DATA-DRIVEN MODELING AND OPTIMIZATION: APPLICATIONS TO SOCIAL COMPUTING

EDITED BY: Chao Gao, Zhanwei Du, Lin Wang and Peican Zhu
PUBLISHED IN: Frontiers in Physics

frontiers Research Topics

ISSN 1664-8714
ISBN 978-2-88976-960-5
DOI 10.3389/978-2-88976-960-5

About Frontiers

Frontiers is more than just an open-access publisher of scholarly articles: it is a pioneering approach to the world of academia, radically improving the way scholarly research is managed. The grand vision of Frontiers is a world where all people have an equal opportunity to seek, share and generate knowledge. Frontiers provides immediate and permanent online open access to all its publications, but this alone is not enough to realize our grand goals.

Frontiers Journal Series

The Frontiers Journal Series is a multi-tier and interdisciplinary set of open-access, online journals, promising a paradigm shift from the current review, selection and dissemination processes in academic publishing. All Frontiers journals are driven by researchers for researchers; therefore, they constitute a service to the scholarly community. At the same time, the Frontiers Journal Series operates on a revolutionary invention, the tiered publishing system, initially addressing specific communities of scholars, and gradually climbing up to broader public understanding, thus serving the interests of the lay society, too.

Dedication to Quality

Each Frontiers article is a landmark of the highest quality, thanks to genuinely collaborative interactions between authors and review editors, who include some of the world's best academicians. Research must be certified by peers before entering a stream of knowledge that may eventually reach the public - and shape society; therefore, Frontiers only applies the most rigorous and unbiased reviews.
Frontiers revolutionizes research publishing by freely delivering the most outstanding research, evaluated with no bias from both the academic and social point of view. By applying the most advanced information technologies, Frontiers is catapulting scholarly publishing into a new generation.

What are Frontiers Research Topics?

Frontiers Research Topics are very popular trademarks of the Frontiers Journals Series: they are collections of at least ten articles, all centered on a particular subject. With their unique mix of varied contributions from Original Research to Review Articles, Frontiers Research Topics unify the most influential researchers, the latest key findings and historical advances in a hot research area! Find out more on how to host your own Frontiers Research Topic or contribute to one as an author by contacting the Frontiers Editorial Office: frontiersin.org/about/contact

DATA-DRIVEN MODELING AND OPTIMIZATION: APPLICATIONS TO SOCIAL COMPUTING

Topic Editors:
Chao Gao, Southwest University, China
Zhanwei Du, The University of Hong Kong, SAR China
Lin Wang, University of Cambridge, United Kingdom
Peican Zhu, Northwestern Polytechnical University, China

Citation: Gao, C., Du, Z., Wang, L., Zhu, P., eds. (2022). Data-driven Modeling and Optimization: Applications to Social Computing. Lausanne: Frontiers Media SA. doi: 10.3389/978-2-88976-960-5

Table of Contents

TYPE Editorial
PUBLISHED 03 August 2022
DOI 10.3389/fphy.2022.987431

Check for updates

OPEN ACCESS

EDITED AND REVIEWED BY
Matjaž Perc,
University of Maribor, Slovenia

*CORRESPONDENCE
Zhanwei Du,
zwdu@hku.hk

†These authors have contributed equally to this work

SPECIALTY SECTION
This article was submitted to Social Physics,
a section of the journal
Frontiers in Physics

RECEIVED 06 July 2022
ACCEPTED 07 July 2022
PUBLISHED 03 August 2022

CITATION
Gao C, Wang L, Zhu P and Du Z (2022), Editorial: Data-driven modeling and optimization: Applications to social computing.
Front. Phys. 10:987431.
doi: 10.3389/fphy.2022.987431

Editorial: Data-driven modeling and optimization: Applications to social computing

Chao Gao[1†], Lin Wang[2†], Peican Zhu[1] and Zhanwei Du[3*]

[1]School of Artificial Intelligence, Optics and Electronics, Northwestern Polytechnical University, Xi'an, China, [2]Department of Genetics, Faculty of Biology, University of Cambridge, Cambridge, United Kingdom, [3]School of Public Health, LKS Faculty of Medicine, The University of Hong Kong, Hong Kong, Hong Kong SAR, China

KEYWORDS

social computing, data-driven modelling, optimization, disease, applications

Editorial on the Research Topic
Data-driven modeling and optimization: Applications to social computing

With the development of computer technology and the exponentially increasing capacity in generating new data, using big data to analyze various problems has become a new norm of research and development. Nowadays, atmospheric science, genomics, biology, remote sensing, medical records, and other fields can generate new data in high-throughput fashion. Big data which has become the characteristics of this era has been integrated into our production and life. Massive data gives rise to challenges as well as opportunities, which promotes the development of new data analysis methods. Data-driven modeling and optimization has become the main method to study practical problems now. Through the analysis of a large amount of data, the underlying patterns and dynamics can be understood much clear which provide a theoretical basis for promoting the better development of society.

This Research Topic has collected a series of studies on different social problems. Those studies mainly focused on establishing models for solving and prediction based on massive data, for instance, Li et al. analyze the network characteristics and predicts the popularity of yoga . Sun et al. establish a new data ownership verification mechanism using Σ-protocol and Pedersen commitment. Wang et al. predict the future development of EV charging piles in China by clustering methods. Yuan et al. study the network generated by urban living and working interaction patterns which impacts the formation of urban structure. Han et al. construct an information dissemination index system from multilevel and complex perspectives. Bai et al. establish a model for identifying the meteorological elements that affect the vegetation coverage change in China. Liu et al. propose a strategy of emergency material allocation in uncertain environments. Qu et al. use visibility graph network to analyze levels of concern about joint punishment for dishonesty.

We also collect some papers on economics and health. Portfolio optimization model, the optimum window of the stock price model and the stand index of the virtual cryptocurrency trading popularity model are established. (Yan et al., Liu et al., Zhu et al.). For COVID-19, there are three models which calculate the number of infections by COVID-19 tests at the Tokyo Olympics, develop a general modeling to estimate the importation risk of COVID-19 and develop a new tool for local surveillance of the COVID-19 outbreak (Vico Lau et al., Xu et al., Liu et al.). And, Hong Hu et al. establish the model for simulating the transmission route of the African swine fever virus in China. Du et al. reveal the influencing factors of cervix cancer.

Last, this Research Topic has collected the studies on deep learning framework. Zhang et al. propose a model which takes the advance of the graph neural networks and overcomes the data sparsity problem for social recommendation. Li et al. propose CR model for state fragility index. Lin et al. propose methods for vehicle detection and vehicle counting. Song et al. construct NMDRL model for studying propagation, game, and cooperation behaviors in networks. Yang et al. propose an approach for accurately estimating the global and local influence of social networks. Sun et al. introduce the semivariogram into geostatistics and study system robustness for three different dynamical models. Sheng et al. solve the maximization of the dynamic influence in low-dimensional latent space by network representation learning. Li et al. develop a network rewire mechanism using multi-objective optimization to enhance the robustness of complex networks.

This Research Topic aims to know the development trend of methods that solve important problems based on massive data at present. The source of papers is diverse and data-driven models and optimization are applied to many fields. The authors have established different models for public health, society, machine learning, and economics, and provide efficient reference values for solving complex problems in different fields.

Author contributions

All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

Funding

Financial support was provided by the AIR@InnoHK Programme from Innovation and Technology Commission of the Government of the Hong Kong Special Administrative Region, the National Natural Science Foundation of China (Nos. 61976181, U1803263, 62073263), the Technological Innovation Team of Shaanxi Province (No. 2020TD-013), the Fundamental Research Funds for the Central Universities under Grant D5000210738.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers
in Physics

ORIGINAL RESEARCH
published: 10 August 2021
doi: 10.3389/fphy.2021.735998

A Novel Edge Rewire Mechanism Based on Multiobjective Optimization for Network Robustness Enhancement

Zhaoxing Li[1,2]*, *Qionghai Liu*[3] *and Li Chen*[2]

[1]College of Information Engineering, Yulin University, Yulin, China, [2]School of Information Technology, Northwest University, Xi'an, China, [3]Yellow River, Shaanxi and Mongolia Supervision Bureau, Yulin, China

A complex network can crash down due to disturbances which significantly reduce the network's robustness. It is of great significance to study on how to improve the robustness of complex networks. In the literature, the network rewire mechanism is one of the most widely adopted methods to improve the robustness of a given network. Existing network rewire mechanism improves the robustness of a given network by re-connecting its nodes but keeping the total number of edges or by adding more edges to the given network. In this work we propose a novel yet efficient network rewire mechanism which is based on multiobjective optimization. The proposed rewire mechanism simultaneously optimizes two objective functions, i.e., maximizing network robustness and minimizing edge rewire operations. We further develop a multiobjective discrete partite swarm optimization algorithm to solve the proposed mechanism. Compared to existing network rewire mechanisms, the developed mechanism has two advantages. First, the proposed mechanism does not require specific constraints on the rewire mechanism to the studied network, which makes it more feasible for applications. Second, the proposed mechanism can suggest a set of network rewire choices each of which can improve the robustness of a given network, which makes it be more helpful for decision makings. To validate the effectiveness of the proposed mechanism, we carry out experiments on computer-generated Erdős–Rényi and scale-free networks, as well as real-world complex networks. The results demonstrate that for each tested network, the proposed multiobjective optimization based edge rewire mechanism can recommend a set of edge rewire solutions to improve its robustness.

Keywords: complex networks, network robustness, network rewire mechanism, multiobjective optimization, partite swarm optimization

OPEN ACCESS

Edited by:
Zhanwei Du,
University of Texas at Austin,
United States

Reviewed by:
Liping Huang,
Nanyang Technological University,
Singapore
Qing Cai,
Nanyang Technological University,
Singapore

****Correspondence:***
Zhaoxing Li
abde0309@163.com

Specialty section:
This article was submitted to
Social Physics,
a section of the journal
Frontiers in Physics

Received: *04 July 2021*
Accepted: *21 July 2021*
Published: *10 August 2021*

Citation:
Li Z, Liu Q and Chen L (2021) A Novel Edge Rewire Mechanism Based on Multiobjective Optimization for Network Robustness Enhancement.
Front. Phys. 9:735998.
doi: 10.3389/fphy.2021.735998

1 INTRODUCTION

In our daily life, complex networks are ubiquitous [1–3]. We rely on diverse kinds of complex networks as they play a very important role in our lives [4–6]. For example, we rely on transportation networks to travel around the world, while we count on power grid networks to provide power supplies to ensure social productions and lives [7]. Complex networks are so important to us. However, complex networks in real-world often suffer from unpredictable disturbances such as random component failures and/or target attacks [8,9]. Those disturbances can lead to the dysfunction of network components [10]. Because the components of a network generally interact with each other, therefore the dysfunction of some components can trigger the dysfunction of other components and system-level disasters even can appear. It is therefore of great scientific and social values to ensure the stabilities and reliabilities of complex networks [11,12].

Up to now, scientists have developed many ways to quantify stabilities and reliabilities of complex networks [13,14]. One of the most effective ways is based on network robustness analysis [15]. The robustness of a network quantifies the capability of the network to survive disturbances. Scientists in the past decade have carried out many studies on network robustness. The majority of existing studies on network robustness of based on graph theories [16,17]. A graph is a straightforward yet effective modelling of a complex network. The nodes of a graph are the entities of a complex network, while the edges of the graph represent the interactions among the entities. Existing studies on network robustness can be roughly grouped into two classes. The first class aims to develop quantitative metrics to measure the robustness of a given network. The second class is dedicated to investigating models and theories to improve the robustness of a given network.

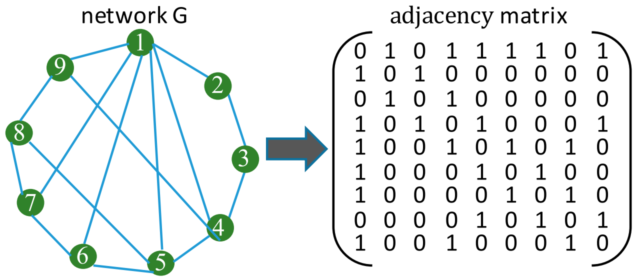
Regarding network robustness quantification, one of the most studied methods is the percolation theories [18,19]. Suppose that $1 - p$ portion of the nodes of a network lose their functionalities due to disturbances. Then the purpose of percolation theories is to mathematically calculate the remaining portion of nodes. Such kind of percolation theories are termed as site percolation [4]. Note that the remaining portion of nodes is no higher than p since the dysfunction of some nodes can lead to the dysfunction of other nodes that rely on those initially failed nodes. Because the dysfunction can be happened to the edges of a network, the corresponding percolation theories are deemed as bond percolation [4]. It has been widely reported that many complex networks show community structures [20]. A network community is commonly regarded as a subnetwork of a given network and the similarities between nodes within the community are high while similarities between communities are low. For a network with community structure, when some of its components are failed, then those failed components will first affect components in their own communities. Therefore, some researchers also investigate the robustness of complex networks to component failures at a mix-level. For example, the work in [21] for the first time investigated the robustness of ecological networks to the species loss of community. Interestingly, they discovered that the community-level robustness of ecological networks has positive correlation with that of node-level robustness.

Percolation theories provide a way to measure the robustness of complex networks. Another challenging question is how to improve the robustness of complex networks. For real-world applications, the back-up mechanism is widely used [16]. Literally, the back-up mechanism improves the robustness of a network by backing up its critical components. Note that to back-up a network component needs extra resources and sometimes it is not easy to do that. For example, it may be difficult to back up a power station due to physical connection issues. Consequently, scientists have developed another effective way, i.e., network rewire mechanism, to improve the robustness of complex networks [22,23]. A network rewire mechanism changes the connections between the nodes of a network so as to improve its robustness. Generally, a rewire mechanism is subject to the constraint that the total number of edges is fixed. However, some researchers propose to discard this constraint when design network rewire mechanisms [24].

Network rewire mechanisms have been proved to be very effective for improving network robustness. However, existing network rewire mechanisms have two major drawbacks. The first one is that a rewire mechanism normally generates a robust network whose structure is quite different from its original one. This drawback makes existing rewire mechanisms hard to be applied to real situations. The second one is that existing rewire mechanisms can only generate single solutions which is not intelligent for smart and personalized decision making.

In order to overcome those two drawbacks mentioned above, in this paper we suggest a novel network rewire mechanism for improving network robustness. Specifically, we propose a multiobjective optimization based network rewire mechanism. The proposed mechanism simultaneously optimizes two objectives. The first objective is to maximize the network robustness. This objective makes sure that a feasible network rewire solution definitely increase the robustness of the studied network. The second objective is to minimize the number of network rewire operations. This objective ensures that a feasible network rewire solution can improve the robustness of the studied network by implementing as small number of rewire operations as possible, making the proposed network rewire mechanism be of practical use. In order to solve the optimization based model, we further develop a multiobjective particle swarm optimization algorithm. Each time we run the developed algorithm, the algorithm can generate a set of solutions. Each solution denotes a network rewire choice with which the robustness of the studied network can be improved. Compared to existing network rewire mechanisms, the newly proposed network rewire mechanism better facilitates decision making. In order to validate the effectiveness of the proposed mechanism, we carry out experiments on Erdős–Rényi and scale-free networks that are generated using computer models. We also carry out experiments on several real-world complex networks. Experiments demonstrate that the proposed optimization-based network rewire mechanism can provide many different choices to improve the robustness of the tested networks. For real-world applications, one may analyze the detailed properties of the given network and decide which choices can be chosen to improve the network's robustness.

The remainder of this paper is organized as follows. **Section 2** presents preliminaries for better understanding of this work. **Section 3** formulates the proposed research problem and **Section 4** delineates the designed algorithm to solve the proposed problem. **Section 5** demonstrates the experiments and **Section 6** concludes the paper.



$$\begin{pmatrix} 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

FIGURE 1 | An example of a simple network together with its corresponding adjacency matrix.

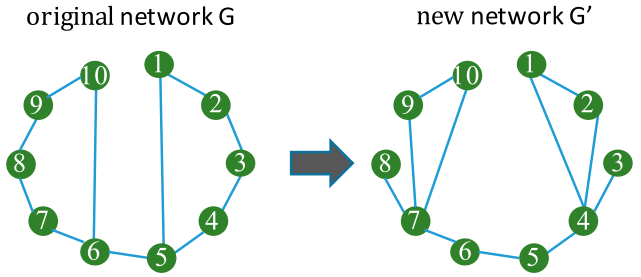


FIGURE 2 | An example of a simple network rewire mechanism.

2 RELATED BACKGROUNDS

2.1 Network Notations

A complex network is usually represented by a graph. A graph consists of nodes and edges. In the literature, a node also can be called a vertex while an edge can be called a link. Mathematically, a graph is generally denoted by $G = \{V, E\}$. In this kind of notation, the symbol V represents the node set and the symbol E represents the edge set. The nodes of a graph represent the entities of a network that the graph models, while the edges represent the relationships between the entities.

The relationships between the nodes of a graph can be depicted using its adjacency matrix which is usually denoted by $\mathbf{A}$. **Figure 1** shows an example of a simple network together with its corresponding adjacency matrix.

For the graph G shown in **Figure 1** with $n = 9$ nodes, its adjacency matrix is normally with size $n \times n$. The element of $\mathbf{A}$ is usually denoted by a_{ij}, representing the relationship between nodes i and j. Note that the value of a_{ij} is problem-specific. For a binary network with $a_{ij} \in \{1, 0\}$, the element a_{ij} only represents whether there is connection between nodes i and j.

2.2 Network Rewire Mechanism

A network rewire mechanism is usually used in the network science domain. For a given network G, a network rewire mechanism aims to generate a new network G'. **Figure 2** shows an example of a simple network rewire mechanism.

In **Figure 2**, the original network has 10 nodes and 11 edges. After adopting a simple rewire mechanism, network G has been changed into network G'. It can be seen from the figure that the network rewire mechanism just changes the edge connections between the nodes and does not change the total number of edges of the original network.

2.3 Particle Swarm Optimization

Many real-world problems can be reformulated as optimization problems. Because real-world problems could be very difficult, therefore when turning a real-world problem into an optimization problem, the optimization formula may not have mathematical properties like gradient. In this case, mathematical methods cannot solve such kind of optimization problems.

In order to solve optimization problems that cannot be solving using traditional mathematical methods, scientists have developed the so called bio-inspired algorithms [25,26] and amongst which is the particle swarm optimization algorithm (PSO) [27–29].

Suppose that one aims to maximize a function $f(x)$ with x being the argument by using PSO. Then a PSO algorithm works with a swarm or a population of individuals. Each individual is a feasible solution x to the optimized problem $f(x)$. Let us use p_i to denote the ith population and $x_j \in \mathrm{p}_i$ to denote the jth individual in p_i. Then a PSO algorithm iteratively evolves p_i to approximate the optimal solutions to $f(x)$. During each iteration, the algorithm updates individual x_j in the following way:

$$\boldsymbol{x}_j = \boldsymbol{x}_j + \boldsymbol{v}_j \tag{1}$$

where v_j is the velocity of individual x_j. The velocity v_j is calculated as follows

$$\boldsymbol{v}_j = w\boldsymbol{v}_j + c_1 r_1 \left(\boldsymbol{L}_j - \boldsymbol{x}_j\right) + c_2 r_2 \left(\boldsymbol{G} - \boldsymbol{x}_j\right) \tag{2}$$

where w, c_1 and c_2 are three constants. r_1 and r_2 are two random number with $r_1, r_2 \in [0, 1]$. The symbol $\boldsymbol{L}_j$ is the personally best individual of x_j, i.e., the best solution that individual x_j has so far found. The symbol $\boldsymbol{G}$ is the globally best individual found by all the individuals.

Note that an individual updates its velocity v_j by referring to its historical information and the information from the whole population. The new velocity v_j then could help individual x_j to explore to promising area. During each iteration, the algorithm updates L_j and $\mathbf{G}$. By iteratively evolving p_i for a prescribed number of iteration, the algorithm stops and $\mathbf{G}$ therefore is regarded as the optimal solution(s) to the original optimization problem.

3 RESEARCH PROBLEM AND PROBLEM FORMULATION

3.1 Network Robustness Measurement

The research aims to enhance the robustness of a given network by proposing a novel network rewire mechanism. To do so, first we need to know how to quantify the robustness of a given network. In the literature, there are many studies on network robustness [16,30]. As a consequence, many methods have been

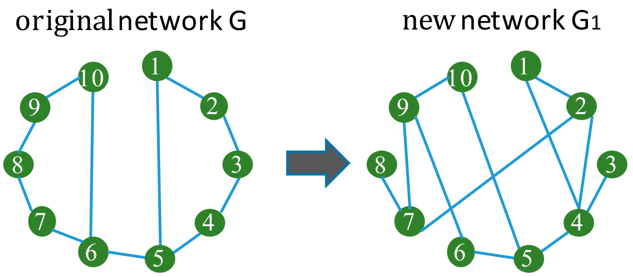


FIGURE 3 | The basic idea of the proposed network rewire mechanism to enhance the robustness of a given complex network. In the proposed rewire mechanism, both edge additions and deletions are allowed without extra constrains.

developed by scientists to quantify the robustness of complex networks [13,31] and amongst which is the method based on spectral analysis [31].

Given a network G with its adjacency matrix being denoted by A. Let us use d_i to represent the degree of node i, i.e., the number of edges attached to i. Then one can easily obtain a diagonal matrix D in which its ith element is d_i. In the literature, the Laplacian matrix L of network G is defined as

$$\mathbf{L} = \mathbf{D} - \mathbf{A} \tag{3}$$

For the matrix L we can work out all its eigenvalues $\boldsymbol{\lambda}$. An eigenvalue λ satisfies the following relation

$$\mathbf{L}\boldsymbol{x} = \lambda \boldsymbol{x} \tag{4}$$

in which x is the corresponding eigenvector. Let $\lambda_2 \in \lambda$ be the second smallest non-negative eigenvalue of matrix L. It has been found that λ_2 has a positive relationship with the network's connectivity [31]. As a consequence, in the literature scientists use it as a metric to quantify the network's robustness. Due to its computationally friendly feature, in this study we use λ_2 as the network robustness quantification metric. Normally, the larger the value of λ_2, the more robust the given network.

3.2 Proposed Network Rewire Mechanism

In this work, we propose a new network rewire mechanism to enhance the robustness of a given complex network. **Figure 3** takes a simple network as an example to show the basic idea of the proposed network rewire mechanism.

On the left hand side of **Figure 3** is a simple network G_0 with 10 nodes. The network G_0 originally carries with 11 edges. In order to enhance its robustness, existing rewire mechanisms change the edge connections. Specifically, they reconnect the nodes and aim to find out a new network G_1 with exactly the same number of edges and higher network robustness.

In this work we propose a novel rewire mechanism as shown on the right hand side of **Figure 3**. The proposed mechanism does not require that G_1 has exactly the same number of G. As can be seen from the right hand side of **Figure 3**, we delete some edges from G. We also add new edges between the nodes. As long as G_1 has higher network robustness than G, then we can both delete edges from and add new ones to G. As compared to existing rewire mechanisms, the newly proposed mechanism is more flexible and could be more of practical usage.

Note that for a given network there are many choices for edge deletion and addition. In order to determine which edges can be deleted and added, we propose a multiobjective optimization model which will be described in detail in the subsequent subsection.

3.3 Proposed Optimization Model

For a given network G we adopt the λ_2 metric to quantify its robustness. The proposed network rewire mechanism aims to enhance G's robustness by removing and adding edges. Specifically, we propose the following multiobjective optimization model.

$$\begin{cases} f_1 = \min \ (\lambda_2^0 - \lambda_2^1) \\ f_2 = \min \ |E^-| \\ f_3 = \min \ |E^+| \end{cases} \tag{5}$$

in which λ_2^0 is the robustness of G, E^- is the set of edges to be deleted from G and E^+ is set of edges to be added to G. For network G the proposed rewire mechanism removes edges in the set E^- and adds edges in the set E^+. Then G becomes G_1 with respect to E^- and E^+ and λ_2^1 is its corresponding robustness.

In the above model, objective f_1 aims to maximize the robustness of G_1. Both objectives f_2 and f_3 aim to minimize the edge operations. For a network in reality, it is not good to remove a large amount of edges from the network as it will change the network's structure a lot. Objective f_2 then constraints this operation. Similarly, it is not practical to add a large amount of edges to the network as adding new edges needs extra workload. Objective f_3 is such a constraint to avoid this behaviour.

Note that existing rewire mechanisms also do edge addition and removal, i.e., they consider both objectives f_2 and f_3. However, they require to remain the total number of edges in the network, i.e., they require that $f_2 = f_3$. In the newly proposed rewire mechanism, we do not pose such constraints. As a consequence, the proposed mechanism is of more practical use. For networks in which adding edges is more convenient than that of deleting edges, then we can minimize f_2 but relax f_3. Similarly, if deleting edges is much easier than adding new ones, then we can pay more attention to f_3.

4 ALGORITHM DESIGN

4.1 Algorithm Framework

One may clearly see from **Eq. 5** that the proposed optimization model does not have a direct relationship between the decision variable (E^+ and E^-) and the objective function values. A concrete solution of E^+ and E^- cannot directly be substituted into the calculation of f_1. Meanwhile, E^+ and E^- are edge sets.

The proposed model contains three objectives. In order to solve it, traditional mathematical methods will apply weighted sum method to do it. However, since the edge sets are composed of links, therefore mathematical methods cannot solve **Eq. 5** as the decision variables are discrete and the objectives do not have any gradient information. Because the proposed optimization model given in **Eq. 5** involves three discrete objectives, in the literature, scientists have developed nature-inspired algorithms such as genetic algorithms (GAs) [32], particle swarm optimization algorithms (PSOs) [33], etc., to solve such kind of problems. Both GAs and PSOs are designed for single objective optimization. In order to solve multiobjective optimization, representative algorithms like the non-dominated sorting genetic algorithm-version II (NSGA-II) [34], multiobjective particle swarm optimization (MOPSO) [35], multiobjective evolutionary algorithm based on decomposition (MOEA/D) [36], etc., have been developed by scientists and been applied to solve many engineering problems.

In order to solve the model shown in **Eq. 5**, we in this work adopt the MOPSO algorithm proposed in [27]. The MOPSO algorithm is chosen because of its simplicity in terms of algorithm understanding and implementation. We redesign some operators of the MOPSO algorithm to make it fit for the model in **Eq. 5**. The whole algorithm framework for solve **Eq. 5** is given in **Algorithm 1**.

Step 4 of **Algorithm 1** involves the individual representation method which will be described in the following subsection. Step 7a of **Algorithm 1** evaluates a particle based on the proposed multiobjective optimization model. Step 10 of **Algorithm 1** is the main for loop which mainly contains the particles' position and velocity information update and this will be described in the following

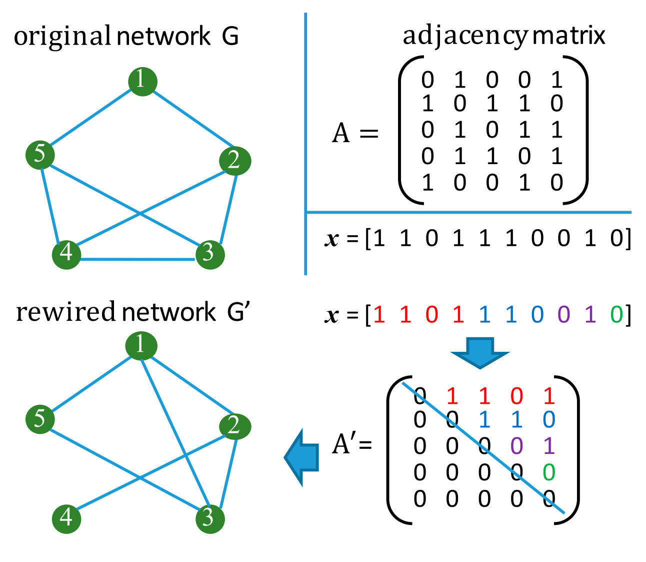


FIGURE 4 | An illustration of a simple network and a possible representation of a particle position.

Algorithm 1 Framework of the modified MOPSO algorithm for solving the proposed optimization based network rewire mechanism.

```
1. $[ps, np, c_1, c_2, w] = InitializeParameters()$;
2. $\mathbf{A} = GenerateNetwork(n, d, \lambda, k_{min}, k_{max})$;
3. $\lambda_2^0 = RobustnessEvaluate(\mathbf{A})$;
4. $\mathbf{X} = InitializePosition()$ // $\mathbf{X} = (x_1, x_2, ..., x_{ps})$ and $x_i$ is the $i$-th individual's position information.
5. $\mathbf{V} = InitializeVelocity()$ // $\mathbf{V} = (v_1, v_2, ..., v_{ps})$ and $v_i$ is the $i$-th individual's velocity information.
6. $\mathbf{L} = InitializePersonalBest()$ // $\mathbf{L} = (L_1, L_2, ..., L_{ps})$ and $L_i$ is the $i$-th individual's personal best position.
7. For all $i \in [1, ps]$, do
     a. $[f_1, f_2, f_3] = FitnessEvaluation(x_i, \mathbf{A})$;
     b. $L_i = UpdatePersonalBest(x_i)$;
8. End For
9. $\mathbf{G} = UpdateGlobalBest(\mathbf{X})$ // $\mathbf{G} = (G_1, G_2, ...,)$ is the global optimal solution set.
10. For all $j \in [1, np]$, do
     a. Forall $i \in [1, ps]$, do
        (1) $G_r = RandomSelect(G)$. // Randomly select one particle from G as the gbest particle.
        (2) $v_i' = UpdateVelocity(v_i, L_i, x_i, G_r)$. // Update $i$-th particle's velocity.
        (3) $x_i' = UpdatePosition(x_i, v_i')$. // Update $i$-th particle's position.
        (4) $[f_1, f_2, f_3] = FitnessEvaluation(x_i', \mathbf{A})$;
        (5) $L_i = UpdatePersonalBest(x_i')$;
     b. $\mathbf{G} = UpdateGlobalBest(\mathbf{X})$
11. End For
```

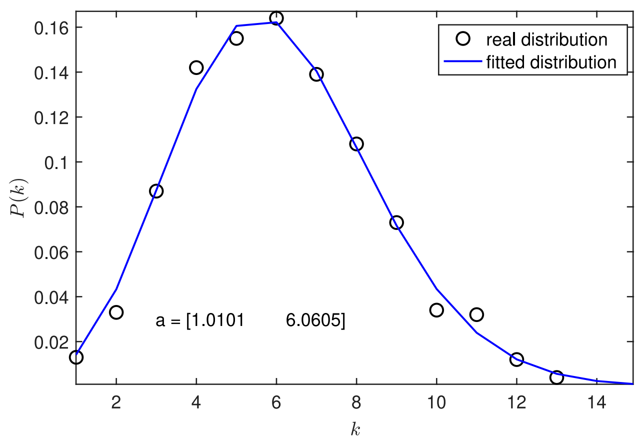


FIGURE 5 | Degree distribution of a randomly generated Erdős–Rényi network (n = 1000, d = 6). The function $P(k) = a_1 e^{-a_2} \frac{a_2^k}{k!}$ is used for curve fitting.

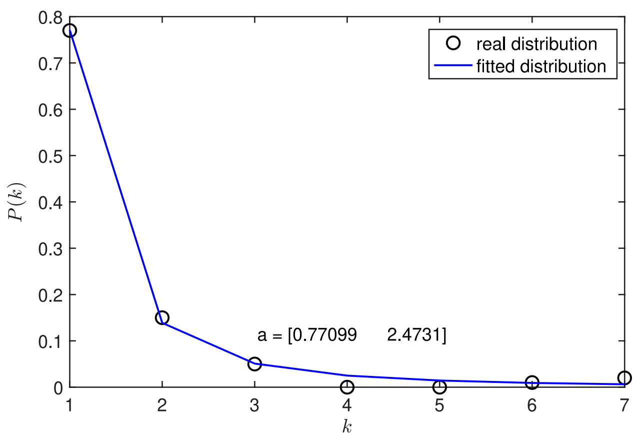


FIGURE 6 | Degree distribution of a randomly generated scale-free network (n = 100, λ = 2.4, $k_{\min}$ = 1, $k_{\min}$ = 10). The function $P(k) = a_1 k^{-a_2}$ is used for curve fitting.

subsection. Note that the details for generating a testing network will be provided in the experiment section and all the parameter settings will also be given in the experiment section.

4.2 Representation of Particle Potion

The algorithm proposed in [27] is for network clustering. The authors therefore developed an integer based particle position representation method. Since in this work we aim to do network rewire to enhance network robustness, we thus need to reconsider the way to represent a particle position.

Keep in mind that the purpose of the work is to find out the edge sets E^+ and E^- so as to improve the robustness of a given network. For a given network G, we know its adjacency matrix A and its edge set E. We then can remove edges from E and E^- can be determined. Based on matrix A, we also know the possible new edges between the nodes and therefore can determine E^+. Considering this, we thus propose the following way to represent the position of the ith particle of a PSO algorithm.

$$\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{im}) \tag{6}$$

In **Eq. 6**, $x_{ij} \in \boldsymbol{x}_i$ is a binary variable, i.e., $x_{ij} \in \{0, 1\}$. The length of x_i is m and $m = n \times (n-1)/2$. **Figure 4** takes a simple network as an example to show how **Eq. 6** works.

In **Figure 4**, the simple network G has five nodes and seven edges. Its adjacency matrix is given as A. In the figure there is a binary sequence x which represents a particle's position. This binary sequence represents a possible edge rewire solution. We first set up an empty matrix A′. Then we turn the binary sequence x into the elements of the upper triangle matrix of matrix A′. The upper triangle matrix then represents the new edge connection relationship and corresponds to the new network G'.

As mentioned earlier, the purpose of this work is to find E^+ and E^-. So the most straightforward way is to represent x_i as an adjacency matrix A′. Based on the difference between A and A′ we can easily work out E^+ and E^- and therefore can calculate the values of the objective functions. However, this kind of representation requires too much computer memories when n is very big. As can be seen from **Figure 4**, the proposed representation mechanism can reduce the memory size by more than half.

4.3 Representation of Particle Velocity

How to represent the velocity of a particle is related to the way how the position of the particle is represented, since the velocity vector help a partite to update its position information. As compared to the matrix representation method, the proposed particle position representation method is more convenient for the velocity representation.

Since in this work we propose to encode the edges in the upper triangle of the adjacency matrix of a given network using binary coding schema, therefore we decide to use binary representation of the velocity vector of a given particle. Specifically, for a given velocity vector $v_i = (v_{i1}, v_{i2}, \ldots, v_{im})$ corresponding to particle x_i, its element $v_{ij} \in \{0, 1\}$. The element v_{ij} represents whether the jth element in the vector x_i will be changed or not.

4.4 Update of Particle Position and Velocity

For the ith particle with x_i being its position vector, we update its position information with the following equation

$$\boldsymbol{x}_i' = \boldsymbol{x}_i \oplus \boldsymbol{v}_i' \tag{7}$$

in which the operator ⊕ defines the operation for x_i’ as follows

$$x_{ij}' = x_{ij} \quad if \quad v_{ij}' = 0 \tag{8}$$

$$x_{ij}' = \sim x_{ij} \quad if \quad v_{ij}' = 1 \tag{9}$$

The velocity vector v_i' of the ith particle is updated as follows

$$\boldsymbol{v}_i' = S(\boldsymbol{v}) \tag{10}$$

in which the speed vector v is calculated as

$$\boldsymbol{v} = w\boldsymbol{v}_i + c_1 r_1 (\boldsymbol{L}_i - \boldsymbol{x}_i) + c_2 r_2 (\boldsymbol{G}_r - \boldsymbol{x}_i) \tag{11}$$

where $\boldsymbol{G}_r \in \boldsymbol{G}$ is a solution randomly chosen from the global best solution set $\boldsymbol{G}$ and the symbol S () is a confining

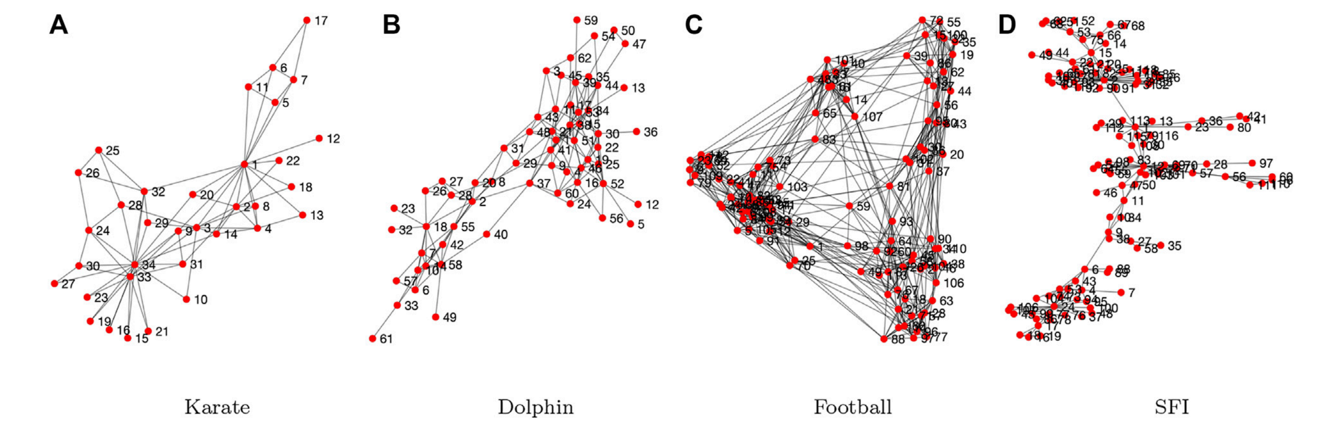


FIGURE 7 | Network structure visualization of (A) the Karate network, **(B)** the dolphin network, **(C)** the Football network and **(D)** the SFI network.

TABLE 1 | Settings of the parameters contained in the algorithm and the generated networks for testing.

Network		**Algorithm**	
Parameter	**Value**	**Parameter**	**Value**
n	(50, 100, 150)	ps	100
d	(4, 5, 6, 7)	np	100
λ	(2, 2.4, 2.6, 2.8)	c_1	1.496
$k_{\min}$	1	c_2	1.496
$k_{\max}$	10	w	0.729

function which maps vector v into v_i'. Specifically, S () works as follows

$$v_{ij}' = 0 \quad if \quad \frac{1}{1+e^{-v_j}} \leq 0.5, \quad v_j \in \boldsymbol{v} \tag{12}$$

$$v_{ij}' = 1 \quad if \quad \frac{1}{1+e^{-v_j}} > 0.5, \quad v_j \in \boldsymbol{v} \tag{13}$$

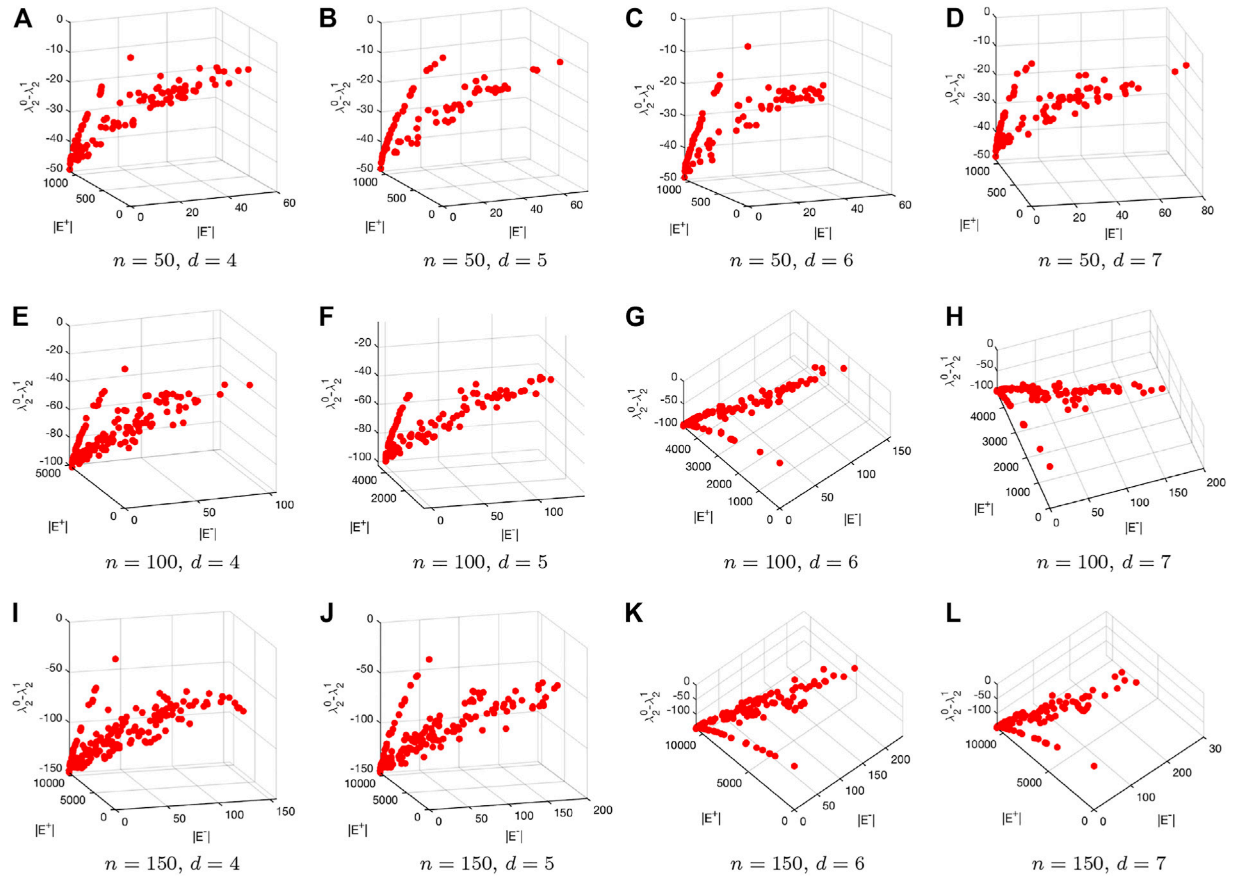
It can be seen from the above equations that v is a real valued vector. By implementing the S () operation we turn v into a binary vector based on which we thus can update the particle's position information. Note that other functions also can be adopted to replace the S () function. However, there is no guarantee that other functions would perform better than the S () function does. The S () function is adopted because it is smooth in the objective space.

4.5 Update of Local and Global Best Particles

Note that PSO is a population based algorithm. During the iterations we need to update the information of the local best individual $\boldsymbol{L}_i$ for all $i \in [1, psize]$ and the global best individual $\boldsymbol{G}$

Algorithm 2 The pseudocode for updating the local and global best particles in the adopted MOPSO algorithm.

```
1. For a new individual $x_i$, do
   a. $[f_1^{x_i}, f_2^{x_i}, f_3^{x_i}] = FitnessEvaluation(x_i, \mathbf{A})$;
   b. $[f_1^{L_i}, f_2^{L_i}, f_3^{L_i}] = FitnessEvaluation(L_i, \mathbf{A})$;
   c. If $\forall j \in [1,3]$, $f_j^{x_i} \leq f_j^{L_i}$, then $L_i \leftarrow x_i$.
2. End For
3. For all $i \in [1, ps]$, do
   a. For all $k \in [1, n_g]$ where $n_g = |G|$ is the number of solutions in $G$, do
      (1) If $\forall j \in [1,3]$, $f_j^{x_i} \leq f_j^{G_k}$, then $G_k \leftarrow x_i$, where $G_k \in G$ and $f_j^{G_k}$ is the value of the $j$-th objective
          with respect to $G_k$.
   b. End For
4. End For
```



$n = 50,\ d = 4$ $n = 50,\ d = 5$ $n = 50,\ d = 6$ $n = 50,\ d = 7$

$n = 100,\ d = 4$ $n = 100,\ d = 5$ $n = 100,\ d = 6$ $n = 100,\ d = 7$

$n = 150,\ d = 4$ $n = 150,\ d = 5$ $n = 150,\ d = 6$ $n = 150,\ d = 7$

FIGURE 8 | Visualization of the Pareto fronts obtained by applying the introduced multiobjective particle swarm optimization algorithm to the generated Erdős–Rényi networks.

since they help the entire swarm to explore for better solutions. **Algorithm 2** presents the pseudocode for updating the local and global best particles in the adopted MOPSO algorithm.

As can be seen from **Algorithm 2**, during each iteration, when a new individual x_i is generated, we then evaluate its fitness. If x_i dominates its historical best $\boldsymbol{L}_i$ (step 1c of **Algorithm 2**), then we replace L_i with x_i, otherwise, we keep L_i as it is. When a new population of individuals are generated, we then update $\boldsymbol{G}$. Specifically, we use the Pareto solutions obtained from the current population to update those in the history. When any of the new Pareto solutions dominate any historical Pareto solutions, we then replace the old solutions with the new one (step 3 of **Algorithm 2**).

5 EXPERIMENTAL STUDY

5.1 Network Datasets

5.1.1 Computer-Generated Networks

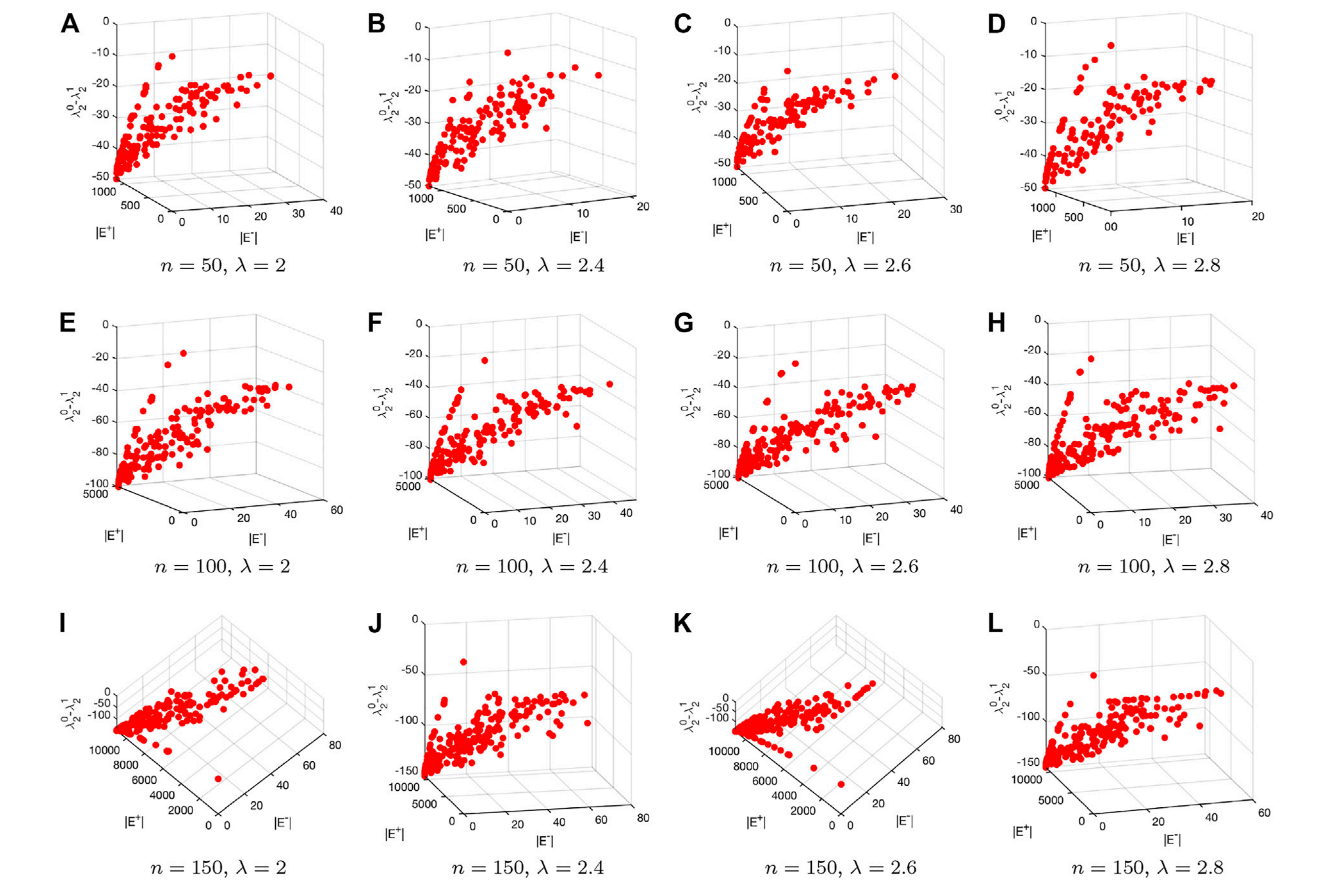
In the experiments we first test the proposed edge rewire mechanism on computer-generated networks. Specifically, we generate two types of networks, i,e., Erdős–Rényi networks and scale-free networks. The degree of an Erdős–Rényi follows Poisson distribution while the degree of a scale-free network follows power law distribution.

In order to generate an Erdős–Rényi network with n nodes, we first generate an empty network G_{ER} with n nodes. We then define a constant d. Then we further define the connection probability r as $r = d/n$. Then we connect each pair of nodes of G_{ER} with probability r. We have proved in our previous paper in [8] that the generated network appropriately follows the following degree distribution

$$p(k) = e^{-d}\frac{d^k}{k!} \tag{14}$$

The network G_{ER} therefore is an Erdős–Rényi network since its degree distribution follows the Poisson distribution and the average degree of G_{ER} is d. **Figure 5** shows the degree distribution and the corresponding curve fitting of an example Erdős–Rényi network generated using the above mentioned method.

In **Figure 5**, the Erdős–Rényi network is generated with $n = 1000$ and $d = 6$. A Poisson distribution function $P(k) = a_1 e^{-a_2}\frac{a_2^k}{k!}$ is used to do the curve fitting. It can be seen



$n = 50,\ \lambda = 2$ $n = 50,\ \lambda = 2.4$ $n = 50,\ \lambda = 2.6$ $n = 50,\ \lambda = 2.8$

$n = 100,\ \lambda = 2$ $n = 100,\ \lambda = 2.4$ $n = 100,\ \lambda = 2.6$ $n = 100,\ \lambda = 2.8$

$n = 150,\ \lambda = 2$ $n = 150,\ \lambda = 2.4$ $n = 150,\ \lambda = 2.6$ $n = 150,\ \lambda = 2.8$

FIGURE 9 | Visualization of the Pareto fronts obtained by applying the introduced multiobjective particle swarm optimization algorithm to the generated scale-free networks.

from **Figure 5** that $a_1 = 1.01$ and $a_2 = 6.06$. The curve fitting result complies well with the distribution given by **Eq. 14**.

In order to generate a scale-free network with n nodes, we first determine the exponent λ in the degree distribution function $p(k) = ck^{-\lambda}$ with k being the nodes degree and c being a constant. We then determine the largest degree $k_{\max}$ and the smallest degree $k_{\min}$. Based on the three parameters λ, $k_{\max}$ and $k_{\min}$ we then solve the following equation

$$\sum_{k_{min}}^{k_{max}} ck^{-\lambda}dk = 1 \tag{15}$$

By solving the above equation we then work out the constant c. With all the above, we then sample a degree sequence $k = (k_1, k_2, \ldots, k_n)$ from the degree distribution function $p(k) = ck^{-\lambda}$. Then we use the method proposed in [37] to work out networks G_{SF} whose has exactly the sampled degree sequence k.

In **Figure 6**, the scale-free network is generated with $n = 100$, $\lambda = 2.4$, $k_{\min} = 1$, and $k_{\min} = 10$. A power-law distribution function $P(k) = a_1k^{-a_2}$ is used to do the curve fitting. It can be seen from **Figure 6** that $a_1 = 0.77$ and $a_2 = 2.47$. By solving **Eq. 15** we get $C = 0.737$. Therefore, the curve fitting result complies well with the theoretical analysis.

5.1.2 Real-World Complex Networks

Apart from the computer-generated networks, in the experiments we also test the proposed edge rewire mechanism on several real-world networks. Specifically, the following networks chosen from the survey paper in [20] are tested.

5.1.2.1 The Karate Network

This network contains 34 nodes and 78 edges. It depicts the relationships between 34 members with the Karate club. The edges represent the relationships among the members. The structure of the Karate network is shown in **Figure 7A**.

5.1.2.2 The Dolphin Network

The dolphin network contains 62 nodes each of which represents a bottlenose dolphin. Three are 159 edges in the network. The edges are built based on statistically significant frequent association, i.e., their social behaviours. The structure of the Dolphin network is shown in **Figure 7B**.

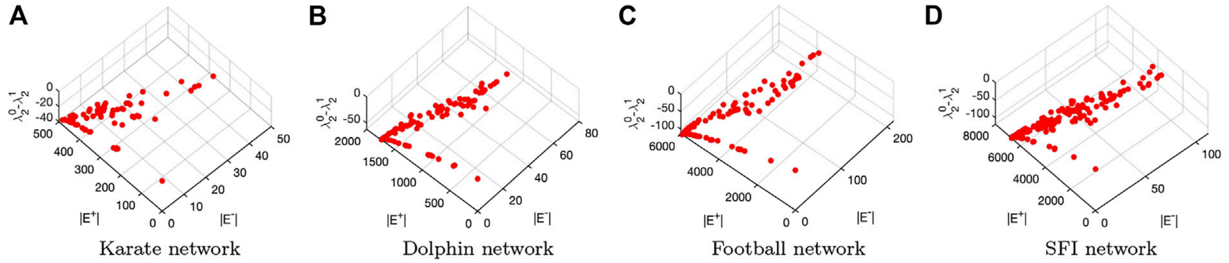


FIGURE 10 | Visualization of the Pareto fronts obtained by applying the introduced multiobjective particle swarm optimization algorithm to **(A)** the Karate network, **(B)** the dolphin network, **(C)** the Football network and **(D)** the SFI network.

TABLE 2 | Robustness improvement made by the two selected extreme solutions from the Pareto front for each of the tested Erdős–Rényi (ER) and scale-free (SF) networks.

Er	$d = 4$	$d = 4$	$d = 5$	$d = 5$	$d = 6$	$d = 6$	$d = 7$	$d = 7$
$n = 50$	55.5537	89.5042	2.6829	6.1284	13.9347	11.4976	4.8016	4.9753
$n = 100$	19.1287	89.7005	24.1805	52.3923	84.2993	90.4997	9.2648	10.4689
$n = 150$	144.3407	148.6653	55.7901	83.6496	17.4103	81.3385	62.9916	71.7342
SF	$\lambda = 2$	$\lambda = 2$	$\lambda = 2.4$	$\lambda = 2.4$	$\lambda = 2.6$	$\lambda = 2.6$	$\lambda = 2.8$	$\lambda = 2.8$
$n = 50$	59.8475	156.2997	66.5295	75.7222	32.2473	33.9289	35.8288	40.4335
$n = 100$	325.3390	653.7876	316.2841	712.6357	219.9658	234.9531	111.1764	155.4250
$n = 150$	1813.5327	1836.7281	242.6360	513.1527	771.9283	829.3302	773.0175	1816.6741

TABLE 3 | Robustness of the real-world networks and their optimized networks with respect to the two selected extreme solutions from the Pareto fronts.

Network	λ_2^0	λ_2'	λ_2''
Karate	0.4685	8.9665	9.0996
Dolphin	0.1730	7.3211	10.0706
Football	1.4590	20.4772	42.9840
SFI	0.0147	18.3791	40.9159

5.1.2.3 The Football Network

The football network represents the American football games between Division IA colleges during regular season Fall 2,000. The football network contains 115 nodes each of which represents a college. Three are 613 edges in the network. An edge is built between two nodes if the corresponding colleges have a game. The structure of the Football network is shown in **Figure 7C**.

5.1.2.4 The SFI Network

The SFI network represents the collaborations between the scientists working with Santa Fe Institute during any part of calendar year 1999 or 2000. The SFI network contains 118 nodes and 200 edges. The structure of the SFI network is shown in **Figure 7D**.

Figure 7 visualizes the structures of the four tested real-world networks. These four networks are chosen because they have been widely used in the network science domain. Note that the four real-world networks are unweighted, i.e., they are binary networks. In the experiments, we only test the proposed mechanism on small-scale networks. On the one hand, the proposed mechanism can be tested on large-scale networks. On the other hand, testing on large-scale network is time consuming as the adopted PSO algorithm is a population based random search algorithm. As the main purpose of this work is to validate the feasibility of the proposed mechanism, therefore we do not carry out experiments on large-scale networks at the moment.

5.2 Parameter Settings

As mentioned above, when generating the networks we need to determine some parameters first. Meanwhile, for the introduced algorithm it also carries with several parameters. In the experiments, we set all the needed parameters as what are shown in **Table 1**.

Note that the main idea of this work is to validate if the proposed optimization based edge rewire mechanism is effective or not. Therefore, we do not set the value of n large enough. For a network with n nodes, there are maximum $n \times (n-1)/2$ edges. If n is too large, then the optimization process will take a long time to finish. In the experiments we set it to be small so as to quickly check if the proposed idea is feasible or not. This is the main reason why the tested real-world networks are small in size.

5.3 Obtained Pareto Fronts

In the experiments, we generate two types of artificial networks. For each type of networks, we generate 12 networks with the configurations of the number of nodes and the degree control parameter, i.e., d for the Erdős–Rényi network and λ for the scale-free network. **Figure 8** and **Figure 9** respectively visualize the Pareto fronts obtained by the introduced multiobjective particle

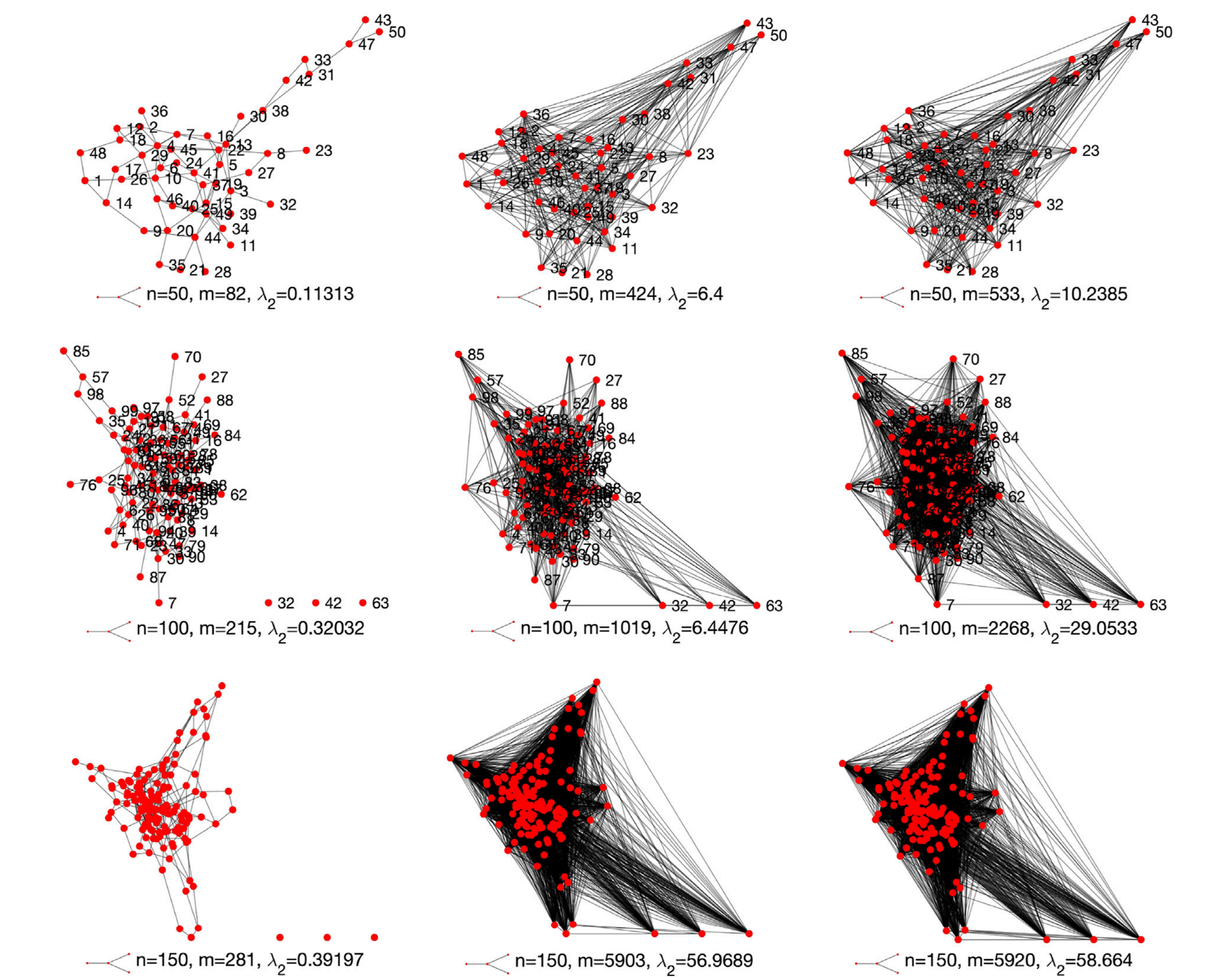

n=50, m=82, λ_2=0.11313
n=50, m=424, λ_2=6.4
n=50, m=533, λ_2=10.2385
n=100, m=215, λ_2=0.32032
n=100, m=1019, λ_2=6.4476
n=100, m=2268, λ_2=29.0533
n=150, m=281, λ_2=0.39197
n=150, m=5903, λ_2=56.9689
n=150, m=5920, λ_2=58.664

FIGURE 11 | Original (first column) and optimized structures (second and third columns) of the Erdős–Rényi networks for d = 4.

swarm optimization algorithm when tested on the Erdős–Rényi and scale-free networks. Note that for each tested network we run the introduced multiobjective particle swarm optimization algorithm for 20 times. After each run we save the Pareto solutions. We eventually merge all the Pareto solutions over the 20 runs and filter out the final Pareto solutions based on Pareto dominance mechanism.

Both in **Figure 8** and **Figure 9** each row represents the results for networks with different sizes. In **Figure 8**, each column corresponds to different d, while each column in **Figure 9** corresponds to different λ. We can clearly see from those two figures that the values of f_1 for all the solutions are negative. This means that the robustness of the original networks have being improved based on the corresponding edge rewire mechanism. One may further see from the figures that the proposed multiobjective optimization based edge rewire mechanism can yield many different solutions to improve the robustness of a given network. Actually, this is one of the biggest advantages of the proposed edge rewire mechanism as compared to existing ones which can only provide single solutions for the decision makers. As can be seen from the two figures, each solution corresponds to edge deletions and additions but the number of added edges does not need to be equal to that of the deleted edges, which makes the proposed mechanism more flexible for real use.

The above experiments are carried out on Erdős–Rényi networks and scale-free networks which are two important types of networks. In the next, we show the results on the four real-world networks. **Figure 10** visualizes of the Pareto fronts obtained by the introduced algorithm when tested on the four networks.

It also can be seen from **Figure 10** similar phenomenon occurs to the tested real-world networks as compared to that of the computer-generated networks. More specifically, there are multiple Pareto solutions for improving the robustness of each of the tested real-world networks. The above experiments

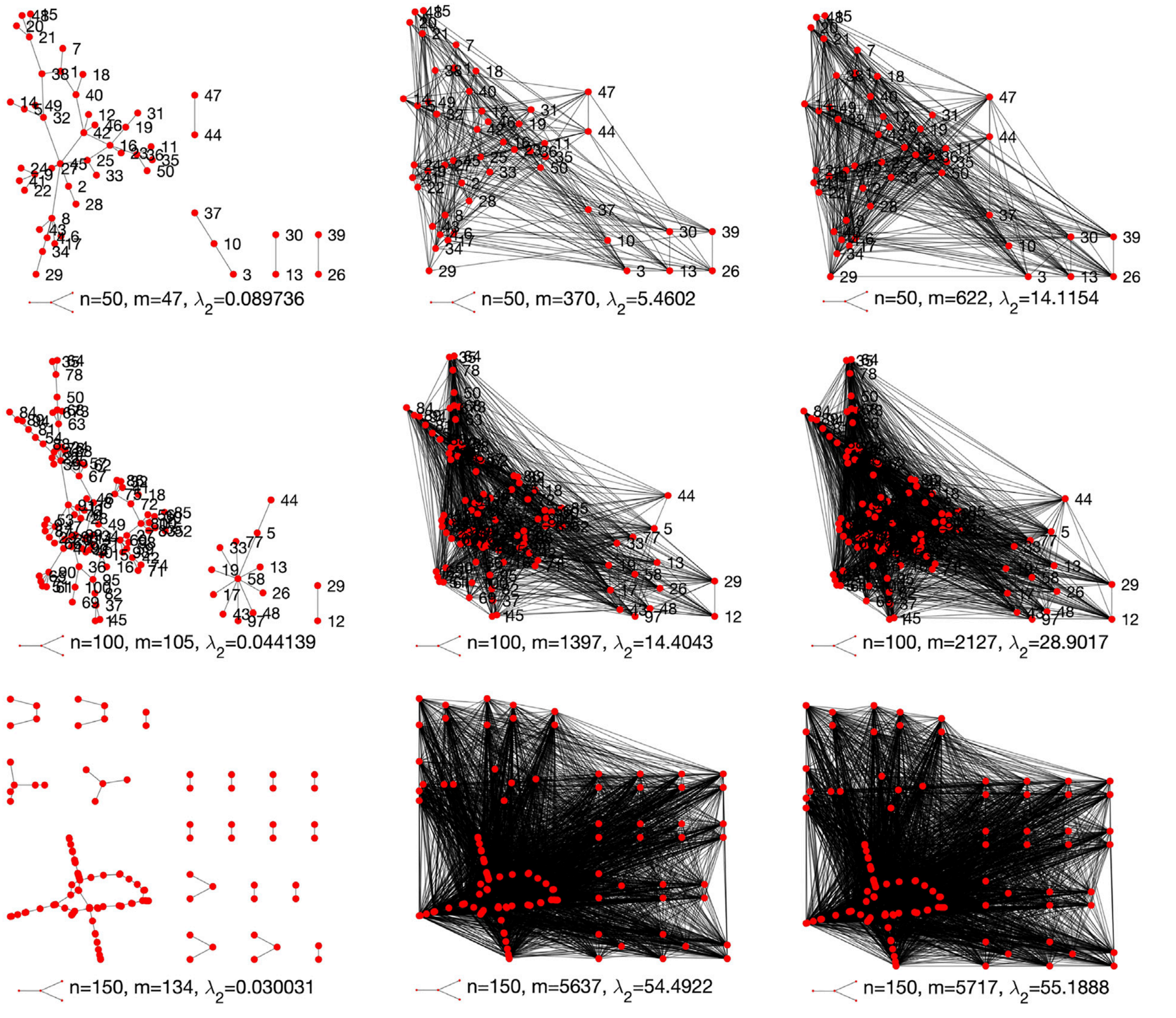

n=50, m=47, λ_2=0.089736
n=50, m=370, λ_2=5.4602
n=50, m=622, λ_2=14.1154
n=100, m=105, λ_2=0.044139
n=100, m=1397, λ_2=14.4043
n=100, m=2127, λ_2=28.9017
n=150, m=134, λ_2=0.030031
n=150, m=5637, λ_2=54.4922
n=150, m=5717, λ_2=55.1888

FIGURE 12 | Original (first column) and optimized structures (second and third columns) the scale-free networks for λ = 2.

prove that the proposed edge rewire mechanism can reconnect those tested networks to improve their network robustness. In what follows, we analyze the robustness of the enhanced networks with respect to the obtained Pareto solutions.

5.4 Enhanced Network Robustness

The above experiments indicate that the proposed multiobjective optimization based edge rewire mechanism can help to improve the robustness of a given network. Note that by optimizing the proposed mechanism, a set of rewire solutions can be obtained. In this section we analyze the robustness improvement of two extreme solutions for each tested network.

As can be seen from **Figures 8–10**, the Pareto front for each network contains a set of solutions. Here, we choose two extreme solutions from each Pareto front. For all the solutions, we sum up f_2 and f_3. Then we choose the two extreme solutions as the two solutions that require the minimum number of edge operations including addition and deletion. We then calculate the ratio of robustness improvement. The results for all the tested artificial networks are shown in **Table 2**.

In **Table 2**, the robustness improvement is calculated as $(\lambda_2^1 - \lambda_2^0)/\lambda_2^0$. In the table, we can see that for all the tested artificial networks, the robustness of the original networks have been greatly improved. We also observe from the table that when the size of a network is increased, then the robustness improvement will also be increased. This is because that a network with n nodes has $n \times (n - 1)/2$ maximum number of edges. Thus, a larger network has larger maximum number of edges and there is larger probability to improve its robustness given the fact that the most robust network is a full connected network.

From **Table 2** we also notice that the robustness improvement for scale-free networks are more significant than that of Erdős–Rényi networks. This is because that a scale-free network has relatively less edges than an Erdős–Rényi network with the same number of nodes. Consequently, the proposed edge

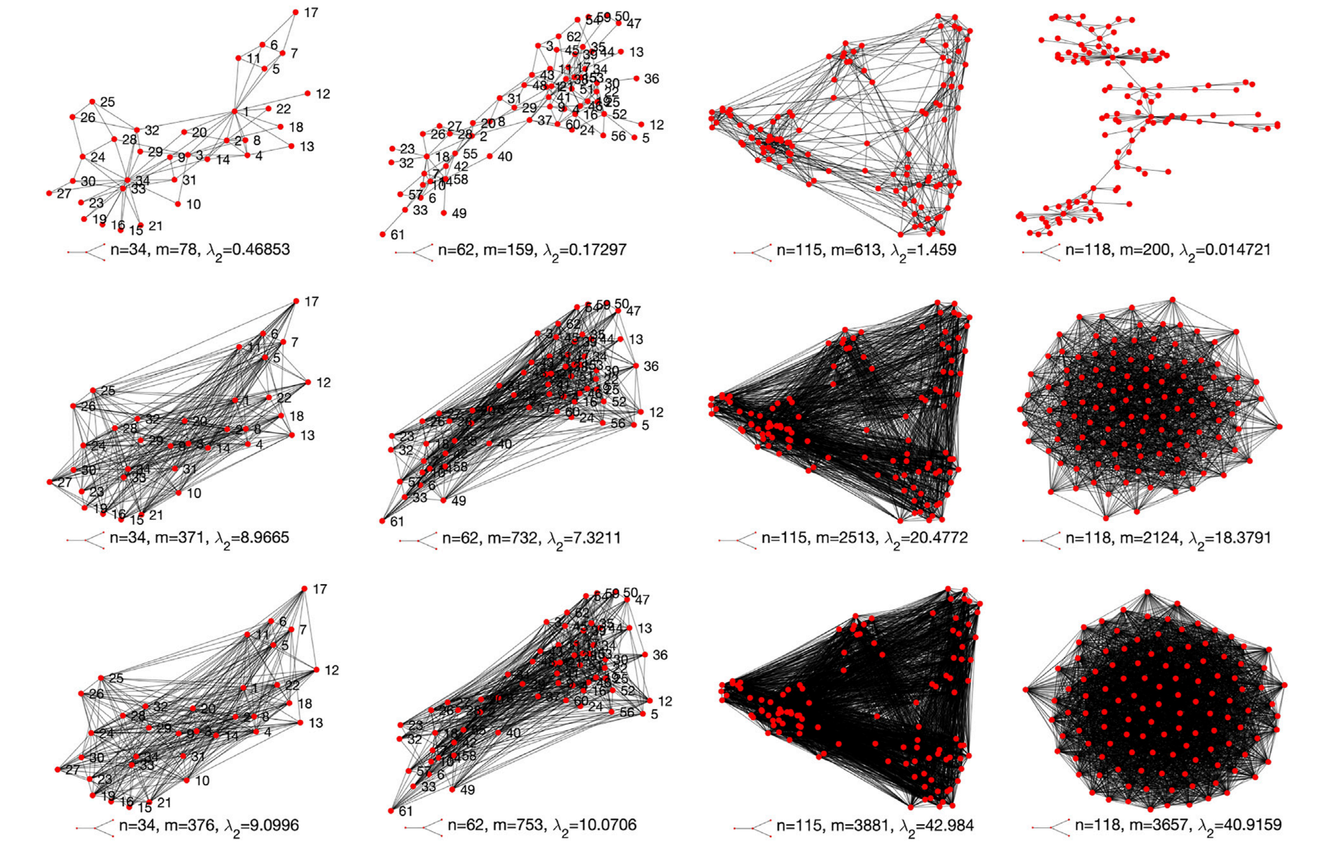

n=34, m=78, λ_2=0.46853
n=62, m=159, λ_2=0.17297
n=115, m=613, λ_2=1.459
n=118, m=200, λ_2=0.014721
n=34, m=371, λ_2=8.9665
n=62, m=732, λ_2=7.3211
n=115, m=2513, λ_2=20.4772
n=118, m=2124, λ_2=18.3791
n=34, m=376, λ_2=9.0996
n=62, m=753, λ_2=10.0706
n=115, m=3881, λ_2=42.984
n=118, m=3657, λ_2=40.9159

FIGURE 13 | Original (first row) and optimized structures (second and third rows) of the four tested real-world networks.

rewire mechanism can work better for improving the robustness of scale-free networks.

Table 3 records the robustness of the tested four real-world networks (recorded by λ_2^0) as well as the robustness of the corresponding optimized networks (recorded by $\lambda_2{}'$ and λ_2'') with respect to the selected extreme solutions. For the Karate network, its original robust is 0.4685 as can be seen from **Table 3**. By rewiring its edges with respect to the two selected solutions, the robustness of the two optimized networks are respectively 8.9665 and 9.0996, which are about 20 times of its original robustness. For the remaining three networks, their robustness also have been significantly improved based on the proposed edge rewire mechanism. The experiments on real-world networks also validate the effectiveness of the proposed rewire mechanism.

5.5 Optimized Network Structure

Tables 2, 3 show that the robustness of the tested networks have been greatly improved. In this section we graphically visualize the optimized structures of the networks against the original ones. For simplicity, here we only visualize the structures of Erdős–Rényi networks for $d = 4$ and scale-free networks for $\lambda = 2$.

Figures 11, **12** respectively visualize the original network structures (first column) and the optimized networks structures (second and third columns) for Erdős–Rényi networks for $d = 4$ and scale-free networks for $\lambda = 2$. We can see from **Figures 11**, **12** that the proposed edge rewire mechanism makes a given network have more edges to improve its robustness. As compared to Erdős–Rényi networks, the optimized scale-free networks seem to be much denser than their original networks. As explained earlier, this is because that a scale-free network has relatively less edges than a same sized Erdős–Rényi network.

Figure 13 visualizes the original network structures (first row) and the optimized networks structures (second and third rows) for the four tested real-world networks. It can be clearly seen from **Figures 11–13** that the selected two solutions suggest to add more links to each of the tested network to improve its network robustness.

5.6 Discussion

One may notice from **Figures 11–13** that the proposed edge rewire mechanism tends to make a given network be denser in order to improve its robustness. This is not necessarily the feature of the proposed model. Actually, the proposed multiobjective optimization based rewire mechanism is very flexible. Note that in the proposed model we do not add in any constraints. We do not limit the number of deleted and added edges. As long as the edge operations can improve the robustness of a given network, then the multiobjective optimization algorithm will determine the goodness of the edge operations based on Pareto dominance mechanism. If the operations are non-dominated with each other, then those operations will be saved as the possible choices.

Note that in the experiments for a tested network, its number of edges is far less than that of the maximum number of edges it can has. This is especially true for the tested scale-free networks. A scale-free network is normally sparse in its connections. Therefore there could be many choices to improve its robustness. In order to avoid making a network be denser, we provide below some possible solutions.

1) One may consider to increase the population size of the particle swarm optimization algorithm and run it for more than 20 times to possibly obtain more Pareto solutions. Then one may expect to select Pareto solutions that do not make many edge reconnections. It is also suggested that one can try to redesign the algorithm operators especially the status update principles for the particle swarm optimization algorithm.
2) One may add extra constraints. For example, one can set the portion of maximum edge operations, i.e., instead of considering all the possible edges, one can just predefine the maximum number of edges allowed in a given network. For example, for a network with n nodes and m edges, it can have a maximum of $n\ (n-1)/2$ edges. Then one may add the constrain of $f_2 + f_3 < \alpha n\ (n-1)/2$ in which $\alpha \in (0, 1)$ is a control parameter.
3) For real applications, one may also consider the cost on the edge operations. For some networks, adding edges could be more costly than deleting edges, while the situation can be the opposite for some networks. Therefore, instead of directly using objectives f_2 and f_3, one can design other cost functions. A straightforward way to achieve this goal is to punish the two objectives with different coefficients. For example, one can consider the objectives βf_2 and $(1-\beta)f_3$ with $\beta \in [0, 1]$.

6 CONCLUSION

The study on network robustness has attracted much attention in the past decade. In reality many complex networks will more or less suffer from external attacks. Those attacks to a network can lead to the failure of network nodes and edges. When some nodes and/or edges fail, the corresponding network could totally fail, which could bring about enormous losses. Network robustness estimates a network's ability to bear with attacks. To do research on network robustness can help the better design of network structures to improve their robustness.

In order to enhance the robustness of a complex network, in the literature one of the most effective ways is the network rewire mechanism which changes the edge connections between the nodes so as to improve the network's robustness. In this work we adopted spectral analysis to measure the robustness of a given network. We then proposed a multiobjective optimization based network rewire mechanism to enhance network robustness. The proposed edge rewire mechanism optimizes three objectives. The first one is the robustness improvement. The second one is to minimize edge deletions and the third is to minimize edge additions. To optimize the proposed mechanism, we further develop a multiobjective discrete partite swarm optimization algorithm to solve the proposed mechanism. Compared to traditional network rewire mechanism, the developed mechanism can generate a set of network rewire choices each of which can improve the robustness of a given network. To validate the effectiveness of the proposed mechanism, we carried out simulations on computer-generated Erdős–Rényi and scale-free networks as well as real-world networks. Experiments have validated the effectiveness of the proposed edge rewire mechanism.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: http://www-personal.umich.edu/~mejn/netdata/.

AUTHOR CONTRIBUTIONS

ZL—Conceptualization, Methodology, Writing-Original draft preparation. ZL and QL—Data generating and collection. ZL, QL, and LC—experiments and discussions, ZL, QL, and LC—Writing—Reviewing and Editing.

FUNDING

This work was supported by the Shanxi Provincial Key Research and Development Project (Grant No. 2020NY-176), the Science Project of Yulin High Tech Zone (Grant No. CXY-2020-32), and the Doctoral Research Startup Fund of Yulin University (Grant No. 16GK25).

REFERENCES

1. Cai Q, Alam S, and Duong VN. A Spatial-Temporal Network Perspective for the Propagation Dynamics of Air Traffic Delays. *Engineering* (2021) 7:452–64. doi:10.1016/j.eng.2020.05.027
2. Hoadley D, Bartolo M, Chesterman R, Faus A, Hernandez W, Kultys B, et al. A Global Community of Courts? Modelling the Use of Persuasive Authority as a Complex Network. *Front Phys* (2021) 9:331. doi:10.3389/fphy.2021.665719
3. Hope SM, Kundu S, Roy C, Manna SS, and Hansen A. Network Topology of the Desert Rose. *Front Phys* (2015) 3:72. doi:10.3389/fphy.2015.00072
4. Newman MEJ. *Networks*. Oxford University Press (2018).
5. Zang T, Gao S, Huang T, Wei X, and Wang T. Complex Network-Based Transmission Network Vulnerability Assessment Using Adjacent Graphs. *IEEE Syst J* (2019) 14:572–81. doi:10.1109/JSYST.2019.2934317
6. Wang C, Zhang F, Deng Y, Gao C, Li X, and Wang Z. An Adaptive Population Control Framework for ACO-Based Community Detection. *Chaos, Solitons & Fractals* (2020) 138:109886. doi:10.1016/j.chaos.2020.109886
7. Ma C, Cai Q, Alam S, Sridhar B, and Duong VN. Airway Network Management Using Braess's Paradox. *Transportation Res C Emerging Tech* (2019) 105:565–79. doi:10.1016/j.trc.2019.06.014
8. Li Z, and Chen L. Continuous Phase Transition of K-Partite Networks. *IEEE Syst J* (2020) 2020:2990663. doi:10.1109/JSYST.2020.2990663

9. Wang C, Deng Y, Yuan Z, Zhang C, Zhang F, Cai Q, et al. How to Optimize the Supply and Allocation of Medical Emergency Resources during Public Health Emergencies. *Front Phys* (2020) 8:1. doi:10.3389/fphy.2020.00383
10. Bianconi G. Dangerous Liaisons? *Nat Phys* (2014) 10:712–4. doi:10.1038/nphys3097
11. Gong M, Ma L, Cai Q, and Jiao L. Enhancing Robustness of Coupled Networks under Targeted Recoveries. *Sci Rep* (2015) 5:8439. doi:10.1038/srep08439
12. Li Z, and Chen L. Robustness of Multipartite Networks in Face of Random Node Failure. *Chaos, Solitons & Fractals* (2019) 121:149–59. doi:10.1016/j.chaos.2019.01.036
13. Newman MEJ, Barabási A-L, and Watts DJ. *The Structure and Dynamics of Networks*. Princeton University Press (2011).
14. Ichinose G, Tsuchiya T, and Watanabe S. Robustness of Football Passing Networks against Continuous Node and Link Removals. *Chaos, Solitons & Fractals* (2021) 147:110973. doi:10.1016/j.chaos.2021.110973
15. Feketa P, and Bajcinca N. On Robustness of Impulsive Stabilization. *Automatica* (2019) 104:48–56. doi:10.1016/j.automatica.2019.02.056
16. Shekhtman LM, Danziger MM, and Havlin S. Recent Advances on Failure and Recovery in Networks of Networks. *Chaos, Solitons & Fractals* (2016) 90: 28–36. doi:10.1016/j.chaos.2016.02.002
17. Shang Y. Localized Recovery of Complex Networks against Failure. *Sci Rep* (2016) 6:30521. doi:10.1038/srep30521
18. Chattopadhyay S, Dai H, Eun DY, and Hosseinalipour S. Designing Optimal Interlink Patterns to Maximize Robustness of Interdependent Networks against Cascading Failures. *IEEE Trans Commun* (2017) 65:3847–62. doi:10.1109/tcomm.2017.2709302
19. Kong LW, Li M, Liu RR, and Wang BH. Percolation on Networks with Weak and Heterogeneous Dependency. *Phys Rev E* (2017) 95:032301. doi:10.1103/PhysRevE.95.032301
20. Cai Q, Ma L, Gong M, and Tian D. A Survey on Network Community Detection Based on Evolutionary Computation. *Ijbic* (2016) 8:84–98. doi:10.1504/ijbic.2016.076329
21. Cai Q, and Liu J. The Robustness of Ecosystems to the Species Loss of Community. *Sci Rep* (2016) 6:35904. doi:10.1038/srep35904
22. Louzada VHP, Daolio F, Herrmann HJ, and Tomassini M. Smart Rewiring for Network Robustness. *J Complex networks* (2013) 1:150–9. doi:10.1093/comnet/cnt010
23. Dunn S, and Wilkinson SM. Increasing the Resilience of Air Traffic Networks Using a Network Graph Theory Approach. *Transportation Res E: Logistics Transportation Rev* (2016) 90:39–50. doi:10.1016/j.tre.2015.09.011
24. Wuellner DR, Roy S, and D'Souza RM. Resilience and Rewiring of the Passenger Airline Networks in the United States. *Phys Rev E Stat Nonlin Soft Matter Phys* (2010) 82:056101. doi:10.1103/PhysRevE.82.056101
25. Črepinšek M, Liu S-H, and Mernik M. Exploration and Exploitation in Evolutionary Algorithms: a Survey. *ACM Comput Surv (Csur)* (2013) 45:35. doi:10.1145/2480741.2480752
26. Trivedi A, Srinivasan D, Sanyal K, and Ghosh A. A Survey of Multiobjective Evolutionary Algorithms Based on Decomposition. *IEEE Trans Evol Comput* (2017) 21:440–62. doi:10.1109/ACCESS.2020.2973670
27. Cai Q, Gong M, Ruan S, Miao Q, and Du H. Network Structural Balance Based on Evolutionary Multiobjective Optimization: A Two-step Approach. *IEEE Trans Evol Computat* (2015) 19:903–16. doi:10.1109/tevc.2015.2424081
28. van den Bergh F, and Engelbrecht AP. A Cooperative Approach to Particle Swarm Optimization. *IEEE Trans Evol Computat* (2004) 8:225–39. doi:10.1109/tevc.2004.826069
29. Coello CAC, Pulido GT, and Lechuga MS. Handling Multiple Objectives with Particle Swarm Optimization. *IEEE Trans Evol Computat* (2004) 8:256–79. doi:10.1109/tevc.2004.826067
30. Xu X, Chen A, and Yang C. An Optimization Approach for Deriving Upper and Lower Bounds of Transportation Network Vulnerability under Simultaneous Disruptions of Multiple Links. *Transportation Res Part C: Emerging Tech* (2018) 94:338–53. doi:10.1016/j.trc.2017.08.015
31. Wu J, Barahona M, Tan Y-J, and Deng H-Z. Spectral Measure of Structural Robustness in Complex Networks. *IEEE Trans Syst Man Cybern A* (2011) 41: 1244–52. doi:10.1109/tsmca.2011.2116117
32. Mitchell M. *An Introduction to Genetic Algorithms*. MIT Press (1998).
33. Kennedy J, and Eberhart R. A Discrete Binary Version of the Particle Swarm Algorithm. In: *Proceedings of 1997 IEEE International Conference on Systems, Man, and Cybernetics*. IEEE (1997). p. 4104–8.
34. Deb K, Pratap A, Agarwal S, and Meyarivan T. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II.. *IEEE Trans Evol Computat* (2002) 6:182–97. doi:10.1109/4235.996017
35. Mohankrishna S, Maheshwari D, Satyanarayana P, and Satapathy SC. A Comprehensive Study of Particle Swarm Based Multi-Objective Optimization. In: *Proceedings of the 2012 International Conference on Information Systems Design and Intelligent Applications* (2012) p. 689–701. doi:10.1007/978-3-642-27443-5_79
36. Zhang Q, and Li H. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Trans Evol Computat* (2007) 11:712–31. doi:10.1109/tevc.2007.892759
37. Bassler KE, Genio CID, Miklós I, Toroczkai Z, and Toroczkai Z. Exact Sampling of Graphs with Prescribed Degree Correlations. *New J Phys* (2015) 17:083052. doi:10.1088/1367-2630/17/8/083052

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers in Physics

ORIGINAL RESEARCH
published: 12 August 2021
doi: 10.3389/fphy.2021.721007

A Network View of Portfolio Optimization Using Fundamental Information

Xiangzhen Yan [1], Hanchao Yang [1]*, Zhongyuan Yu [2] and Shuguang Zhang [1]

[1]Department of Statistics and Finance, School of Management, University of Science and Technology of China, Hefei, China, [2]School of Systems and Enterprises, Stevens Institute of Technology, Hoboken, NJ, United States

This article proposes the use of a novel approach to portfolio optimization, referred to as "Fundamental Networks" (FN). FN is an effective and robust network-based fundamental-incorporated method, and can be served as an alternative to classical mean-variance framework models. As a proxy for a portfolio, a fundamental network is defined as a set of "interconnected" stocks, among which linkages are a measure of similarity of fundamental information and are referred to asset allocation directly. Two empirical models are provided in this paper as applications of Fundamental Networks. We find that Fundamental Networks efficient portfolios are in general more mean-variance efficient in out-of-sample performance than Markwotiz's efficient portfolios. Specifically, portfolios set for profitability goals create excess return in a general/upward trending market; portfolios targeted for operating fitness perform better in a downward trending market, and can be considered as a defensive strategy in the event of a crisis.

Keywords: portfolio optimization, fundamental analysis, financial networks, asset allocation, minimum spanning tree

OPEN ACCESS

Edited by:
Peican Zhu,
Northwestern Polytechnical University, China

Reviewed by:
Jingrui Sun,
Southern University of Science and Technology, China
Fujun Lai,
Yunnan University of Finance And Economics, China

****Correspondence:***
Hanchao Yang
yanghanchao44@gmail.com

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *05 June 2021*
Accepted: *21 July 2021*
Published: *12 August 2021*

Citation:
Yan X, Yang H, Yu Z and Zhang S (2021) A Network View of Portfolio Optimization Using Fundamental Information. Front. Phys. 9:721007. doi: 10.3389/fphy.2021.721007

1 INTRODUCTION

The problem of portfolio optimization is one of the most important issues in asset management [1]. Modern Portfolio Theory (MPT) has served as the foundation and industry standard for portfolio optimization, being capable of demonstrating the concepts of diversification and risk-return efficiency based on a mean-variance framework [2, 3]. However, within the last few decades, practitioners and academics have become aware of its drawbacks. First, the high sensitivity of the estimated mean-variance efficient portfolio to estimation errors in expected return may lead to non-robust results [4–12]. Second, the mean-variance optimization requires the inversion of a positive-definite co-variance matrix which would lead to large estimation errors and will further offset the benefits of diversification. Hence, the portfolio cannot be constructed for a large number of stocks [13]. Third, the lack of hierarchical structure in a correlation matrix allows weights to vary freely in unintended ways, which is a root cause to the mean-variance efficient portfolio's instability [14]. Moreover, MPT only involves the analysis of prices and is not able to incorporate the fundamental analysis which has been deemed value-relevant to stock return by not only well-known investors mentioned earlier, but also multiple researchers in the past [15–19].

To our knowledge, substitution of mean-variance framework has not been well discussed in the area of portfolio optimization. Studies introduced fundamental analysis to mean-variance model are still confined by the drawbacks. The application of financial networks started a new branch in optimizing portfolios. However, few of them conveyed fundamental information. Therefore, the main purpose of our research is to bridge portfolio optimization to fundamental analysis with network structure.

In this paper, we propose a novel approach for portfolio optimization, referred to as "Fundamental Networks" (FN), an effective and robust network-based fundamental-incorporated method. It can be used as an alternative to the classical Markowitz's model. Compare to it, FN avoids the estimation errors of expected return and standard deviation,and also removes the requirement of a positive definite co-variance matrix. As a proxy for a portfolio, a fundamental network is defined as a set of "interconnected" stocks, among which linkages are a measure of similarity of fundamental information among stocks. The capital allocated to a stock is proportionate to the concept of weighted degree in network theory.

The fundamental networks bridge portfolios to fundamental information with network theory. The process of portfolio optimization is decomposed into steps as asset selection and capital allocation [20]. iterature referred to this paper is classified as fundamental analysis and financial networks. However, few of them addressed the both steps.

Fundamental analysis is the most common way to assess firm value [19]. However, the application of fundamental analysis in portfolio optimization tends to focus on the stock selection rather than developing an effective and innovative weighting schema based on fundamental information [18, 20–24]. Attempts introduced an additional fundamental condition into Markowitz's model for both optimal selection and allocation. Although these models are improved by selecting stocks of good economic condition for the portfolio, major limitations from Markowitz's model have not been addressed [25–27]. Another area of studies which is related to our work is fundamental indexation, which assigns portfolio weights using metrics in financial statements instead of stock price and capitalization [28–30].

Pioneered by the remarkable work of [31]; network-based methods have found their way into finance literature, and Recent studies such as [32–35] explore their usefulness for optimal investment purposes. Indeed, as noted in [36]; many financial optimization problems including Markowitz's, had an underlying network structure. The topology of network encodes the complex dependency structure of financial equities, extracts hierarchical and clustering properties, and reduces data complexity while preserving fundamental characteristics of information [34]. Because the functionality of complex networks relies on their underlying structure and structural stability, most empirical studies chose the minimum spanning tree (MST) as the base structure for their financial networks [37–41].

In network theory, information about the relative importance of nodes in a network can be obtained through centrality measures, which are the most fundamental and frequently used measures to reveal network structure [42–46]. Investment decisions are usually derived from choosing "central" or "non-central" assets from the network. "Non-central" investment strategy claims that a portfolio containing the outskirts of the network have greater diversification potential, and thus are exposed to less risk [32, 34, 35, 45]. However, these studies only select stocks, the allocation process is still under the Markowitz's mean-variance framework, therefore its drawbacks inherited from the framework remain unsolved.

Our proposed FN can address both asset selection and capital allocation for portfolio investment. In the selection stage, a portfolio objective is set, such as maximizing changes of return on equity (ROE). In the allocation stage, we construct a fundamental network with proper network structure. To start with, we define each selected stock as a node in the network, and each node is described using its fundamental information set which may include changes in net profit, asset turnover and leverage ratio. Next, we define potential links in the network using similarity measure of information sets among stocks, similarity measure can be Euclidean distance or other common distance measures. Finally, a network structure such as the minimum spanning tree (MST) is applied to form connections among stocks, and weighted degrees are derived for capital allocation. Network efficiency is considered to be a measure of network instability and error tolerance [47].

Two empirical models are provided in this paper as applications of the FN. Model 1 is based on DuPont framework where the objective of selecting stocks with high ROE, changes in net profit, asset turnover and leverage ratio serve as fundamental information set to describe a node. Model 2 is focused on operating income, where stock nodes are presented as changes of gross profit. Both of these models use Euclidean distance to construct a MST to be used as the desired network structure. We evaluate model performance using Sharpe Ratio from year 2006 to 2018. Model 1 outperforms the mean-variance efficient portfolios for all risk cases, and Model 2 outperforms the mean-variance efficient portfolios in the low-risk cases. In other words, FN efficient portfolios are more mean-variance efficient in out-of-sample performance than Markwotiz's efficient portfolios. In addition, the annual return of Model 1 is positively correlated to the annual return of the S&P 500 Index, while Model 2 is negatively correlated to that of the S&P 500 Index. Unlike the one fixed optimization goal of mean-variance framework, FN provide solutions to complex market conditions. In our study, Model 1 is suggested for general market condition, while Model 2 serves as a defensive strategy in the event of a crisis.

The major contributions of our proposed FN are:

- **Proposing a Novel Framework for Portfolio Optimization:** creating an network-based fundamental-incorporated method for portfolio optimization, completely different from the classical Markowitz's model. Empirically, the out-of-sample performance is suggested that network efficient portfolios reached higher mean-variance efficiency than Markowitz's solutions.
- **Expanding Source of Information:** incorporating fundamental information and providing flexibility in selecting portfolio objective. It can be any fundamental variable or financial ratio representing profitability, liquidity, solvency, operating efficiency etc.
- **Lifting Restriction:** removing portfolio size constraint in mean-variance framework, referred to the positive-definite requirement of co-variance matrix, so that enable large-scale portfolio optimization.

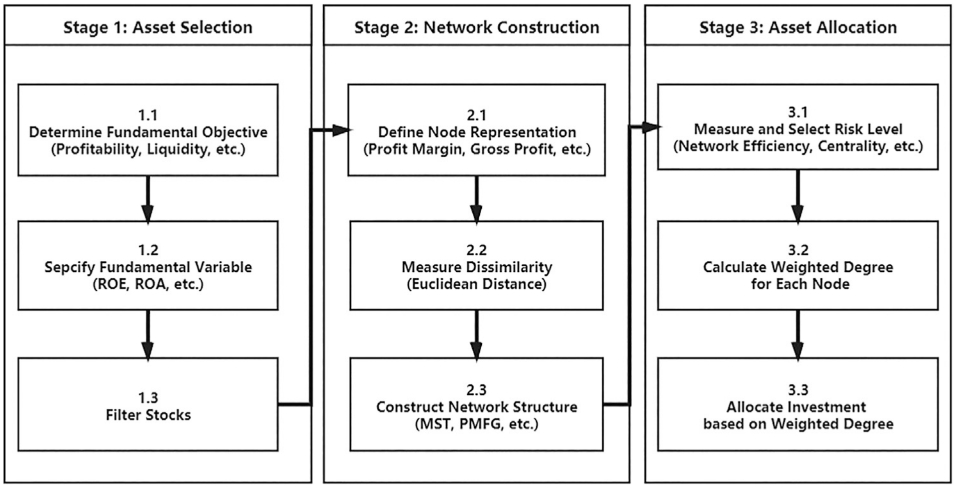

Stage 1: Asset Selection
1.1
Determine Fundamental Objective
(Profitability, Liquidity, etc.)
1.2
Sepcify Fundamental Variable
(ROE, ROA, etc.)
1.3
Filter Stocks
Stage 2: Network Construction
2.1
Define Node Representation
(Profit Margin, Gross Profit, etc.)
2.2
Measure Dissimilarity
(Euclidean Distance)
2.3
Construct Network Structure
(MST, PMFG, etc.)
Stage 3: Asset Allocation
3.1
Measure and Select Risk Level
(Network Efficiency, Centrality, etc.)
3.2
Calculate Weighted Degree
for Each Node
3.3
Allocate Investment
based on Weighted Degree

FIGURE 1 | Flowchart for Portfolio Optimization under Fundamental Network Framework. Two empirical examples are specified in **Table 2**.

- **Enhancing Robustness of Asset Allocation:** increasing stability and robustness of optimal capital allocation on stocks by adopting optimal network structure.

The remainder of the paper is organized as follows: **section 2** introduces the general mathematical model of fundamental networks. In **section 3**, two empirical studies are provided and evaluated. **Section 4** presents our conclusions and avenues of future research.

2 METHODOLOGY

2.1 Stock Portfolio as a Network

In our FN model, the asset selection rule is determined by the chosen fundamental objective. The asset allocation process is to minimize stock similarity under an optimal network structure. For example, ROE is an indicator of corporate profitability. To maximize profitability of a portfolio, we choose the changes of ROE as the fundamental objective and filter a set of high performance companies out of the entire market. We then construct a network based on fundamental information set, using the same example as in Model 1. Elements in the set include changes in net profit, asset turnover and leverage ratio among others. The allocation result is expected to be positively correlated to the weighted degree of each node in the network.

The main novelty of this paper lies in the flexibility choosing objective, information sets and risk measures based on fundamentals. Objectives and information sets can be any fundamental variable or financial ratio representing profitability, liquidity, solvency, operating efficiency etc. Common network structures can refer to, but are not limited to minimum spanning trees, planar maximally filtered graphs, market graphs [48]. Variations in node representation, network structure and distance definition allow a variety of potential risk measures to be applied. Such risk measures can be derived as centrality, assortativity, and network efficiency among others. Since, centrality, assortativity and network efficiency are measures of similarity and stability in graph theory, these can be considered as diversification and robustness of allocation in portfolio theory [33, 34].

2.2 Fundamental Network Framework

In this paper, we introduce the concept of “Fundamental Network Framework” as a general process equating portfolio optimization with finding optimal network topology features (**Figure 1**). The framework is consisted of three stages: asset selection, fundamental network construction and asset allocation.

2.3 Fundamental Network Construction

According to **Figure 1**, Stage 1 and 2 is to define and construct fundamental networks by modeling a portfolio with topological features. A fundamental network is referred to a connected, undirected and weighted graph $G(V, E, W)$.

Definition 1 **(Fundamental Network)** Define a connected, undirected and weighted graph

$$G(V,E,W) \tag{1}$$

where $V = \{v_1, \ldots, v_n\}$ *denotes a set of stocks. Every stock and its fundamental/financial ratio information is represented as a node.* $E = \{e_1, \ldots, e_n\}$ *is the edge set.* W *is the weighted adjacency matrix and* $w_{i,j} \in W$, $w_{i,j}$ *represents the edge weight between nodes* v_i *and* v_j.

In stage 1 (**Figure 1**), we need to define the fundamental objective such as profitability, and then refer a specific variable for the objective. In this paper, we only allow one dimension objective. Definition 2 is to mapping a stock to the fundamental variable. For instance, ROE is selected to describe profitability noted as the utility $U = u(V) = ROE$.

Definition 2 **(Utility)** *Define a utility as*

$$U = u(V) \tag{2}$$

where $u()$ *is a utility function.*

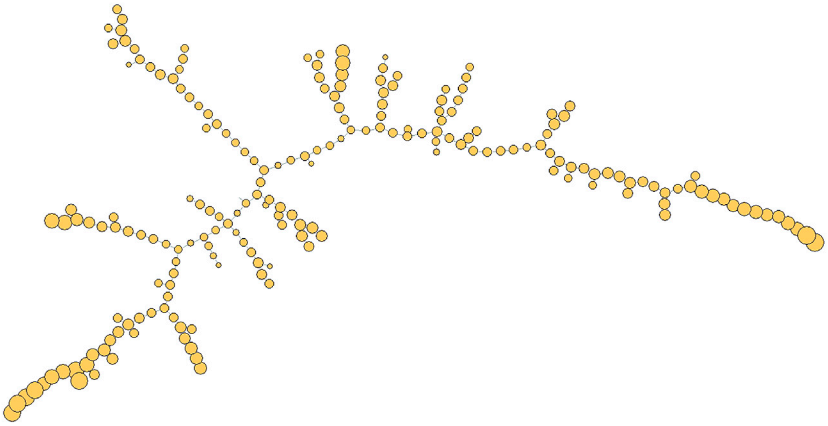


FIGURE 2 | Fundamental Network: Model 1, 200 stocks in 2009. Size of node is positively correlated to weighted degree and investment allocation.

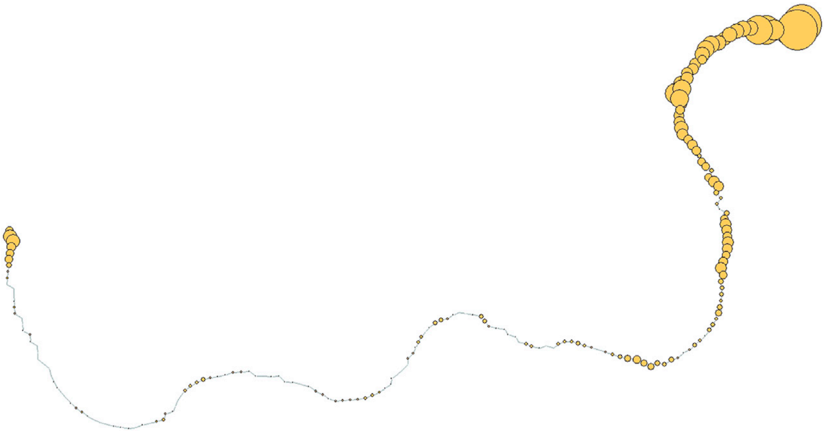


FIGURE 3 | Fundamental Network: Model 2, 200 stocks in 2009. Weighted degree is higher for the nodes (larger) near the end points than those in the middle since no hierarchical structure is detected.

Then filter stocks with the variable selected to establish a stock pool for later optimization. Notice that V_U are the selected stocks based on U and $U_{threshold}$. It can represent any predetermined objective such as profitability, liquidity, solvency, or operating efficiency.

Definition 3 **(Stock Pool)** Select a subset of stocks satisfying

$$V_U \in V, \forall v \in V_U, u(v) \geq U_{threshold} \tag{3}$$

where $U_{threshold}$ is a threshold level of U.

Stage 2 is mainly to construct fundamental networks based on selected stock pool. Specifically, each node represents the fundamental(s) of a stock. A linkage between two nodes represents the dissimilarity of the corresponding stocks. In this paper, dissimilarity measure is referred to Euclidean distance.

Definition 4 **(Fundamental Node)** *Define $V = \{v_1, \ldots, v_n\}$, a node is*

$$v_i = \{I_{(i,q)} | q = 1, 2, \ldots, m\} \tag{4}$$

where v_i is the fundamental information set of stock$_i$ and $I_{(i,q)}$ is referred to qth fundamental variable representing a stock. $I_{(i,q)}$ ranges from financial ratios to the variables in the financial statements such as leverage ratio, net income,Δgrossprofit, etc. Note that m is noted as the number of variables to define a node.

Definition 5 **(Dissimilarity)** Define Euclidean distance as the measure of dissimilarity between two nodes i, j, $D = \{d(i,j), \forall i, j \in [1, n]\}$.

$$d(i,j) = \sqrt{I_{(i,q)} - I_{(j,q)}}, q = 1, 2, \ldots, m; \tag{5}$$

A network with the set V_U can be constructed by Definition 1–3. Given node set V_U, network structure E is derived from network generation algorithms such as small-world networks, scale-free networks and minimum spanning tree etc. The

TABLE 1 | Performance Comparison.

	$\hat{R}$			$\hat{\sigma}$			$\widehat{Sharpe}_t$		
Year	**Model 1**	**Model 2**	**MV**	**Model 1**	**Model 2**	**MV**	**Model 1**	**Model 2**	**MV**
2007	−0.10	0.09	−0.10	0.25	0.22	0.22	−0.40	0.40	−0.48
2008	−0.81	−0.28	−0.41	0.55	0.37	0.43	−1.48	−0.75	0.94
2009	0.94	0.34	0.41	0.43	0.23	0.18	2.20	1.45	2.24
2010	0.13	0.28	0.26	0.24	0.22	0.31	0.55	1.30	0.82
2011	0.21	0.06	0.16	0.38	0.23	0.28	0.55	0.27	0.58
2012	0.30	0.10	0.15	0.22	0.15	0.23	1.34	0.65	0.68
2013	0.54	0.20	0.14	0.19	0.13	0.15	2.90	1.58	0.93
2014	0.22	0.07	0.07	0.17	0.12	0.23	1.31	0.59	0.31
2015	−0.01	0.09	−0.08	0.21	0.21	0.23	−0.05	0.43	−0.33
2016	1.12	0.00	0.14	0.43	0.15	0.16	2.60	−0.02	0.88
2017	0.27	0.17	0.09	0.15	0.17	0.20	1.83	0.99	0.44
2018	0.05	0.02	−0.41	0.21	0.24	0.32	0.25	0.09	−1.26
Average	0.24	0.10	0.04	0.29	0.20	0.25	0.97	**0.58**	**0.32**

On average, both Model 1 and 2 are more mean-variance efficient than the benchmark. Model 1 is featured with high return and Model 2 has the lowest volatility.

TABLE 2 | Model Specifications.

FN framework	**Steps**	**Model 1**	**Model 2**
Stage 1	Fundamental Objective	Profitability	Operating Fitness
	Fundamental Variable	Δ ROE	Δ Operating Income
Stage 2	Node Representation	Δ Profit Margin Δ Asset Turnover Δ Leverage Ratio	Δ Gross Profit
	Dissimilarity Measure	Euclidean	Euclidean
	Network Structure	MST	MST
Stage 3	Risk Measure	TNE	TNE
	Allocation	Weighted Degree	Weighted Degree

*In accordance with **Figure 1**, this table specifies fundamental information and network related variables for different optimal objectives. Δ is the difference of the variable between year t and year t-1. The FN framework allows nodes to be defined in multi-dimensional space such that Model 1 is in $I R^3$ and Model 2 is in $I R$.*

topology features of $G(V_U, E, W)$ vary when choosing different network models. It is found a more complex topology often comes with a greater systemic risk [37, 49, 50] and a structural optimized network structure will also lead to an optimized portfolio [35, 51].

Fundamental network structural stability and dissimilarity in graph theory is defined in our model as portfolio risk in portfolio theory. Network efficiency is considered to be a risk tolerance measure in graph theory [47]. By bridging stock portfolios to networks, we are able to use network efficiency as a risk measure of a portfolio.

In this paper, we use minimum spanning tree (MST) as the base network structure. A minimum spanning tree (MST) is a tree $T(V_U, E_T, W_T) \in G(V, E, W)$ with minimum total edge weight.

Definition 6 **(Minimum Spanning Tree)** Given a connected, undirected weighted graph $G(V_U, E, W)$, the minimum spanning tree (MST) is a tree $T(V_U, E_{MST}, W_{MST}) \subseteq G$ with minimum total edge weight defined as:

$$w_{MST} = \sum_{e_{i,j}\in E_{MST}} w_{i,j}, \forall \mathcal{G} \subseteq G \qquad (6)$$

where $\mathcal{G}$ is subgraph of G and $E_{MST} \subseteq E$, $W_{MST} \subseteq W$.

2.4 Risk Measurement and Asset Allocation

In stage 3 (**Figure 1**), after the network is constructed, we introduce total network efficiency (TNE), which is a measure of systemic attack and error tolerance, as the risk measure in our model. A network with higher network efficiency has a higher tolerance to errors and attacks [47].

Definition 7 Total Network Efficiency (TNE)

$$TNE(G) = \sum_{i \neq j \in G} \frac{1}{d_{i,j}} \qquad (7)$$

where $d_{i,j}$ is the distance of all paths from node v_i to $v_j \in G(V_U, E, W)$, $\forall i \neq j$. Note that $G(V_U, E, W)$ is a undirected graph, so $d_{i,j} = d_{j,i}$.

TNE is positively related to the number of nodes and edges since it is defined as the sum of reciprocal of all edges in a network. Therefore, increasing size of an FN will increase its TNE, indicating a reduction of instability and an increase in diversification.

In addition, given a certain number of nodes, a minimum spanning tree has the maximum of TNE $\forall \mathcal{G} \in G$. Mathematically, according to Definition 7, a minimum spanning tree T minimizes $\sum_{i,j} d_{i,j}$, $\forall i, j \in G$ such that $TNE_T \geq TNE_{\mathcal{G}}, \forall \mathcal{G} \in G$, where $\mathcal{G}$ is arbitrary subgraph of G. Given a total number of n stocks where

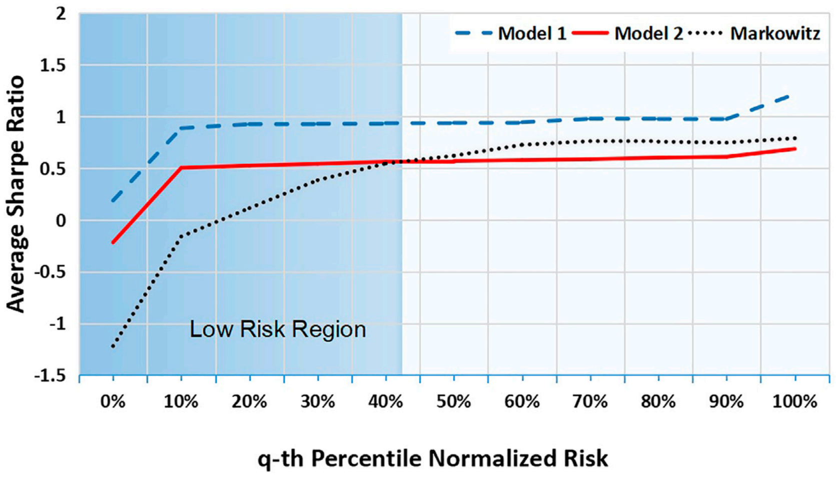


FIGURE 4 | Fundamental Network Efficient Frontiers Model Out-of-Sample Performance. Model 1 (blue) is generally more mean-variance efficient than the benchmark while Model 2 (red) only outperforms in the low risk region ($q \leq 45\%$). Mean-variance efficient frontier is the benchmark (black).

TABLE 3 | Regressions Analysis and Diagnostics.

	β Coefficient	α	R^2	p-value
Model 1	0.7812	0.0622	0.2205	3.172E-15
Model 2	-0.5435	0.044	0.1102	6.73E-08

A positive β indicates that the premium between Model 1 and Markowitz's is positively correlated with market return while Model 2 is opposite with a negative β.

$V_{U_i} = \{v_1, \ldots, v_n\}$, $u(v_i) \geq U_{threshold}$, the minimum spanning tree set $T = \{T_i(V_{U_i}, E_i, W)\}$, $i = 1, \ldots, n$ are the portfolios with maximized.

Finally, investment allocation is calculated from the weighted degree of the nodes in the network. From a network perspective, a stock with high weighted degree is systemically important as it is assigned a large portion of investment.

Definition 8 **(Investment Allocation)** *Given a connected weighted graph.*

> *$G(V_U, E, W)$, $V_U = \{v_1, \ldots, v_n\}$ is a n-stocks portfolio satisfying $\forall v_i \in V_U$, $U(v_i) \geq U_{threshold}$, where v_i represents the fundamental information set of a stock I_i.*
> *Define $X = (x_1, x_2, \ldots, x_n)$ as the vector of investment allocation, x_i as the proportion of investment in stock i, and $\sum_{i=1}^{n} x_i = 1$.*

$$x_i = \frac{\sum_{j \in N(v_i)} w_{i,j}}{\sum_{i,j \in V} w_{i,j}}, \tag{8}$$

where $N(v_i)$ is the neighbor set of v_i, $w_{i,j}$ represents the edge weight between two stocks' fundamentals v_i and v_j.

3 EMPIRICAL ANALYSIS AND RESULTS

In this section, we present two empirical solutions in portfolio optimization as applications of Fundamental Networks. While minimizing fundamental network structural instability, the models are distinct from their fundamental objectives and node variables to adapt to different market scenarios. Our optimized portfolios outperforms Markwotiz's under certain market conditions.

3.1 Data and Models

Financial statement data are from the Compustat database,the sampled stocks are S&P 1,500 index members from 2006–2018. There are around 550 to 580 stocks each year after excluding those missing necessary data. The backtest started from 04/15/2007 and the positions are adjusted at the first trading day right after each April 15th when all companies had released their financial statements. Risk free rate refers to 1-year t-bill rate. Price returns are calculated annually between each April 15th. Fundamental variables are selected from annual financial statements. A detailed description of fundamental variable settings is shown in **Table 2**. All fundamental values can be found in the supplementary data set.

To illustrate the flexibility of FN framework in selecting portfolio objectives and fundamental information, we introduce two models in the empirical analysis **Table 2**. Model 1 maximizes company profitability described by changes of Return on Common Equity (ROE). In accordance with DuPont Analysis, each node is defined by changes in net profit, asset turnover and leverage ratio. Model 2 is focused on operating fitness, fundamental objective is defined as changes of operating income, and stock nodes are presented as changes of gross profit.

In the FN models, Euclidean distance is referred to the measure of diversification. Therefore, instead of restricted by the positive-definite requirement of co-variance matrix in mean-variance framework, portfolio size in FN models can be as large as the stock Universe. In our examples, the portfolio size ranges up to around 550 stocks.

The FN networks displayed in **Figures 2**, **3** are portfolios selected from efficient frontiers of Model 1 and 2 in 2009, each consisted of 200 stocks. Each node represents a stock while linkages measure similarity of their fundamental information. Size of node is positively correlated to weighted degree and investment allocation.

Apparently, the structural complexity of the two networks varies significantly that Model 1 has a hierarchical structure and higher interconnectedness than Model 2. This outcome is related to the complexity-stability debate [52] that whether a positive correlation exists between them. Interconnectedness is a key feature in measuring network complexity. In finance, many studies suggested that interconnectedness conveys higher systemic risk in bank networks [53]. According to [54]; there is no economic theory at hand that can be used to answer the question of the optimal level of interconnectedness for a financial system from a general equilibrium perspective. However, our empirical results suggest that the portfolios with high interconnectedness have higher volatility in out-of-sample performance (**Table 1**).

3.2 Empirical Results

The assessment of the performance is based on out-of-sample statistics in the holding period which is 252 trading days. In stead of individual portfolios, this paper is to compare all the portfolios on the efficient frontiers. Therefore, we investigate the performance from both risk and time perspectives.

Specifically, we note Sharpe Ratio as $Sharpe(q, t)$, where t is the time and q represents qth percentile of the risk distribution, $q = \{0, 5\%, 10\%, \ldots, 1\}$. To compare models with different risk measures, we normalize risk into [0, 1]. Then Sharpe ratio can be represented at intervals of 5% from 0 to 1, for a total number of 21 risk levels.

$$Sharpe_q = \frac{1}{n} \sum_{t=1,2,\ldots,n} Sharpe(q, t) \tag{9}$$

where $Sharpe_q$ is the average Sharpe ratio of qth percentile portfolios over time t.

For example, efficient frontiers are constructed for each year. $Sharpe(5\%, 2008)$ is referred to Sharpe ratio of the portfolio on the frontier with the lowest 5% risk at the year 2008. $Sharpe_{5\%}$ is the average of all the portfolios with lowest 5% risk from the years between 2007–2019 **Figure 4**.

In addition, we introduce $\hat{Sharpe}_t$ as the average of Sharpe ratios for all portfolios on the efficient frontier at time t. Similarly, $\hat{R}$ and $\hat{\sigma}$ represent average return and volatility of the efficient frontier at time t.

In **Table 1**, the Model 1 significantly outperforms Markwotiz's in the year of 2012–2014, 2016, and 2017. These years are often considered to be "good" years where the market earnings growth are positive. The model 2 maximizes operating income and outperforms Markowitz's in "bad" years (2007, 2008, 2015, 2018). The result is consistent with [55] who suggested that high operating performance companies have low volatility and outperforms high volatility stocks.

Define $R = R(q, t), \forall q, t$ to be the actual return of the efficient frontier for all time t; $R_0 = R_0(q, t), \forall q, t$ is the actual return of mean-variance efficient frontier for all time t; R_{Market} is the annual return of the stock market which is represented by the annual return of the S&P 500 Index in this paper.

Let the premium between fundamental network frontier and mean-variance frontier to be:

$$R(q, t) - R_0(q, t) = \beta R_{Market}(q, t) + \alpha, \forall q, t \tag{10}$$

As a result, Model 1 creates excess return when market grows but may suffer loss when market drops. Meanwhile, Model 2 has a negative β which means it outperforms Markowitz's when market goes down. Model 1 refers to an aggressive strategy by optimizing company profitability. Model 2 is a defensive strategy with emphasis on operating income (**Table 3**).

4 CONCLUSION AND DISCUSSION

We propose a novel approach, named Fundamental Networks (FN), for asset allocation under a network framework, which allows network stability measures to become applicable for portfolio optimization. FN avoids confrontation of estimation errors and the positive definite requirement of the covariance matrix in mean-variance framework. Our FN model provides robust and well-diversified solutions for investment.

Two examples with different fundamental objectives and variables are demonstrated in the paper. We conclude that network efficient portfolios, when properly defined with fundamental variables, are also mean-variance efficient in out-of-sample performance under certain market conditions. Moreover, our approach is adaptive to different market conditions. Portfolios set for profitability goals create excess return in an upward trending market, and outperform Markowitz's benchmark when averaged across years. On the other hand, portfolios targeted for operating fitness produce better returns in the down trending market, and can be used as a defensive strategy in times of crisis. This empirical result also suggests that fundamental networks with lower interconnectedness are less volatile than those with higher complexity.

In conclusion, FN takes into consideration all filtered stocks and their interconnections, including some stocks which are insignificant but indispensable for the diversification of portfolios. The fundamentals-integrated network reveals both interesting known structures (similarity among stocks) and other structural patterns that are typically lost in the mean-variance framework. When combined with fundamentals and financial ratio information, such patterns can be developed as strategies under different market conditions. This coherent and principled network approach should prove useful for various forms of portfolio construction not limited to fundamental information. More generally, the proposed model is expected to be applied to other complex systems, because of its ability to generalize and connect networks from arbitrary data sets.

There are several avenues for future work. Firstly, a theoretical interpretation of fundamental networks is desired. Secondly, we can investigate, compare, discover patterns for optimal solutions with various fundamental objectives and variables. Thirdly, we can test and analyze more network structures (e.g., planar maximally filtered graph, market graph, maximum cliques among others), and network topological features (e.g., density, assortativity and community structure and others). Finally, fundamental networks open the door for multi-layer stock networks modeling fundamental-price dynamics, a potential interpretation for market efficiency.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

XY and HY contributed to conception and design of the study. XY organized the database. HY performed the statistical analysis. XY, HY, ZY and SZ wrote sections of the manuscript. XY, HY, and ZY wrote sections of the manuscript. All authors contributed to manuscript revision, read, and approved the submitted version.

REFERENCES

1. Elton EJ, Gruber MJ, Brown SJ, and Goetzmann WN. *Modern Portfolio Theory and Investment Analysis*. New York: John Wiley & Sons (2009).
2. Markowitz H. Portfolio Selection. *J Finance* (1952) 7:77. doi:10.2307/2975974
3. Markowitz HM. Portfolio Selection: E cient Diversification of Investments. *Cowles Foundation Monograph* 16 (1959). Diversification, 1959.
4. Mainik G, Mitov G, and Rüschendorf L. Portfolio Optimization for Heavy-Tailed Assets: Extreme Risk index vs. Markowitz. *J Empirical Finance* (2015) 32:115–34. doi:10.1016/j.jempfin.2015.03.003
5. Kolm PN, Tütüncü R, and Fabozzi FJ. 60 Years of Portfolio Optimization: Practical Challenges and Current Trends. *Eur J Oper Res* (2014) 234:356–71. doi:10.1016/j.ejor.2013.10.060
6. Chopra VK, and Ziemba WT. The Effect of Errors in Means, Variances, and Covariances on Optimal Portfolio Choice. In: *Handbook of the Fundamentals of Financial Decision Making: Part I (World Scientific)* (2013). p. 365–73. (Singapore: World Scientific). doi:10.1142/9789814417358_0021
7. Jagannathan R, and Ma T. Risk Reduction in Large Portfolios: Why Imposing the Wrong Constraints Helps. *J Finance* (2003) 58:1651–83. doi:10.1111/1540-6261.00580
8. Jorion P. Bayesian and Capm Estimators of the Means: Implications for Portfolio Selection. *J Banking Finance* (1991) 15:717–27. doi:10.1016/0378-4266(91)90094-3
9. Michaud RO. The Markowitz Optimization Enigma: Is 'Optimized' Optimal? *Financial Analysts J* (1989) 45:31–42. doi:10.2469/faj.v45.n1.31
10. Jorion P. Bayes-stein Estimation for Portfolio Analysis. *J Financial Quant Anal* (1986) 21:279–92. doi:10.2307/2331042
11. Jorion P. International Portfolio Diversification with Estimation Risk. *J Bus* (1985) 58:259–78. doi:10.1086/296296
12. Barry CB. Portfolio Analysis under Uncertain Means, Variances, and Covariances. *J Finance* (1974) 29:515–22. doi:10.1111/j.1540-6261.1974.tb03064.x
13. López de Prado M. Building Diversified Portfolios that Outperform Out of Sample. *Jpm* (2016) 42:59–69. doi:10.3905/jpm.2016.42.4.059
14. Raffinot T. Hierarchical Clustering-Based Asset Allocation. *Jpm* (2017) 44: 89–99. doi:10.3905/jpm.2018.44.2.089
15. Nissim D, and Penman SH. Ratio Analysis and Equity Valuation: From Research to Practice. *Rev Account Stud* (2001) 6:109–54. doi:10.1023/a:1011338221623
16. Sloan RG. Do stock Prices Fully Reflect Information in Accruals and Cash Flows about Future Earnings? *Account Rev* (1996) 289–315.
17. Lev B, and Thiagarajan SR. Fundamental Information Analysis. *J Account Res* (1993) 31:190–215. doi:10.2307/2491270
18. Ou JA, and Penman SH. Financial Statement Analysis and the Prediction of Stock Returns. *J Account Econ* (1989) 11:295–329. doi:10.1016/0165-4101(89)90017-7
19. Ball R, and Brown P. An Empirical Evaluation of Accounting Income Numbers. *J Account Res* (1968) 6:159–78. doi:10.2307/2490232
20. Edirisinghe NCP, and Zhang X. Generalized Dea Model of Fundamental Analysis and its Application to Portfolio Optimization. *J Banking Finance* (2007) 31:3311–35. doi:10.1016/j.jbankfin.2007.04.008
21. Ho CTB. Performance Measurement Using Data Envelopment Analysis and Financial Statement Analysis. *Ijor* (2007) 2:26–38. doi:10.1504/ijor.2007.011441
22. Küçükşahin H, and Coşkun E. The Performance of Fundamental Indexes: An Application on Istanbul. *Ege Acade Rev* (2020) 20:1-18. doi:10.21121/eab.595407
23. Xidonas P, Mavrotas G, and Psarras J. A Multicriteria Methodology for Equity Selection Using Financial Analysis. *Comput Operations Res* (2009) 36: 3187–203. doi:10.1016/j.cor.2009.02.009
24. Yu Z, Serban N, and Rouse WB. The Demographics of Change: Enterprise Characteristics and Behaviors that Influence Transformation. *J Enterprise Transformation* (2013) 3:285–306. doi:10.1080/19488289.2013.860346
25. Tarczyński W. Different Variants of Fundamental Portfolio. *Folia Oeconomica Stetinensia* (2014) 14:47–62.
26. Lyle MR, and Yohn TL. Fundamental Analysis and Mean-Variance Optimal Portfolios. *Forthcoming*, The Accounting Review (2019). Lakewood Ranch: American Accounting Association.
27. Zhang H, and Yan C. Modelling Fundamental Analysis in Portfolio Selection. *Quantitative Finance* (2018) 18:1315–26. doi:10.1080/14697688.2017.1418520
28. Blitz D, and Swinkels L. Fundamental Indexation: An Active Value Strategy in Disguise. *J Asset Manag* (2008) 9:264–9. doi:10.1057/jam.2008.23
29. Arnott RD, Hsu J, and Moore P. Fundamental Indexation. *Financial Analysts J* (2005) 61:83–99. doi:10.2469/faj.v61.n2.2718
30. Hsu JC. Cap-weighted Portfolios Are Sub-optimal Portfolios. *J investment Manage* (2004) 4.
31. Mantegna RN. Hierarchical Structure in Financial Markets. *Eur Phys J B* (1999) 11:193–7. doi:10.1007/s100510050929
32. Peralta G, and Zareei A. A Network Approach to Portfolio Selection. *J empirical Finance* (2016) 38:157–80. doi:10.1016/j.jempfin.2016.06.003
33. Kaya H. Eccentricity in Asset Management. *Jntf* (2015) 1:1–32. doi:10.21314/jntf.2015.003
34. Pozzi F, Di Matteo T, and Aste T. Spread of Risk across Financial Markets: Better to Invest in the Peripheries. *Sci Rep* (2013) 3:1665. doi:10.1038/srep01665
35. Onnela JP, Chakraborti A, Kaski K, Kertész J, and Kanto A. Dynamics of Market Correlations: Taxonomy and Portfolio Analysis. *Phys Rev E Stat Nonlin Soft Matter Phys* (2003) 68:056110. doi:10.1103/PhysRevE.68.056110
36. Nagurney A. Networks in Economics and Finance in Networks and beyond: A Half century Retrospective. *Networks* (2019) 77 (1), 50–65. (New York: Wiley Online Library).
37. Khashanah K, and Yang H. Evolutionary Systemic Risk: Fisher Information Flow Metric in Financial Network Dynamics. *Physica A: Stat Mech its Appl* (2016) 445:318–27. doi:10.1016/j.physa.2015.10.012
38. Heiberger RH. Stock Network Stability in Times of Crisis. *Physica A: Stat Mech its Appl* (2014) 393:376–81. doi:10.1016/j.physa.2013.08.053
39. Eom C, Kwon O, Jung W-S, and Kim S. The Effect of a Market Factor on Information Flow between Stocks Using the Minimal Spanning Tree. *Physica A: Stat Mech its Appl* (2010) 389:1643–52. doi:10.1016/j.physa.2009.12.044
40. Coelho R, Gilmore CG, Lucey B, Richmond P, and Hutzler S. The Evolution of Interdependence in World Equity Markets-Evidence from Minimum Spanning Trees. *Physica A: Stat Mech its Appl* (2007) 376:455–66. doi:10.1016/j.physa.2006.10.045
41. Situngkir H, and Surya Y. *On Stock Market Dynamics Through Ultrametricity Of Minimum Spanning Tree*. Tech. Rep. Bandung: Bandung Fe Institute (2005).
42. Zhu S, Kou M, Lai F, Feng Q, and Du G. The Connectedness of the Coronavirus Disease Pandemic in the World: A Study Based on Complex

Network Analysis. *Front Phys* (2021) 8:642. doi:10.3389/fphy.2020.602075

43. Gao C, Su Z, Liu J, and Kurths J. Even central Users Do Not Always Drive Information Diffusion. *Commun ACM* (2019) 62:61–7. doi:10.1145/3224203
44. Newman MEJ. Mathematics of Networks. *New palgrave encyclopedia Econ* (2008) 2:1–8. doi:10.1057/978-1-349-95121-5_2565-1
45. Chi KT, Liu J, and Lau FC. A Network Perspective of the Stock Market. *J Empirical Finance* (2010) 17:659–67.
46. Hu J, Chengyi X, Huijia L, Peican Z, and Wenjun X. Properties and structural analyses of USA's regional electricity market: A visibility graph network approach. *Applied Mathematics and Computation* (2010) (Elsevier) 385, 125434.
47. Latora V, and Marchiori M. Efficient Behavior of Small-World Networks. *Phys Rev Lett* (2001) 87:198701. doi:10.1103/physrevlett.87.198701
48. Kalyagin VA, Koldanov AP, Koldanov PA, Pardalos PM, and Zamaraev VA. Measures of Uncertainty in Market Network Analysis. *Physica A: Stat Mech its Appl* (2014) 413:59–70. doi:10.1016/j.physa.2014.06.054
49. Yalamova R, and McKelvey B. Explaining what Leads up to Stock Market Crashes: A Phase Transition Model and Scalability Dynamics. *J Behav Finance* (2011) 12:169–82. doi:10.1080/15427560.2011.602484
50. He J, and Deem MW. Structure and Response in the World Trade Network. *Phys Rev Lett* (2010) 105:198701. doi:10.1103/physrevlett.105.198701
51. Shi Y, Zheng Y, Guo K, Jin Z, and Huang Z. The Evolution Characteristics of Systemic Risk in China's Stock Market Based on a Dynamic Complex Network. *Entropy* (2020) 22:614. doi:10.3390/e22060614
52. McCann KS. The Diversity-Stability Debate. *Nature* (2000) 405:228–33. doi:10.1038/35012234
53. Battiston S, and Martinez-Jaramillo S. *Financial Networks and Stress Testing: Challenges and New Research Avenues for Systemic Risk Analysis and Financial Stability Implications* (2018). Amsterdam: Elsevier.
54. Martinez-Jaramillo S, Carmona CU, and Kenett DY. Interconnectedness and Financial Stability. *J Risk Manage Financial Institutions* (2019) 12 (2), 168–83. Henry Stewart Publications.
55. Dutt T, and Humphery-Jenner M. Stock Return Volatility, Operating Performance and Stock Returns: International Evidence on Drivers of the 'low Volatility' Anomaly. *J Banking Finance* (2013) 37:999–1017. doi:10.1016/j.jbankfin.2012.11.001

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers in Physics

ORIGINAL RESEARCH
published: 07 September 2021
doi: 10.3389/fphy.2021.739259

An Ownership Verification Mechanism Against Encrypted Forwarding Attacks in Data-Driven Social Computing

Zhe Sun[1], *Junping Wan*[1], *Bin Wang*[2]*, *Zhiqiang Cao*[1], *Ran Li*[1] *and Yuanyuan He*[3]

[1]Cyberspace Institute of Advanced Technology, Guangzhou University, Guangzhou, China, [2]College of Electrical Engineering, Zhejiang University, Hangzhou, China, [3]School of Cyber Science and Engineering, Huazhong University of Science and Technology, Wuhan, China

Data-driven deep learning has accelerated the spread of social computing applications. To develop a reliable social application, service providers need massive data on human behavior and interactions. As the data is highly relevant to users' privacy, researchers have conducted extensive research on how to securely build a collaborative training model. Cryptography methods are an essential component of collaborative training which is used to protect privacy information in gradients. However, the encrypted gradient is semantically invisible, so it is difficult to detect malicious participants forwarding other's gradient to profit unfairly. In this paper, we propose a data ownership verification mechanism based on Σ-protocol and Pedersen commitment, which can help prevent gradient stealing behavior. We deploy the Paillier algorithm on the encoded gradient to protect privacy information in collaborative training. In addition, we design a united commitment scheme to complete the verification process of commitments in batches, and reduce verification consumption for aggregators in large-scale social computing. The evaluation of the experiments demonstrates the effectiveness and efficiency of our proposed mechanism.

Keywords: ownership verification, cryptography-based privacy-preserving, pedersen commitment, Σ-protocol, social computing

Edited by:
Peican Zhu,
Northwestern Polytechnical University, China

Reviewed by:
Wei Du,
University of Arkansas, United States
Cheng Huang,
University of Waterloo, Canada

****Correspondence:***
Bin Wang
bin_wang@zju.edu.cn

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *10 July 2021*
Accepted: *05 August 2021*
Published: *07 September 2021*

Citation:
Sun Z, Wan J, Wang B, Cao Z, Li R and He Y (2021) An Ownership Verification Mechanism Against Encrypted Forwarding Attacks in Data-Driven Social Computing. Front. Phys. 9:739259. doi: 10.3389/fphy.2021.739259

INTRODUCTION

Social computing, a field highly relevant to human behavior, has made considerable progress in recent years. With the proliferation of cell phones and IoT devices, information about humans, behaviors, and interactions is being recorded in unprecedented detail. Combined with deep learning models, the application of social computing can profoundly solve various industry challenges such as epidemic prediction [1, 2], social network service [3], and hot topic recommendations [4]. For example, data-driven epidemic propagation can help governments and hospitals prepare resources in advance [1]. Moreover, some well-designed pre-diagnosis models of COVID-19 may alleviate physician shortages [2].

As a data-driven technology, deep learning has a natural desire for data to be more accurate and higher-quality. Since user data in social computing is closely tied to privacy, providing the data directly would cause extremely serious privacy damage. Google [5] proposed the concept of federated learning in 2016, which is an innovation in collaborative training. Without directly obtaining local data from multiple parties, it can take full advantage of the value of the data by sharing gradients.

However, transmitted gradients are subject to model inversion attacks, attribute inference attacks, membership inference attacks, etc. Thus, the privacy of users requires further protection.

To resolve the problem of privacy leakage in transmitted gradients, researchers have proposed a wide range of solutions that can be roughly divided into two categories: differential privacy [6] and cryptographic algorithms [7]. Differential privacy is a method of adding noise so that the attacker cannot determine the user's exact gradient. However, the differential privacy-based method modifies the original data, resulting in a loss of model accuracy. Unlike differential privacy methods, cryptographic algorithms such as homomorphic encryption do not alter the value of the gradient. But instead, they prevent third parties and model aggregators from obtaining a single participant's model parameters by encryption or concealment. Most cryptography-based parameter protection methods have a higher arithmetic overhead and many researchers are committed to optimizing their overhead.

Apart from performance issues, cryptography-based privacy-preserving methods will also introduce a derivative problem. Because of the semantic invisibility of encrypted data, data aggregators cannot judge whether a problem exists. The most obvious problem associated with data invisibility is that data tampering is difficult to detect, and some work has been done to try to resolve this. Li et al. [8] proposed to store verification tags on the blockchain and generate a Merkle tree as proof. The transaction information must be uploaded to the blockchain, which prevents malicious users from tampering during the training process. Weng et al. [9] proposed a blockchain-based auditing model called Deepchain that considers the malicious behavior of model aggregators.

However, another problem of semantic invisibility in encrypted data was left behind. Cryptography methods blur the user's ownership of the real information corresponding to the ciphertext data during the transaction. Malicious users can steal historically encrypted gradients from publicly audited information or previous transactions, and upload them elsewhere to gain benefits, in what we call an encrypted forwarding attack. The attack will destroy the fairness of the entire system, and make fewer users willing to participate in collaborative training. Recently, researchers have proposed extracting user data features through the first few layers of deep neural networks, making them adaptable to multiple uncertain deep learning tasks [10, 11]. This will lead to a change in traditional collaborative training from focusing on a single defined task to completing multiple uncertain tasks, further expanding the reach of encrypted forwarding attacks.

In this paper, we aim to resolve the problem of encrypted forwarding attacks in collaborative training. To prevent users from submitting others' encrypted gradients maliciously, users need to use Pedersen commitment to commit on the uploaded gradient. The model aggregator can verify the ownership of the encrypted gradient by determining whether the user has plaintext. Our main contributions are as follows:

- We propose a data ownership verification mechanism to counter encrypted forwarding attacks caused by cryptography-based privacy-preserving methods. We use an interactive Σ-protocol and Pedersen commitment algorithm to prove that the user has the plaintext corresponding to the encrypted gradients.
- We design a united commitment scheme to complete the verification process of commitments in batches, thereby reducing verification consumption for the aggregator.
- We verify the effectiveness of the model on a social face dataset CelebA and a digital recognition dataset MNIST, then provide a safety analysis. The experimental results demonstrate that the commitment scheme does not impose an additional burden on secure aggregation of social applications.

The rest of this paper is organized as follows. In *Related Work*, we introduce the existing work of cryptography-based privacy-preserving methods and derivative audits approaches of encrypted data. *Preliminaries* introduces the relevant background knowledge of the Σ-protocol, Pedersen commitment, and attack models. *System Design* elaborates the details of the ownership verification mechanism. In *Security Analysis*, we analyze the correctness and safety of our mechanism. *Experimental Results* evaluates and analyses the performance of Paillier algorithm and Pedersen commitment in our mechanism. Finally, we conclude our paper in *Conclusion*.

RELATED WORK

Cryptography-Based Privacy-Preserving Methods in Collaborative Training

Privacy concern is one of the main problems in collaborative training. Although some collaborative training allows collaborative participants to only share model updates rather than raw training data, such as federated learning. There are still some privacy problems that are not completely solved [12–14]. Attackers may infer whether a sample exists in the training dataset or contain certain privacy attributes from the user gradient, called member inference attacks [15] and attribute inference attacks [16].

Secure aggregation [17] is a cryptography-based privacy-preserving method that prevents aggregators from gaining direct access to individual participants' gradients and committing privacy attacks. Homomorphic encryption is a common security aggregation algorithm that allows users to operate on ciphertext and decrypt the results of operations, where the decryption result is the same as for operating on plaintext. Participants can first encrypt the gradient *via* homomorphic encryption. The parameter aggregator then aggregates all encrypted gradients and decrypts the aggregated result, thereby indirectly obtaining a global model update contributed by the participants. The existing homomorphic encryption methods are mainly based on full-homomorphic encryption [18] and semi-homomorphic encryption [19].

Fully homomorphic encryption is mainly based on ideal lattices theory [20]. It relies on a large number of polynomial-based power operations and modulo operations, which greatly increases the consumption of implementation. It is still difficult to apply directly in the existing work.

Paillier encryption [19] is a representative work of semi-homomorphism that is widely used because of its simple structure. Phong et al. [21] proposed the method combining asynchronous stochastic gradient descent and Paillier homomorphic encryption. They proved that the system could prevent the aggregator from learning the data privacy of the participants, and ensure the availability of the model training with the acceptable system overhead. Zhou et al. [22] applied a homomorphic encryption scheme to fog computing. They proposed a scheme that combines Paillier encryption and blindness technology, which can resist collusion attacks by multiple malicious entities.

In order to improve the efficiency of the homomorphic encryption system, Zhang et al. [23] proposed a federated learning model called BatchCrypt. They encoded batches of gradients and perform homomorphic encryption with less time, which greatly reduces the overhead of the whole system. LWE has more extensive applicability because of its full-homomorphism. Hao et al. [24] utilized and improved the BGV algorithm based on LWE, which achieves the secure aggregation of gradients to prevent privacy disclosure. It makes the scheme based on homomorphic encryption practical.

The above works complete the data privacy protection for the participants and make the system feasible, but another potential problem has appeared: it is difficult for the aggregator to determine whether a problem exists due to the semantic invisibility of the encrypted data.

Verification of Secure Aggregation

Semantic invisibility of encrypted data presents numerous challenges, including transmission errors, malicious tampering by intermediaries, and the exclusion of some user-generated gradients by dishonest aggregators.

The correctness audit of the transmission gradient becomes very difficult in the encrypted state. Weng et al. [9] provided an audit mechanism based on ∑-protocol [25] which generates proofs of correctness for each gradient. Guo et al. [26] proposed a Paillier-based zero-knowledge proof algorithm. The server and users can jointly calculate the statement of encrypted gradients to prove the correctness of gradients.

To ensure that the gradients provided by each participant are indeed aggregated correctly, Xu et al. [27] use a homomorphic hash technique combined with pseudorandom function to help users verify the correctness of aggregation. To extend the applicability of the algorithm, Guo et al. [28] proposed a commitment scheme based on linearly homomorphic hash to verify the integrity of aggregation. Before sending gradient to the server, the users need to generate and exchange the hash value of their gradient. The users can verify whether the gradient is tampered with by checking hash values.

Despite the foregoing, there is a serious problem that has not been addressed. Encrypted data may be accessed by multiple users as public data. Malicious users can forward them to other collaborative tasks with the same purpose for improper profit. There are even cases where multiple users claim ownership of ciphertext in the same task. This will undermine the willingness of honest users to engage in collaborative training and thus lead to a shortage of training data for social applications. In this paper, we propose an ownership verification mechanism based on the Patterson commitment that can prove that the user owns the plaintext of the data without providing the plaintext.

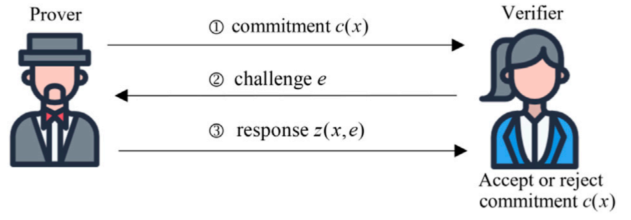


FIGURE 1 | The general process of Σ-protocols.

PRELIMINARIES

In this section, we briefly recall the definition of Σ-protocol and Pedersen commitment, and introduce the attack models.

Σ-protocol

Σ-protocol [25] is a two-party interaction protocol in zero-knowledge proof field. It is used to prove that someone knows a secret without disclosing it. There are many classic examples such as Schnorr's protocol [29], which is used for authentication.

Consider a binary relation R and an element $(x, y) \in R$, and we have $f(x) = y$, where x is called the pre-image of y under a mapping f. In Σ-protocol, we think of x as a witness and y as the corresponding instance for x, where x and y are finite values.

Suppose that there are two parties, one of which is prover, and the other is verifier. Without exposing x, the prover wants to prove to the verifier that he knows the pre-image value x of y under the mapping f. As shown in **Figure 1**, they need to follow Σ-protocol with the steps below:

Step 1: Prover computes a commitment c with the value of, and sends it to verifier.

Step 2: Verifier returns a random value e called challenge to prover.

Step 3: Prover sends a response z to verifier. It is computed with x and e.

Step 4: Verifier uses instance y and the message (c, e, z) generated in previous steps to compute the verification result.

Since the challenge e is randomly selected, if the prover does not know the witness x, he cannot use challenge e to compute a correct response z in Step 3. When the verification passes, the verifier is convinced that the prover knows the witness x. The witness in our ownership verification mechanism can be a secret gradient used for social computing. A user can prove that it knows the secret gradient x corresponding to the commitment$c$

Pedersen Commitment

Pedersen commitment [30] is a homomorphic commitment scheme, in which one computes a commitment bound to a chosen value. Pedersen commitment scheme can be used to prove that a committed value is not tampered with. According to its homomorphism, it is usually used to prove that the secret committed data satisfies certain binding relationships. The scheme consists of a commitment phase and a verification phase. In the beginning, A trusted third party selects a multiplicative group G with the order of large prime q, then selects two prime elements g and h of group G, where no one knows the value x to make $g = x * h$. After the committer and verifier agree on two elements g and h, they follow the process as follows:

Commitment: To commit to secret value v, the committer chooses a random value r, computes commitment $comm(v, r) = v * g + r * h$ and sends it to verifier.

Verification: Committer sends (v, r) to verifier. Verifier use (v, r) to computes $comm'(v, r) = v * g + r * h$ and checks whether it is equal to $comm(v, r)$. If the verification passed, it means that v is not tampered with.

Pedersen commitment scheme is designed that the sum of two commitments can also be seen as a commitment. It can be represented by $comm(v_3, r_3) = com(v_1, r_1) + com(v_2, r_2)$, where $v_3 = v_1 + v_2$, $r_3 = r_1 + r_2$. It seems like someone use a random value r_3 to commit to v_3.

Pedersen commitment scheme has the following properties: 1) Perfectly hiding: the r in the commitment computation is randomly chosen, so two commitments to one value will be different. No one can find a correlation between commitment and committed value. 2) Computationally binding: the commitments on different messages are different, which means that the malicious party cannot deny the value he committed. A more detailed introduction and proof can be seen in [30].

Different from Σ-protocol, the committed value in Pedersen commitment scheme needs to be revealed in the verification phase. It cannot be directly used to prove that the single encrypted gradient is not tampered with. In our scheme, we use the homomorphism of Pedersen commitment scheme to check whether the aggregated commitment is correctly corresponding to the aggregated gradient, further confirming the binding relationship between the commitment and encrypted gradient.

Threat Models

As we attempt to solve the problem of data silos in social computing through cooperative training, privacy protection and data ownership have become unavoidable difficulties. Cryptography-based secure aggregation methods can be a good solution to keep private information from being accessed by unauthorized users, but they also make it more difficult to verify data ownership. Two main roles are included in the current collaborative training:

Model Aggregator is a service provider that publishes the request for data training. It asks multiple users to provide trained gradients to conduct social computing.

Data Owners are users who have datasets and are willing to participate in collaborative training. They submit the trained models or gradients to the model aggregator for model updating.

Once we use a cryptography-based secure aggregation algorithm that prevents model aggregators from obtaining plaintext data from a single data owner. It is difficult to distinguish whether the user who submits encrypted data is its real owner or not. As shown in **Figure 2**, attackers can download encrypted data from elsewhere or historically, and then participate in the current cooperative training for improper profit, which we call encrypted forwarding attacks. In addition, we present several threat scenarios for encrypted forwarding attacks.

Threat 1. Attackers can only obtain encrypted data and claim ownership of the encrypted data. Malicious users access encrypted data through public channels, such as the available audit information or publicly accessible blocks on the blockchain. The encrypted data is being downloaded by attackers and forwarded to similar tasks for unfair gain.

Threat 2. Attackers can obtain encrypted data as well as its verification information. As an intermediate forwarder, a malicious user receives not only the encrypted data but also the previous verification information. By masquerading as a data owner and participating in social computing, traditional mechanisms of ownership verification may be bypassed.

Security Goal: To prevent encrypted forwarding attacks by malicious attackers, we need an ownership verification mechanism for encrypted data. For privacy reasons, it can conclusively confirm ownership of data without directly obtaining the plaintext of user's data. In addition, verification information must be real-time to prevent falsification by forwarding historical verification information.

SYSTEM DESIGN

In this section, we present the specific construction of our data ownership verification mechanism. Our mechanism prevents the leakage of user information to curious aggregators while resisting encrypted forwarding attacks by malicious users, as detailed in the threat model.

System Overview

We suppose that there exist many users who possess private training data in a community. They all agree on the structure and configuration of a common task model. When a user wants to update its model, it claims to be a model aggregator and requests collaborative training from other users. Other data owners will respond to the model aggregator. We denote the model aggregator as ag and data owners as $ow_i, i \in \{1, \cdots, N\}$

The aggregator ag then collects gradients from these data owners to update its task model. The process of gradient collecting is shown in **Figure 3**.

Phase 1. Initialization

A user claims to be an aggregator to all other users in the community. Others will respond to the claim and form a

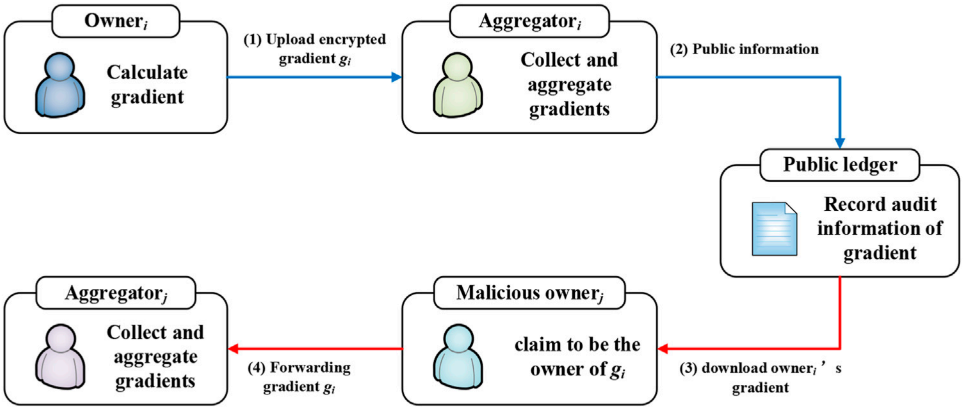

Owner$_i$
Calculate gradient
(1) Upload encrypted gradient g_i
Aggregator$_i$
Collect and aggregate gradients
(2) Public information
Public ledger
Record audit information of gradient
(3) download owner$_i$ ' s gradient
Malicious owner$_j$
claim to be the owner of g_i
(4) Forwarding gradient g_i
Aggregator$_j$
Collect and aggregate gradients

FIGURE 2 | Encrypted forwarding attacks in cooperative training.

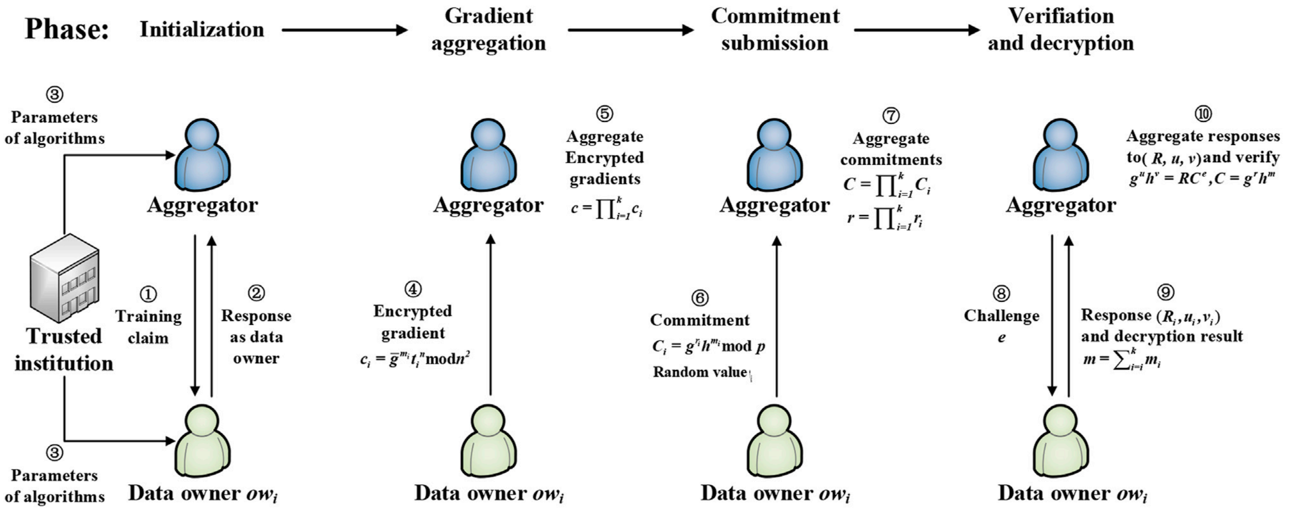

Phase: Initialization
Gradient aggregation
Commitment submission
Verifiation and decryption
③ Parameters of algorithms
Aggregator
Trusted institution
① Training claim
② Response as data owner
③ Parameters of algorithms
Data owner ow_i
④ Encrypted gradient $c_i = \bar{g}^{m_i} t_i^{\,n} \bmod n^2$
⑤ Aggregate Encrypted gradients $c = \prod_{i=1}^{k} c_i$
⑥ Commitment $C_i = g^{r_i} h^{m_i} \bmod p$ Random value
⑦ Aggregate commitments $C = \prod_{i=1}^{k} C_i$ $r = \prod_{i=1}^{k} r_i$
⑧ Challenge e
⑨ Response (R_i, u_i, v_i) and decryption result $m = \sum_{i=i}^{k} m_i$
⑩ Aggregate responses to(R, u, v)and verify $g^u h^v = RC^e, C = g^r h^m$

FIGURE 3 | The workflow of our system.

group with the aggregator by some means [31]. After that, a trusted institution executes the initialization of threshold Paillier algorithm and Pedersen commitment scheme for the group.

Phase 2. Gradient Aggregation

After each data owner ow_i gets gradient g_i through local training, it uses the threshold Paillier algorithm to encrypt it to cipher c_i, then uploads it to the aggregator ag. The aggregator ag will aggregate the gradients.

Phase 3. Commitment Submission

Each data owner ow_i who has uploaded gradient g_i to compute a commitment $comm_i$ based on Pedersen commitment scheme and uploads it to the aggregator ag.

Phase 4. Verification and Decryption

The aggregator ag conducts the verification of commitment with each data owner ow_i. If the verification passes, the aggregator ag asks data owners to collaboratively decrypt the aggregated gradient. At last, the aggregator ag confirms the ownership of gradients by checking commitments and gradients.

After the aggregator ag and all data owners have finished the process as above, the aggregator will pay each data owner for the model update. We can think that they have finished a round of secure aggregation.

Gradient Aggregation Based on Paillier Algorithm

Initialization: After the aggregator and data owners form a group together, a trusted institution will execute the initialization process. Concretely, it confirms the scale of the group and initializes the (l, N) threshold Paillier algorithm, where N is the number of data owners and $l \in \{\frac{N}{3} + 1, \ldots, N\}$ is the least number of data owners together to decrypt a cipher. The concrete process is shown in **Algorithm 1**.

Algorithm 1 The initialization of threshold Paillier algorithm

```
1: Select large prime $p = 2p' + 1, q = 2q' + 1$
   where $p'$and $q'$ is prime, $gcd(pq, (p-1)(q-1)) = 1$
2: Set $n = pq,\ \lambda = lcm(p-1, q-1),\ m = p'q'$
3: Define $L(x) = \frac{(x-1)}{n}$
4: Randomly choose $\beta \in {Z_n}^*, (a, b) \in {Z_n}^* \times {Z_n}^*$
5: Set $\bar{g} = (1+n)^a \times b^n, \theta = L(\bar{g}^{m\beta}) = am\beta \, mod \, n$
6: Set public key $PK = (\bar{g}, n, \theta)$, private key $SK = m\beta$
7: Let $a_0 = m\beta$, set $f(X) = \sum_{i=0}^{t} a_i X^i$ where $0 \le a_i \le nm - 1$
8: Set $SK_i = s_i = f(i) \, mod \, nm$ as the share of $a_0$ for each $worker_i$
9: Randomly choose $r \in {Z_{n^2}}^*$, set $VK = v = r^2 \, mod \, n^2$,
   denote $\Delta = l!$, compute $VK_i = v^{\Delta s_i} \, mod \, n^2$
```

The trusted institution publishes the public key $PK = (\overline{g}, n, \theta)$. It also distributes shares of private key SK_i and verification key VK_i to each data owner ow_i, respectively.

Encoding of gradient: The homomorphism of Paillier algorithm can be represented as $\prod_i Enc(m_i) = Enc(\sum_i m_i)$, where $Enc()$ denotes the encryption. The aggregator ag can aggregate encrypted gradients $Enc(m_i)$ and decrypt it to obtain aggregated gradient $\sum_i m_i$. Suppose that a data owner ow_i has the gradient $g_{\mathrm{i}} = (g_{i1,} \cdots, g_{ij}, \cdots, g_{in})$ to upload. Each element g_{ij} represents the gradient of the $j-th$ component of the model. First of all, each data owner ow_i encodes the gradient. They divide each element g_{ij} into several vectors $g_{ij}^{(\mathrm{k})}$ and encode them into a specific format which can be encrypted by Paillier algorithm. The data owner ow_i encrypts the gradient by encrypting each vector $g_{ij}^{(\mathrm{k})}$. To explain the encryption of gradients briefly, we use m_{i} to represent the gradient encrypted by Paillier algorithm.

Encryption and aggregation: We assume that there is a group composed of k data owners. Given the public parameter $PK = (\overline{g}, n, \theta)$ and threshold l, each data ownerow_i encrypts its gradient g_{i} by computing $c_i = \overline{g}^{m_i} t_i^n \mathrm{mod} n^2$, where t_i is a random element chosen from Z_n^*. Then they together upload the cipher c_i to the aggregator ag. The aggregator ag will aggregate all the encrypted gradient c_i by computing $c = \prod_{i=1}^{k} c_i$ and return the aggregation result c to each data owner ow_i to request the decryption. To ensure that the aggregator ag correctly aggregates all the encrypted gradient c_i, we introduce the commitment scheme in [28].

Decryption and update: In the decryption phase, each data owner ow_i provides his decryption share $d_i = c^{2\Delta s_i} \mathrm{mod} n^2$, where s_i is its share of the private key. In addition, each data owner ow_iwill also publish the proof $s_i = \log_{v^\Delta} VK_i = \log_{c^{4\Delta}} (d_i)^2$. If at least l data owners are verified to provide correct decryption shares, the aggregator ag can obtain the decryption result by computing $m = L(\prod_{i \in S} d_i^{2u_i} \mathrm{mod} n^2) \times \frac{1}{4\Delta^2\theta} \mathrm{mod} n$, where $u_i = \Delta \times \lambda_{0,i}^S \in Z$, $\lambda_{x,i}^S = \prod_{i' \in S \setminus \{i\}} \frac{x-i'}{i-i'}$ and $L(u) = \frac{u-1}{N}$. The proof of correctness can be seen in [32]. To request data owners to join in the decryption process, the aggregator ag should pay for the gradients to them. According to the addictive homomorphism of Paillier algorithm, the decryption result m is equal to $\sum_{i=1}^{k} m_i$. The aggregator ag will use it to update the task model.

Ownership Verification

In the collaborative training based on homomorphic encryption such as Paillier encryption, the aggregator is set to obtain only the decrypted aggregation result but not any gradient from individual data owner. Only in this way, the privacy in gradients will not be directly obtained by the aggregator. If a malicious user obtains an encrypted gradient, he may forward the encrypted gradient to another aggregator to profit. To solve the forwarding problem, the aggregator is required to verify the ownership of each received encrypted gradient.

The verification mechanism is based on Σ-protocol and Pedersen commitment scheme. As for Pedersen commitment scheme, a trusted institution is required to select two prime elements $g, h \in G$, where G is a cyclic group and $\log_g h$ is unknown to others. The process of verification is shown in **Algorithm 2**:

Commitment submission: After data owner ow_i has uploaded the encrypted gradient c_i to the aggregator ag, it computes a commitment $comm(r_i, m_i)$: $= C_i = g^{r_i} h^{m_i}$, where r_i is a random value and m_i is the gradient. Then it submits (C_i, r_i) to the aggregator ag.

Commitment verification: The aggregator ag will compute the aggregated gradient $c = \prod_{i=1}^{k} c_i$ and the aggregated commitments $C = \prod_{i=1}^{k} C_i$, $r = \sum_{i=1}^{k} r_i$. Then it sends a random challenge value e back to each data owner. Each *data* owner ow_i needs to use the challenge value e to compute a response $R_i = g^{k_i} h^{l_i}$, $u_i = k_i + er_i \mathrm{mod} p$, $v_i = l_i + em_i \mathrm{mod} p$. Then each data owner submits (R_i, u_i, v_i) to the aggregator ag. After that, the aggregator ag aggregates all (R_i, u_i, v_i) to obtain (R, u, v) by computing $R = \prod_{i=1}^{k} R_i, u = \sum_{i=1}^{k} u_i, v = \sum_{i=1}^{k} v_i$. Then it verifies whether $g^u h^v = RC^e$ holds. If it passes, it means that each data owner ow_i proves it knows the secret value m_i bound to the commitments C_i.

Algorithm 2 The interaction of commitment between aggregator and owners

Given a group G of order p and two generators g, h.

Each $owner_i$ has gradient m_i and encrypted gradient c_i

Aggregator ag	Data owner ow_i
	Upload encrypted gradient c_i
	Compute commitment $C_i := g^{r_i} h^{m_i} \, mod \, p$
	Send (C_i, r_i) to $aggregator$
Aggregate gradients $c = \prod_{i=1}^{k} c_i$	
Aggregate commitments	
$C = \prod_{i=1}^{k} C_i$,	
$r = \sum_{i=1}^{k} r_i$,	
Send challenge value e to data owner ow_i	
	Compute response
	$R_i = g^{k_i} h^{l_i} \, mod \, p$,
	$u_i = k_i + er_i \, mod \, p$,
	$v_i = l_i + em_i \, mod \, p$
	Send (R_i, u_i, v_i) to $aggregator$
Aggregate response	
$R = \prod_{i=1}^{k} R_i$,	
$u = \sum_{i=1}^{k} u_i$,	
$v = \sum_{i=1}^{k} v_i$,	
Verify whether $g^u h^v = RC^e$ holds	
	Upload decryption shares of c
Obtain gradient $m = \sum_{i=1}^{k} m_i$	
Check whether $C = g^r h^m$	

Ownership verification: To verify the committed value is the correct gradient, the aggregator *ag* requests owners to decrypt the gradient $c = \prod_{i=1}^{k} c_i$, and obtains the aggregated gradient $m = \sum_{i=1}^{k} m_i$. Then he checks whether $C = g^r h^m$. If it passes, it means that the secret value m_i is indeed the gradient corresponding to cipher c_i. The aggregator *ag* confirms that each data owner owns the plaintext of his gradient.

In our scheme, the aggregator can reduce the consumption of verification through checking the aggregated commitments. If the verification failed, it means that some data owners cannot provide the correct plaintext of their gradients. The aggregation of gradients and commitments will be rescheduled. Concretely, the aggregator can divide the group into several subgroups, then repeat the gradient aggregation and commitment verification. The malicious data owner will be identified through a series of verifications.

SECURITY ANALYSIS

The Protection of Gradient

In order to maintain the correctness of gradient and prevent it from being obtained directly by others, we use the addictive threshold Paillier homomorphic encryption algorithm. Firstly, we discuss the feasibility of the threshold Paillier encryption algorithm for gradient protection in our scheme.

Lemma 1: $\varepsilon_g(x, y) \rightarrow g^x * y^n \bmod n^2$ is bijective when the order of g is a non-zero multiple of n, and it is also a homomorphic mapping that $\varepsilon_g(x_1, y_1) * \varepsilon_g(x_2, y_2) = \varepsilon_g(x_1 + x_2, y_1 + y_2)$.

Lemma 2: A number x is said to be an nth residue modulo n^2 if there exists a number $y \in Z_{n^2}^*$ such that $z = y^n \bmod n^2$, and it is hard to find the value of z.

The proof of Lemma 1 and Lemma 2 can be seen in [19]. Lemma 1 indicates that the aggregated ciphertext of gradient can be decrypted to the aggregated gradient. The aggregator is able to acquire aggregated gradient without knowing individual gradients. From Lemma 2, we can conclude that the ciphertext of gradient is hard to be cracked.

Each data owner in our threshold Paillier algorithm has the ability to decrypt a cipher only in groups, so whenever a data owner obtains an individual encrypted gradient, he can't decrypt it unless others collude with him. The individual gradient can be well protected in our encryption scheme.

The Σ-protocol can make one prove that he knows the secret without revealing it. Specifically, a prover can state a commitment and prove that he knows the secret in the commitment. In the process of our protocol, the data owner ow_i states $C_i{:} = g^{r_i} h^{m_i}$, and then generates two random values $(k_{\mathrm{i}}, l_{\mathrm{i}})$ to hide the information in $(R_i,\ u_i, v_i)$ to prevent gradient m_i from being exposed to others.

Lemma 3: Only when the generated random values k_{i} and l_{i} are different, no one can recover the secret value from the $(R_i,\ u_i, v_i)$, that is, a commitment will not disclose any information about the committed value.The concrete proofs can be seen in [33]. As long as the data owner ensures that the random values k_{i} and l_{i} are different, we can think that the gradient is protected in the commitment according to Lemma 3.

TABLE 1 | The structure of model.

Smile recognition	Digit recognition
2 × Conv3-64	Conv1-16
2 ×Conv3-128	Conv1-32
3 × Conv3-256	FC-10
3 × Conv3-256	—
2 × FC-4096	—

The Validity of Commitment

Lemma 4: Two commitments for two different messages are different, otherwise the relationship between g and h can be calculated, which is not in line with the discrete logarithm hypothesis.

According to Lemma 4, we know each commitment is corresponding to a unique gradient. If a data owner submits a commitment, it means that it states the ownership of a gradient. We suppose that there exists a user who forwards another data owner's encrypted gradient. As we stated in *Threat Models*, we consider two kinds of treat model.

If an attacker only obtains encrypted gradient, it needs to state a commitment. Consider that it does not know the plaintext of gradient g_i, it may commit to a secret fake gradient g_{fake} and upload the commitment C_{fake}. In the following steps, it needs to respond to the challenge e, which proves the binding relationship between fake gradient g_{fake} and commitment C_{fake}. In other words, if the verification passes, the aggregator can think that the user has committed to the plaintext of gradient, which can be denoted as g_i. After that, the aggregator uses the decryption result to check whether the committed g_{fake} is equal to the gradient g_i. If not, the user's malicious behavior of forwarding will be discovered.

If an attacker obtains encrypted data as well as its historical verification information, it needs to respond to the dynamic challenge e in the verification process. Due to it even does not know the committed value in commitments, the verification will fail. In other words, the historical verification information is not reusable for an attacker.

Despite each data owner may obtain encrypted gradient from the aggregator in the decryption phase, the forwarding attack still does not work because of our commitment scheme.

EXPERIMENTAL RESULTS

In this section, we introduce the experiments in our scheme, including the performance evaluation of Paillier algorithm and Pedersen commitment scheme. We deploy an aggregator and multiple data owners to simulate the process of collaborative training.

We build the deep learning model with Python(version3.83), Numpy(version 1.18.5), Pytorch(1.6.0) on GPUs. We use CelebFaces Attributes Dataset (CelebA) to conduct smile recognition. The dataset includes 202,599 face images with 40 binary attributes. We randomly select 60,000 images and averagely assign them to the data owners. We also use the famous MNIST dataset to conduct handwritten digit recognition. We set the epoch of training as 3. The structure of the network is shown in **Table 1**.

After each iteration of the training, we output the gradient and evaluate the performance of Paillier algorithm. We compress the gradient by setting the precision of each gradient at 3–5, and use the gradient to update the model. Concretely, the data owners train

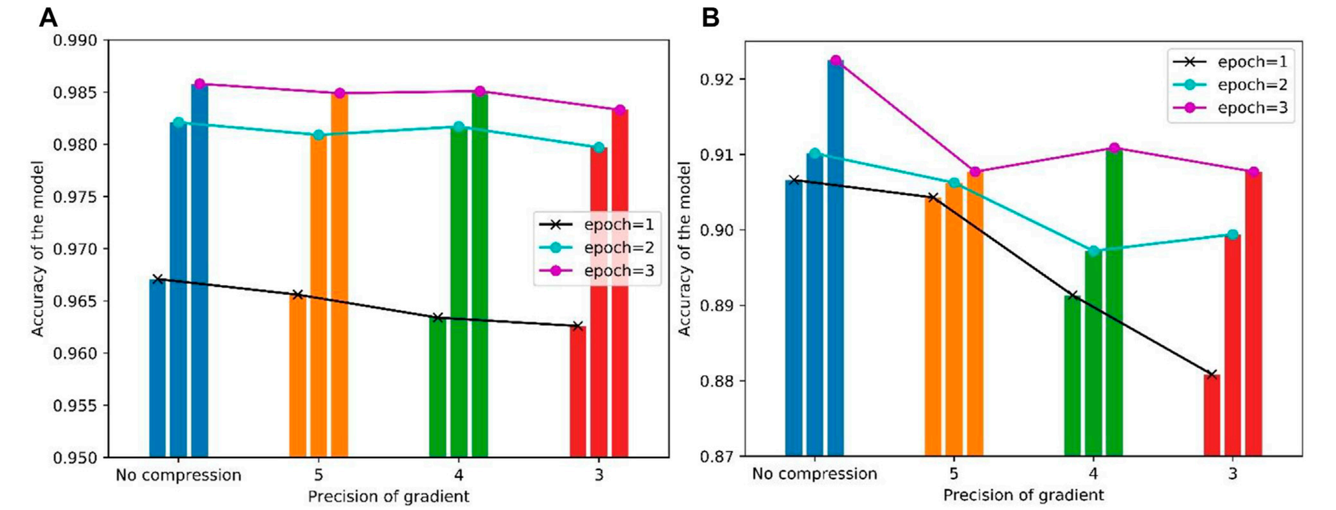


FIGURE 4 | The accuracy with different precision of gradient. **(A)**; On MNIST dataset **(B)** On CelebA dataset;

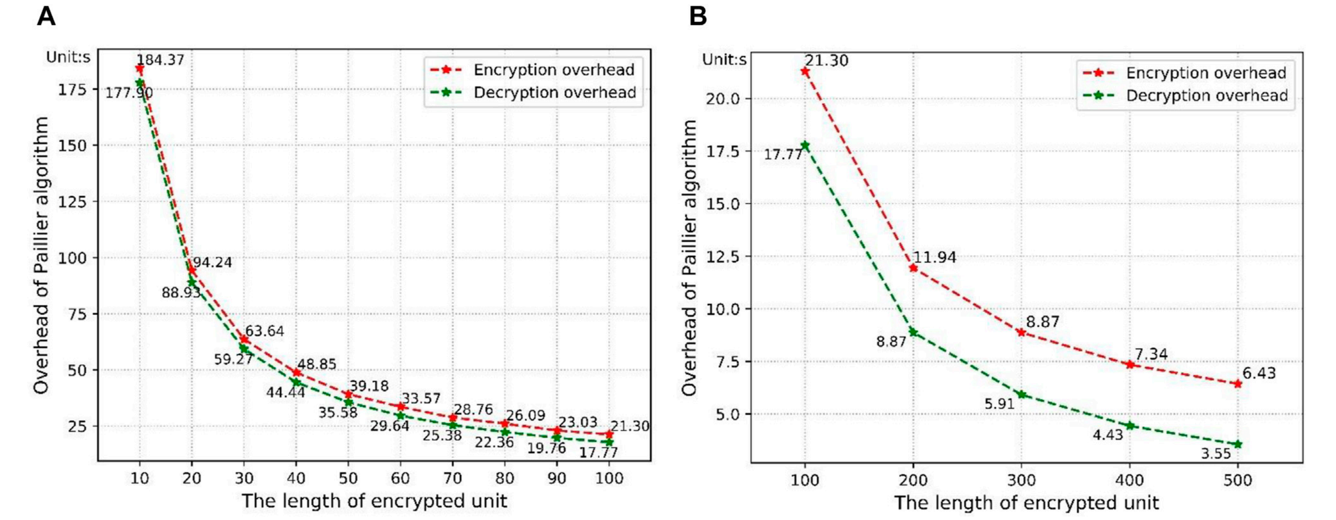


FIGURE 5 | The overhead of Paillier algorithm. **(A)** encrypted unit ∊[10,100]; **(B)** encrypted unit ∊[100,500].

their local model and output the gradient in given precision. Then we aggregate these gradients o update the model. With the change of given precision, the accuracy of the model is shown in **Figure 4**.

If we do not compress the gradient, the accuracy on CelebA dataset in each epoch is 90.66, 91.01, and 92.24%, respectively. The accuracy on MNIST dataset in each epoch is 96.71, 98.21, and 98.58%, respectively. When we control the precision of gradient at 5, 4, 3, the accuracy on CelebA dataset in the third epoch changes to 90.77, 91.08, and 90.41%, which is approximately 1.5% lower compared to the accuracy with no gradient compression. With gradient compression, the accuracy decline is not obvious on MNIST dataset, where the accuracy in the third epoch is 98.49, 98.51, and 98.33%. decreasing within 0.2%. The change of gradient precision has a more significant impact on large-scale training. When the precision decreases, the accuracy in different epochs is lower, which makes the training more difficult to converge. To ensure the quality and stability of training, we choose to set the precision at five in our subsequent experiment.

As for gradient encryption, we use the Paillier algorithm implemented in Python based on CPU. We use the Charm-crypto.[1] library to perform the encryption and decryption process of Paillier algorithm. Charm-crypto is a Python library for fast encryption on large numbers. We encode the gradient as a vector of integers that can be encrypted by Paillier algorithm. By

[1] The encryption library called Charm-crypto can be download in https://github.com/JHUISI/charm.

TABLE 2 | Overhead of Pedersen commitment scheme with one data owner.

phase	Length of committed unit				
	10 (s)	50 (s)	100 (s)	200 (s)	300 (s)
Committing	0.925	0.311	0.256	0.204	0.188
Verifying	2.329	0.392	0.296	0.224	0.202

TABLE 3 | Overhead of large-scale commitment aggregation.

Length of committed unit	Number of data owners		
	10 (s)	50 (s)	100 (s)
100	0.294	0.331	0.377
200	0.222	0.241	0.264

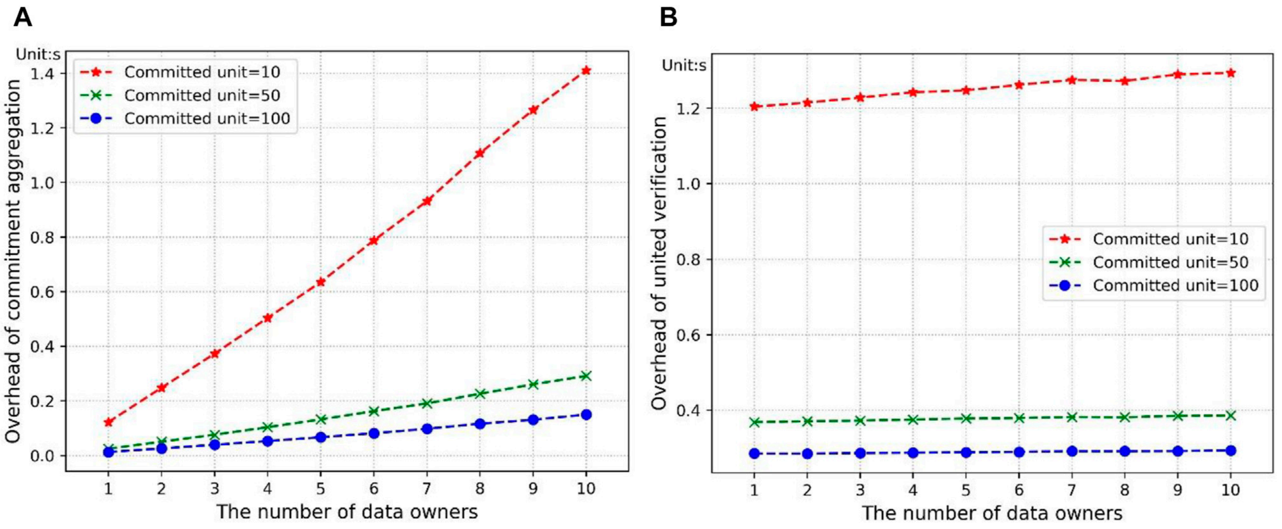


FIGURE 6 | The overhead of Pedersen commitment scheme. **(A)** commitment aggregation; **(B)** commitment verification.

TABLE 4 | The Overhead of large-scale commitment verification.

Length of committed unit	Number of data owners		
	10 (s)	50 (s)	100 (s)
100	0.148	1.091	3.090
200	0.077	0.568	1.585

controlling the precision of gradients, we get multiple vectors of the gradient in various lengths. Because the precision changes, the total number of parameters varies. We choose to evaluate the Paillier algorithm on a message with a length of 10^5 digits. The evaluation of overhead is shown in **Figure 5**.

To maintain the correctness of decryption results after cipher aggregation, the length of the encrypted unit is limited by the number of ciphers and security parameters of the Paillier algorithm. On the premise that we maintain the correctness of decryption, we divide the message into multiple units to encrypt. When the length of an encrypted unit ranges from 10 to 100, as the length is doubled, the time consumption is almost halved. The length of the unit does not have a significant impact on unit encryption time. When the length ranges from 100 to 500, the unit encryption time obviously increases as the length increases, however, the total time consumption still decreases. It is better to choose a larger encryption unit if possible. Since training overhead is much higher than the gradient encryption in cooperative training, the overhead of Paillier algorithm above is acceptable.

As for the Pedersen commitment scheme, we use cryptographic algorithms based on discrete logarithm to deploy it. We divide the message with a length of 10^5 digits into multiple units again and evaluate the time consumption of the scheme. We first simulate the process of commitment and verification of a device. The overhead is shown in **Table 2**.

The time consumption of committing is much less than the time consumption of encryption and decryption. When the unit is larger, the total overhead of the commitment scheme is at a lower level. To evaluate the impact of the number of data owners on the commitment scheme, we set the number of data owners from 1 to 10, respectively. Each data owner needs to commit to a message with a length of 10^5 digits. The experimental results can be seen in **Figure 6**. The total time of commitment aggregation increases because the number of devices increases. The time consumption of verification increases slightly as the number of devices increases, which almost does not influence the total time consumption of our commitment scheme.

To better show the practicality of this commitment scheme in large-scale cooperative learning, we set the committed unit at 100, 200, respectively, and expand the number of devices to 50 and 100. The evaluation result can be seen in **Tables 3**, **4**. The total time consumption is still much lower than the encryption and decryption. It means that the increase in data owners will not impose a clear burden on the commitment scheme. Our experimental results indicate that our commitment scheme is feasible in the cooperative training.

CONCLUSION

In this paper, we propose an ownership verification mechanism against encrypted forwarding attacks in data-driven social computing. It can defend against the malicious gradient stealing and forwarding behavior in cryptography-based privacy-preserving methods. Based on the premise of maintaining gradient privacy, we present a protocol based on Σ-protocol and Pedersen commitment to achieve our security goal. Specifically, we design a united commitment algorithm to make participants cooperate to submit gradients and provide proof of data ownership. If any user submits other's gradient, it will fail to provide correct proof to pass the verification process. The experiment results validate the security of our proposed mechanism and demonstrate the practicality of our solution.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: the Large-scale CelebFaces Attributes (CelebA) Dataset, http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html, and THE MNIST DATABASE of handwritten digits, http://yann.lecun.com/exdb/mnist/.

AUTHOR CONTRIBUTIONS

ZS: Conceptualization and methodology, writing—original draft preparation; JW: formal analysis, validation, writing—original draft preparation; ZC: visualization; BW: project administration, supervision, and writing—review and editing; RL and YH: writing—review and editing.

FUNDING

This work was supported in part by the National Key R&D Program of China (No. 2018YFB2100400), in part by the National Natural Science Foundation of China (No. 62002077, 61872100), in part by the China Postdoctoral Science Foundation (No. 2020M682657), in part by Guangdong Basic and Applied Basic Research Foundation (No. 2020A1515110385), in part by Zhejiang Lab (No. 2020NF0AB01), in part by Guangzhou Science and Technology Plan Project (202102010440).

ACKNOWLEDGMENTS

The authors would like to thank reviewers for their critical reading and improvement of the manuscript.

REFERENCES

1. Zeroual A, Harrou F, Dairi A, and Sun Y. Deep Learning Methods for Forecasting COVID-19 Time-Series Data: A Comparative Study. *Chaos, Solitons & Fractals* (2020) 140:110121. doi:10.1016/j.chaos.2020.110121
2. Liang W, Yao J, Chen A, Lv Q, Zanin M, Liu J, et al. Early Triage of Critically Ill COVID-19 Patients Using Deep Learning. *Nat Commun* (2020) 11:3543–7. doi:10.1016/j.physrep.2007.04.00410.1038/s41467-020-17280-8
3. Li F, Sun Z, Li A, Niu B, Li H, and Cao G. Hideme: Privacy-Preserving Photo Sharing on Social Networks. In: IEEE INFOCOM 2019-IEEE Conference on Computer Communications; 2019, April 29-May 2; Paris, France (2019). doi:10.1109/INFOCOM.2019.8737466
4. Han W, Tian Z, Huang Z, Li S, and Jia Y. Topic Representation Model Based on Microblogging Behavior Analysis. *World Wide Web* (2020) 23:3083–97. doi:10.1007/s11280-020-00822-x
5. Konečný J, McMahan HB, Ramage D, and Richtárik P. Federated Optimization: Distributed Machine Learning for On-Device Intelligence. (2016) arXiv preprint arXiv:1610.02527.
6. Abadi M, Chu A, Goodfellow I, McMahan HB, Mironov I, Talwar K, and Zhang L. Deep Learning with Differential Privacy. In: Proceedings of the 2016 ACM SIGSAC conference on computer and communications security (2016). doi:10.1145/2976749.2978318
7. Li T, Sahu AK, Talwalkar A, and Smith V. Federated Learning: Challenges, Methods, and Future Directions. *IEEE Signal Process Mag* (2020) 37:50–60. doi:10.1109/MSP.2020.2975749
8. Li J, Wu J, Jiang G, and Srikanthan T. Blockchain-based Public Auditing for Big Data in Cloud Storage. *Inf Process Manage* (2020) 57:102382. doi:10.1016/j.ipm.2020.102382
9. Weng J, Weng J, Zhang J, Li M, Zhang Y, and Luo W. Deepchain: Auditable and Privacy-Preserving Deep Learning with Blockchain-Based Incentive. *IEEE Trans Dependable Secure Comput* (2019) 1. doi:10.1109/TDSC.2019.2952332
10. Li A, Duan Y, Yang H, Chen Y, and Yang J. TIPRDC: Task-independent Privacy-Respecting Data Crowdsourcing Framework for Deep Learning with Anonymized Intermediate Representations. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining; 2020, July 6-10; Virtual Event, CA, USA (2020). doi:10.1145/3394486.3403125
11. Sun Z, Yin L, Li C, Zhang W, Li A, and Tian Z. The QoS and Privacy Trade-Off of Adversarial Deep Learning: An Evolutionary Game Approach. *Comput Security* (2020) 96:101876. doi:10.1016/j.cose.2020.101876
12. Aghasian E, Garg S, Gao L, Yu S, and Montgomery J. Scoring Users' Privacy Disclosure across Multiple Online Social Networks. *IEEE access* (2017) 5: 13118–30. doi:10.1109/ACCESS.2017.2720187
13. Li S, Zhao D, Wu X, Tian Z, Li A, and Wang Z. Functional Immunization of Networks Based on Message Passing. *Appl Maths Comput* (2020) 366:124728. doi:10.1016/j.amc.2019.124728
14. Du J, Jiang C, Chen K-C, Ren Y, and Poor HV. Community-structured Evolutionary Game for Privacy protection in Social Networks. *IEEE Trans.Inform.Forensic Secur.* (2018) 13:574–89. doi:10.1109/TIFS.2017.2758756
15. Shokri R, Stronati M, Song C, and Shmatikov V. Membership Inference Attacks against Machine Learning Models. In: IEEE Symposium on Security and Privacy; 2017, May 22-26; San Jose, CA, USA (2017). doi:10.1109/SP.2017.41
16. Melis L, Song C, De Cristofaro E, and Shmatikov V. Exploiting Unintended Feature Leakage in Collaborative Learning. In: IEEE Symposium on Security and Privacy. (2017); 2019, May 19-23; San Jose, CA, USA (2019). doi:10.1109/SP.2019.00029
17. Yin L, Feng J, Lin S, Cao Z, and Sun Z. A Blockchain-Based Collaborative Training Method for Multi-Party Data Sharing. *Comput Commun* (2021) 173: 70–8. doi:10.1016/j.comcom.2021.03.027
18. Brakerski Z, and Vaikuntanathan V. Efficient Fully Homomorphic Encryption from (Standard) $\Mathsf{LWE}$. *SIAM J Comput* (2014) 43:831–71. doi:10.1137/120868669
19. Paillier P. Public-key Cryptosystems Based on Composite Degree Residuosity Classes. In: International Conference on the Theory and Applications of Cryptographic Techniques; 1999, May 2-6; Prague, Czech Republic (1999). p. 223–38. doi:10.1007/3-540-48910-X_16
20. Gentry C. Fully Homomorphic Encryption Using Ideal Lattices. In: Proceedings of the forty-first annual ACM symposium on Theory of computing; 2009, May 31-June 2; Bethesda, MD, USA (2009). doi:10.1145/1536414.1536440

21. Phong LT, Aono Y, Hayashi T, Wang L, and Moriai S. Privacy-Preserving Deep Learning via Additively Homomorphic Encryption. *IEEE Trans.Inform.Forensic Secur.* (2018) 13:1333–45. doi:10.1109/TIFS.2017.2787987
22. Zhou C, Fu A, Yu S, Yang W, Wang H, and Zhang Y. Privacy-preserving Federated Learning in Fog Computing. *IEEE Internet Things J* (2020) 7: 10782–93. doi:10.1109/JIOT.2020.2987958
23. Zhang CL, Li SY, Xia JZ, and Wang W. Batchcrypt: Efficient Homomorphic Encryption for Cross-Silo Federated Learning. In: USENIX Annual Technical Conference. (2020); 2020, July 15-17; Boston, MA, USA (2020).
24. Hao M, Li H, Luo X, Xu G, Yang H, and Liu S. Efficient and Privacy-Enhanced Federated Learning for Industrial Artificial Intelligence. *IEEE Trans Ind Inf* (2020) 16:6532–42. doi:10.1109/TII.2019.2945367
25. Damgård I. *On Σ-protocols. Lecture Notes*. University of Aarhus, Department for Computer Science (2002).
26. Guo JL, Liu ZY, Lam KY, Zhao J, Chen YQ, and Xing CP. Secure Weighted Aggregation for Federated Learning. (2020) arXiv preprint arXiv: 2010.08730.
27. Xu G, Li H, Liu S, Yang K, and Lin X. Verifynet: Secure and Verifiable Federated Learning. *IEEE Trans.Inform.Forensic Secur.* (2020) 15:911–26. doi:10.1109/TIFS.2019.2929409
28. Guo X, Liu Z, Li J, Gao J, Hou B, Dong C, et al. VeriFL: Communication-Efficient and Fast Verifiable Aggregation for Federated Learning. *IEEE Trans.Inform.Forensic Secur.* (2021) 16:1736–51. doi:10.1109/TIFS.2020.3043139
29. Schnorr CP. Efficient Signature Generation by Smart Cards. *J Cryptology* (1991) 4:161–74. doi:10.1007/BF00196725
30. Pedersen TP. Non-interactive and Information-Theoretic Secure Verifiable Secret Sharing. In: Annual international cryptology conference; 1991, July 11-15; Santa Barbara, CA, USA (1991). p. 129–40. doi:10.1007/3-540-46766-1_9
31. Li S, Jiang L, Wu X, Han W, Zhao D, and Wang Z. A Weighted Network Community Detection Algorithm Based on Deep Learning. *Appl Maths Comput* (2021) 401:126012. doi:10.1016/j.amc.2021.126012
32. Schoenmakers B, and Veeningen M. Universally Verifiable Multiparty Computation from Threshold Homomorphic Cryptosystems. In: International Conference on Applied Cryptography and Network Security; 2015, June 2-5; New York, USA (2015). p. 3–22. doi:10.1007/978-3-319-28166-7_1
33. Yu G. Simple Schnorr Signature with Pedersen Commitment as Key. *IACR Cryptol Eprint Arch* (2020).

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

ORIGINAL RESEARCH
published: 13 September 2021
doi: 10.3389/fphy.2021.741106

Analysis of Stock Price Data: Determinition of The Optimal Sliding-Window Length

Xuebin Liu[1], *Xuesong Yuan*[2], *Chang Liu*[3], *Hao Ma*[4] *and Chongyang Lian*[5]*

[1]School of Law, Central University of Finance and Economics, Beijing, China, [2]Ansteel Company Limited Cold-Rolling Silicon Steel Mill, Anshan, China, [3]School of Finance, Zhongnan University of Economics and Law, Wuhan, China, [4]School of Computer Science and Engineering, Northwestern Polytechnical University, Xi'an, China, [5]School of Law, Xinjiang University, Urumqi, Xinjang

Over the recent years, the study of time series visualization has attracted great interests. Numerous scholars spare their great efforts to analyze the time series using complex network technology with the intention to carry out information mining. While Visibility Graph and corresponding spin-off technologies are widely adopted. In this paper, we try to apply a couple of models derived from basic Visibility Graph to construct complex networks on one-dimension or multi-dimension stock price time series. As indicated by the results of intensive simulation, we can predict the optimum window length for certain time series for the network construction. This optimum window length is long enough to the majority of stock price SVG whose data length is 1-year. The optimum length is 70% of the length of stock price data series.

Keywords: time series visualization, complex network, sliding window-based visibility graph, multiplex visibility graph, stock price

OPEN ACCESS

Edited by:
Chao Gao,
Southwest University, China

Reviewed by:
Jun Hu,
Fuzhou University, China
Jiwei Xu,
Xi'an University of Posts and Telecommunications, China

****Correspondence:***
Chongyang Lian
1392265185@qq.com

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *14 July 2021*
Accepted: *31 August 2021*
Published: *13 September 2021*

Citation:
Liu X, Yuan X, Liu C, Ma H and Lian C (2021) Analysis of Stock Price Data: Determinition of The Optimal Sliding-Window Length. Front. Phys. 9:741106. doi: 10.3389/fphy.2021.741106

INTRODUCTION

Along with the big data era, time series widely exists in practice and is a popular data representation means, e.g., the stock price, the carbon price, white Gaussian noise, surface concentration ozone and etc. Specifically, time series is a sequence of data points represented in time order, while the time intervals between any consecutive points are always the same [1]. Due to the nonlinear and discrete properties, a bunch of analyzing approaches have been proposed [2]. Afterwards, complex network theory is developing rapidly [3] and applied to the analysis of time series data [1–9]. Hence, a technique, i.e., time series data visualization, and some improved versions, are developed by constructing complex networks from the initial data. Hence, sufficient analysis of the time series data can be performed accordingly.

Among those approaches, a technique, named Visibility Graph (VG), is widely adopted, and attracts the intensive interests [10]. This is initially proposed by Lacasa and his coworkers when investigating the time series data of robot movement [10]. Through VG, a corresponding complex network can be constructed, while the inherent properties and implied information of the original data can be preserved properly, such as Hurst coefficient, fractal properties [8, 11]. It is proved to be an efficient tool for the analysis of times series data [12–14]. Hence, the VG-based complex network and corresponding derivative theories are becoming a hot topic and various scholars have devoted their endless efforts into applying such theories into the various studies.

Initially, VG-based analysis mainly focuses on one-dimensional time series data. Recently, scholars start to investigate multiple time series data jointly to reveal inclined information. For instance, the authors in [10, 15] proposed a Multiplex Visibility Graphs (MVG) approach and conducted analysis of the surface concentration ozone, while complex networks are constructed for two time series data sets, i.e., the surface concentration ozone and the concentration of NO2 which are closely related with each other. Similarly, with the development of VG, a bunch of improved approaches have been proposed, such as sliding window-based Visibility Graph (SVG) [16], Multiplex Visibility Graph (MVG) [10, 15], Horizontal Visibility Graph (HVG) [17], and Limited Penetrating Visibility Graph (LPVG) [14, 18]. With the application of these approaches, it becomes easier for us to extract implied information from time series data.

Stock price time-series data is also one of the common time-series data. The analysis of stock price data, especially stock price trend prediction based on the analysis result, attracts the interests of various scholars [19, 20]. The authors in [20, 21] performed stock price forecasts and trend research study of stock price time-series data through machine learning approaches. While, we analyzed the stock price time-series data by complex network theory in which the corresponding complex network is constructed for stock time-series data, and relevant information can be studied accordingly. Here, we mainly adopt the SVG model to visualize the time series data of stock price. As revealed in [16], the appropriate window length for the analysis of different time series data sets varies. Hence, analyses of different stock price data are performed to determine the appropriate window length of SVG. Furthermore, corresponding multi-layer networks are constructed through MVG, then the correlations between time series data of multiple stock prices are thoroughly studied.

MODEL DESCRIPTION

Firstly, the VG and corresponding spin-off technologies are introduced. For VG, it is typically an undirected graph with the corresponding weight of each link equals to 1. For an original time series sequence, each data point is assigned an index indicating the time flag, i.e., X_i, while the data value for X_i equals to Y_i. Hence, each data point can be indicated by (X_i, Y_i) for simplicity. Aiming to construct the network through VG, we are supposed to determine whether a link exists between two data points from the original time series data while the corresponding criterion is provided as [10]:

1) If two data points A (X_a, Y_a) and B (X_b, Y_b) are consecutive, i.e., no data points exist between these two points, there definitely exists a link connecting A and B.
2) If two data points are not consecutive, i.e., a point C (X_c, Y_c) exists where ($X_a < X_c < X_b$), and the relationship described by **Eq. 1** is satisfied, then we can obtain a link connecting A and B.

$$Y_c < Y_a + (Y_b - Y_a)\frac{(X_c - X_a)}{(X_b - X_a)} \tag{1}$$

For any data point combination, the above criterion is applied to discriminate the existence of links, and the corresponding VG network can be derived accordingly which can be further indicated by an adjacent matrix. If a link exists between two data points, then the corresponding value in the adjacent matrix equals to 1; otherwise, it is 0. An illustrative example is shown in **Figure 1A** indicating the network construction process of a time series data set consisting of 10 points.

Sliding-window is widely applied in various areas and related algorithms are proved to be of high computational efficiency and able to reduce the required storage [22]. Hence, an improved method is developed as in [16] by introducing the sliding-window idea into the network constructing process of VG to improve construction efficiency. Because of sliding-window, the aforementioned criterion is only necessary to be applied between a data point and a certain point within the sliding-window. Thus, the necessary times of applying the above discriminate criteria will be reduced tremendously. As in [16], we suppose the time series data is composed of N data points while the selected sliding-window length equals to W. Then, the network construction procedure through SVG is provided as:

1) Step 1: For the first W data points, the discriminate criteria of the original VG algorithm are applied to determine the existence of links;
2) Step 2: The window moved forward by the distance of a data point, and a new data point enters the window. Thus, the sliding-window covers the new data point and the previously existed W-1 data points. Hence, the discriminate criteria of the original VG will be applied.
3) Step 3: Repeat Step 2 until we reach the end of the time series data.

Examples are provided in **Figure 1** which illustrates the construction process through VG and SVG with a window length of 4. For **Figures 1B–D**, the data points indicated by red columns are within the sliding-window, whereas those represented by blue columns are outside the sliding-window.

Accordingly, the computational complexity of SVG is largely determined by the required times of applying the discriminate criteria (fundamentally affected by the sliding-window length). For a time series consisting of N data points and a provided window length W, the required times of applying the discrimination criteria to construct the complex network, i.e., S, is calculated as

$$S = \frac{W \times (W-1)}{2} + (N - W) \times (W - 1) \tag{2}$$

where W*(W-1)/2 indicates the times of applying the discrimination criteria to the first W data points, while (N-W) refers to the total number of times when moving forward, and the discrimination criteria is anticipated to be applied for W-1 for each movement. When W is infinitely close to 1, the computational time complexity will be O (n). In practice, it is unlikely for W to be close to 1, then the practical complexity will

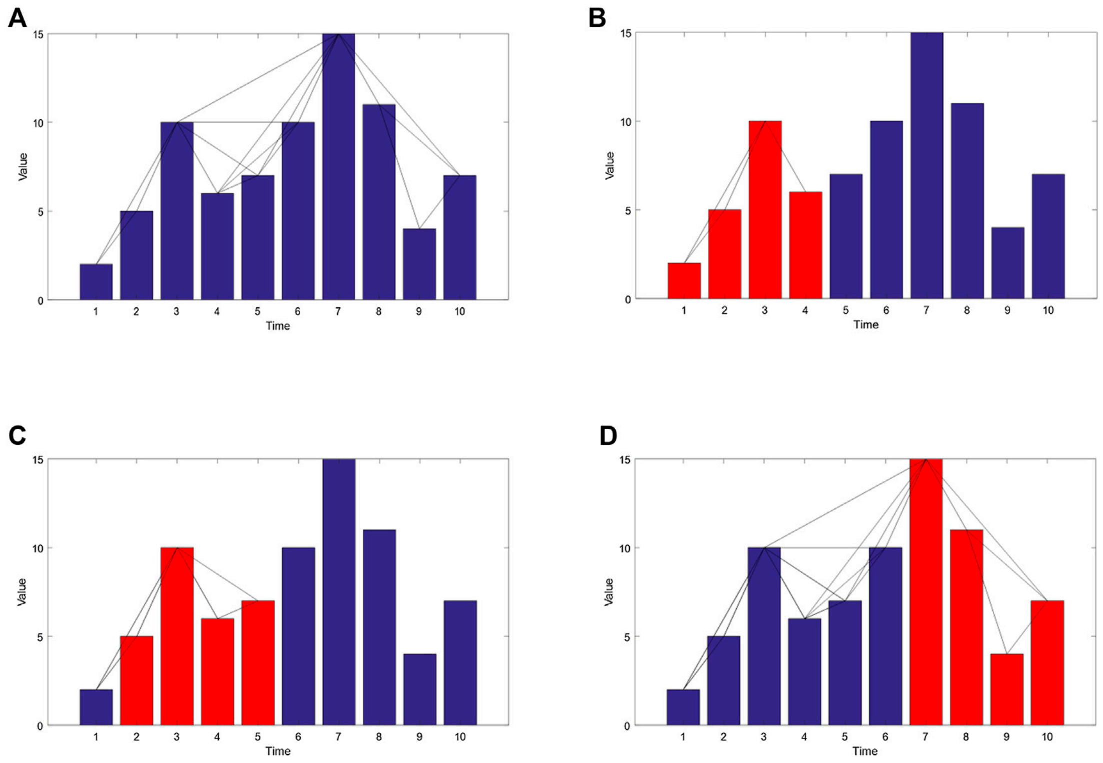
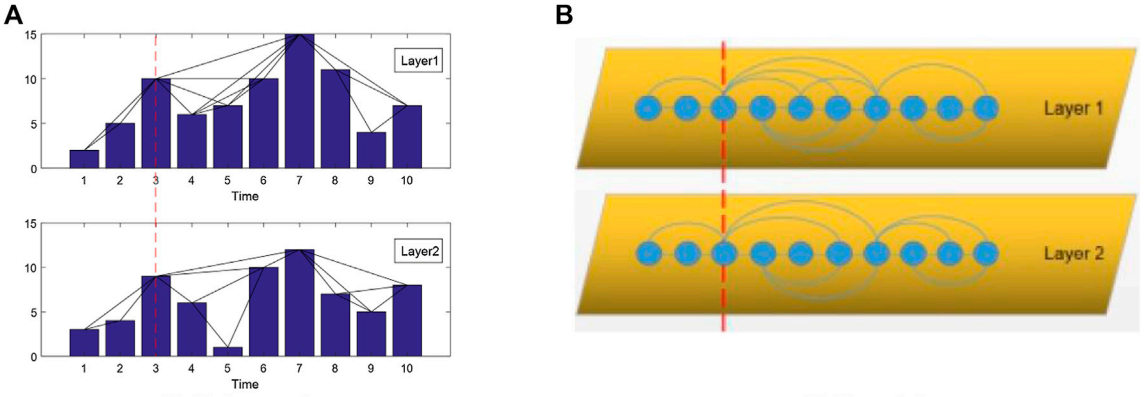


FIGURE 1 | (A) An example of the network construction process through VG. **(B–D)** A network construction process through SVG. Here, the lines connecting the top of data columns indicate the existed links.



Multi-time series

MVG model

FIGURE 2 | An illustration of the MVG.

fall into the range of O (n)and O (n^2). Generally, the average time complexity is around O (nlogn) [23].

In this manuscript, we also study time series data sets of multiple stocks, thus the MVG is also introduced [15]. For MVG, there exists one time axis in common reflecting the varying of different types of data at the same time. Such types of data have inclined relationships which can be analyzed through calculating corresponding network parameters of the MVG. An example of MVG is provided as in **Figure 2**. As illustrated by **Figure 3B**, the 3rd data points on different layers seem to possess similar properties.

As in [24], similar analysis can be performed to explore the implicit information of MVG. Here, two parameters are adopted aiming to investigate the interlayer information, i.e., Average Edge Overlap (AEO) and Interlayer Mutual Information (IMI) [25]. AEO is the average of the existence probabilities of a common link in all layers of the MVG which reflects the similarity of links on different

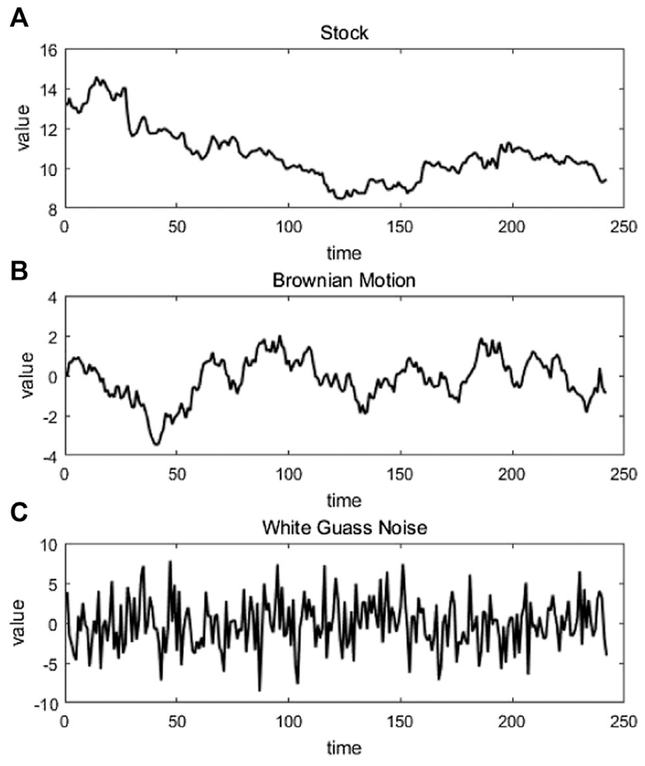


FIGURE 3 | Three types of representative time series data. **(A)** stock, **(B)** Brownian motion, and **(C)** Guass noize.

layers (being denoted as ω). Corresponding value is calculated as

$$\omega = \frac{\sum_{\mathrm{i}}\sum_{j>i}\sum_{\alpha}aij^{[\alpha]}}{M\sum_{\mathrm{i}}\sum_{j>i}\left(1-\delta\left(0,\sum_{\alpha}aij^{[\alpha]}\right)\right)} \tag{3}$$

where the numerator indicates the total number of the appearance of the link between any two data points i and j in the layers of MVG. While M represents the total number of layers for the MVG. If δ equals to 1, this indicates the link between the two data points does not exist in any of the layers. According to (**Eq. 3**), the maximum value of ω equals to 1, this indicates all layers of the MVG are identical. Correspondingly, the minimum value of ω equals to 1/M which corresponds to the scenario that every link only exists in one layer.

Another metrics, i.e., Interlayer Mutual Information (IMI), is introduced to reflect the relationship between the degree distributions of different layers [25]. Here $I(\alpha,\beta)$ indicates the IMI for two layers α and β which is provided as

$$I(\alpha,\beta) = \sum_{k[\alpha]}\sum_{k[\beta]} P(k[\alpha],k[\beta])\log\frac{P(k[\alpha],k[\beta])}{P(k[\alpha])P(k[\beta])} \tag{4}$$

where $P(k[\alpha], k[\beta])$ denotes the occurring probability of the case that a point on layer α possesses a degree of $k[\alpha]$ while the corresponding data point on layer β is of a degree of $k[\beta]$. P $(k[\alpha])$ represents the probability of a data point on layer α possessing a degree of $k[\alpha]$, while $P(\mathrm{k}\ [\beta])$ indicates the probability of a data point on layer β possessing a degree of $k[\beta]$. A higher $I(\alpha,\beta)$ indicates that the degree distributions of the two layers seem to be even more similar.

ANALYSIS OF STOCK PRICES

In this section, we focus on analyzing the time series data of stock price through the afore-mentioned approaches. Three representative types of data are selected for illustrations. **Figure 3A** illustrates the time series data for the stock opening price of Ping An Bank Co., Ltd. consisting of a total number of 242 data points. For comparison, **Figure 3B** and **Figure 3C** indicate the data by adding Brownian Motion with Hurst coefficient of 0.5 and one-dimensional White Gaussian noise of 10 dB, respectively. For ease of reference, the data series are assumed to be of the same lengths. Among the three data sets, the transition of the data indicated by **Figure 3A** seems to be the smoothest; while the varying trend of the data indicated by **Figure 3C** is the most violent.

As afore-mentioned, networks obtained through SVG for different sliding-window lengths are likely to be of different properties. First, we investigated the relationship between the maximum degree of the obtained network and the sliding-window length with corresponding results being presented in **Figures 4A–C**, respectively. As illustrated, the maximum degree varies if a different sliding-window length is adopted. Whereas, once the sliding window length arrives at a certain threshold, the maximum degree maintains. However, for different types of data, the maximum degree varies. For the stock opening price of Ping An Bank Co., Ltd., the maximum degree is approximately 60, while the maximum degrees for data incorporating Brownian Motion and White Gaussian noise are 40 and 20, respectively. Furthermore, the corresponding velocity of convergence also varies. For the stock opening price of Ping An Bank Co., the maximum degree converges until W increases to approximately 70% of the total number of data points (W is supposed to be larger than 164 which is approximately 68% of the total data points). For data incorporating Brownian Motion, the maximum degree converges when W approximately equals to 35% of the total number. While for the data with White Gaussian noise, the corresponding value converges when W is around 20% of the total number.

The discrepancy of the maximum degree or the velocity of convergence can reflect the characteristics of different types of data. Compared with the other types of data, the transition of the stock opening price of Ping An Bank Co., Ltd. seems to be the smoothest; thus, it is likely for more data points to meet the discriminate criteria. Hence, the derived network is likely to possess a large maximum degree. In other words, it is highly likely for data points that are far from each other to be connected if the transition is smooth. Whereas, for the data with Gaussian white noise, the discriminate criteria condition is less likely to be met due to the sudden variance of the original data series. Thus, the maximum degree is relatively small. Reversely, if the maximum degree of an obtained network is relatively small, we can predict that the transition of the original data is sharp.

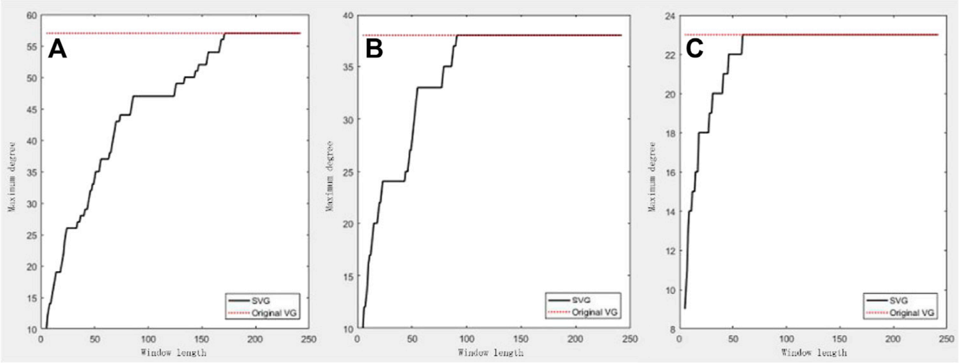


FIGURE 4 | Illustrations of the relationships between the maximum degrees of the obtained complex networks through SVG and *W* for different types of time series data **(A)** Original data; **(B)** Brownian Motion incorporated **(C)** Gaussian white noise considered.

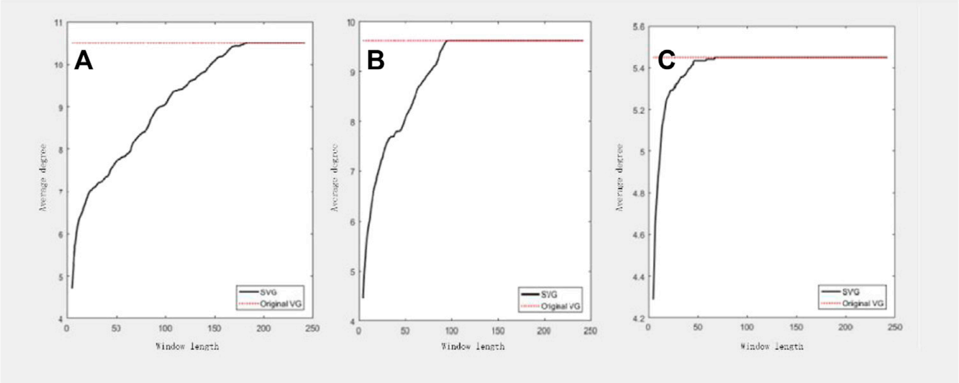


FIGURE 5 | Relationships between the average degree of the obtained complex network through SVG and the sliding-window length for different types **(A)** Original data; **(B)** Brownian Motion incorporated **(C)** Gaussian white noise considered.

TABLE 1 | Comparison of network construction through SVG for different types of time series data.

Data description	Optimal window length/Total number of data points (%)	S For SVG/S for original VG (%)
Stock opening price of Ping An Bank Co.	69.4	90.7
Data with Brownian Motion (Hurst coefficient = 0.5)	40.4	64.7
Data with White Gaussian noise of 10 dB	28.1	48.4

Previously, we mainly investigated the maximum degree of the obtained network, whereas, the optimum window length is also of great significance. Afterwards, we also investigated the relationship between the average degree of the obtained network and *W* to provide information regarding the determination of the optimum *W* with the corresponding results being provided in **Figure 5**.

The criteria of optimum *W* are provided as: for a given *W*, if the primary parameters of the obtained network, such as maximum degree, is approximately the same as the corresponding value obtained through original VG, and the percentage of varying velocity is smaller than 5% with the increase of *W*, we can regard it as the optimum value. Accordingly, we find that the optimum *W* for the original stock price data is also approximately 168 (about 69% of the total data points). Similarly, analyses can also be conducted on the other types of data to find the optimum *W*. Corresponding results are provided in **Table 1** which illustrate the computational efficiency of SVG and original VG.

Moreover, the degree distribution of the obtained network is provided as in **Figure 6**. We see that the derived network for the stock opening price data follows power-law distribution while the

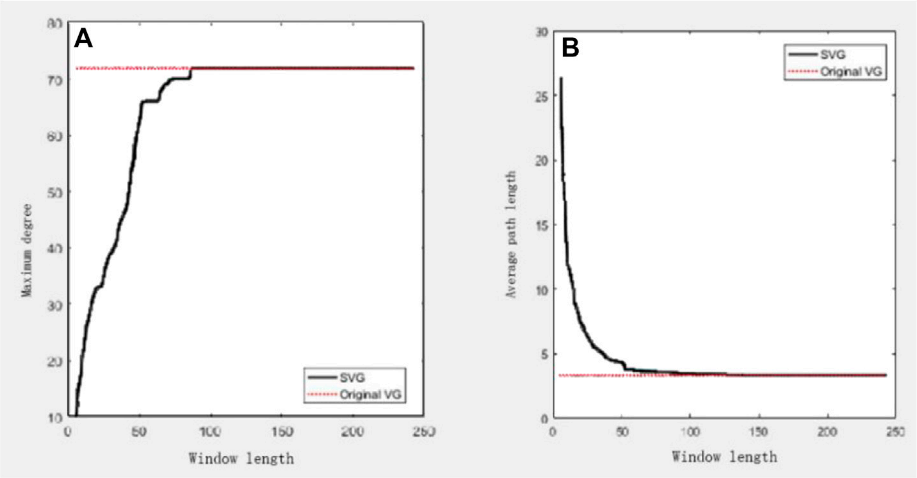


FIGURE 6 | Illustration of the relationships between different parameters and window length for the stock opening price data of Shenzhen Cau Technology Co., Ltd. in 2018 **(A)** maximum degree; **(B)** maximum average length.

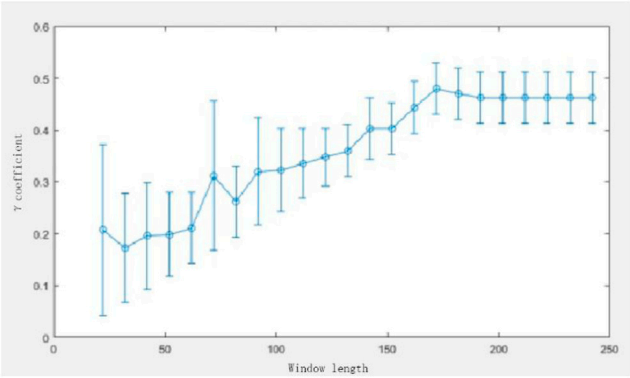


FIGURE 7 | Illustration of the coefficient γ vs. W.

relationship between γ and W is given in **Figure 7**. As indicated, a sliding-window length of 168 (approximate 69% of the total number of data points) seems to be the appropriate value for the construction of the complex network from the stock opening price when considering parameter γ.

In order to derive a general conclusion, we also take the stock opening price data for 500 stocks from the A-share market. After sufficient analyses, we find that for the one-year-long data, a window length of 75% of the total data points is sufficient for the construction of the network. Here, sufficient length means it is safe and incurs no information loss, but it does not necessarily to be the optimum window length. After further analysis, we find that the optimum window length might be smaller than 60% of the total points for the data of some stocks. Another stock of Shenzhen Cau Technology Co., Ltd. is taken for an illustration. This company mainly focus on computer software and bio-pharmacy technology which is likely to be affected by market fluctuations. Hence, the stock price data is likely to fluctuate rapidly [17]. The optimum W for constructing a network through SVG is only 100 for Shenzhen Cau Technology Co., Ltd. as illustrated in **Figure 6**. This validates the previous conclusion that when the data fluctuate rapidly, the maximum degree of the network obtained through SVG is likely to be smaller. Whereas for stock prices of bank and real estate companies, the optimum window length is around 160. This verifies the conclusion that the optimum window length is largely affected by the characteristic of the original data.

Furthermore, we also performed an analysis of the stock opening price data for Ping An Bank Co. from 2018 to 2019. The relationships between incorporated parameters and window length are provided in **Figure 8**. As presented, for a two-year-long data, the optimum window length is approximately 378 (which is about 77.8% of the total data points) according to the above criteria of discriminating the optimum W. We can find that for data of different lengths, the percentage obtained by dividing the obtained window length with the total data points varies slightly. Furthermore, to construct the network through SVG for different data lengths, the obtained optimum window lengths are provided in **Table 2**.

As aforementioned, it is necessary to analyze multiple time series data to mine implicit information. Hence, experiments are conducted into the investigation of different stock price data by applying MVG. First, a two-layered network is constructed from the opening stock price and the highest stock price of Ping An Bank Co. **Figure 9A** illustrates the corresponding original time series data, while the obtained adjacent matrices are provided in **Figures 9B,C**. As presented in **Figure 9A**, the opening stock price and the highest stock price of Ping An Bank Co. are of a similar trend; this can also be observed by similar adjacent matrices of the networks for different data series.

Regarding the obtained two-layered networks, the aforementioned parameters can be calculated, being listed as ω

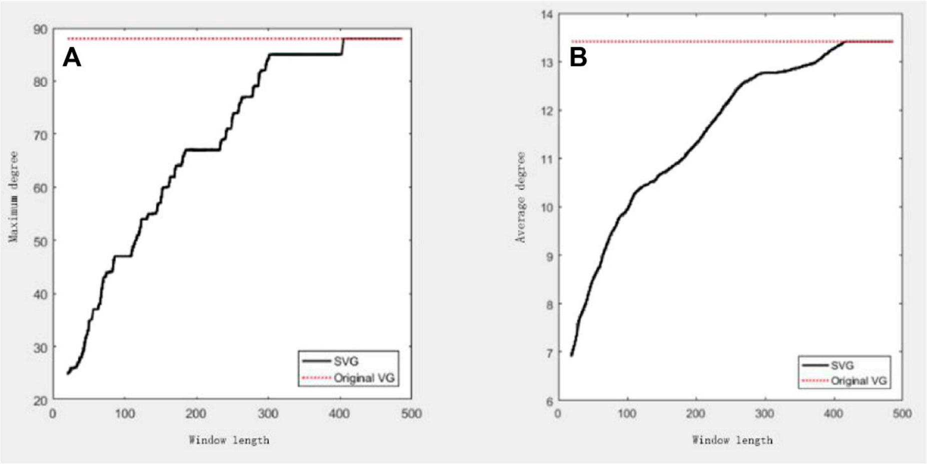


FIGURE 8 | Illustration of the relationship between incorporated parameter and *W* through SVG. Here, a two-year stock opening price data for Ping An Bank Co. is considered. **(A)** maximum degree; **(B)** average degree.

TABLE 2 | Optimum window length for constructing network through SVG for stock opening price of Ping An Bank Co. with different total data points.

Total points of the time series data	Optimum window length/Total data points (%)
242 (one-year long data)	69.4
361 (one-year and a half data)	72.5
486 (two-year long data)	77.8

= 0.7285 and $I(\alpha,\beta)$ = 1.3096. These parameters can be used to predict the correlations of the provided data series. ω can be used to indicate the link distributions of different layers; thus, the obtained networks are similar. Later, we performed an analysis of different time series data combinations and the corresponding parameters are calculated, provided in **Table 3**. As illustrated, the correlations of different data combination for the same stock price varies. But even the scenario with the least correlation, the corresponding value is much higher than the correlation between No2 and surface concentration ozone.

Moreover, we concern about the relationship between the stock prices of different stocks. Thus, we build an MVG network for the price data of different stocks. For example, we build a two-layer complex network based on the time-series of the opening prices of Ping An Bank and Vanke Co. Ltd. Class A. Similarly, **Figure 10A** below shows the opening stock price time-series data of two stocks in 2018, and **Figures 10B,C** shows the non-zero elements' distribution of the complex network adjacency matrix generated by the opening price data of two stocks. After calculating the interlayer parameters of MVG, we can obtain ω = 0.6426 and $I(\alpha,\beta)$ = 1.2836 for the two-layer network. Such values almost reach the value of the two-layer network of surface ozone concentration and nitrogen dioxide concentration mentioned earlier. This means that the two stocks of Ping An Bank Co. and Vanke Co. Ltd. Class A have a relatively close relationship in the trend of stock data. More results are provided in **Table 4**. Obviously, the opening data is consistent with the above conclusion, while conclusions hold true for all the other price data. The close relationship between Ping An Bank Co. and Vanke Co. Ltd. A on the trend of stock data can be explained from the perspective of economics as the relationship between finance and real estate. The investment cost and investment income of the real estate industry are closely related to the financial environment, while the market in turn affects the economy and finance [15, 17]. Therefore, this mutual influence relationship in economics can be seen on the interlayer parameters of the two-layer MVG of stock prices.

In contrast, there exists no such strong correlation between Ping An Bank Co. and the biopharmaceutical stock Shenzhen CAU Technology Co. Ltd. **Table 5** below shows the interlayer parameters of the two-layer MVG networks obtained for the opening prices of some other stocks and Ping An Bank Co.

In **Table 5**, both Vanke Co. Ltd. A and Shenzhen Zhenye Co. Ltd. A are real estate stocks. According to the previous analysis, after building a two-layered MVG network for other stock data and Ping An Bank Co., the inter-layer parameters tend to indicate the tightness of the relationship between the two stocks. In contrast, Shenzhen CAU Technology Co. Ltd. is a biopharmaceutical stock, while Digital China Group Co. Ltd. is an Internet stock. They are not closely related to Ping An Bank Co. from the perspective of stock, and therefore we can see a relatively low correlation. After analyzing other stocks, we found similar conclusions. For example, after constructing a two-layer network with the opening price data of Changan Automobile stock and Daye Special Steel stock, the average edge overlap ω obtained equals 0.6489. This value almost even exceeds the ω value of the two-layer network constructed with Ping An Bank Co. and Vanke Co. Ltd. A. Daye Special Steel Co. Ltd. belongs to steel and metal shares, while Chongqing Changan Automobile Co. Ltd. belongs to industrial machinery shares. The industrial production of the latter depends on the raw materials provided by

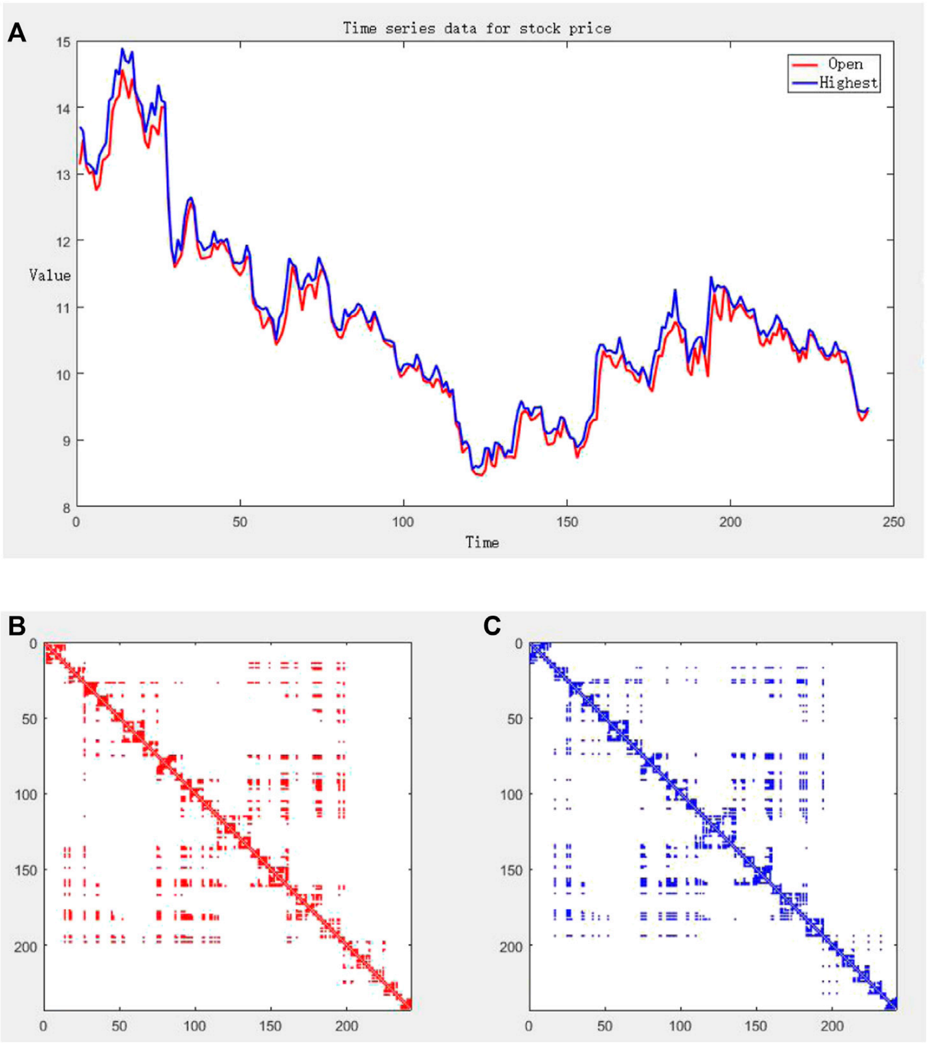


FIGURE 9 | (A) The opening stock price and highest stock price of Ping An Bank Co.; **(B)** Adjacent matrices of the networks obtained through MVG for opening price; **(C)** Adjacent matrices of the networks obtained through MVG for highest price.

TABLE 3 | Parameter obtained for the two layered networks for different combinations of time series data for Ping An Bank Co.

	ω	$I(\alpha,\beta)$
Opening and highest	0.7285	1.3096
Opening and lowest	0.7567	1.3714
Opening and close	0.6649	1.1782
Highest and lowest	0.7309	1.4094
Highest and close	0.7692	1.3027
Lowest and close	0.7238	1.2566

the former type of enterprises. It is the correlation between the two in the background of the stock industry that causes the inter-layer parameters of the two-layer network constructed by the two stock price data also show a relatively close correlation.

COMPLEX ANALYSIS

When I = 0, the inner loop executes n times; when I = 1, the inner loop executes n−1 times, and when I = n−1, the total execution times can be calculated as follows:

$$
\begin{aligned}
& n + (n-1) + (n-3) + \ldots\ldots + I \\
&= (n-1) + [(n-1)+2] + [(n-2)+3] + [(n-3)+4] + \ldots\ldots \\
&= (n-1) + (n-1) + (n+1) + (n+1) + \ldots\ldots \\
&= (n+1)n/2 \\
&= (n+1)/2 \\
&= n^2/2 + n/2
\end{aligned}
$$

According to the second rule of derivation of large order o previously mentioned: only the highest order is reserved, so n2/2 is reserved. According to the third article, if the constant of this

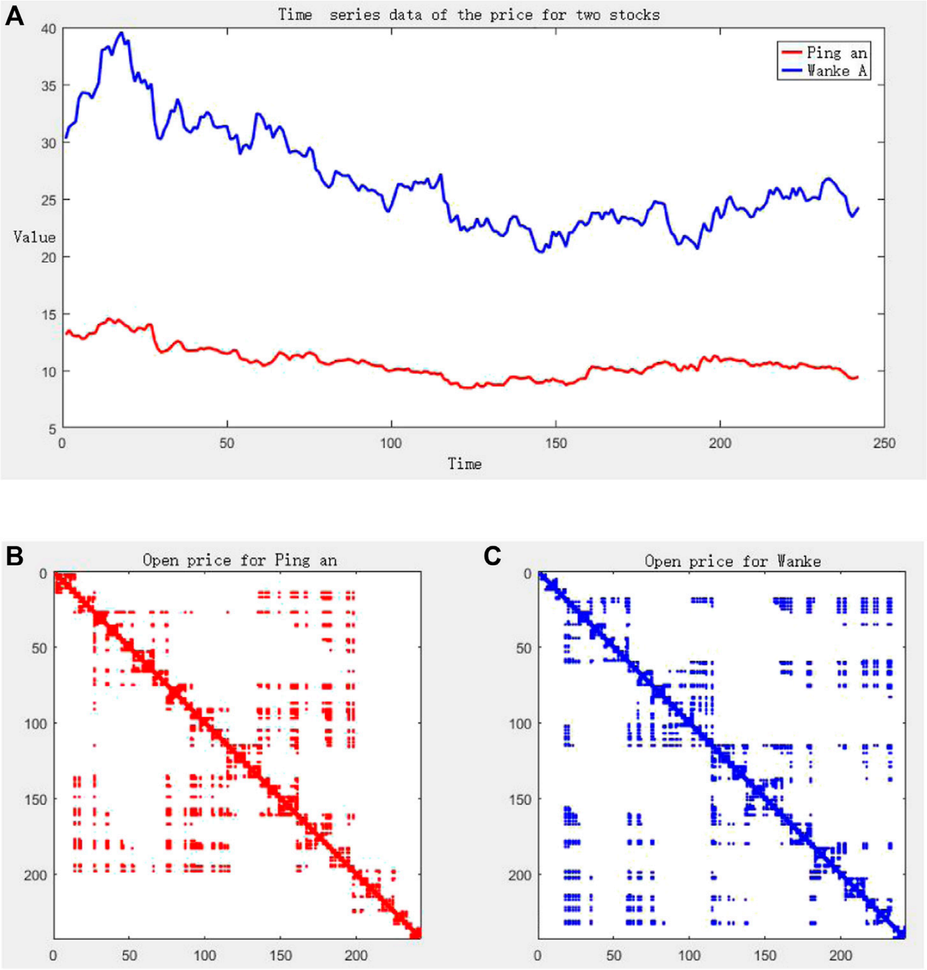


FIGURE 10 | (A) Two stocks' opening price time-series data; **(B,C)** denote the adjacent matrices of two-layer MVG network built with provided time-series data for Ping An Bank Co. and Vanke Co. Ltd., respectively.

TABLE 4 | Parameters obtained for the price data for Ping An Bank Co. and Vanke Co. Ltd. Class A.

	ω	$I(\alpha,\beta)$
Opening	0.6426	1.2836
Close	0.6469	1.2396
Highest	0.6407	1.1834
Lowest	0.6379	1.2674

TABLE 5 | Parameters obtained for the two layered networks for different combinations of Ping An Bank Co. and four different stocks.

Stock name	ω	$I(\alpha,\beta)$
Vanke Co. Ltd. A	0.6426	1.2836
Shenzhen CAU Technology Co. Ltd.	0.5921	1.1656
Shenzhen Zhenye Co. Ltd.	0.6381	1.2780
Digital China Group Co. Ltd.	0.6068	0.9865

item is removed, then 1/2 of the time complexity of this code will be removed. Finally, the timev complexity of this code is O (n2).

CONCLUSION

Tvhrough VG and related techniques (SVG and MVG) for analyzing time-series data, we conducted intensive experiments on various stocks, and we also combine the knowledge of securities and social economics to obtain more meaningful research results. In this paper, we try to find out the size of the window length W that should be selected when constructing the network through SVG for stock price time-series with the length of N. According to the above analysis, for one-year-long stock price time-series data, the length of the security window that does not lose the original data information in most cases due to the establishment of the SVG network is approximately $W/N = 70\%$. At this time, compared with the traditional VG model, the reduction in the amount of calculation when constructing the network is about 10%. Although such a window length may compromise the effect of using the SVG algorithm, such a window length is safe and sufficient. Such a long window length is not always necessary, in other words, it is not optimal. The actual optimal window length for some stocks can even be W/N < 50%. And this optimal window length has been proved in this paper to be related to the type and nature of stocks. Different types of stock data may have different optimal window length values, which requires further research. Besides, it is found that for stock price time series data of different lengths, the optimal value when applying the SVG model and the value of the security window length ratio W/N is different, which calls for further research. We believe that the SVG algorithm will play a more significant advantage in building a complex network for stock price data with further research conducted.

REFERENCES

1. Zou Y, Donner RV, and Marwan N. Complex Network Approaches to Nonlinear Time Series Analysis[J]. *Phys Rep* (2018) 787:1–97. doi:10.1016/j.physrep.2018.10.005
2. Pavón-Domínguez P, Jiménez-Hornero FJ, and Gutiérrez de Ravé E. Joint Multifractal Analysis of the Influence of Temperature and Nitrogen Dioxide on Tropospheric Ozone. *Stoch Environ Res Risk Assess* (2015) 29(7):1881–9. doi:10.1007/s00477-014-0973-5
3. Albert R, and Barabási A-L. Statistical Mechanics of Complex Networks. *Rev Mod Phys* (2002) 74(1):47–97. doi:10.1103/revmodphys.74.47
4. Zhou C, Ding L, Skibniewski MJ, Luo H, and Jiang S. Characterizing Time Series of Near-Miss Accidents in Metro Construction via Complex Network Theory. *Saf Sci* (2017) 98:145–58. doi:10.1016/j.ssci.2017.06.012
5. Fan X, Li X, Yin J, Tian L, and Liang J. Similarity and Heterogeneity of price Dynamics across China's Regional Carbon Markets: A Visibility Graph Network Approach. *Appl Energ* (2019) 235:739–46. doi:10.1016/j.apenergy.2018.11.007
6. Hu J, Xia C, Li H, Zhu P, and Xiong W. Properties and Structural Analyses of USA's Regional Electricity Market: A Visibility Graph Network Approach. *Appl Maths Comput* (2020) 385:125434. doi:10.1016/j.amc.2020.125434
7. Gao Z, Li S, and Dang W. Wavelet Multiresolution Complex Network for Analyzing Multivariate Nonlinear Time Series[J]. *Int J Bifurcation Chaos* (2017) 27(8):1750123. doi:10.1142/s0218127417501231
8. Carmona-Cabezas R, Ariza-Villaverde AB, Gutiérrez de Ravé E, and Jiménez-Hornero FJ. Visibility Graphs of Ground-Level Ozone Time Series: A Multifractal Analysis. *Sci Total Environ* (2019) 661:138–47. doi:10.1016/j.scitotenv.2019.01.147
9. Dai P-F, Xiong X, and Zhou W-X. Visibility Graph Analysis of Economy Policy Uncertainty Indices. *Physica A: Stat Mech its Appl* (2019) 531:121748. doi:10.1016/j.physa.2019.121748
10. Lacasa L, Luque B, Ballesteros F, Luque J, and Nuño JC. From Time Series to Complex Networks: The Visibility Graph. *Pnas* (2008) 105(13):4972–5. doi:10.1073/pnas.0709247105
11. Lacasa L, Luque B, Luque J, and Nuño JC. The Visibility Graph: A New Method for Estimating the Hurst Exponent of Fractional Brownian Motion. *Europhys Lett* (2009) 86(3):30001. doi:10.1209/0295-5075/86/30001
12. Liu K, Weng T, Gu C, and Yang H. Visibility Graph Analysis of Bitcoin price Series. *Physica A: Stat Mech its Appl* (2020) 538:122952. doi:10.1016/j.physa.2019.122952
13. Lacasa L, Nuñez A, Roldán É, Parrondo JMR, and Luque B. Time Series Irreversibility: a Visibility Graph Approach. *Eur Phys J B* (2012) 85(6):217. doi:10.1140/epjb/e2012-20809-8
14. Gao Z-K, Cai Q, Yang Y-X, Dang W-D, and Zhang S-S. Multiscale Limited Penetrable Horizontal Visibility Graph for Analyzing Nonlinear Time Series. *Sci Rep* (2016) 6:35622. doi:10.1038/srep35622
15. Carmona-Cabezas R, Gómez-Gómez J, Ariza-Villaverde AB, Gutiérrez de Ravé E, and Jiménez-Hornero FJ. Multiplex Visibility Graphs as a Complementary Tool for Describing the Relation between Ground Level O3 and No2. *Atmos Pollut Res* (2020) 11(1):205–12. doi:10.1016/j.apr.2019.10.011
16. Carmona-Cabezas R, Gómez-Gómez J, Gutiérrez de Ravé E, and Jiménez-Hornero FJ. A Sliding Window-Based Algorithm for Faster Transformation of Time Series into Complex Networks. *Chaos* (2019) 29(10):103121. doi:10.1063/1.5112782
17. Luque B, Lacasa L, Ballesteros F, and Luque J. Horizontal Visibility Graphs: Exact Results for Random Time Series. *Phys Rev E Stat Nonlin Soft Matter Phys* (2009) 80:046103. doi:10.1103/PhysRevE.80.046103
18. Ren W, and Jin N. Sequential Limited Penetrable Visibility-Graph Motifs. *Nonlinear Dyn* (2020) 99(3):2399–408. doi:10.1007/s11071-019-05439-y
19. Davis EP, and Zhu H. Bank Lending and Commercial Property Cycles: Some Cross-Country Evidence. *J Int Money Finance* (2011) 30(1):1–21. doi:10.1016/j.jimonfin.2010.06.005
20. Du K, Fu Y, and Qin Z. Regime Shift, Speculation, and Stock Price[J]. *Res Int Business Finance* (2010) 52:101181.
21. Rapach DE, Strauss JK, and Zhou G. International Stock Return Predictability: What Is the Role of the United States?. *J Finance* (2013) 68(4):1633–62. doi:10.1111/jofi.12041
22. Tanbeer SK, Ahmed CF, Jeong B-S, and Lee Y-K. Sliding Window-Based Frequent Pattern Mining over Data Streams. *Inf Sci* (2009) 179(22):3843–65. doi:10.1016/j.ins.2009.07.012
23. Lan X, Mo H, Chen S, Liu Q, and Deng Y. Fast Transformation from Time Series to Visibility Graphs. *Chaos* (2015) 25:083105. doi:10.1063/1.4927835
24. Lacasa L, Nicosia V, and Latora V. Network Structure of Multivariate Time Series. *Sci Rep* (2015) 5:15508. doi:10.1038/srep15508
25. Dimitri GM, Agrawal S, Young A, Donnelly J, Liu X, Smielewski P, et al. A Multiplex Network Approach for the Analysis of Intracranial Pressure and Heart Rate Data in Traumatic Brain Injured Patients. *Appl Netw Sci* (2017) 2(1):29. doi:10.1007/s41109-017-0050-3

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: http://www.10jqka.com.cn/.

AUTHOR CONTRIBUTIONS

XL: Visualization, Software, Computation, Drawing Writing, XY: Investigation, CL: Writing-Reviewing and Editing, HM: Visualization, Software, CL: Conceptualization, Methodology, Validation.

Conflict of Interest: XY was employed by Ansteel Company Limited Cold-Rolling Silicon Steel Mill.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

ORIGINAL RESEARCH
published: 01 October 2021
doi: 10.3389/fphy.2021.719012

A Study on the Attention of Yoga and Its Development Based on Complex Network Theory

Pengchao Li[1], Qinghong Miao[2]*, Yuchi Meng[3], Jie Ning[4], Jing Long[5] and Junya Huang[1]*

[1]School of Art, Beijing Sport University, Beijing, China, [2]Beijing Jingkong Technology Co., Ltd., Beijing, China, [3]Chinese Skating Association, Beijing, China, [4]School of Physical Education, Zhoukou Normal University, Zhoukou, China, [5]Zhongshan Experimental Middle School, Zhongshan, China

Taking Baidu search index as the data source, this research collects yoga-related data in various provinces in China, analyzes the public's attention to yoga on the Internet from the perspective of the complex network theory, so as to dig out characteristics of those who are interested in yoga as well as the temporal and spatial change of yoga attention from 2011 to 2020. Then, by transforming the time series into a network through the VG model and HVG model, the paper analyzes the network characteristics and predicts the popularity of yoga. Furthermore, the publicity of yoga and the public's attention to it are analyzed, considering the policy, national education level, the influence of TV, mobile phone and other communication equipment, so as to help the market to provide corresponding products and services in a targeted manner and to promote the healthy development of the yoga industry.

Keywords: complex network, social network, yoga, attention, visualization graph

Edited by:
Peican Zhu,
Northwestern Polytechnical University, China

Reviewed by:
Mingyang Zhou,
Shenzhen University, China
Wei Wang,
Sichuan University, China

****Correspondence:***
Qinghong Miao
miaoqh_1101@163.com
Junya Huang
junyahuang@sina.com

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *01 June 2021*
Accepted: *26 July 2021*
Published: *01 October 2021*

Citation:
Li P, Miao Q, Meng Y, Ning J, Long J and Huang J (2021) A Study on the Attention of Yoga and Its Development Based on Complex Network Theory. Front. Phys. 9:719012. doi: 10.3389/fphy.2021.719012

INTRODUCTION

In recent years, the government has been vigorously developing national fitness. The *"Healthy China 2030" Planning Outline* and the *"Opinions on Promoting National Fitness and Sports Consumption to Promote the High-quality Development of the Sports Industry"* issued by the Party Central Committee and the State Council both recognize the necessity of carrying out national fitness campaign, activating the fitness training market and building a better platform for the sports industry. Yoga is a physical, mental, and spiritual exercise with a history of more than 5,000 years and it represents "harmony" and "consistency". Its essence is to connect "self" and "superego," to transform the soul and to improve physical and mental health. Since the 19th century, modern yoga has been attracting groups of all ages rapidly with its functions, such as improving posture and self-cultivation. It is estimated that there were 2.5 million yoga practitioners in the United Kingdom and 15 million in the US in 2008 [1], and the number has been increasing rapidly in recent years.

At present, the number of researches on yoga at home and abroad has been increasing significantly year by year. Though its research content covers the history of yoga, the feasibility of introducing yoga and the efficacy of yoga, the majority of them are purely theoretical description and analysis based on transplantation of related theories and adoption of conceptual introduction and explanatory methods. These researches focus on theoretical explanations and abstract categories, featuring low-level repeated studies and weakness of absorbing new research methods. The relationship between yoga and human health is one of the important directions of domestic research. Wang Min (2005) conducted a 15 weeks experiment on female college students, showing that yoga practice can effectively improve their respiratory system, circulatory system skills, physical flexibility, balance ability, and mental health [2]. Wu Minyue (2010) reviewed scientific research papers on yoga and health at home and abroad, showing that long-term insistent practice of yoga has a positive influence on preventing and treating chronic diseases such as cardiovascular diseases and diabetes [3]. Over time, the medical field has recognized the efficacy of yoga and has been adopting it into the treatment and management of diseases, including chronic disease [4], depression [5–7], cancer [8, 9], and rehabilitation. Later, the study of yoga is no longer restricted to its exercising and health function and covers the development of the entire industry, causing an in-depth impact on yoga research. Huang Min (2010), Chen Xiaoying (2010), and some other researchers analyzed the development status of yoga-related industries from the perspective of yoga industrialization and marketization and put forward suggestions for the sustainable development of yoga industry [10, 11]. Yu Jingjing (2011), Liu Min (2013), and Zhang Maomao (2014) discussed the social background and value orientation of yoga and the building of yoga teaching teams in universities [12–14]. VG model is also used in stock and venture capital [15, 16].

Compared with domestic research, foreign research presents the characteristics of overlap, penetration, and integration of multidiscipline. It pays special attention to the adoption of empirical research methods such as randomized controlled trials, double-blind trial intervention, meta-analysis, follow-up, prospective studies, and nationwide surveys. The research content is systematic and target-orientated. For instance, Cramer H (2013) analyzed 12 randomized controlled trials (619 participants) to study the relationship between yoga and its influence on relieving depression and anxiety and improving the quality of life. He argues that yoga can be considered as one of the adjunctive treatment methods for patients with depression or individuals with elevated levels of depression [17]. Stemlieb B (2011) conducted a randomized controlled experimental intervention on breast cancer patients, discovering that targeted yoga intervention can significantly improve the persistent fatigue symptoms [18]. Bussing A (2012) believes that although yoga has not been proven to be an independent treating method due to some research conditions, the beneficial effects of yoga intervention on physical and mental health related to pain do exist. As adjunctive therapy, yoga can improve body function and self-confidence [19]. With the widespread concept of practicing yoga, more research has been conducted on the relations between yoga and the treatment of diseases including obesity [20, 21], chronic blood disease [22], mental health [23, 24], cancer [25, 26], and even COVID - 19 prevention [27, 28]. The increasing popularity of yoga can be partly attributed to the Internet and social media. Social media bloggers have been attracting more and more yoga practitioners. Chen et al. (2014) designed a yoga pose recognition model to help yoga practitioners to practice with more appropriate postures. The Internet also provides researchers a better platform to conduct their studies. K Firestone (2014), Johnson (2014) conducted online surveys to collect data [29, 30], which has some similarities to our study.

This paper breaks the boundaries of domestic literature, adopts search behavior data to measure public's attention on yoga, and attempts to investigate the time change trend of yoga attention in a specific time and space environment and the characteristics of the practitioner, so as to understand the public's need for yoga and its changing trend and analyze the evolution of yoga attention by establishing the time series prediction model of yoga attention degree, so as to propose suggestions on the future yoga development directions and strategies. The second part of the paper is a description of the methods used. The third part is a complex network-based analysis and visualization of the Baidu search index data of yoga from 2011 to 2020. The fourth part is the prediction of the popularity of yoga and relevant suggestions.

MODEL DESCRIPTION

Data Extraction and Processing

Attention, which refers to the degree to which a thing receives interests from social groups, is an important indicator of the strength of the internal relationship between things and group behavior, exerting an important influence on public opinion, culture, and policy. Attention can be reflected by different parameters, such as the number of searches, views, reposts, comments, and favorites of an event on the Internet. Based on Baidu's massive data, the Baidu index, on the one hand, analyzes the hotness of keywords and, on the other hand, explores in-depth the data characteristics of public opinion, market demand, and user characteristics. The Baidu index reflects the active search demands of internet users, and all activities that affect their search behavior may affect the Baidu index. Therefore, in order to show the public's attention to yoga and those who are interested in yoga, this study selects the visualization results of the Baidu index search as the data basis for analyzing domestic users' attention to yoga. The interpolation method is adopted to obtain equal interval data in case missing sampling or uneven sampling interval of the data occurs.

Modeling of Complex Networks

A complex network describes a system composed of a large number of interacting individuals. The analysis of a complex network can help describe the structure of the network system and understand the law of information evolution on the network, so as to finally realize the intervention and optimization of information evolution process on the network. This paper uses the time series visualization method proposed by Lacasa et al. [31] to construct the network, which consists of the Baidu search index time series subsystem with the keyword "yoga" in 31 provinces in China. First, the discrete time series data of the subsystem $X(t)$ corresponds to the nodes of the network, and the connection edges are constructed according to the visual criterion. The connection edge can be established by

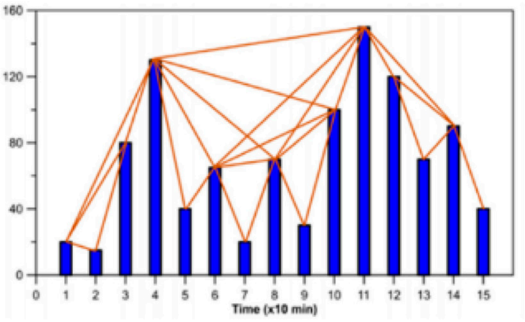


FIGURE 1 | The theory of time series data visualization.

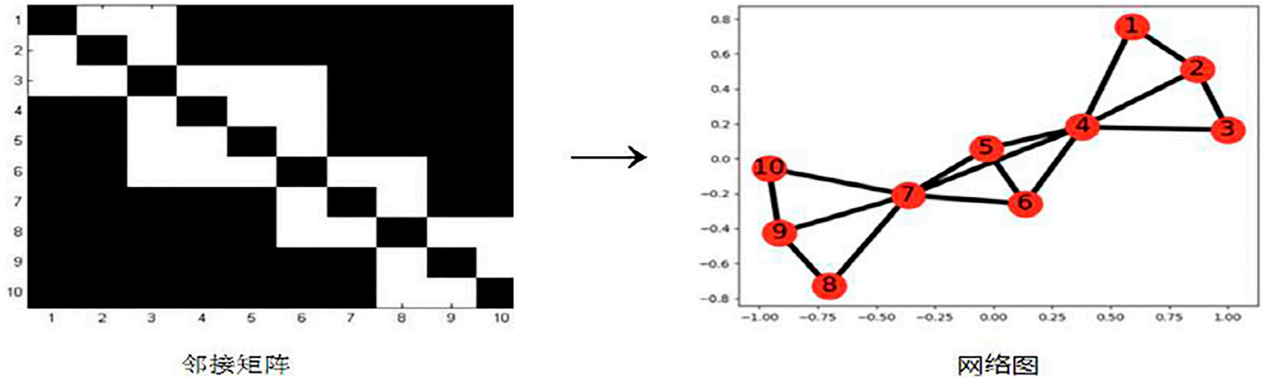


FIGURE 2 | Visual network construction process and characteristics extraction.

visualizing the data of any two points. In the time series $(X(t))$ any point between (t^a, x^a) and (t^c, x^c) can be linked, and when $t^a < t^b < t^c$ any point t^b, t^b between t^a, t^a and t^c, t^c, all satisfy:

As shown in **Figure 1**, the height of the bar represents the data value at each time point. If the tops of the 2 bars are visible to each other, the corresponding two points are connected by the network in the figure.

Secondly, to construct an adjacency matrix based on time series nodes and edges, and to form a network graph, as shown in **Figure 2**.

The important characteristics of the complex network are calculated as follows [32–34]:

(1) Degree and degree distribution.

The number of edges connected by a node is called the degree of the node, the node i is shown as k_i:

$$K_i = \sum_j a_{ij}$$

Among them, a_{ij} is the number of connecting edges between nodes i and j. In the network, the greater the degree, it means the more nodes are connected to it, then the greater influence of the node, and thus the stronger time correlation. The average degree is defined as follows:

$$K = \frac{1}{N} \sum_i k_i$$

Among them, N represents the number of nodes in the network.

(2) Average path length and diameter.

The path connecting two reachable nodes in the network with the least number of edges is called the distance between nodes, and the longest path between two nodes is called the diameter of the network d_{ij}. Both the average network path length and the network diameter can reflect the network transmission efficiency. The smaller the value, it means that the transmission effect can be achieved with fewer nodes in the network, thus the higher network efficiency. The calculation formula for the average network path length is as follows:

$$L = \frac{1}{\frac{1}{2}N(N-1)} \sum_{i \geq j} d_{ij}$$

(3) Clustering coefficient and clustering.

The clustering coefficient, which describes the degree of clustering of all nodes in the network, represents the tightness of the network. Its calculation formula is as follows:

$$C_i = \frac{2E_i}{K_i(K_i - 1)}$$

TABLE 1 | Descriptive statistics of Baidu Index of Yoga in 2020.

Province	Average			Maximum			Minimum			Standard error		
	Search index	PC	Mobile	Search index	PC	Mobile	Search index	PC	Mobile	Search index	PC	Mobile
Anhui	290.9559	49.1176	241.8382	441	74	367	136	0	136	85.3160	24.3436	70.2625
Macau	7.6176	0.0000	7.6176	60	0	60	0	0	0	19.6522	0.0000	19.6522
Beijing	437.1618	69.8676	367.2941	592	101	518	181	0	181	106.9644	18.3107	96.2235
Fujian	260.2794	52.3088	207.9706	350	78	287	144	0	123	57.8735	22.3395	46.2014
Gansu	175.6471	31.6029	144.0441	250	63	205	89	0	89	53.1994	29.1722	33.0266
Guangdong	704.0588	79.9412	624.1176	992	142	858	348	57	287	184.4647	16.2957	174.1870
Guangxi	238.5000	50.3971	188.1029	339	70	278	115	0	115	51.5927	22.4839	40.7710
Guizhou	163.8676	22.2941	141.5735	269	63	208	92	0	83	47.0779	28.5689	30.2188
Hainan	128.1618	26.5147	101.6471	191	63	139	67	0	67	37.8889	29.2026	18.7464
Hebei	501.1912	59.4706	441.7206	710	85	683	208	0	206	167.9120	19.9046	158.7823
Henan	491.6029	58.9559	432.6471	726	89	656	201	0	201	166.1601	19.4778	158.3369
Heilongjiang	274.8971	49.2647	225.6324	403	70	337	138	0	119	77.7641	24.3743	66.9417
Hubei	335.6912	56.5000	279.1912	527	76	453	128	0	128	104.3891	21.1861	93.0765
Hunan	281.3529	46.5000	234.8529	398	80	324	148	0	148	74.3916	27.3777	59.5925
Jilin	225.8235	43.5147	182.3088	333	74	267	92	0	92	62.9944	27.4783	44.9006
Jiangsu	468.7059	65.9118	402.7941	667	104	591	232	0	201	125.1420	21.4293	111.7364
Jiangxi	247.9412	47.8235	200.1176	387	78	326	109	0	109	68.6181	24.7844	54.2321
Liaoning	345.2206	59.3088	285.9118	479	80	419	157	0	157	87.5737	15.8060	80.3566
Inner mongolia	216.2794	41.1618	175.1176	337	66	278	95	0	95	67.1258	27.8233	49.5706
Ningxia	97.8235	11.0147	86.8088	176	61	121	63	0	63	30.6876	22.8307	15.5604
Qinghai	85.3529	9.4559	75.8971	153	61	100	57	0	57	26.4551	21.6963	10.4869
Shandong	636.7206	69.0000	567.7206	910	110	825	312	0	255	196.8528	15.6205	188.8028
Shanxi	286.9412	50.5000	236.4412	395	83	338	125	0	125	82.4659	23.9574	69.7311
Shaanxi	295.4706	58.4559	237.0147	401	70	337	125	0	119	72.0341	13.1324	65.0191
Shanghai	288.7794	63.2794	225.5000	374	104	313	122	0	122	51.5004	15.9382	42.4276
Sichuan	410.0882	59.6029	350.4853	587	80	521	186	0	186	120.6645	17.8494	110.3189
Taiwan	16.0147	3.3529	12.6618	115	57	58	0	0	0	29.4454	13.5115	23.9786
Tianjin	197.6324	40.4118	157.2206	282	70	219	86	0	86	52.4248	28.2691	33.5936
Tibet	63.1324	2.5735	60.5588	126	61	78	0	0	0	17.3609	12.0748	15.9658
Hongkong	53.5441	5.0882	48.4559	120	59	69	0	0	0	30.9542	16.4804	23.8924
Xinjiang	189.0441	36.1912	152.8529	269	64	212	99	0	99	51.2443	28.7341	35.3428
Yunnan	199.7059	31.1765	168.5294	291	66	227	109	0	109	53.6976	29.6668	34.0910
Zhejiang	403.8824	64.1176	339.7647	537	87	469	200	0	179	97.2186	19.8709	86.1114
Chongqing	223.0882	44.2353	178.8529	307	78	249	102	0	102	60.4621	26.9400	42.2273

Among them, C_i represents the clustering coefficient of node i, K_i represents the number of nodes connected to node i, and E_i represents the actual number of edges between nodes in the network. It can be deduced that the average clustering coefficient C of the network is as follows:

$$C = \frac{1}{N}\sum_{i=1}^{N} C_i$$

The value of the clustering coefficient is generally between 0 and 1. The larger the coefficient, the better the connectivity of the network and the greater the degree of network aggregation. In general, some real-world network clustering coefficients are larger than random network clustering coefficients of the same size, which means that real networks have better clustering tendency, which features the nature of clustering.

(4) Density.

The density of the network is equal to the ratio of the actual number of edges in the network to the maximum possible number of edges in the network. The greater the density, the more the number of edges in the network and the denser the network.

EMPIRICAL ANALYSIS

We analyze the time series data of the Baidu index for 31 provinces and cities from 2011 to 2020. The descriptive statistics, visualization of time series data, and complex network method are also applied to the analysis to find out the characteristics of the visualized networks.

Descriptive Statistics

As showed in **Table 1**, based on the search result of “yoga” in the Baidu index, Guangdong, Shandong, Hebei, Henan, and Jiangsu are the top 5 provinces that pay the most attention to yoga. These five are all provinces with large population and economy. But we noticed that the standard error of the index in these provinces is in the top position too. It is found that the data of these provinces are more prone to fluctuations due to Internet searches. **Figure 3** shows the variation tendency of the Baidu index of the top 5 and the bottom 3 provinces in 2020. The increase of the Baidu index in 2021 is, on the one hand, due to the COVID-19 pandemic, which brings exercises and health issues to the public’s attention

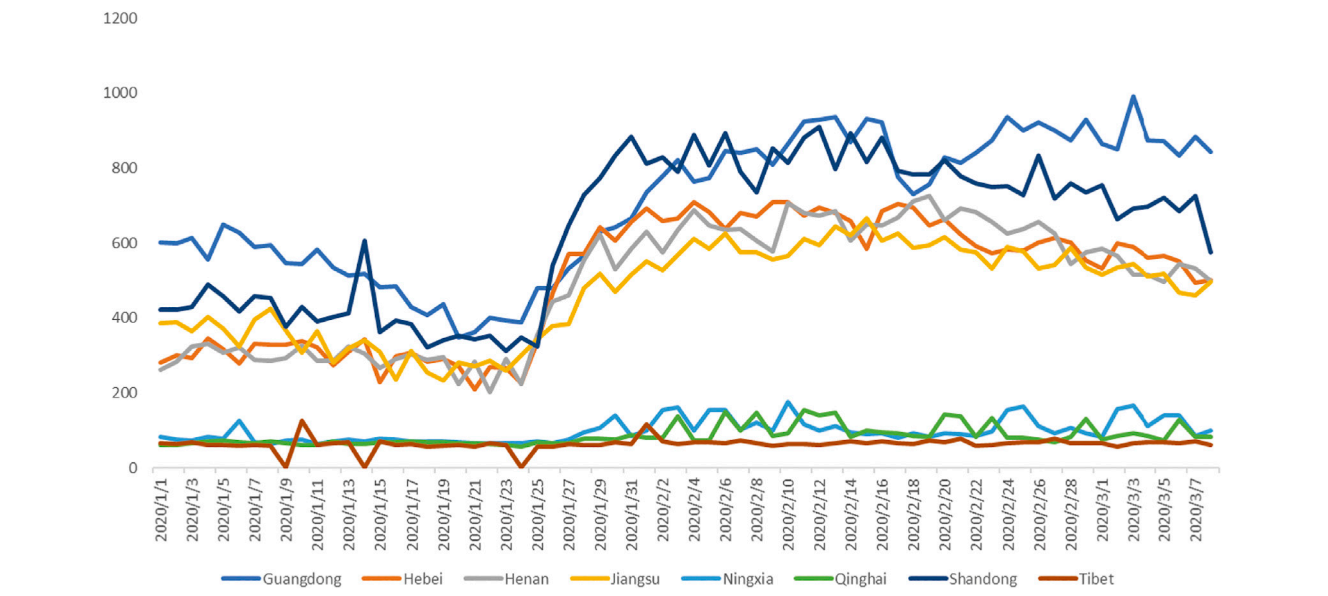


FIGURE 3 | Top 5 and last 3 provinces by the Baidu index of yoga in 2020.

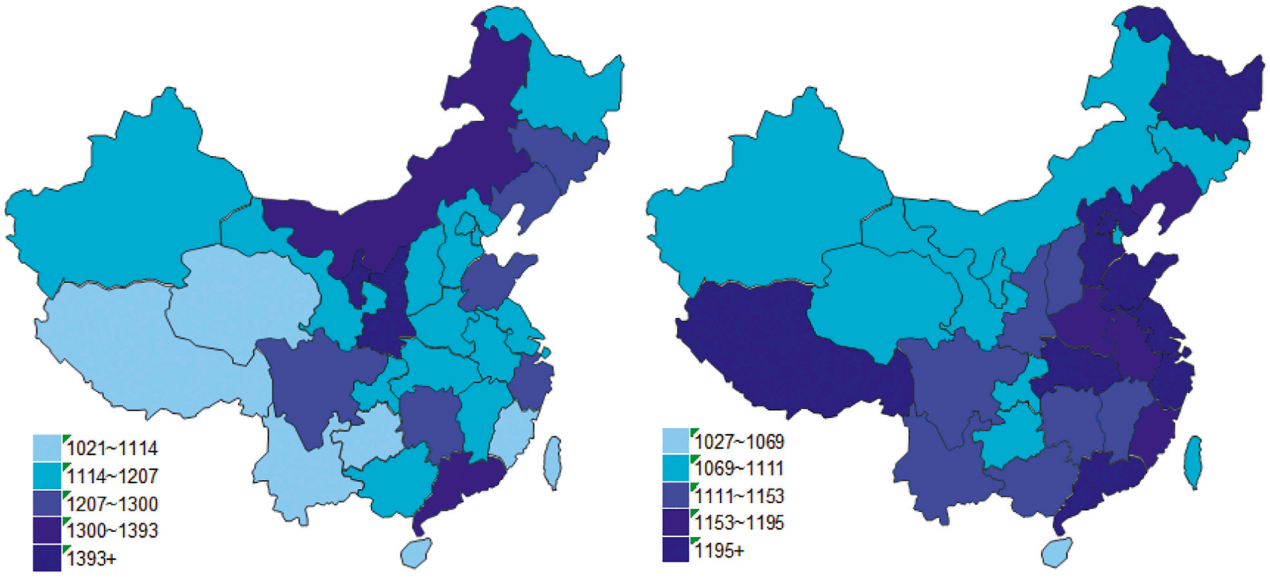


FIGURE 4 | The comparison of the total number of edges between 2011 and 2019.

and, on the other hand, benefited from the improvement of network infrastructure construction.

In addition, with the rapid development of the mobile phone and information technology, the mobile phone starts to become the main channel for information because it is more convenient and easy operating especially when doing exercise following the video on the Internet.

Characteristic Evolution

Calculating the characteristic of visualized networks, the information is found out including the total number of edges, average degree, average path, diameter, clustering coefficient, and the annual results of density. Given space limitation, it only shows the comparison chart of 2011 and 2019 in **Figures 4–9**.

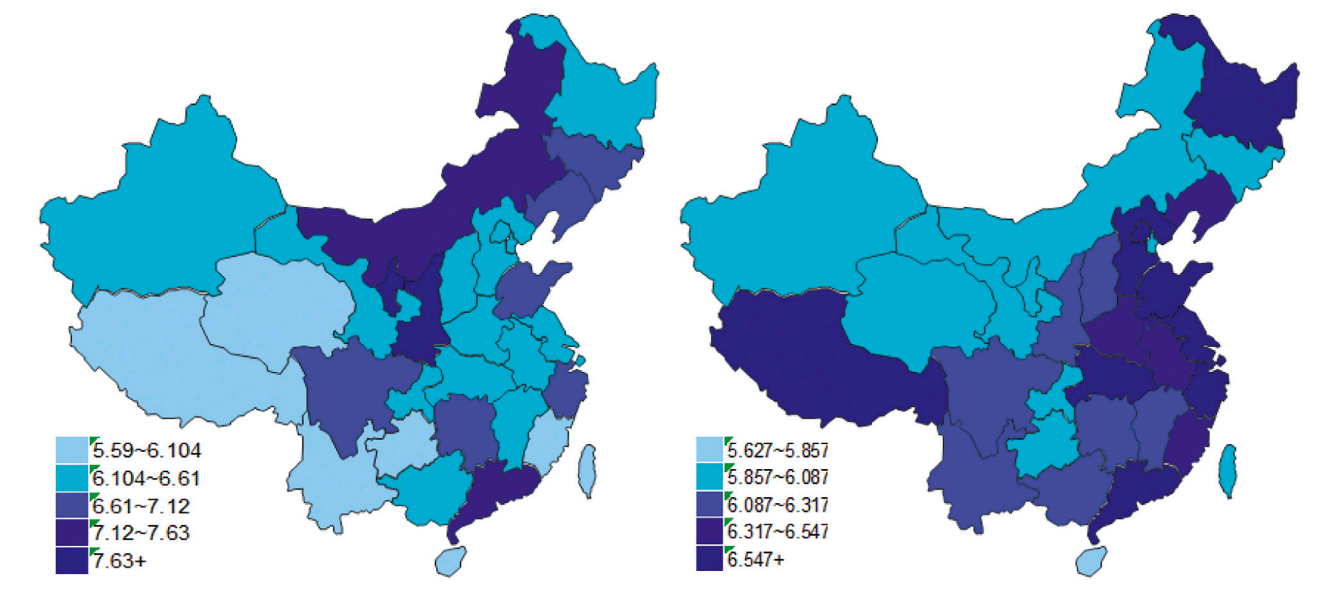


FIGURE 5 | The comparison of average network degree between 2011 and 2019.

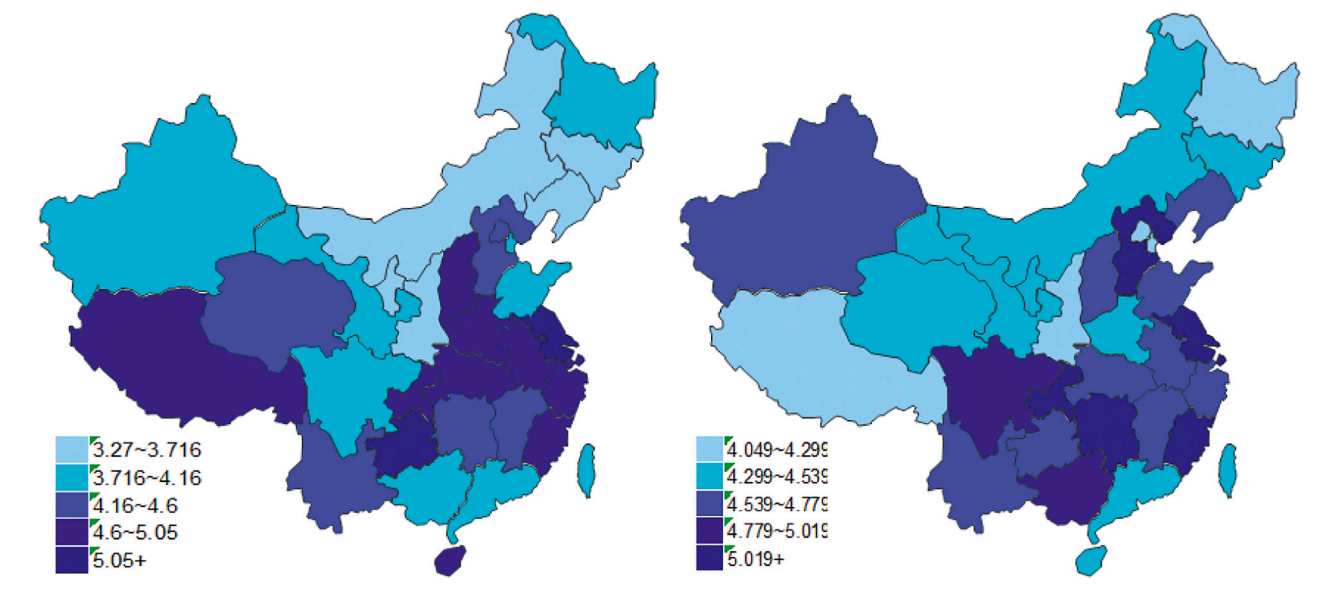


FIGURE 6 | The comparison of the average network path between 2011 and 2019.

Annual Change Characteristics of Yoga Attention

1) Geographical distribution characteristics of yoga attention

According to the analysis of relevant charts in 2011, we can find that most of the provinces and cities which pay great attention to yoga fall into the area of ethnic minority settlements or its neighboring provinces and cities, while the central and southwestern regions paid relatively little attention to yoga. Since 2012, the peaks of attention appeared in Anhui, Hubei, Tibet, Yunnan, Guangxi, Shanxi, Hunan, and other places. Among them, Tibetans pay the greatest attention to yoga in China. In recent years, the attention to yoga in coastal areas has been rising.

We found that the degree of attention to yoga changes over time segments. The local people may search "yoga" a lot at a certain point in time, causing the increase of the index, while after the peak arrives, a sharp decline may occur because the exercise habit of the local people has not been cultivated due to exercises conditions or time limits. At the same time, first-tier cities and

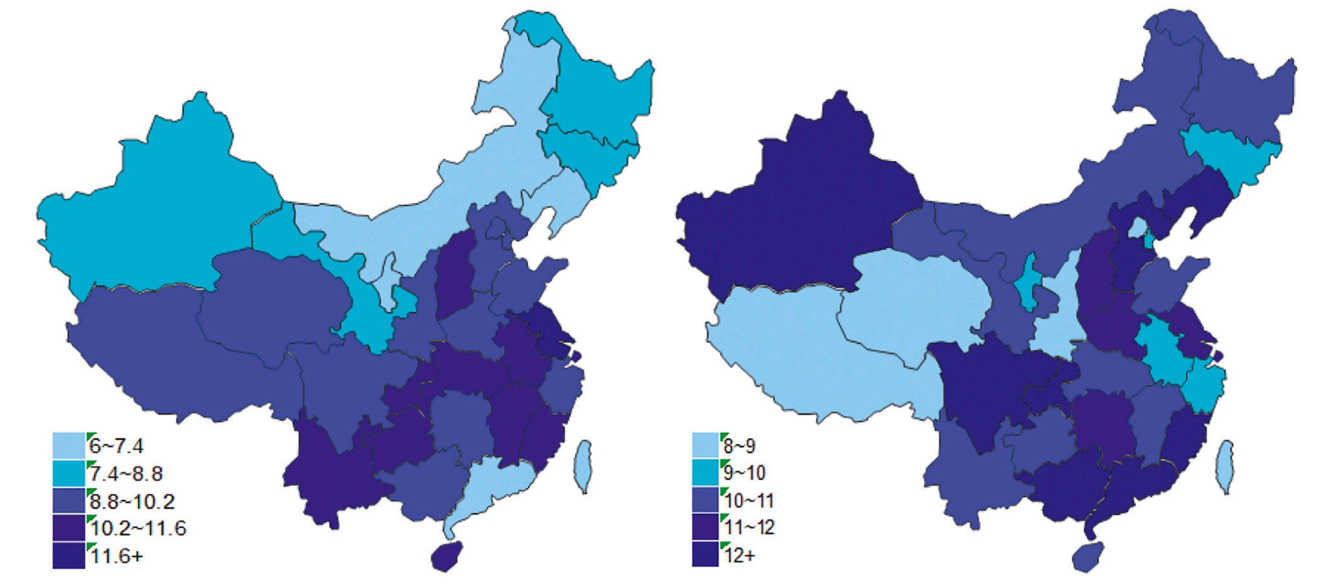


FIGURE 7 | The comparison of average network diameter between 2011 and 2019.

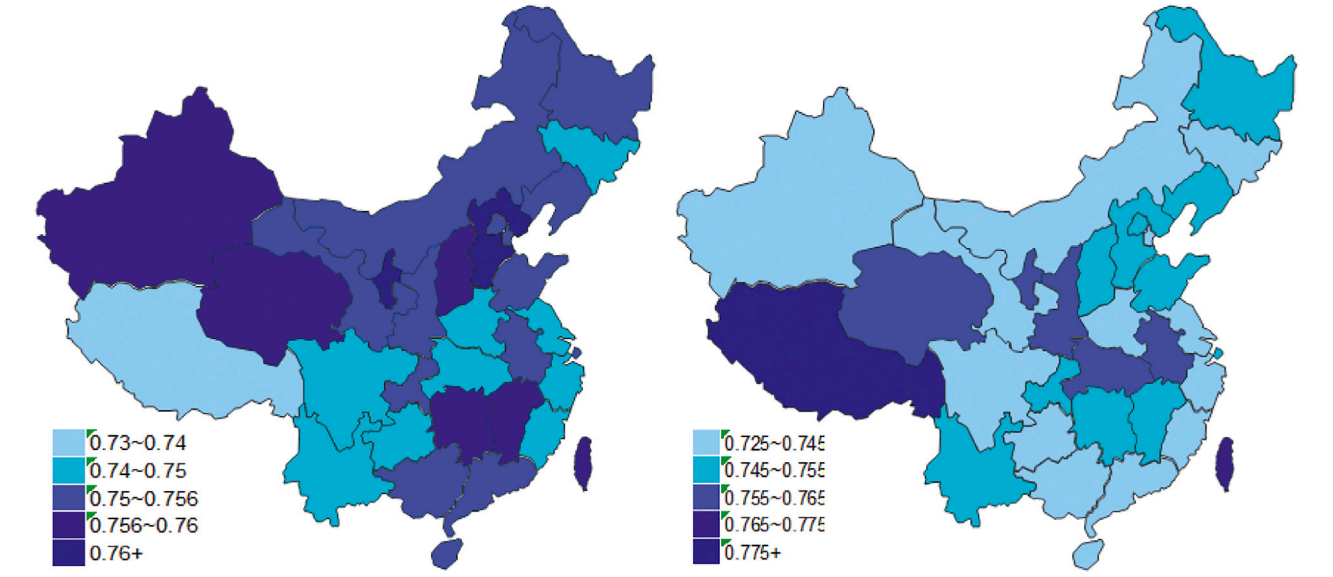


FIGURE 8 | The comparison of clustering coefficient between 2011 and 2019.

provinces pay less attention to yoga and among them, Shanghai pays comparatively great attention.

2) Temporal characteristics of yoga attention

Compared with 2011, it is found that the attention to yoga in most provinces has increased in recent years due to the development of yoga industry. In particular, with the development of the Internet, the retention time of information is longer, and the influence of each point of time on the others and the correlation among them are more prominent. Collectively, it is mainly reflected in the following aspects:

①The network average degree has improved. That is, as time changes, the average influence of each node increases, and its time correlation is stronger.

②The total number of edges is relatively reduced. The reduction in the number of edges indicates that the visible amount of data at the top of the data bar is reduced, which indicates that the peak and bottom of the region's attention that year frequently occurred, meaning greater fluctuations.

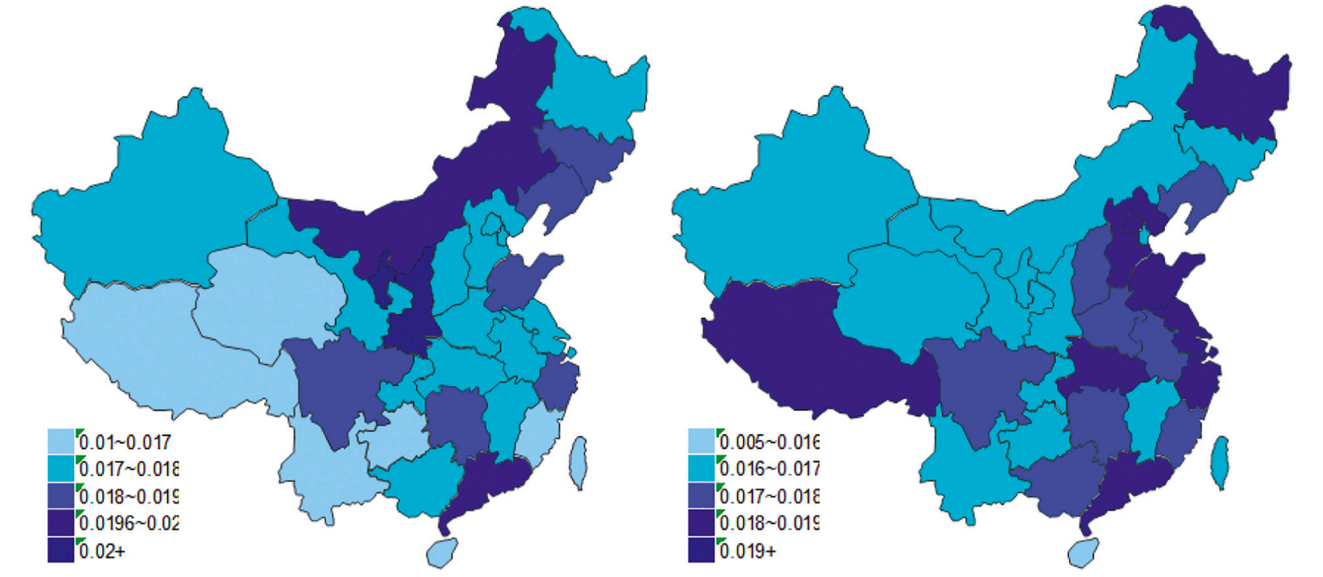


FIGURE 9 | The comparison of network density between 2011 and 2019.

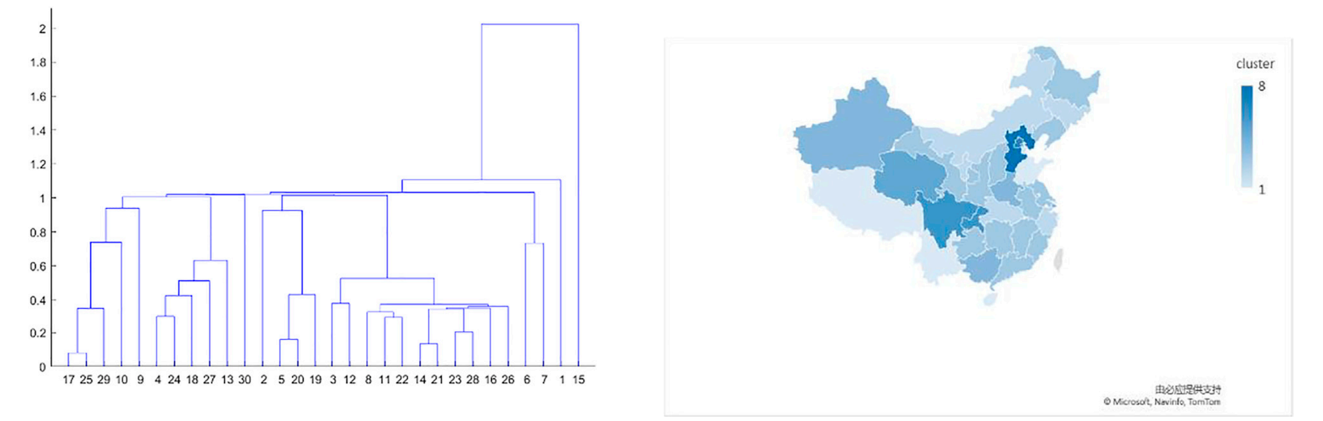


FIGURE 10 | Systematic clustering.

③The average path and diameter of the network are longer and have been relatively stable in recent years. This shows that network efficiency declines with the passage of time and the rise of emerging technologies.

Relevant information can also be found from the existing literature. With the development of yoga, people start to analyze yoga from various aspects such as theoretical training, method learning, and practical skills. Before 2011, yoga literature focused on theoretical studies, including the fitness effects of yoga and physical education research, the comparison of traditional sports and culture between China and India, the study of religious Buddhism and the interpretation of philosophical thoughts, which did not arouse the public's interest, hence attracted little attention from the public. Nowadays, the promotion of yoga focuses on people's daily life, such as soothing emotions, improving posture, which is the pursuit of people of all ages. News from all aspects attracts more public attention and the development of communication technology speeds up the spread of news in a short period of time, which has led to an increase in the frequency of related searches.

The diversified development of yoga presents different fashion trends at different times. The gimmick marketing could bring huge attention to yoga in a short period of time, but the attention will decrease quickly because the content of related articles is either exaggerated or unprofessional, resulting in a relative decrease in the total number of edges, which directly affects the average path and diameter of the network.

In this paper, we analyze the cluster data of 31 provinces in China, as shown in **Figure 10**.

It is found that Hebei, Beijing, and Qinghai are the same cluster; Sichuan and Chongqing are the same cluster; Shanghai, Henan, Xinjiang, and Guangxi are the same cluster; Guangdong, Jiangsu, Fujian, Heilongjiang, Shaanxi, Liaoning, Shanxi, Hunan, Jiangxi, Anhui, Gansu, and Guizhou are the same cluster; Hubei, Zhejiang, Jilin, Neimenggu, and Ningxia are the same cluster; Tianjin, Shandong, Yunnan, Hainan, and Xizang are the same cluster.

CONCLUSION

Using the complex network to visualize the Baidu search index, we can clearly see the distribution of yoga attention in various provinces and cities and the changes that have occurred in recent years. The development of the network environment has provided a good carrier for the spread of yoga. The development of various public accounts and APPs has made the barriers for yoga practitioners continue to shrink, while the targeted group continues to expand, and the attention of yoga continues to rise. But at the same time, it has also led to the spread of distorted information and the decrease of public's interest in yoga. More importance shall be attached to the implementation of basic and popular tutorials so as to boost the development of yoga in China. The public should also choose wisely and get information from relatively authoritative content. All social organizations such as schools, yoga education institutions, and the local government should formulate the corresponding standards and system for the healthy development of yoga. Firstly, different standards shall be formulated based on the characteristic of different age groups and provide different tutorials and guidance methods for teenagers, middle-aged, and the elderly so as to help them practice yoga reasonably. The second is to promote yoga among the targeted group with practical content and tell them about the physical and mental health effects of yoga. Though the theoretical interpretation of yoga culture is of no interest to the public, its positive effect on various diseases and physical and mental health is obvious for them to understand. The positive function of yoga shall be publicized to attract more targeted groups. Thirdly, the government should strengthen guidance and supervise relevant online content to avoid distorted and false information. At the same time, the government should encourage the adoption of new social media forms such as short video platforms to facilitate the production of online information in a healthy and upward way. It is also seen from the study of the geographical characteristic of yoga attention that the yoga attention has not been raised to a higher level because all provinces are promoting yoga in their own local areas with little cooperation, causing development barriers due to region limitations, which does not contribute to the development of yoga in China as a whole. All kinds of activities such as offline exercise can be organized to help expand the influence of yoga, giving full play to the advantages of different provinces to raise the attention of yoga.

Visualizing time series data is more intuitive and convenient to analyze the changing trend of data and explore the deep laws and characteristics of the development of the industry. This paper applies the time series visualization method to the study of yoga attention, which helps to grasp the current situation of yoga sports development and propose the policy to guide the future development direction. This study provides a paradigm and sample for the application of time series visualization methods in the field of social science research for further advancement of other analogous studies.

DATA AVAILABILITY STATEMENT

The datasets presented in this article are not readily available. Requests to access the datasets should be directed to https://index.baidu.com/v2/index.html#/ and the corresponding authors.

AUTHOR CONTRIBUTIONS

PL: conceptualization, methodology, software, validation, drawing, and writing; QM: visualization, software; YM: computation, drawing, and writing; JN: visualization, software; JL: investigation; JH: writing—reviewing and editing.

REFERENCES

1. Singleton M. Suggestive Therapeutics: New Thought's Relationship to Modern Yoga. *Asian Med Tradition Modernity* (2017) 3(1):64–84. doi:10.1163/157342107x207218
2. Wang M. The Study on the Influence of Yoga Body Training on Female College Students' Physical and Mental Health. *J Guangzhou Inst Phys Educ* (2005) 25(2):84–7. doi:10.4028/www.scientific.net/amr.187.164
3. Satyapriya M, Nagendra HR, Nagarathna R, and Padmalatha V. Effect of Integrated Yoga on Stress and Heart Rate Variability in Pregnant Women. *Int J Gynecol Obstet* (2009) 104(3):218–22. doi:10.1016/j.ijgo.2008.11.013
4. Qi J, Zhang Y, and Du Q. Analysis of the Biomechanical Mechanism of Yoga in the Rehabilitation of Patients with Chronic Low Back Pain [J]. *Chin J Med Phys* (2017) 34(7):748–52. doi:10.3969/j.issn.1005-202X.2017.07.020
5. He J, Wang G, and Wang H. Meta-analysis of Yoga Intervention for Major Depressive Disorder. *J Int Psychiatry* (2018) 56(3):432–6. doi:10.1016/j.jad.2017.02.006
6. Ma K, Liu J, and Fu C. Research Progress on the Interventional Effect and Mechanism of Exercise on Depression. *China Sports Sci Technol* (2020) 56(No.11):12–24. doi:10.16470/j.csst.2020132
7. Han Y, Liu X, and Xiang H. Clinical Progress of Exercise Therapy for Depression. *[J]Sports Res Educ* (2020) 35(5):85–90. doi:10.16207/j.cnki.2095-235x.2020.05.016
8. Pan R, Xu L, and Yao X. Effect of Yoga Breathing on Perioperative Psychology of Lung Cancer Patients. *[J]Chinese Gen Pract Nurs* (2017) 15(2):2610–1.
9. Li Y, Li Y, and Zeng F. Study on the Application of Yoga Breathing Exercise in Lung Rehabilitation of Patients after Lung Cancer Operation. *Nurs Integrated Traditional Chin West Med* (2020) 6(10):265–9.
10. Huang M. Research on the Development of Yoga Industry. *J Guide Sports Culture* (2010) 9:79–81. doi:10.3969/j.issn.1671-1572.2010.09.021

11. Raghuraj P, and Telles S. Immediate Effect of Specific Nostril Manipulating Yoga Breathing Practices on Autonomic and Respiratory Variables. *Appl Psychophysiol Biofeedback* (2008) 33(2):65–75. doi:10.1007/s10484-008-9055-0
12. Telles S, Naveen KV, and Dash M. Yoga Reduces Symptoms of Distress in Tsunami Survivors in the Andaman Islands [J]. *Evid Based Complement Altern Med* (2007) 4:503. doi:10.1093/ecam/nem069
13. McCaffrey R, Ruknui P, Hatthakit U, and Kasetsomboon P. The Effects of Yoga on Hypertensive Persons in Thailand. *Holist Nurs Pract* (2005) 19(4): 173–80. doi:10.1097/00004650-200507000-00009
14. Zhang Y-J, Meng K, Gao T, Song Y-Q, Hu J, and Ti E-P. Analysis of Attention on Venture Capital: A Method of Complex Network on Time Series. *Int J Mod Phys B* (2020) 34(29):2050273. doi:10.1142/S0217979220502732
15. Cui X, Hu J, Ma Y, Wu P, Zhu P, and Li H-J. Investigation of Stock price Network Based on Time Series Analysis and Complex Network. *Int J Mod Phys B* (2021) 35:2150171. doi:10.1142/S021797922150171X
16. Zhang Y-J, Meng K, Gao T, Song Y-Q, Hu J, and Ti E-P. Analysis of Attention on Venture Capital: A Method of Complex Network on Time Series. *Int J Mod Phys B* (2020) 34(29):2050273. doi:10.1142/s0217979220502732
17. Cramer H, Lanche R, and Haller H. A Systematic Review and Meta-Analysis of Yoga for Low Back Pain [J]. *Clin J Pain* (2013) 29(5):450–60. doi:10.1142/s0217979220502732
18. Bower JE, Garet D, Sternlieb B, Ganz PA, Irwin MR, Olmstead R, et al. Yoga for Persistent Fatigue in Breast Cancer Survivors. *Cancer* (2011) 118(15):3766–75. doi:10.1002/cncr.26702
19. Bussing A, Michalsen A, and Khalsa SB. Effects of Yoga on Mental and Physical Health: A Short Summary of Review [J]. *Evidence-Based Complement Altern Med* (2012) 2012:165410. doi:10.1155/2012/165410
20. Alaguraja K, and Yoga P. Influence of Yogasana Practice on Flexibility Among Obese Adolescent School Boys. *Int J Yoga, Physiother Phys Educ* (2017) 2:70–1. doi:10.1155/2012/165410
21. Alaguraja K, and Yoga P. Impact of Yogic Package on Body Mass index Among Obese People. *Int J Phys Educ Exerc Sports* (2019) 1:4–6.
22. Alaguraja K, and Yoga P. Combination of Naturopathy and Yoga on VO2 Max Among Hypertensive Patient. *Indian J Public Health Res Dev* (2020) 11:4.
23. Danhauer SC, Addington EL, Sohl SJ, Chaoul A, and Cohen L. Review of Yoga Therapy during Cancer Treatment. *Support Care Cancer* (2017) 25:1357–72. doi:10.1007/s00520-016-3556-9
24. Cramer H, Lauche R, Klose P, Lange S, Langhorst J, and Dobos GJ. Yoga for Improving Health-Related Quality of Life, Mental Health and Cancer-Related Symptoms in Women Diagnosed with Breast Cancer. *Cochrane Database Syst Rev* (2017) 1:CD010802. doi:10.1002/14651858.CD010802.pub2
25. Park CL, Finkelstein-Fox L, Groessl EJ, Elwy AR, and Lee SY, Exploring How Different Types of Yoga Change Psychological Resources and Emotional Well-Being across a Single Session. *Complement therapies Med* (2020) 49:102354. doi:10.1016/j.ctim.2020.102354
26. Novaes MM, Palhano-Fontes F, Onias H, Andrade KC, Lobão-Soares B, Arruda-Sanchez T, et al. Effects of Yoga Respiratory Practice (Bhastrika Pranayama) on Anxiety, Affect, and Brain Functional Connectivity and Activity: A Randomized Controlled Trial. *Front Psychiatry* (2020) 11:467. doi:10.3389/fpsyt.2020.00467
27. Sharma N. The Yoga for Physical and Mental Health - Can Possibly Aid in Prevention and Management of COVID19 Infection? *Dev Sanskriti Interdis Internat J* (2020) 16:22–31. doi:10.36018/dsiij.v16i.161
28. Zhu P, Wang X, Li S, Guo Y, and Wang Z. Investigation of Epidemic Spreading Process on Multiplex Networks by Incorporating Fatal Properties. *Appl Math Comput* (2019) 359:512–24. doi:10.1016/j.amc.2019.02.049
29. Firestone K, Carson J, Mist S, Carson K, and Jones K. Interest in Yoga Among Fibromyalgia Patients: An International Internet Survey. *Int J yoga Ther* (2014) 24:117–24. doi:10.17761/ijyt.24.1.p8l0750h1r1832q2
30. Johnson CC, Taylor AG, Anderson JG, Jones RA, and Whaley DE. Feasibility and Acceptability of an Internet-Based, African Dance-Modified Yoga Program for African-American Women with or at Risk for Metabolic Syndrome. *J Yoga Phys Ther* (2014) 4:1000174. doi:10.4172/2157-7595.1000174
31. Hu J, Xia C, Li H, Zhu P, and Xiong W. Properties and Structural Analyses of USA's Regional Electricity Market: A Visibility Graph Network Approach. *Appl Math Comput* (2020) 385:125434. doi:10.1016/j.amc.2020.125434
32. Chen X, Gong K, Wang R, Cai S, and Wang W. Effects of Heterogeneous Self-protection Awareness on Resource-Epidemic Coevolution Dynamics. *Appl Math Comput* (2020) 385:125428. doi:10.1016/j.amc.2020.125428
33. Pan L, Yang D, Wang W, Cai S, Zhou T, and Lai YC. Phase Diagrams of Interacting Spreading Dynamics in Complex Networks [J]. *Phys Rev Res* (2020) 2(2):023233. doi:10.1103/physrevresearch.2.023233
34. Pan L, Wang W, Cai S, and Zhou T. Optimal Interlayer Structure for Promoting Spreading of the Susceptible-Infected-Susceptible Model in Two-Layer Networks [J]. *Phys Rev E* (2019) 100:022316. doi:10.1103/physreve.100.022316

Conflict of Interest: QM was employed by Beijing Jingkong Technology Co., Ltd.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers in Physics

ORIGINAL RESEARCH
published: 04 October 2021
doi: 10.3389/fphy.2021.746660

Analyzing Levels of Concern About Joint Punishment for Dishonesty Using the Visibility Graph Network

Zhiqiang Qu[1], *Yujie Zhang*[2]* and *Fan Li*[3]

[1]*School of Law, Central University of Finance and Economics, Beijing, China,* [2]*School of Finance, Shanxi University of Finance and Economics, Taiyuan, China,* [3]*School of Public Administration and Policy, Renmin University of China, Beijing, China*

Joint punishment for dishonesty is an important means of administrative regulation. This research analyzed the dynamic characteristics of time series data from the Baidu search index using the keywords "joint punishment for dishonesty" based on a visibility graph network. Applying a visibility graph algorithm, time series data from the Baidu Index was transformed into complex networks, with parameters calculated to analyze the topological structure. Results showed differences in the use of joint punishment for dishonesty in certain provinces by calculating the parameters of the time series network from January 1, 2020 to May 27, 2021; it was also shown that most of the networks were scale-free. Finally, the results of K-means clustering showed that the 31 provinces (excluding Hong Kong, Macao and Taiwan) can be divided into four types. Meanwhile, by analyzing the national Baidu Index data from 2020 to May 2021, the period of the time series data and the influence range of the central node were found.

Keywords: joint punishment for dishonesty, visibility network, Baidu index, social credit system, time series network analysis

OPEN ACCESS

Edited by:
Peican Zhu,
Northwestern Polytechnical University, China

Reviewed by:
Manman Yuan,
University of Science and Technology Beijing, China
Xiaopeng Li,
Tianjin University of Technology, China

****Correspondence:***
Yujie Zhang
zhangyjaye@163.com

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *24 July 2021*
Accepted: *13 September 2021*
Published: *04 October 2021*

Citation:
Qu ZQ, Zhang YJ and Li F (2021) Analyzing Levels of Concern About Joint Punishment for Dishonesty Using the Visibility Graph Network. Front. Phys. 9:746660. doi: 10.3389/fphy.2021.746660

INTRODUCTION

Credit tools play a key role in the context of big data as a means of government regulation and must follow the principle of the rule of law. Since the 18th CPC National Congress, central government has put forward a series of new requirements for the construction of social credit, which have resulted in a new emphasis on the social credit system. In 2014, the State Council issued the "Planning Outline for the Construction of a Social Credit System (2014–2020)"[1], which proposed to build a credit reference system covering all the population by 2020. In the same year, the State Council's "Government Work Report"[2] set out a clear intention "to establish a blacklist system for enterprises that violate the principle of market competition and infringe on the rights and interests of consumers, so as to make it difficult for those who break faith." In 2017, Shanghai took the lead in issuing social credit regulations, which established the social credit management system in the form of local laws and legislation. However, credit regulation is based on credit evaluation. Credit is not a legal concept in the strictest sense but is a product of China's social construction process. In recent years, the joint punishment mechanism based on the social credit system has become common, although the concept of credit is not clear, which results in the generalization of punishment for dishonesty. The newly revised "Civil Servant Law" stipulates that those "listed as the Joint Disciplinary object of dishonesty according to law" shall not be

[1] http://www.gov.cn/xinwen/2014-06/27/content_2708964.htm

[2] http://www.gov.cn/guowuyuan/2014-03/14/content_2638989.htm

employed as civil servants. Although the joint disciplinary mechanisms of credit have a profound impact on the rights of citizens in practice, the meaning of this revision is not clear.

This paper uses the Baidu Index as the research object with the keyword "joint punishment for dishonesty." Current research exists on ways of monitoring using the browser search index, for example, on the detection and prediction of diseases [1–7]. As the largest search engine in China, Baidu has more than 80 per cent of the market share[3]. At present, there are many applications of Baidu index analysis in China, such as forecasting the number of tourists[4] [8, 9] and stock market prices [10–12], and of more recent significance, estimating the prevalence of influenza and other diseases [13, 14], predicting the incidence of Hand, Foot and Mouth Disease (HFMD) in real time [15–17], and monitoring the AIDS epidemic [18].

Related literature maps the time series analysis of complex networks [19–21]. For example, a specific period is extracted from a non-cyclical time series to use as a node. For a pair of nodes, the shortest one moves along the other, and the strongest correlation is the coupling strength between the two nodes. If the coupling strength is greater than the threshold, the two nodes are connected [22–24]. There are several variations of this method. For example, all possible segments with a specified length can simply be seen as nodes [25–29] and then each node linked to its nearest neighbor of the same length [30]. The network graph generated is embedded in the two-dimensional space in the pane filter, and the correlation is strong and the relationship maintained as much as possible [31]. One exciting task is to deconstruct the initial sequence into components through multi-resolution analysis to use as nodes [32]. In addition, scholars explored non-liner and uncertain complex valued networks [33, 34]. In the empirical study, scholars have also tried to analyze United States regional power market, search index and stock price by visualization graph method for time series data [35–37].

In recent years, punishment for dishonesty has increased along with more social awareness about dishonesty. Therefore, the study of the fluctuation characteristics and influence mechanisms of the Baidu Index using the keywords "joint punishment for dishonesty" can assist state organs to formulate more effective measures and policies, to improve people's awareness of the regulations and their rights, and to build a society ruled by law.

THE SYSTEM CONCEPT OF JOINT DISCIPLINARY MEASURES FOR DISHONESTY

Social credit is understood as a tool to protect market economy transactions in a form of market credit. Since 2011, social credit and social credit tools have gradually become important innovative means of social governance and have been incorporated into the government legislative plan, which is an important measure of the system's socialist core values.

The System Concept of Joint Disciplinary Measures for Dishonesty from a Functional Perspective

To a certain extent, China's urbanization can be seen as the process of transformation from acquaintance society to stranger society. Credit in acquaintance society is based on personality, and in particular, on moral constraints. In stranger society, the information asymmetry between the two sides of the social market transaction requires a third-party credit guarantee, which is the same for third-party institutions, e.g., certification and accreditation. From a national governance perspective, China first proposed social credit to protect economic transactions, focusing on financial credit. Therefore, a social credit tool results in the spontaneous formation of a social market economy from the outset, which ensures fairness and symmetry of market transaction information. Currently, credit is a governance tool, focusing on the security of economic transactions, market expansion, transaction costs and other information.

With the rapid development of the economy and the transformation of society, credit as a governance tool is not limited to ensuring secure economic transactions; it has been extended to the social public domain, becoming a valuable tool for social governance. For example, in the State Council's "Planning Outline for the Construction of a Social Credit System (2014–2020)"[5], it is clearly stated that this system is an important part of both the socialist market economic system and social governance system. The credit tool is a means of social management.

Joint Disciplinary Measures for Dishonesty as a Means of Government Regulation

Regulation refers to the restriction of the activities of individuals and economic subjects in a specific society according to certain rules. As a means of government regulation, a credit tool is also important for improving market failure and enhancing social governance. In terms of the system setting, disciplinary measures for dishonesty are an important way to improve the socialist market economy, solve market failure and ease the information asymmetry between the transaction subjects. According to the subject classification, regulation can be divided into public and private. In the current joint punishment of dishonesty, such as Ant's credit system and that of other private organizations, credit is not included in the joint punishment system. Therefore, the joint disciplinary measures for dishonesty can be understood generally to be a regulatory tool for the collection, evaluation, classification,

[3] http://www.cnnic.net.cn/hlwfzyj/hlwxzbg/ssbg/201401/P020140127366465515288.pdf

[4] https://www.chinainternetwatch.com/17415/search-engine-2012-2018e/

[5] http://www.gov.cn/xinwen/2014-06/27/content_2708964.htm

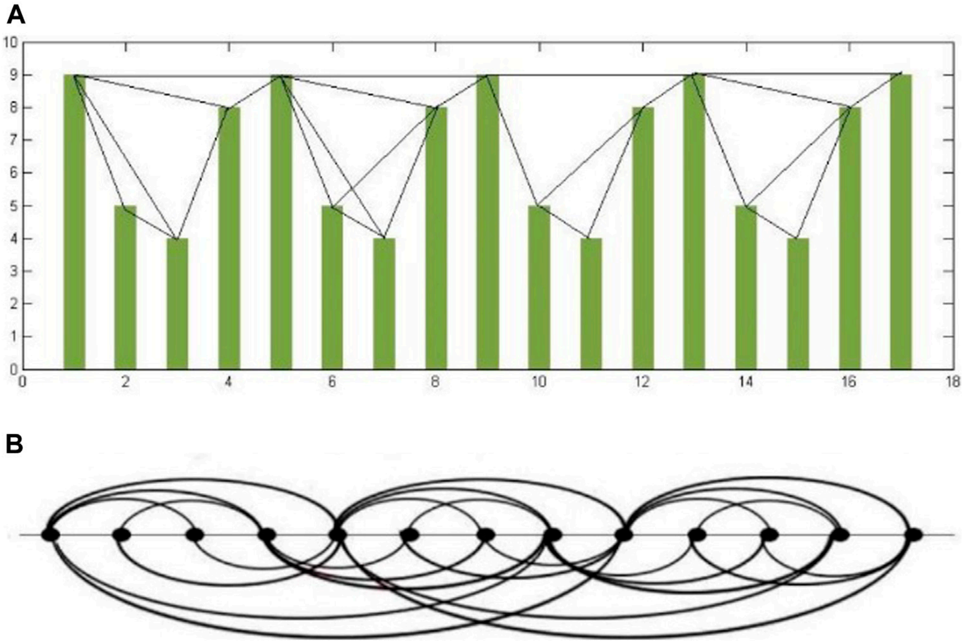


FIGURE 1 | Visibility graph algorithm.

sharing and making public information about the credit of citizens, legal professionals or other organizations in the course of their duties.

Using the normative and working documents produced by the central administrative departments and local governments, joint disciplinary measures for dishonesty can be divided into the following types: 1) cancelling an individual's qualification, which has an impact similar to "market prohibition"; 2) reducing "entry" opportunities and thereby reducing access to these opportunities; 3) greater supervision of the subject to "increase the frequency of inspection" and "strengthen on-site verification"; 4) publishing a "blacklist" in relation to dishonesty, thereby impacting on reputation. These four types are characterized by administrative, punitive punishment. According to guidance[6] issued by the State Council in 2016 on establishing and improving the joint incentive and punishment system for promise keeping and building social integrity, the specific joint punishment for breaking a promise generally includes four elements: administrative constraints and punishment, market constraints and punishment, industrial constraints and punishment, and social constraints and punishment. Typical administrative restrictions and punishments include "market and industry prohibition measures for enterprises with serious dishonesty and their legal representatives, main responsible persons and registered practitioners who are directly responsible for dishonesty." Market constraints and disciplinary measures include "restricting exit and purchase of real estate, flying, taking high-grade trains and seats, traveling and vacationing, staying in star rated hotels and other high consumption behaviors." Industry regulation and punishment include "supporting industry associations and chambers of commerce to implement disciplinary measures such as warning, criticism in the industry, public condemnation, rejection, and persuasion against dishonest members according to industry standards, industry rules, and trade agreements, depending on the seriousness of the case." Social constraints and punishment include "encouraging fair, independent and conditional social institutions to carry out big data public opinion monitoring of dishonesty and preparing and publishing regional and industrial credit analysis reports." Therefore, it is not difficult to see that joint disciplinary measures for dishonesty have the same characteristics as administrative punishment. In this case, social credit legislation is needed. The fundamental feature of China is that the people are the leaders of the country, and legislation is based on their concerns. Therefore, people's interest in social credit is an important way of promoting social credit legislation.

METHODOLOGY AND MATERIALS

Complex Networks

Many complex systems in nature can be described as networks. A typical network is composed of many nodes and the edges between them, with the nodes used to represent different individuals, and the edges representing the relationship between these. A complex network can be abstracted as a graph $G = (V,\ E)$ with the node set $V\ (g)$ and edge set $E\ (g)$. The number of nodes is defined as $N = |V|$ and the number of edges is $M = |E|$. Each edge in $E\ (g)$ has a pair of corresponding nodes in $V\ (g)$.

[6] http://www.gov.cn/zhengce/content/2016-06/12/content_5081222.htm

TABLE 1 | Original data from the Baidu search index in various provinces and cities.

Province	Average value	Standard error	Median	Standard deviation	Variance	Maximum value	Minimum value	Sum
Beijing	187.4721	2.961697	304	157	33341.03	962	87	712394
Shanghai	163.7863	2.600913	276	160.331	25712.81	802	66	622388
Guangdong	257.6874	4.428736	399	273.0056	74551.69	1245	151	979212
Tianjin	96.755	1.444423	180	89.04019	7930.243	448	0	367669
Henan	149.9637	2.384882	271	147.014	21618.81	722	91	569862
Sichuan	142.9887	2.281	251	140.6103	19776.46	903	74	543357
Chongqing	107.6476	1.657722	194	102.1889	10445.31	401	58	409061
Jiangsu	216.2671	3.991742	332	246.0675	60565.15	4028	105	821815
Hubei	116.9863	1.79056	206	109.7627	12051.03	641	65	444548
Zhejiang	159.7435	3.045229	324	187.7205	35246.67	930	132	734820
Fujian	174.1613	2.478682	277	152.7962	26527.16	1158	83	661813
Heilongjiang	82.57947	1.287911	155	79.39215	6304.773	359	57	313802
Shandong	159.3055	2.435159	278	150.1133	22539.94	851	129	605361
Shaanxi	92.55632	1.37918	171	85.01837	7230.025	355	0	351714
Hebei	114.4392	1.764219	220	108.7537	11830.49	692	73	434869
Liaoning	105.4563	1.642114	183	101.2267	10249.55	811	61	400734
Jilin	71.35184	1.122542	138	69.19812	4789.64	275	0	271137
Yunnan	91.17184	1.423843	165	87.77156	7705.874	401	57	346453
Xinjiang	64.40132	1.034316	126	63.7595	4066.345	265	0	244725
Guangxi	87.12263	1.324623	161	81.65526	6669.336	395	0	331066
Shanxi	85.33763	1.303746	161	80.36829	6460.762	365	57	324283
Hunan	82.1625	1.585768	194	97.75332	9558.228	582	66	394149
Jiangxi	97.58263	1.497132	181	92.28941	8519.577	448	60	370814
Anhui	121.2776	1.869917	223	115.2695	13290.54	543	73	460855
Gansu	60.08605	0.936133	129	62.95182	3963.974	231	0	228327
Hainan	61.11053	1.035654	131	63.13735	3987.374	260	0	232220
Guizhou	74.52	1.177976	144	72.61532	5274.373	266	57	283176
Ningxia	46.99632	0.886105	120	54.62317	2984.476	184	0	178586
Qinghai	29.85684	0.724157	60	42.58405	1813.878	152	0	113456
Inner Mongolia	75.92553	1.173236	140	72.32313	5235.723	292	0	288517
Tibet	18.15658	0.545759	57	119.4583	1132.137	185	0	68995

Structural Characteristics of Complex Networks

The three robust measures of network topology are average path length, clustering coefficient and degree distribution.

Average Path Length

Average path length L is the average number of steps along the shortest paths for all possible pairs of nodes i and j in the network. It is a measure of the efficiency of information or mass transport on a network. The formula of average path length is shown below.

$$L = \frac{1}{\frac{1}{2}N(N+1)} \sum_{ij} d_{i,j}$$

This shows that the average path length depends on the system size but does not change drastically with it.

Clustering Coefficient

In a network, the clustering coefficient of nodes is the proportion of the number of edges between all nodes adjacent to the node, to the maximum possible number of edges between these adjacent nodes. If node i has k_i edges linked to other nodes in the network, the actual number of edges between these k_i nodes is E_i, the maximum possible number of edges between nodes is $k_i(k_i - 1)/2$. Therefore, the clustering coefficient C_i is defined as:

$$C_i = \frac{2E_i}{k_i(k_i - 1)}$$

The clustering coefficient of the whole network refers to the mean value of the clustering coefficient of all nodes in the network, which reflects the local characteristics of the network, i.e., the probability that two adjacent nodes to the same node are still adjacent. The clustering coefficient C of the whole network is calculated as:

$$C = \frac{1}{N} \sum_{i=1}^{N} C_i$$

Degree Distribution

The degree k_i of a node i is the number of other nodes adjacent to that node, which is the same as the number of edges connected to the node. The degree of the network refers to the average of all node degrees in that network.

Degree distribution $P(k)$ is the probability distribution of the degrees of each node in the network and is an overall description of the degree of the nodes in that network. For

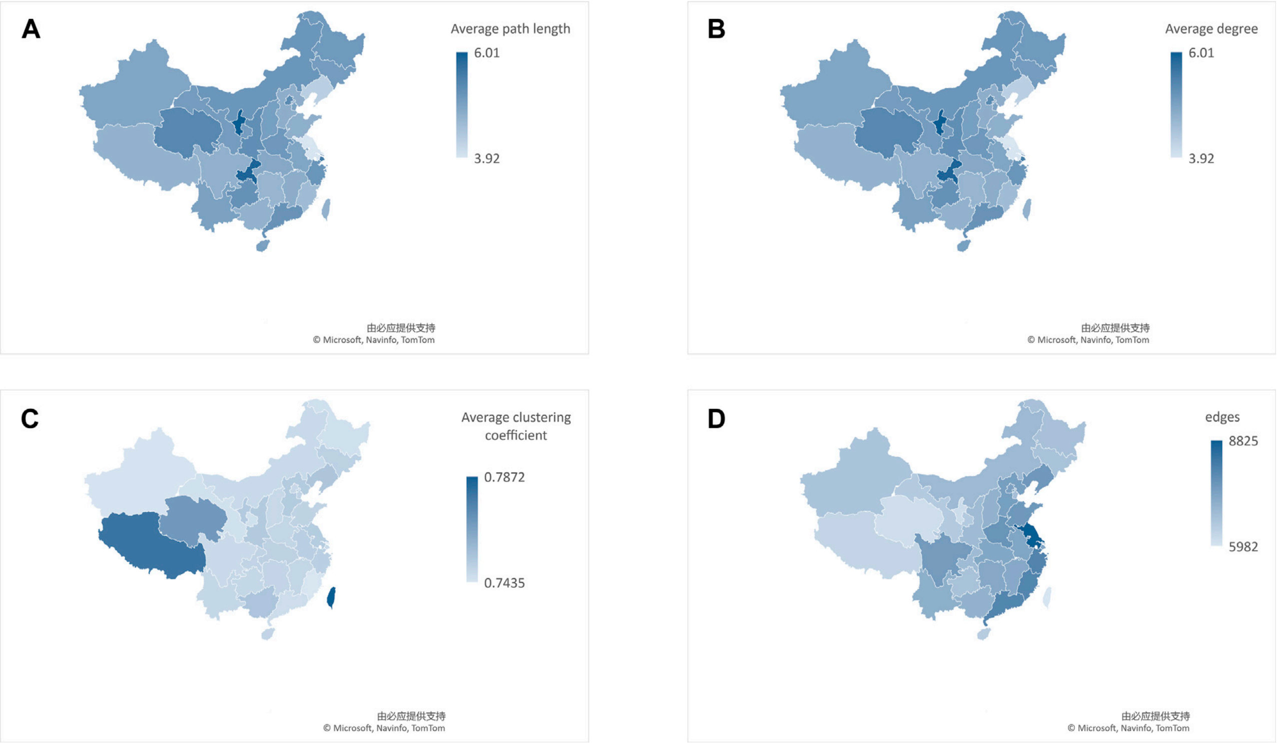

A
Average path length
6.01
3.92
由必应提供支持
© Microsoft, Navinfo, TomTom
B
Average degree
6.01
3.92
由必应提供支持
© Microsoft, Navinfo, TomTom
C
Average clustering coefficient
0.7872
0.7435
由必应提供支持
© Microsoft, Navinfo, TomTom
D
edges
8825
5982
由必应提供支持
© Microsoft, Navinfo, TomTom

FIGURE 2 | Visibility network parameters.

example, if the degree distribution conforms to the power law distribution $p(k) \sim k^{-y}$, then the network is scale-free.

Visibility Graph Algorithm for Time Series Data

This research applied a visibility graph algorithm proposed by Lacasa to construct the network. The time series data from the Baidu search index of 31 provinces, autonomous regions and cities in China (excluding Hong Kong, Macao and Taiwan) with the keywords "joint punishment for dishonesty" is transformed into complex networks.

Figure 1 presents the principle of the visibility graph algorithm for time series data, with the time series data transferred to a bar chart. The column shows the data at each time point. If the tops of two columns are visible (i.e., they can directly connect) to each other, the two points are connected. Based on the above theory, time series data can be transferred to a complex network.

First, the time nodes $x\ (t)$ are defined in the network, and edges are established by visualizing the principle. That is to say, for any point (t^b, x^b)between two points (t^a, x^a) and (t^c, x^c), when $t^a < t^b < t^c$ and $x^b < (x^c - x^a)\frac{t^b - t^a}{t^c - t^a}$, the edge can be established.

Second, the adjacency matrix is constructed according to the time series nodes and edges.

Finally, the network graph is created.

K-Means Clustering

The k-means clustering algorithm is an iterative clustering algorithm and is the most used based on Euclidean distance. It assumes that the shorter the distance between the two targets, the greater the similarity. The steps are as follows.

First, the data are divided into k groups, then "K" samples are randomly selected as the initial clustering center.

$$a = \{a_1, a_2, a_3 \cdots a_n\}$$

Second, for each sample x_i in the data set, the distance from the sample x_i to the k cluster centers is calculated, and the sample x_i is divided into the clusters corresponding to the closest cluster centers. The cluster centers and the objects assigned to them represent a cluster.

Third, for each category a_j, the cluster center (the centroid of the sample) is recalculated as:

$$a_j = \frac{1}{|c_i|} \sum_{x \in c_i} x$$

This process will continue to cycle until the following conditions are satisfied: 1) no (or minimum) objects are reassigned to different clusters; 2) no (or minimum) clustering centers change again, and the sum of squared errors is locally minimum.

Data and Materials

This paper uses data generated by the Baidu Index, which is one of the main statistical data analysis platforms in the era of big data

TABLE 2 | Visibility network parameters.

Province	Average clustering coefficient	Average path length	Average degree	Number of sides
Beijing	0.7476	5.33	5.33	7416
Shanghai	0.7469	5.54	5.54	7320
Guangdong	0.7461	5.18	5.18	8000
Tianjin	0.7493	4.96	4.96	6793
Henan	0.7484	5.13	5.13	7689
Sichuan	0.7471	4.76	4.76	7556
Chongqing	0.7490	5.90	5.90	6871
Jiangsu	0.7515	3.92	3.92	8825
Hubei	0.7500	4.87	4.87	7304
Zhejiang	0.7503	5.16	5.16	8034
Fujian	0.7435	4.56	4.56	7995
Heilongjiang	0.7451	5.10	5.10	6738
Shandong	0.7494	4.77	4.77	7603
Shaanxi	0.7520	5.26	5.26	6881
Hebei	0.7518	4.79	4.79	7471
Liaoning	0.7536	4.26	4.26	7609
Jilin	0.7500	5.06	5.06	6710
Yunnan	0.7480	4.98	4.98	7129
Xinjiang	0.7439	4.92	4.92	6687
Guangxi	0.7541	4.73	4.73	7067
Shanxi	0.7476	5.02	5.02	7072
Hunan	0.7487	4.69	4.69	7266
Jiangxi	0.7483	4.79	4.79	7262
Anhui	0.7508	4.92	4.92	7342
Inner Mongolia	0.7473	5.15	5.15	6913
Gansu	0.7448	5.06	5.06	6528
Hainan	0.7486	5.00	5.00	6503
Guizhou	0.7480	5.21	5.21	6725
Ningxia	0.7519	6.01	6.01	6159
Qinghai	0.7682	5.36	5.36	6134
Tibet	0.7798	4.71	4.71	6323

and, as such, is an important basis for analysis and decision-making. The so-called search index is based on the volume of searches generated by Baidu users, using keywords as statistical index parameters. Through a series of scientific calculations, the weighted sum of a keyword is calculated. The general public's concern about dishonesty can also demonstrate their concern to build a credit society. As internet searching is an important source of information for the public, the level of the Baidu Index reflects the level of public awareness about the construction of a credit society.

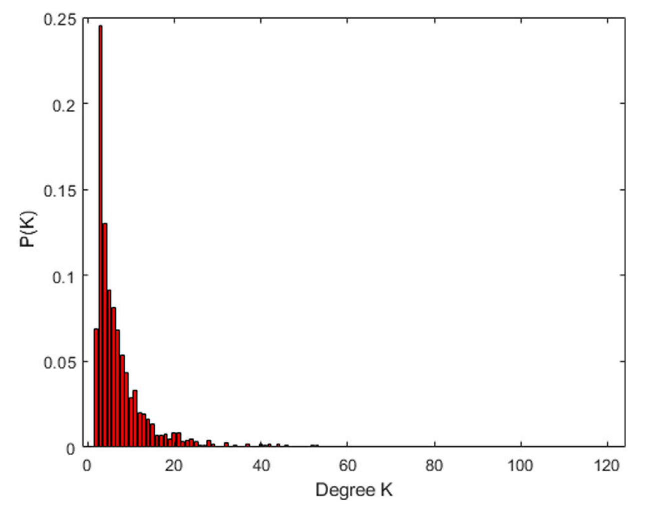


FIGURE 3 | Power distribution of visibility network in Beijing.

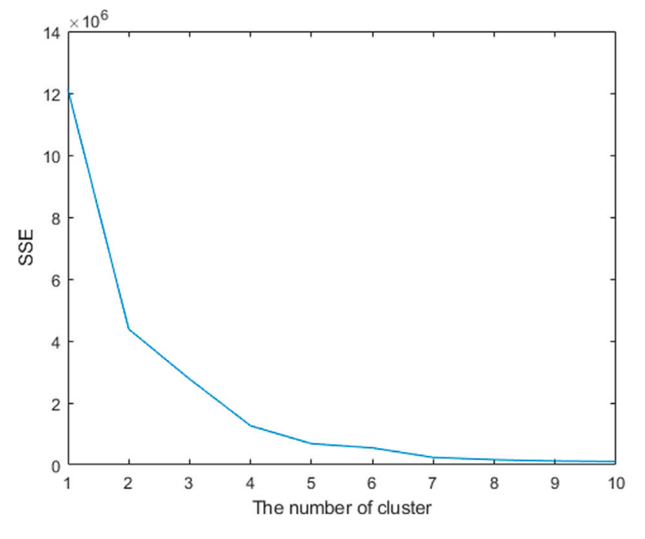


FIGURE 4 | Elbow method result for k-means.

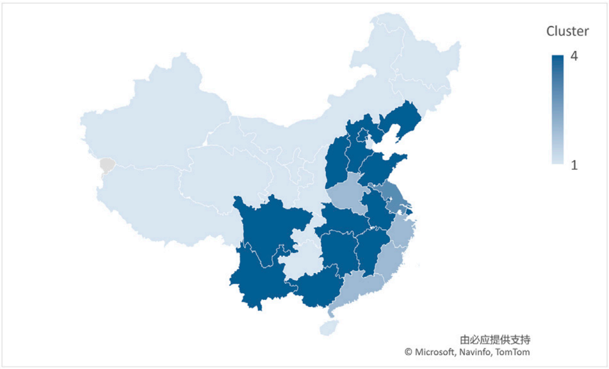


FIGURE 5 | K-means clustering result by K = 4.

RESULTS

This paper uses the Baidu search index from the 31 provinces from January 1, 2020, to May 27, 2021, and the keywords "joint punishment for dishonesty". The parameters of the original data are shown in **Table 1**.

An examination of the original data clearly shows that the average value, standard error, median, minimum value and sum of the economically developed areas, such as Guangdong and Jiangsu, are larger, and the average value, standard error, median, minimum value and sum of the economically backward areas, such as Qinghai and Tibet, are lower. The maximum, variance and standard deviation are also large in Guangdong and Jiangsu, while other areas are low, for example, Guangdong standard deviation reaches 273.0056 and Qinghai standard deviation is only 42.58405.

DISCUSSION

Construction of a Visibility Map Network in China

By constructing the Baidu search index network of the 31 provinces, the visibility network diagram was produced. See **Figure 2** for the schematic diagram.

The parameters are shown in **Table 2**. The higher the average clustering coefficient and the greater number of edges, the closer the relationship between time nodes. The larger the average path length and degree, the less close is the relationship between time and search behavior.

The data shows a positive correlation between the daily search volume of each province; therefore, an analysis of past data can provide a predictive function of future search volume.

Network Degree Distribution

Figure 3 represents the power-law distribution of the complex network in Beijing, showing that the visibility graph network is scale-free. There are more time nodes with fewer edges, and the proportion of nodes with larger degree values is smaller. As the degree increases, the number of nodes decreases.

The degree distribution shows that the Baidu comprehensive search index in Beijing is a fractal time series with long-term correlation. The original time series are in different time ranges, however, due to long-term correlation, any future changes to the Baidu search index in Beijing may result in similar time ranges to previous ones. See the **Supplementary Appendix** for the visibility network distribution map of other provinces and cities.

K-Means Clustering

The k-means clustering method is used to cluster the data from the 31 provinces, autonomous regions and cities. Using the elbow method, it is found that the slope increases significantly in four places (as shown in **Figure 4**). The schematic diagram of clustering results is shown in **Figure 5**. The provinces, autonomous regions and cities under each category display similar attention to the joint punishment of dishonesty from January 1, 2020, to May 27, 2021.

Figure 5 shows that the 31 provinces and cities are divided into four categories: Xinjiang, Tibet, Gansu, Qinghai, Hainan, Ningxia, Guizhou, Heilongjiang and Jilin are combined into the first group; Guangdong, Fujian, Zhejiang, Sichuan, Liaoning, Hebei, Henan and Shandong are the second group; and Jiangsu Province is the third group. Inner Mongolia, Shanxi, Shaanxi, Hunan, Hubei, Anhui, Jiangxi, Guangxi, Yunnan, Chongqing, Beijing and Shanghai make up the fourth group.

The Baidu search index, which is a combination of dishonesty and punishment, reflects changes in attention at the province and city levels on social credit. The first group of networks has a larger diameter, smaller edges, and lower average degree and clustering coefficient reflecting a weak association between nodes, relatively backward economic and social credit degree, and the underdevelopment of social credit legislation. The second group is in the mid-range in terms of diameter, edge, average, cluster coefficient and density, and there is a relationship between nodes; this reflects that in 2021, there is a high level of concern about joint punishment for dishonesty. In some places, relevant local regulations and administrative normative documents have been issued, which standardize the punishments for dishonesty, however, the standard density is not as good as that of the third group. For the fourth group of provinces and cities, the network edge, average degree, clustering coefficient and density are relatively high, indicating that the relationship between nodes is close. In 2021, there was great concern about joint punishment for dishonesty in such places, shown by the amount of normative documentation and research published. For example, Hunan first established the credit risk management college to study and analyze the credit risk.

CONCLUSION

Legislation should reflect the concerns of the people. Therefore, by analyzing the focus on "joint punishment for dishonesty", this paper provides evidence-based, theoretical support for the further promotion of social credit legislation and the construction of a credit society. Based on the Baidu search index of 31 provinces, this paper transforms the original time series into a visibility graph network, studying its dynamic characteristics to offer a new perspective from which to analyze the time series of "joint punishment for dishonesty". Results show that there are differences in the degree of concern in China's provinces, autonomous regions and municipalities on this issue. Cluster analysis allows the similarity of each province to be clearly seen. At the same time, through the division of time series, the visibility graph algorithm is used to analyze and predict the people's legislative demands.

This paper innovatively introduces visibility graph method to advance research in social credit legislation. Limited to the data, only the social attention of joint punishment for dishonesty is analyzed. Next, we will further mine the credit data, using HVD (horizontal visibility graph) and other complex network methods, combining the time data of credit legislation, to explore the relationship and mechanisms between credit attention and credit legislation.

DATA AVAILABILITY STATEMENT

The datasets presented in this article are not readily available because they are charged by Baidu Index database (https://index.baidu.com/v2/index.html#/). Requests to access the dataset should only be downloaded with Baidu's consent and restrictions.

AUTHOR CONTRIBUTIONS

QZ: Visualization, Software, Computation, Drawing and Writing ZY: Conceptualization, Investigation, Visualization, Software LF: Methodology, Validation, Writing-Reviewing and Editing.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fphy.2021.746660/full#supplementary-material

REFERENCES

1. Kandula S, and Shaman J. Reappraising the utility of Google flu trends. *Plos Comput Biol* (2019) 15(8):e1007258. doi:10.1371/journal.pcbi.1007258
2. Aguilera AM, Fortuna F, Escabias M, and Di Battista T. Assessing Social Interest in Burnout Using Google Trends Data. *Soc Indic Res* (2021) 156(2): 587–99. doi:10.1007/s11205-019-02250-5
3. Polgreen PM, Chen Y, Pennock DM, Nelson FD, and Weinstein RA. Using internet searches for influenza surveillance. *Clin Infect Dis* (2008) 47:1443–8. doi:10.1086/593098
4. Kang M, Zhong H, He J, Rutherford S, and Yang F. Using Google trends for influenza surveillance in South China. *PloS one* (2013) 8(1):e55205. doi:10.1371/journal.pone.0055205
5. Dugas AF, Jalalpour M, Gel Y, Levin S, Torcaso F, Igusa T, et al. Influenza forecasting with Google flu trends. *PloS one* (2013) 8(2):e56176. doi:10.1371/journal.pone.0056176
6. Thompson LH, Malik MT, Gumel A, Strome T, and Mahmud SM. Emergency department and 'Google flu trends' data as syndromic surveillance indicators for seasonal influenza. *Epidemiol Infect* (2014) 142:2397–405. doi:10.1017/S0950268813003464
7. Chan EH, Sahai V, Conrad C, and Brownstein JS. Using web search query data to monitor dengue epidemics: a new model for neglected tropical disease surveillance. *Plos Negl Trop Dis* (2011) 5(5):e1206. doi:10.1371/journal.pntd.0001206
8. Huang X, Zhang L, and Ding Y. The Baidu Index: Uses in predicting tourism flows -A case study of the Forbidden City. *Tourism Manag* (2017) 58:301–6. doi:10.1016/j.tourman.2016.03.015
9. Yang X, Pan B, Evans JA, and Lv B. Forecasting Chinese tourist volume with search engine data. *Tourism Manag* (2015) 46:386–97. doi:10.1016/j.tourman.2014.07.019
10. Sun S, Wei Y, Tsui K-L, and Wang S. Forecasting tourist arrivals with machine learning and internet search index. *Tourism Manag* (2019) 70:1–10. doi:10.1016/j.tourman.2018.07.010
11. Zhao R. Inferring private information from online news and searches: Correlation and prediction in Chinese stock market. *Physica A: Stat Mech its Appl* (2019) 528:121450. doi:10.1016/j.physa.2019.121450
12. Yuan Z, and Wenbin B. The impact of investors' attention on stock returns - Study based on Baidu Index [Conference presentation]. In: *International Conference on Service Systems & Service Management,*. New Jersey: IEEE (2014).
13. Dong J, Dai W, Liu Y, Yu L, and Wang J. Forecasting Chinese Stock Market Prices using Baidu Search Index with a Learning-Based Data Collection Method. *Int J Info Tech Dec Mak* (2019) 18:1605–29. doi:10.1142/S0219622019500287
14. Yuan Q, Nsoesie EO, Lv B, Peng G, Chunara R, and Brownstein JS. Monitoring influenza epidemics in China with search query from Baidu. *PloS one* (2013) 8: e64323. doi:10.1371/journal.pone.0064323
15. Su K, Xu L, Li G, Ruan X, Li X, Deng P, et al. Forecasting influenza activity using self-adaptive AI model and multi-source data in Chongqing, China. *Ebiomedicine* (2019) 47:284–92. doi:10.1016/j.ebiom.2019.08.024
16. Zhao Y, Xu Q, Chen Y, and Tsui KL. Using Baidu index to nowcast hand-foot-mouth disease in China: a meta learning approach. *BMC Infect Dis* (2018) 18: 398. doi:10.1186/s12879-018-3285-4
17. Chen S, Liu X, Wu Y, Xu G, Zhang X, Mei S, et al. The application of meteorological data and search index data in improving the prediction of HFMD: A study of two cities in Guangdong Province, China. *Sci Total Environ* (2019) 652:1013–21. doi:10.1016/j.scitotenv.2018.10.304
18. Li K, Liu M, Feng Y, Ning C, Ou W, Sun J, et al. Using Baidu Search Engine to Monitor AIDS Epidemics Inform for Targeted intervention of HIV/AIDS in China. *Sci Rep* (2019) 9(1):320. doi:10.1038/s41598-018-35685-w
19. Gao Z-K, Small M, and Kurths J. Complex network analysis of time series. *Europhys Lett* (2006) 116(5):50001.
20. Zhao L-L, Tang Z, Wang J-Y, Wang J-B, and Yang H-J. Time series analysis based upon complex network. *J Univ Shanghai Sci Technol* (2011) 33(1):47–52. (in Chinese).
21. Zou Y, Donner RV, Marwan N, Donges JF, and Kurths J. Complex network approaches to nonlinear time series analysis. *Phys Rep* (2019) 787:1–97. doi:10.1016/j.physrep.2018.10.005
22. Zhang J, and Small M. Complex Network from Pseudoperiodic Time Series: Topology versus Dynamics. *Phys Rev Lett* (2006) 96(23):238701. doi:10.1103/PhysRevLett.96.238701
23. Zhang J, Luo X, Nakamura T, Sun J, and Small M. Detecting temporal and spatial correlations in pseudoperiodic time series. *Phys Rev E* (2007) 75(1): 016218. doi:10.1103/PhysRevE.75.016218
24. Zhang J, Sun J, Luo X, Zhang K, Nakamura T, and Small M. Characterizing pseudoperiodic time series through the complex network approach. *Physica D: Nonlinear Phenomena* (2008) 237(22):2856–65. doi:10.1016/j.physd.2008.05.008
25. Yang Y, and Yang H. Complex network-based time series analysis. *Physica A: Stat Mech its Appl* (2008) 387(5):1381–6. doi:10.1016/j.physa.2007.10.055
26. Gao Z, and Jin N. Flow-pattern identification and nonlinear dynamics of gas-liquid two-phase flow in complex networks. *Phys Rev E* (2009) 79:066303. doi:10.1103/PhysRevE.79.066303
27. Marwan N, Donges JF, Zou Y, Donner RV, and Kurths J. Complex network approach for recurrence analysis of time series. *Phys Lett A* (2009) 373: 4246–54. doi:10.1016/j.physleta.2009.09.042

28. Donner RV, Zou Y, Donges JF, Marwan N, and Kurths J. Recurrence networks-a novel paradigm for nonlinear time series analysis. *New J Phys* (2010) 12(3):033025. doi:10.1088/1367-2630/12/3/033025
29. Wang J, and Yang H. Complex network-based analysis of air temperature data in China. *Mod Phys Lett B* (2009) 23(14):1781–9. doi:10.1142/S0217984909019946
30. Pham TD. From fuzzy recurrence plots to scalable recurrence networks of time series. *Epl* (2017) 118(2):20003. doi:10.1209/0295-5075/118/20003
31. Xu X, Zhang J, and Small M. Superfamily phenomena and motifs of networks induced from time series. *Proc Natl Acad Sci* (2008) 105(50):19601–5. –05. doi:10.1073/pnas.0806082105
32. Tumminello M, Aste T, Di Matteo T, and Mantegna RN. A tool for filtering information in complex systems. *Proc Natl Acad Sci* (2005) 102(30):10421–6. doi:10.1073/pnas.0500298102
33. Yuan M, Wang W, Wang Z, Luo X, and Kurths J. Exponential Synchronization of Delayed Memristor-Based Uncertain Complex-Valued Neural Networks for Image Protection. *IEEE Trans Neural Netw Learn Syst* (2021) 32(1):151–65. doi:10.1109/tnnls.2020.2977614
34. Gao Z-K, Li S, Dang W-D, Yang Y-X, Do Y, and Grebogi C. Wavelet multiresolution complex network for analyzing multivariate nonlinear time series. *Int J Bifurcation Chaos* (2017) 27(08):1750123. doi:10.1142/S0218127417501231
35. HuXia JC, Xia C, Li H, Zhu P, and Xiong W. Properties and structural analyses of USA's regional electricity market: A visibility graph network approach. *Appl Math Comput* (2020) 385(2020):125434. doi:10.1016/j.amc.2020.125434
36. Zhang Y-J, Meng K, Gao T, Song Y-Q, Hu J, and Ti E-P. Analysis of attention on venture capital: A method of complex network on time series. *Int J Mod Phys B* (2020) 34:2050273. doi:10.1142/S0217979220502732
37. Cui X, Hu J, Ma Y, Wu P, Zhu P, and Li H-J. Investigation of stock price network based on time series analysis and complex network. *Int J Mod Phys B* (2021) 35:2150171. doi:10.1142/S021797922150171X

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

PERSPECTIVE
published: 07 October 2021
doi: 10.3389/fphy.2021.757361

COVID-19 Importation Risk From Olympic Athletes Prior to the Tokyo 2020 Olympics

Hoi Yat Vico Lau[1,2], *Mingda Xu*[1,2], *Lin Wang*[3], *Benjamin J. Cowling*[1,2] *and Zhanwei Du*[1,2]*

[1]*WHO Collaborating Centre for Infectious Disease Epidemiology and Control, School of Public Health, Li Ka Shing Faculty of Medicine, The University of Hong Kong, Hong Kong, Hong Kong, SAR China,* [2]*Laboratory of Data Discovery for Health, Hong Kong Science and Technology Park, Hong Kong, Hong Kong, SAR China,* [3]*Department of Genetics, University of Cambridge, Cambridge, United Kingdom*

The COVID-19 pandemic delayed the Tokyo 2020 Olympics for 1 year and sparked an unprecedented outbreak in Japan in early July 2021 due to the relaxation of social distancing measures for foreign arrivals. Approximately 11,000 athletes from 205 countries would gather at the Tokyo Olympics held from July 23 through August 8, 2021. Based on the prevalence of infection in different source locations and athlete numbers, we estimated that seven countries would introduce least one infection of COVID-19 to Tokyo and at most eleven unidentified infections after the three requested COVID-19 tests.

Keywords: Coronavirus, importation, infections, prevention, testing

Edited by:
Hui-Jia Li,
Beijing University of Posts and Telecommunications (BUPT), China

Reviewed by:
Dong Li,
Shandong University, China
Chijun Zhang,
Jilin University of Finance and Economics, China

***Correspondence:**
Zhanwei Du
zwdu@hku.hk

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *12 August 2021*
Accepted: *01 September 2021*
Published: *07 October 2021*

Citation:
Vico Lau HY, Xu M, Wang L, Cowling BJ and Du Z (2021) COVID-19 Importation Risk From Olympic Athletes Prior to the Tokyo 2020 Olympics. Front. Phys. 9:757361. doi: 10.3389/fphy.2021.757361

INTRODUCTION

The risk of overwhelming fragile public health systems still lies in the coronavirus disorder 2019 (COVID-19), especially in developing countries. The COVID-19 in Japan rose sharply in early July 2021. As of August 22, 2021, there are 1.28 million COVID-19 confirmed cases reported, coupled with 16 thousand deaths. Globally, there have been over 211 million reported cases and 4.42 million deaths [1].

In September 2013, Tokyo in Japan was awarded the privilege to host the 2020 Olympics for the second time [2]. Hosting the Olympics has held great importance for Japan as the 1964 Games were seen as a symbol of revival for the country's rehabilitation and rebuilding process after World War Two, especially when Japan is currently facing a prolonged economic stagnation and recession [3]. However, the World Health Organization (WHO) declared the Coronavirus outbreak as a global public health emergency in January 2020 only months before the Olympics were scheduled to be held, this pressured the International Olympic Committee (IOC) to postpone the Games by a year with a cost of approximately $2.8 billion USD [2]. As of January 2021, Japan's government decided to continue this event amid the pandemic to avoid $15.4 billion USD in loss if canceled [2, 4].

In May 2021, the Tokyo Medical Practitioners Association, as well as Japanese firms, called to cancel the Games with the fear of facing another wave of infections [2]. This fear quickly turned to reality as Japan faced the fourth wave of infections in mid-June 2021 and outbreaks appeared in several cities in Japan, including the host city, Tokyo [5]. The Tokyo Olympics has taken measures to prevent the spread of the virus. In July 2021, IOC announced that there will be no spectators, which would see $820 million USD in lost revenue, to mitigate the potential outbreak [6]. But as to how specific and efficient the policies are still questionable. Athletes are required to undergo three COVID-19 tests before entering the Olympic Village, two of which should be done within 96 h of their flights to Tokyo and one test done on arrival [7]. However, COVID-19 tests can be negative during the latent period, and may not be able to identify all the infected athletes and prevent all introductions of infection into the Olympic village and potentially the local community.

Over 11,000 athletes from 205 countries would gather at the Tokyo Olympics held from July 23 through August 8, 2021 [8]. To build a COVID-19 bubble isolated from the Japanese population, there have been measures tailored towards preventing the spread of the virus. However, some measures were criticized by experts as they believe that the "safety protocols do not adequately protect athletes" and the "measures may not be strong enough to prevent outbreaks" [9]. The facilities in place were not designed to mitigate the risks that the pandemic might pose. The control measures and facilities provided could potentially be insufficient to prevent an outbreak from happening inside the COVID-19 bubble in the Olympic village. To quantify the transportation risk of athletes from each of the 205 other countries prior to the Tokyo 2020 Olympics, we would evaluate the probability of at least one infection and the number of infections arriving in Tokyo even after the three requested tests among athletes prior to the Tokyo 2020 Olympics.

MATERIALS AND METHODS

Risk of Imported Infections From Participating Countries to Tokyo

We used the number of athletes who traveled into Tokyo and the infection rate per person from each of the participating countries during the period from July 1 to July 23, 2021 [10, 11], to calculate the probable number of imported infected cases from participating countries. The following equation is used to deduce the probability of infected athletes traveling into Tokyo from active cases in the country i by

$$\gamma_i = \alpha_i \beta_i \tag{1}$$

Here α_i denotes the average probability of a person being infected in the country i, estimated bythe country's active cases per person. The number of active cases was the average active cases between the athletes' traveling period, July 1 to July 23, 2021. Then, it is multiplied by the number of participating athletes (β_i) to evaluate the probable number of infected athletes that the country would import into Tokyo.

Testing Strategy

The IOC playbook for players has adopted a testing strategy where foreign players would take three COVID-19 tests and local players would take two [7]. Since the test would most likely be taken on three different days for foreign players and the probability of a missed infection differs from day to day post-infection, the probability of a missed infection is deduced with the probability of the day of the infection, the day post-infection, and 2 days post-infection. Whereas the probability of Japanese athletes would be calculated by the day of the infection and the next day. The test chosen is the antigen test as it is the test that athletes would most likely take and it is also the test that would be used on arrival. The antigen test is also the least effective between the eligible tests that could be taken which are molecular and antibody tests [12].

We can quantify the overall probable number of unidentified infections of all the countries combined after each COVID-19 test by accumulating the number of unidentified infections after each test from each country other than Tokyo. The probable number of unidentified infected athletes after testing (ζ) is given by

$$\zeta = \gamma_i \prod_j \left(1 - \epsilon_j\right) \tag{2}$$

Here γ_i denotes the probable number of imported infected athletes of the country i. ϵ_j represents the sensitivity of COVID-19 tests with day j post-infection.

Alternative Strategy

A more effective measure could be mandatory quarantine and testing. A successful example would be the NBA Bubble (July 12–October 11, 2020) where players were required to self-isolate in their rooms for up to 48 h until receiving two negative COVID-19 tests with only two infections in the first week (July 22–28, 2020) and no infections after [13]. Inspired by this, we would estimate the probable number of unidentified COVID-19 cases if all athletes are self-isolated for more days after arriving. The probable number of unidentified cases would be given by

$$\theta_k = \theta_{k-1} \cdot \iota_k \tag{3}$$

Here θ_k represent the probable unidentified cases on the day k. ιk denotes the sensitivity of COVID-19 tests on day k

RESULTS

Risk of Imported Infections From Participating Countries to Tokyo

The risk for the transportation of COVID-19 to Tokyo varies across countries informed by the number of active cases during the studied period between July 1 to July 23, 2021 (**Figure 1**). The number of infected athletes arriving in Tokyo between July 1 to July 23, 2021, ranges from approximately two to six infections depending on the day of arrival. With limited outliers in the data, the average is taken to represent the probable number of infections that would be imported to Tokyo from Great Britain. Hence, the probable number of imported cases is 3.81, so approximately four cases.

The United States and Great Britain have the highest probable cases with nine and four imported cases, respectively, followed by Spain, Russia, Netherlands, Brazil, and Argentina with at least one COVID-19 case into Tokyo. The estimated overall accumulated infections can be approximately rounded to 32 infected athletes in the studied period from July 1 to July 23, 2021. This would imply that the United States and Great Britain possess the highest risk of bringing multiple infected athletes into Tokyo along with Spain, Russia, Netherlands, Brazil, and Argentina.

Testing Strategies

We compare IOC's testing strategy in various days post-infection with an increase in sensitivity of the antigen COVID-19 test in the

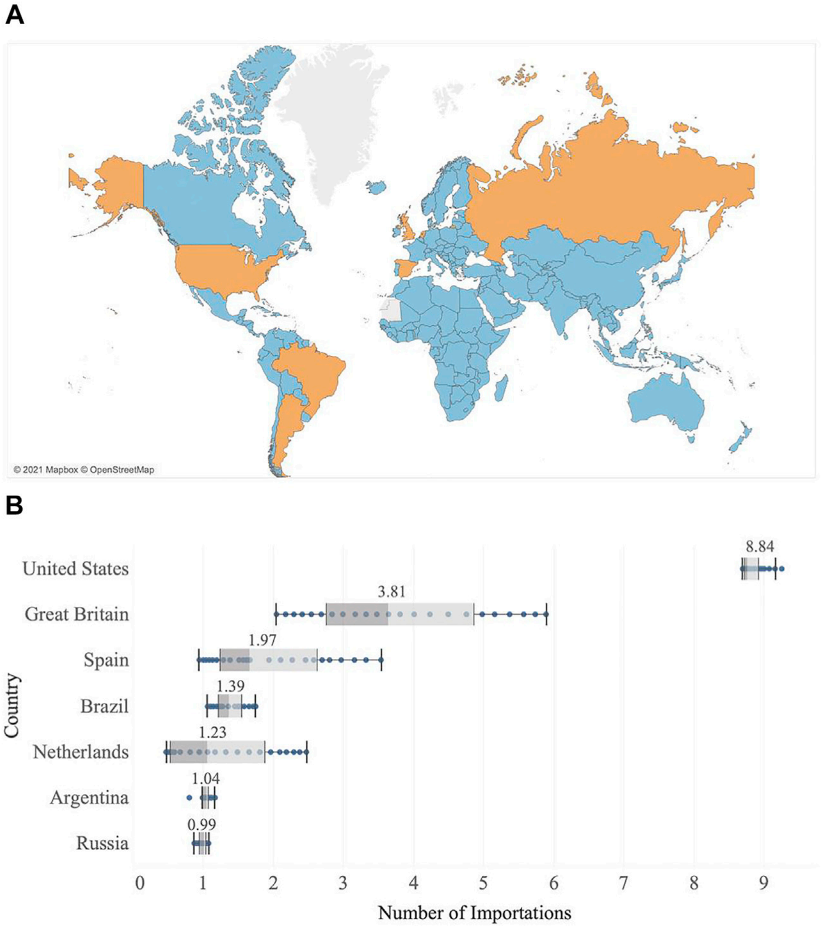


FIGURE 1 | Risks for transportation of COVID-19 to Tokyo, Japan, between July 1 to July 23, 2021. **(A)** Estimated median probable number of athletes imported to Tokyo from participating countries into Tokyo during the studied period between July 1 to July 23, 2021. For each country, we estimate probable numbers by assuming athletes all travel to Tokyo on a day during the studied period, resulting in 23 estimates. The seven countries with a median probable number >1 are indicated in shades of orange; 198 countries below that threshold are indicated in shades of blue. **(B)** Estimated probable numbers of athletes in seven countries with the largest median estimates of probable numbers. The box plots denote a five-number summary of minimum, lower quartile, median, upper, and maximum values based on the 23 estimated numbers. The labeled value above the box plot is the average number of importations. The map was created using Tableau Software for Desktop version 2021.2 (https://www.tableau.com/support/releases/desktop/2021.2). The layouts were modified with Keynote version 11.1 (https://www.apple.com/hk/en/keynote/).

following days (**Figure 2A**). The sensitivity of the antigen COVID-19 test ranges from 26.3 to 100% with a steady increase in sensitivity daily during the first 6 days post-infection [12]. Given there are 32 athletes estimated to be imported into Tokyo in Japan, we estimate the number of unidentified infections across three COVID-19 tests (**Figure 2A**). There would still be 11 unidentified cases after three antigen COVID-19 tests if the first test was taken coincidently on the day of infection. Specifically, the test on the first, second and third day post-infection has sensitivities of 26.3, 30.0, and 36.3%, respectively, resulting in reduced importation from 32 cases to 24, 17, and 11, respectively. In contrast, there would only be five unidentified infections finally if the first tests were all taken on the second day post-infection and can be zero if the first tests are all conducted 3 days or later post-infection.

We illustrate the decrease in the number of unidentified infections with the use of the increased sensitivity of the antigen COVID-19 test in the following days (**Figure 2B**). This shows that with daily testing for 5 days, the risk of an infected person is closer to zero than one with the day of infection on the day of the first test. The day of infection on the day of the first test was chosen to portray the most pessimistic assumption for COVID-19 testing since the test is least effective when the person is recently infected. Mandatory quarantine of the player in their rooms was chosen to avoid the risk of the virus transmitting through contact and interaction. We can imply that the risk of an infected person entering the Olympic Village after quarantine and daily testing is extremely low. With implementing a 5 days quarantine and daily testing for the entry of athletes, the risk of the virus entering the Olympic Village has been driven down significantly.

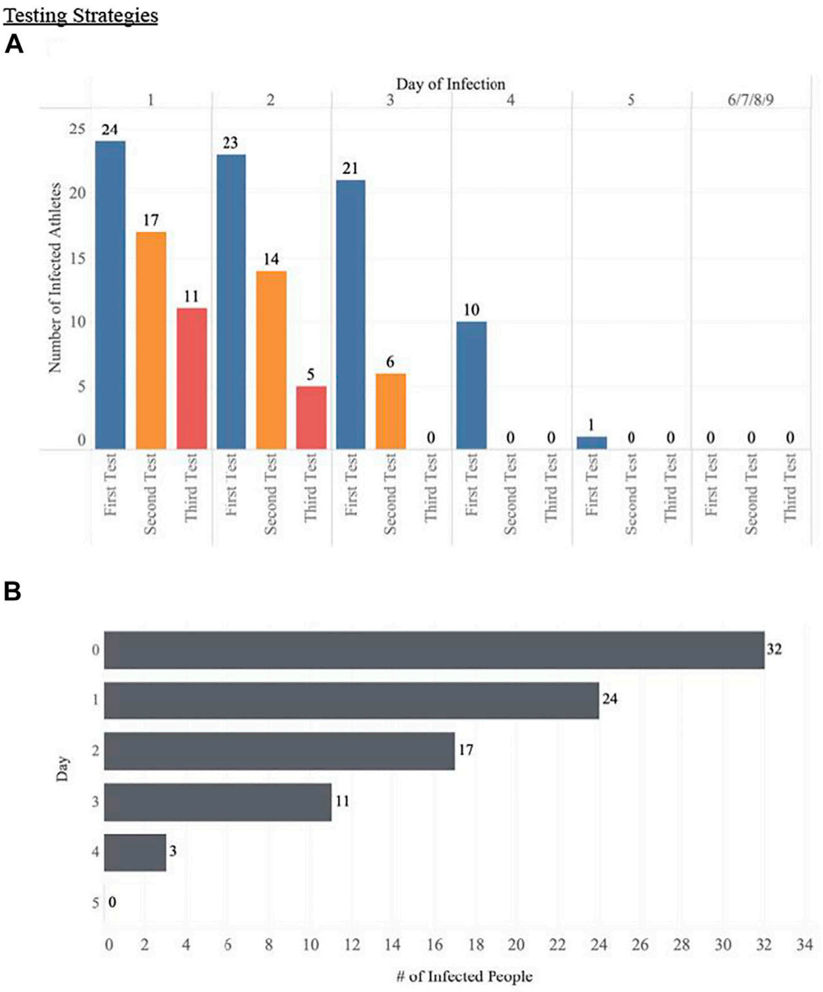


FIGURE 2 | The number of infected athletes after each test for the respective strategies. **(A)** The probable number of unidentified cases with the IOC's strategy by using the probable number of unidentified cases and the sensitivity of each test executed during post-infection [12]. We assume tests are taken on the first, second, and third day post-infection, which are colored by blue, orange, and red, respectively. This value in the *y*-axis infers the number of unidentified infected athletes after each test. **(B)** The number of infected athletes after 5 days of quarantine and regular testing. This figure extrapolates the idea of how the number of quarantine days decreases the chance of unidentified infections while regularly tested in isolation. The bar graph and line graph were created using Tableau Software for Desktop version 2021.2 (https://www.tableau.com/support/releases/desktop/2021.2). The layouts were modified with Keynote version 11.1 (https://www.apple.com/hk/en/keynote/).

DISCUSSION AND CONCLUSION

Despite the efforts of the implementation of guidelines to contain the spread of the virus via the official playbook, the measures and facilities in place cannot prevent the burst of the coronavirus bubble. Even though vaccinated players have an increase in immunity, they can also be infected along with the unvaccinated players. Accurate testing before entry to the coronavirus bubble is essential as the virus would assuredly spread among athletes inside the bubble with its shared space and loosened coronavirus protocols while competing. The unaddressed aerosol fine particle transmission and the contact tracing strategies are not adequate to fully monitor and safely control the risk of transmissions. It would be advised to enforce more specific and effective measures before the continuation of the Games.

Are the facilities and measures within the Olympic Village sufficient to prevent the break of the bubble?

The Tokyo Olympics 2020 attempts to construct a coronavirus bubble with several measures in place to control the spread of the virus such as regular testing, contact tracing, and social distancing [7]. According to the official playbook within the bubble, there are policies to be followed, but without enough details [7].

For example, the playbook of Tokyo Olympics 2020 does not address specific guidelines to prevent the most important transmission mode, aerosol fine particle transmission [7]. The significant features of exposure to aerosol inhalation are the concentration of infectious particles in the air and the length of time spent in contact with those particles [14]. Organizers were confident that it is sufficient to keep the athletes' room ventilated, but the specifics as to how frequently the air within indoor spaces was refreshed and replaced every hour were not given in detail [15]. There is also a lack of information about the review, adaptation, and modification of HVAC systems, which would pose a potential risk of unmonitored aerosol transmissions in various spaces inside the coronavirus bubble.

Contact tracing mainly relies on smartphones rather than wearable technology [7]. As smartphones would not always be present with the athletes while they are competing, wearable technology would be more efficient and safe for close contact tracking [14]. The current policy in the playbook to track contacts is by surveying with questions over the phone (how long were they together, whether they were wearing a mask, etc.,) [7]. This strategy is not feasible as the player's participation would be at stake if they are verified as close contact, therefore the player could simply fabricate their statement for the continuation of their participation. Wearable technology backed up by video monitoring for finding close contact could and should be implemented into contact tracing as another option.

Other issues may arise too. Though the Olympic Village is expected to be a bubble isolated from the Japanese public, there would still be inevitable contact with service providers such as cleaning, transport, COVID-19 testing, security, and catering outside the bubble [16]. As seen in Australia, bubbles are still vulnerable to infections despite strict hygiene practices [16].

Limitation of the Predicted Outcome of International Olympic Committee's Testing Plan

There is also a risk of athletes being infected during their travels to Tokyo between the three COVID-19 tests taken in their country. Once the case is not identified by the first two tests, the virus could be easily transmitted during the flight, a confined space. There is a high probability that the infection would be unidentified as they would only take one COVID-19 test on arrival. With no quarantine measures taken, the virus could potentially enter the Olympic Village and spread amongst athletes.

The active cases per one million of the population could not accurately quantify the number of infected athletes as the correlation between the active cases and the athletes is quite low. It could only be roughly estimated as athletes could not be represented by the general population of their country. The vaccination rate of the athletes is not taken into account. With the athletes less prone to be infected by the virus after vaccination, the number of infected athletes might be even lower than the number predicted. Since the playbook and IOC did not disclose specifics for the infrastructure of their facilities and the actions that will be taken to execute these measures, the implications made under the ambiguous information given cannot be definitive.

We estimate there are approximately 32 infected athletes in the Tokyo Olympics 2020, which is close to the reported 28 athletes [17]. As our prediction only takes into account the measures inside the Olympic Village, only the infected athletes who live inside will be considered. Furthermore, our prediction was the probable number of infected athletes entering the Olympic Village after testing. The number of athletes reported might have been infected after entering the Village by other athletes or service providers, so the number of infected athletes reported would be higher than our estimated value.

Overall, if there are positive cases amongst the athletes inside the Olympic Village, it can suggest that the policies settled in the playbook for Olympic Village are insufficient and the intended COVID-19 bubble is broken. The intent of the COVID-19 bubble was to create a COVID-19 free environment. With the bubble broken, the risk of the athletes contaminating each other is created and it would no longer be a COVID-19 free environment.

The COVID-19 bubble is undeniably broken with not only one but multiple positive cases among athletes before and during the Games [17]. The spread within the Olympic Village between athletes could also be a possibility as many athletes infected are from the same country and some even the same sport. This includes two South African football players, three beach volleyball players, four Czech Republican players, and five Dutch players in particular [17]. This is a testament to how the prevention measures in place from the playbook failed to truly keep out the virus to create a safe environment for the competing athletes.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: https://www.eurosport.com/olympics/athletes/country/all.

AUTHOR CONTRIBUTIONS

VY, LW, BC and ZD: conceived the study, designed statistical and modeling methods, conducted analyses, interpreted results, wrote and revised the article; MX: interpreted results, and revised the article.

ACKNOWLEDGMENTS

We acknowledge the financial support from the Health and Medical Research Fund, Food and Health Bureau, Government of the Hong Kong Special Administrative Region (grant no. COVID190118), and Seed Fund for Basic Research for New Staff of the University of Hong Kong (202009185062). National Natural Science Foundation of China (grant no. 72104208).

REFERENCES

1. ArcGIS. *COVID-19 Dashboard by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University (JHU)* (2020). Available at: https://gisanddata.maps.arcgis.com/apps/opsdashboard/index.html#/bda7594740fd40299423467b48e9ecf6 (Accessed September 28, 2020).
2. Reuters. *Olympics-Tokyo's Delayed and Disrupted 2020 Games* (2021). Available at: https://www.reuters.com/article/us-olympics-2020-timeline-idCAKCN2EE1IN (Accessed July 21, 2021).
3. Illmer A. *Tokyo Olympics: Why Doesn't Japan Cancel the Games? BBC* (2021). Available at: https://www.bbc.com/news/world-asia-57097853 (Accessed July 21, 2021).
4. Wade S. *Official Costs of Tokyo Olympics up by 22% to $15.4 Billion*. Associated Press (2020). Available at: https://apnews.com/article/tokyo-coronavirus-pandemic-2020-tokyo-olympics-japan-olympic-games-3c46bce81928865d9aae0832b5ddd9e3 (Accessed July 21, 2021).
5. Toyo Keizai. *Coronavirus Disease (COVID-19) Situation Report in Japan* (2021). Available at: https://toyokeizai.net/sp/visual/tko/covid19/en.html (Accessed July 21, 2021).
6. Ryall J. *Tokyo Olympics: "Japanese Only" Signs Spark Outrage as Sponsors Count Cost of Spectator Ban and Covid-19 State of Emergency*. South China Morning Post (2021). Available at: https://www.scmp.com/week-asia/health-environment/article/3140792/tokyo-olympics-japanese-only-signs-spark-outrage (Accessed July 21, 2021).
7. IOC. *Third Version of Tokyo 2020 Playbooks Published* (2021). Available at: https://olympics.com/ioc/tokyo-2020-playbooks (Accessed July 21, 2021).
8. Tokyo. *Summer Olympics Participating Countries: 206 NOCs Nations List* (2020). Available at: https://www.whereig.com/olympics/summer-olympics-participating-countries.html (Accessed August 2, 2021).
9. Keith M. *Here's what Medical Experts Say Could Improve COVID-19 Safety Measures at the Tokyo Olympics*. Insider (2021). Available at: https://www.insider.com/covid-19-safety-measures-at-the-tokyo-olympics-2021-7 (Accessed July 23, 2021).
10. Worldometers. *COVID Live Update: 201,777,519 Cases and 4,281,876 Deaths from the Coronavirus - Worldometer* (2021). Available at: https://www.worldometers.info/coronavirus/(Accessed August 6, 2021).
11. EuroSport. *Athletes by Country* (2021). Available at: https://www.eurosport.com/olympics/athletes/country/all (Accessed August 6, 2021).
12. Du Z, Pandey A, Bai Y, Fitzpatrick MC, Chinazzi M, Pastore Y, et al. Comparative Cost-Effectiveness of SARS-CoV-2 Testing Strategies in the USA: a Modelling Study. *The Lancet Public Health* (2021) 6:e184–e191. doi:10.1016/s2468-2667(21)00002-5
13. Sporting News. *NBA Bubble, Explained: A Complete Guide to the Rules, Teams, Schedule & More for Orlando Games* (2020). Available at: https://www.sportingnews.com/us/nba/news/nba-bubble-rules-teams-schedule-orlando/zhap66a9hcwq1khmcex3ggabo (Accessed July 28, 2021).
14. Sparrow AK, Brosseau LM, Harrison RJ, and Osterholm MT. Placebo-Controlled Trials of Covid-19 Vaccines - Why We Still Need Them. *N Engl J Med* (2021) 384:e2. doi:10.1056/nejmp2033538
15. Tokyo YA, Sturmer J, Sturmer J, Hoshiko E, and Kyung-Hoon K. *Japan Unveils its Olympic Village, but Health Experts Remain Concerned about the Spread of Coronavirus*. ABC News (2021). Available at: https://www.abc.net.au/news/2021-06-21/japan-unveils-its-coronavirus-safe-olympic-village/100225266 (Accessed July 21, 2021).
16. Dalton CB, and Taylor J. Are COVID-19-safe Tokyo Olympics and Paralympics Really Possible? *Med J Aust* (2021) 215:54–5. doi:10.5694/mja2.51159
17. Tokyo. *COVID-19 Positive Case List*. Tokyo Organising Committee of the Olympic and Paralympic Games (2021). Available at: https://olympics.com/tokyo-2020/en/notices/covid-19-positive-case-list (Accessed July 22, 2021).

Conflict of Interest: BC is supported by the AIR@innoHK program of the Innovation and Technology Commission of the Hong Kong SAR Government.

No potential competing interest from any commercial affiliation was reported by the authors.

The remaining author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers
in Physics

ORIGINAL RESEARCH
published: 25 October 2021
doi: 10.3389/fphy.2021.755932

Comprehensive Analyses of the Spatio-Temporal Variation of New-Energy Vehicle Charging Piles in China: A Complex Network Approach

*Maoze Wang, Fan Wu and Junhua Chen**

School of Management Science and Engineering, Central University of Finance and Economics, Beijing, China

This study collects data on electric vehicle (EV) charging piles for various provinces in China and analyzes the development of the network of EV chargers from the perspective of a complex network. Features of the distribution of EV charging piles for the period from May 2016 to April 2019 and the spatio-temporal variations across provinces are thus analyzed. The study then transforms time-series data of the EV charging piles into a complex network by applying a visibility graph, uses several clustering methods to categorize different provinces, and predicts the future development of the network of EV charging piles in China. Additionally, the distribution of EV charging piles across time is analyzed for a combination of national policies and new-energy vehicles. The results of the study will guide provincial governments in creating policies that develop relevant industries progressively and promote the sustainable development of EVs and green-energy industry.

Keywords: complex network, EV charging piles, visibility graph, policy effects, new energy vehicles

OPEN ACCESS

Edited by:
Peican Zhu,
Northwestern Polytechnical University, China

Reviewed by:
Jun Hu,
Fuzhou University, China
Fan Meng,
Peking University, China

****Correspondence:***
Junhua Chen
junhuachen@cufe.edu.cn

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *09 August 2021*
Accepted: *05 October 2021*
Published: *25 October 2021*

Citation:
Wang M, Wu F and Chen J (2021) Comprehensive Analyses of the Spatio-Temporal Variation of New-Energy Vehicle Charging Piles in China: A Complex Network Approach. Front. Phys. 9:755932. doi: 10.3389/fphy.2021.755932

INTRODUCTION

Electronic vehicles (EVs) are universally recognized as a practical solution to the problems of reducing carbon emissions and improving air quality in the global sector of transportation [1]. Many powerful economies are shifting their vehicle preference to electronic vehicles for eco-friendly purposes, and the development of the supporting infrastructure of EVs is rapidly progressing. However, short traveling distances and limited battery volumes due to current technical barriers are holding back the expansion of EVs, and the construction of EV chargers is thus considered the most effective way of promoting the adoption of EVs [2]. In recent years, European countries, along with the United States, have expanded their distributions of EVs and EV charging piles [3], as shown in **Figure 1**. From the viewpoint of the owners of EVs, the locations of EV charging piles are critical for the convenience of recharging EV batteries. Home chargers have the lowest cost, while public charging piles are becoming a necessary option for off-home charging [4]. Furthermore, public charging piles are mostly high power and provide faster charging in urban areas, which is more suitable for high-power charger installation than homes [5].

While wealthy countries are developing their EV infrastructure, China, a country with a large population and massive land area, is also creating a nationwide distribution of EVs. The charging pile industry is in full development with the expansion of the investment blueprint of new infrastructure in China. As new-energy vehicles are being promoted in China, the construction of charging piles, as important infrastructure, has gradually attracted attention. The central government, provinces, and

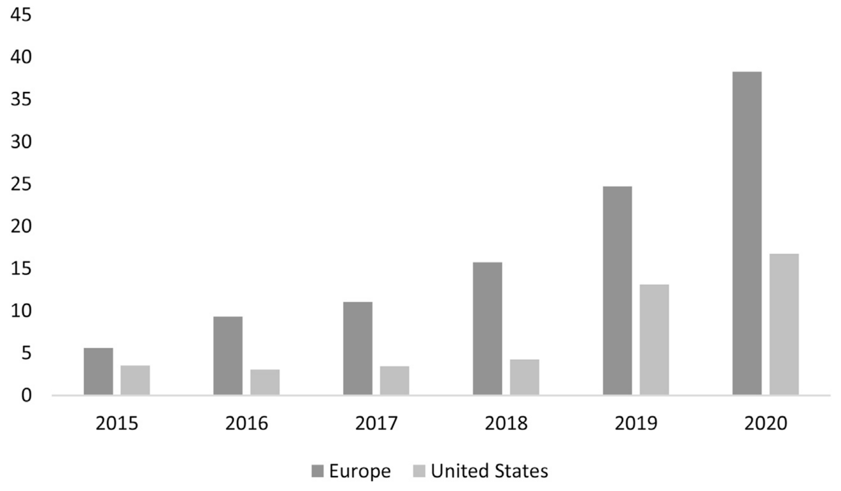


FIGURE 1 | Numbers of EV chargers in Europe and the United States (unit: thousands).

cities have successively introduced preferential policies and measures that promote the development of the charging pile industry, and the construction of charging piles in China has undergone explosive growth, from 33,000 piles in 2014 to 777,000 piles in 2018, which is growth of more than 200% in 4 years. Statistics show that the 2017 new-energy vehicle ownership, public charging pile number, car pile ratio compared with before 2012 decreased, but the rate of construction of charging piles is not keeping up with the manufacture of new-energy vehicles. China has built 55.7% of the world's new-energy charging piles, but the shortage of public charging resources and user complaints about charging problems continues. Additionally, there are many other problems; e.g., the layout of the charging pile is unreasonable, there is an imbalance between supply and demand, and the time required for investment to turn into profit is uncertain.

This paper gives a new perspective of complex network to study the growing distribution of EVs and charging piles in China. This study investigates the historical development of China's new-energy vehicles and charging piles from May 2016 to April 2019 and how local policies have affected the distribution of EVs in China. The data are analyzed by adopting time series visualization, complex networks, and several clustering methods. Combined with the model results, policies and characteristics of provinces, it is believed that the results of this study will provide a reference for the rapid development of charging piles in China.

The remainder of the paper is organized as follows. *Literature Review* reviews previous research on new-energy vehicles and piles. *Data and Model* presents the models and methods used in the paper, including the methodology and data collection. *Results and Discussion* presents the estimation results and analysis. *Results and Discussion* presents empirical research and analyzes features and reasons that lead to these results. *Conclusion* presents what we do in this study, our findings and expectations.

LITERATURE REVIEW

Environmental problems have become a major concern in recent years. Many papers have suggested that the cause of environmental problems, such as environmental deterioration and frequent haze, lies in automobile exhaust emissions, which has encouraged the development of new-energy vehicles and their related industries [6,7,8]. The traditional-automobile industry is driven by oil and consumes many precious resources. Therefore, the promulgation of appropriate policies that promote the innovative development of the new-energy vehicle industry will greatly help solve environmental problems.

However, there are many problems to be solved in developing new-energy vehicles. One problem is the development of new-energy charging technology while another is the gulf between the rate of manufacture of new-energy vehicles and the rate of construction of new-energy vehicle charging piles, which continues to grow.

Scholars have found that the construction of charging pile facilities plays a positive role in the development of new-energy vehicles. Policies supporting EV construction cultivate the EV market, with technical advances and subsidies in China promoting future progress of the EV industry [9]. [10] found that improving the supporting infrastructure has a more obvious effect on the market promotion of new-energy vehicles than factors of technological progress [11]. showed that the construction of charging pile infrastructure provides a stronger incentive to the new-energy vehicle market than government subsidies for vehicle companies. Improvements to charging piles and the supporting facilities of charging stations can affect the customer's intention to purchase new-energy vehicles [12–16].

There is a lack of relevant empirical studies in the literature, with most studies considering simulated scenarios. The situation in the simulated scenario tends to be more or less different from the actual situation. Additionally, most studies have focused on different factors and perspectives of the planning layout of the site selection, operation mode, and system improvements of charging piles whereas there have been few tracing studies on actual construction or studies providing a macroscale or comparative perspective.

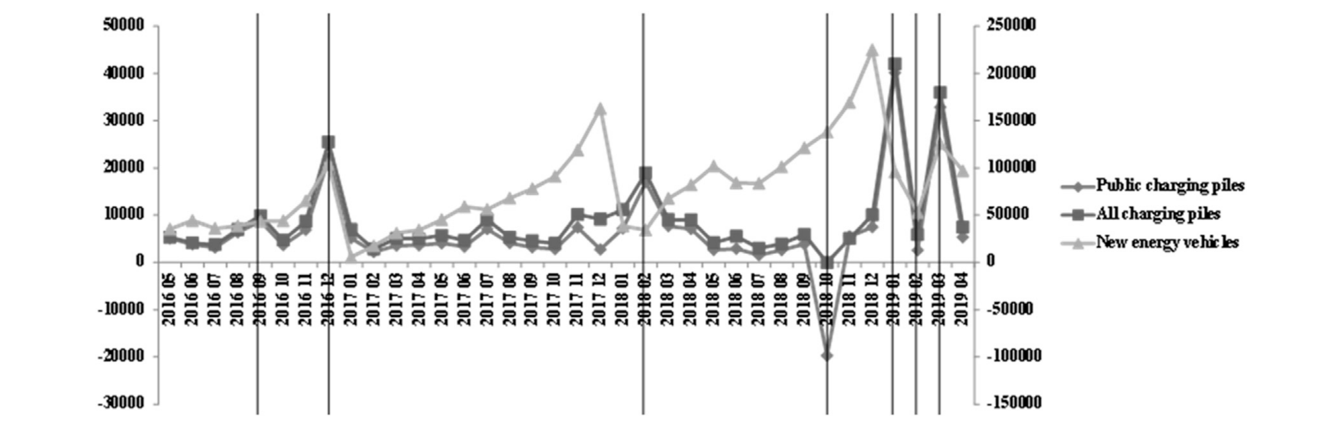


FIGURE 2 | Trends of the numbers of new-energy vehicles, all charging piles, and public charging piles.

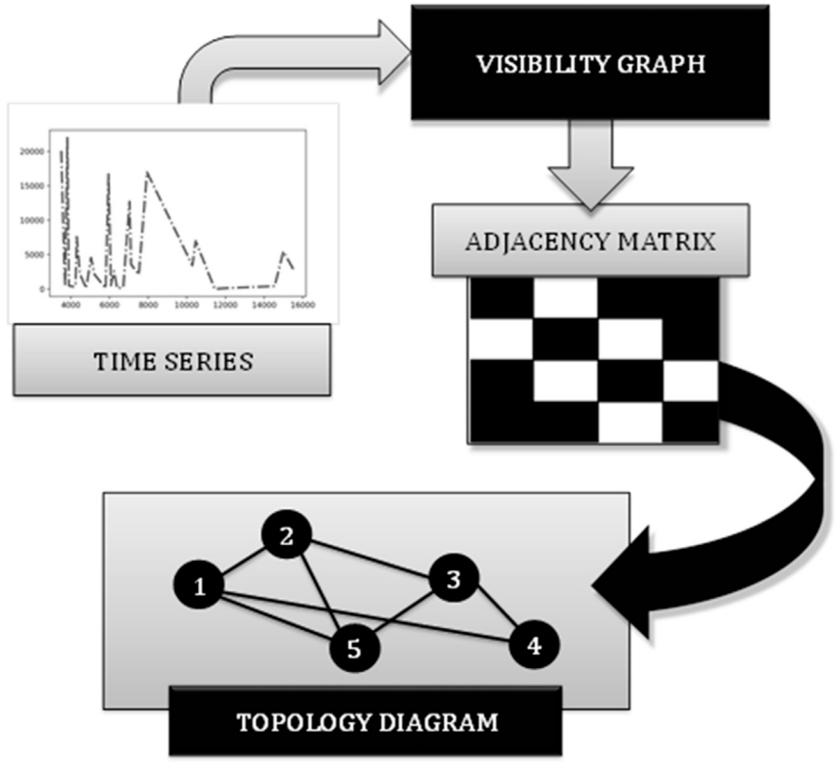


FIGURE 3 | Construction of the time-series network and extraction of features.

This paper adopts real-world data to conduct a visual network analysis of the overall development of new-energy vehicles and charging piles based on the Chinese background of the development of new-energy vehicle charging piles. In addition, considering the formulation of new-energy vehicles and charging pile development policies by province, complex network clustering analysis is conducted on data of the development of public charging piles in 31 provinces and cities in China. And **Figure 3** shows the process of construction of the time-series network and extraction of features.

DATA AND MODEL

Data

Data are collected from the National Bureau of Statistics of China and the China Electric Vehicle Charging Infrastructure Promotion Alliance[1]. Eliminating the missing data and outliers, this study

[1] All data of EVs in China were collected from online sources from website of EVCIPA (http://www.evcipa.org.cn/)

TABLE 1 | Policies relating to new-energy vehicles and charging piles.

Time	National ministries and commissions	The name of policy
2016/08	the National Development and Reform Commission and the National Energy Administration	The 13th Five-Year New Energy vehicle charging infrastructure incentive policy
2016/12	the National Development and Reform Commission and the Ministry of Housing and Urban-rural Development	Notice on accelerating the construction of charging piles and Supporting Facilities for electric vehicles in residential areas
2018/11	the National Development and Reform Commission and the National Energy Administration	Action Plan to Improve the Charging Capacity of New Energy Vehicles
2019/03	General Office of the Ministry of Industry and Information Technology and General Office of the State Development Bank	Circular of the General Office of the Ministry of Industry and Information Technology of the People's Republic of China on Accelerating Industrial Energy Conservation and Green Development

TABLE 2 | Statistical results of collected data of EVs, all charging piles, and public charging piles.

Attribute	New energy vehicles	All charging piles	Public charging piles
Mean	77688.36111	8853.86	6569.916667
Variance	2206957726	75863784.35	88279834.65
Standard Deviation	46978.27	8709.982	9395.735
Standard Error	7829.711	1451.664	1565.956
Kurtosis	4.157038	7.613911667	6.739286817
Sum	2796781	318739	236517
Median	67889.00	5807.00	4112.00
Minimum	5682	−14	−19701
Maximum	225000	42011	40181
Jarque-Bera	8.854177	103.1986	61.25077
Skewness	1.068182	2.617011	1.466934

analyzes the data of new-energy vehicles and charging piles in China for the period from May 2016 to April 2019.

Time series statistics of EVs in China are processed and generated in MATLAB algorithms, and Gephi has been applied to output the visibility graph and relevant coefficients.

Figure 2 shows obvious good trends of the manufacture of new-energy vehicles and charging piles on the whole. However, the difference in scale of the left and right vertical axes directly reflects the mismatch of the manufacture of charging piles and new-energy vehicles in China. The statistical results for all charging piles and public charging piles in China are similar, and different from those for new-energy vehicles.

The new monthly increase in the number of charging piles of new-energy vehicles had small peaks in September 2016, December 2018, February 2018, January 2019, and March 2019. **Table 1** shows that important relevant policies were launched before and after some of these peaks.

China's ratio of new-energy vehicles to charging piles still does not meet the requirements of the development guide. Accelerating the planning and implementation of the reasonable construction of charging piles is the cornerstone of further development.

Table 2 displays the statistical results of EVs, all charging piles, and public charging piles in China during 2016–2019.

Principles of Time-Series Visualization

A time series is a series of data points indexed by the observation time. Common tasks of time-series data mining are dimension reduction, similarity measurement, classification, cluster analysis, pattern discovery, and visualization. Different time-series analysis methods, such as chaos analysis, fractal analysis, recursion graph, complexity measurement, multi-scale entropy, and time-frequency representation, have been developed. In the past decade, scholars have increasingly adopted complex networks to analyze dynamic systems based on time series, such as investigating USA's electricity market, stock prices and even global efforts against terrorism [17-19].

The VG method adopted in this paper is based on the complex network model proposed by [20]. Time series can be divided into univariate time series (UTSs) and multivariate time series (MTSs) according to the number of variables. Traditional methods such as K-Shape and K-MS can be used for the rapid and accurate clustering and classification of UTS data sets but they are unsuitable for MTS data mining [21]. Used different frequencies as multiple variables to construct complex networks, detect community structure characteristics, and analyze the relationship between the clustering coefficient and system evolution. By constructing a common projection axis as the prototype of each cluster, the tail removal algorithm Mc2PCA of the method is given and its time complexity is analyzed.

The constructed new graph inherits essential properties and inherent features of the time-series data, allowing scholars to conduct analyses and further interpret the original data with the application of theoretical methods of complex networks and graph theory. The VG algorithm transforms a time series $\{x_i\}$, $i = 1,\ldots, n$ into a VG $G = (V,E)$, where $V(G) = \{v_i\}$, $i = 1,\ldots, n$ is a set of vertices with vertex v_i corresponding to data point x_i. $E(G)$ is

TABLE 3 | Network parameter analysis.

	New energy vehicles	All charging piles	Public charging piles
Edge	122	120	125
Average degree	6.778	6.667	6.944
Diameter	3	4	4
Average path length	2.075	2.111	2.129
Density	0.194	0.19	0.198
Modularity	0.444	0.422	0.428

the edge of the graph. We define *A = {a* $A = \{a_{i,j}\}$, $i, j = 1,\ldots, n$ as the adjacent matrix of the VG with $a_{i,j} = 1$ for connected vertices and $a_i,j = 0$ for disconnected vertices. The element $a_{i,j} = 1$ when the geometrical criterion

$$x_k < \left(x_i - x_j\right)\frac{t_i - t_k}{t_i - t_j} - x_i \tag{1}$$

is fulfilled.

The principle of the transition is stated as follows. The graph is deemed as a set of zeniths, which are nodes linked to each other by lines called edges. The numbers of new-energy vehicles and charging piles are first counted according to the set time. Statistical histograms are then produced accordingly. The height of the histogram reflects the volume at each time point or month from May 2016 to April 2019.

The bottom line is the criterion determining whether two points are set as being connected. The prerequisite for connecting the two points in the network is whether the peaks of the two histograms can be seen from each other (i.e., whether a straight line can connect the peaks without crossing all the histograms). It can then be transformed into the corresponding relationship between the two pairs, which is shown as the connection of each time points on the time dimension.

We next acquire the adjacency matrix by working on the time-series nodes and edges, and we calculate the various features of the subject networks.

By analyzing the original data, we visualize the time series data and obtain the complex network characteristics of the new-energy vehicles, all charging piles, and public charging piles.

As previously mentioned, we transform the time series data into complex network forms. The network parameters are given in **Table 3**. We find that among the three networks, the number of edges and average degree are largest for the public charging piles, reflecting that there are more peaks and troughs for these piles. The diameters of the networks of the new-energy vehicles, all charging piles, and public charging piles are respectively 3, 4, and 4; these values are the lowest number of edges between the two time points with the longest distance. The average path lengths are respectively 2.075, 2.111, and 2.129 for the three networks; these values indicate the average number of edges between any two time points.

Analysis of Centrality

Using data for the period from May 2016 to April 2019, we conduct a quantification analysis of the daily networks, including analyses of the degree centrality, betweenness centrality, eigenvector centrality, strength, and clustering coefficient of the complex network.

Degree of Centrality

The degree of centrality is the most direct measurement in the analysis of a network and is the simplest measure characterizing the connectivity properties of a single vertex in theory [22]. The calculation of the degree of centrality of a node is simply the counting of the number of edges associated with the subject node *n*. The degree of centrality of a node is positively correlated to the importance of the node in the network. In network *N* with *k* nodes, the degree of centrality of node *n*, denoted $D_c(n)$, is expressed as

$$D_c(n) = \frac{Deg(n)}{k-1}. \tag{2}$$

In graph theory, $\Theta(V^2)$ and $\Theta(E)$ are respectively the complexities of calculating the degree of centrality in the dense adjacency matrix and sparse adjacency matrix, where *V* is representative of all nodes and *E* makes reference to all edges.

The definition of centrality can be extended from the node to the graph [22]. Assume that n^* is the node with the highest degree of centrality in network *N*. $X:=(Y,Z)$ is then defined as the maximum:

$$H = \sum_{j=1}^{|n|}\left[D_c(n^*) - D_c\left(n_j\right)\right]. \tag{3}$$

The centrality of network *N* is defined as

$$D_c(W) = \frac{\sum_{j=1}^{|n|}\left[D_c(n^*) - D_c\left(n_j\right)\right]}{H}. \tag{4}$$

If a node is linked to all other nodes in network *W* and all other nodes only link to this central node, *H* of network *W* (which will be a star graph) reaches a maximum [22]. Here, $H = (k - 1)(k - 2)$, and the centrality of network *N* can be simplified as

$$D_c(W) = \frac{\sum_{j=1}^{|n|}\left[D_c(n^*) - D_C\left(n_j\right)\right]}{(k-1)(k-2)}. \tag{5}$$

Betweenness Centrality

The extent to which the location of a node is within the scope of other nodes on a graph is a measure of betweenness. Nodes that have higher betweenness centrality values are on the shortest path from other nodes.

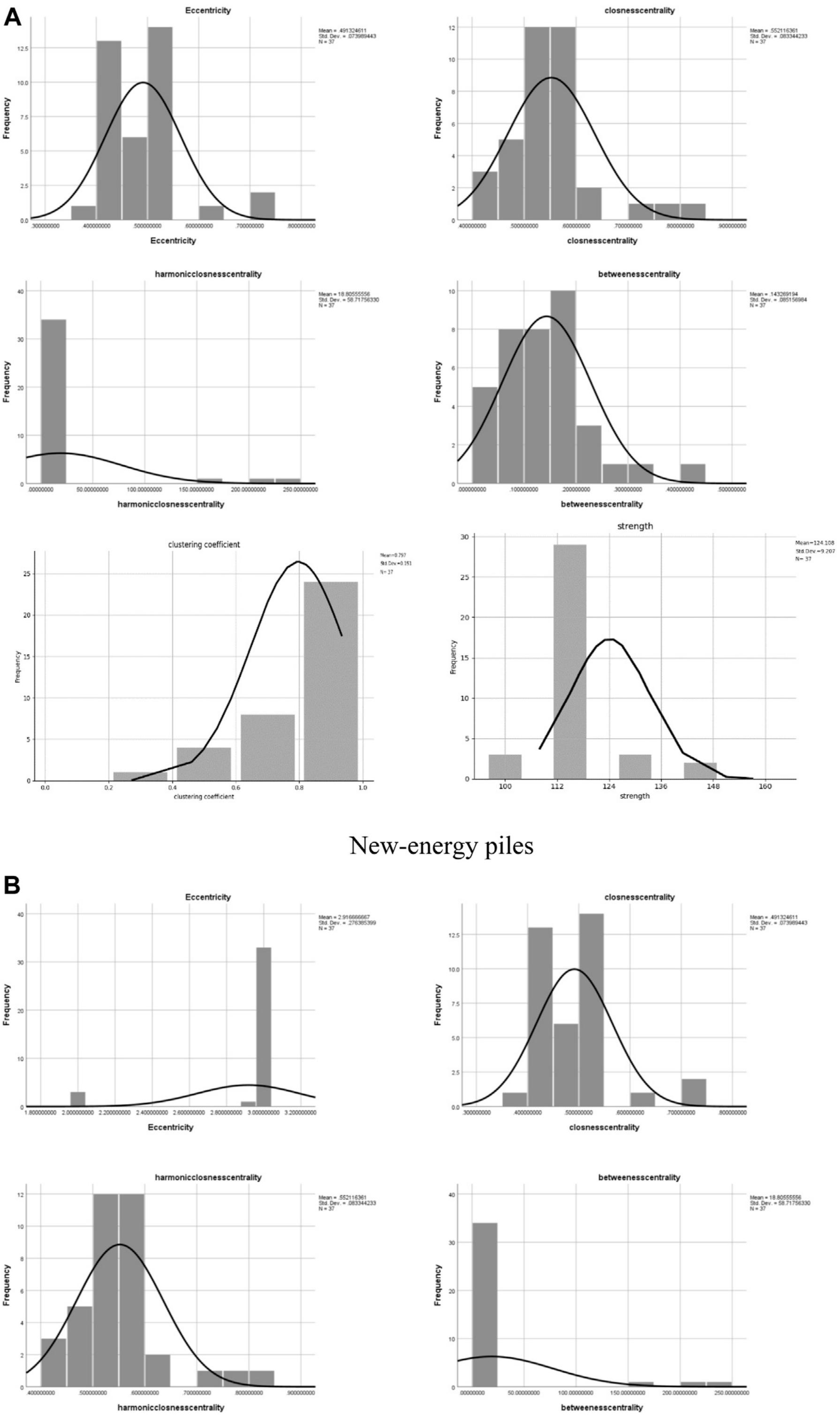
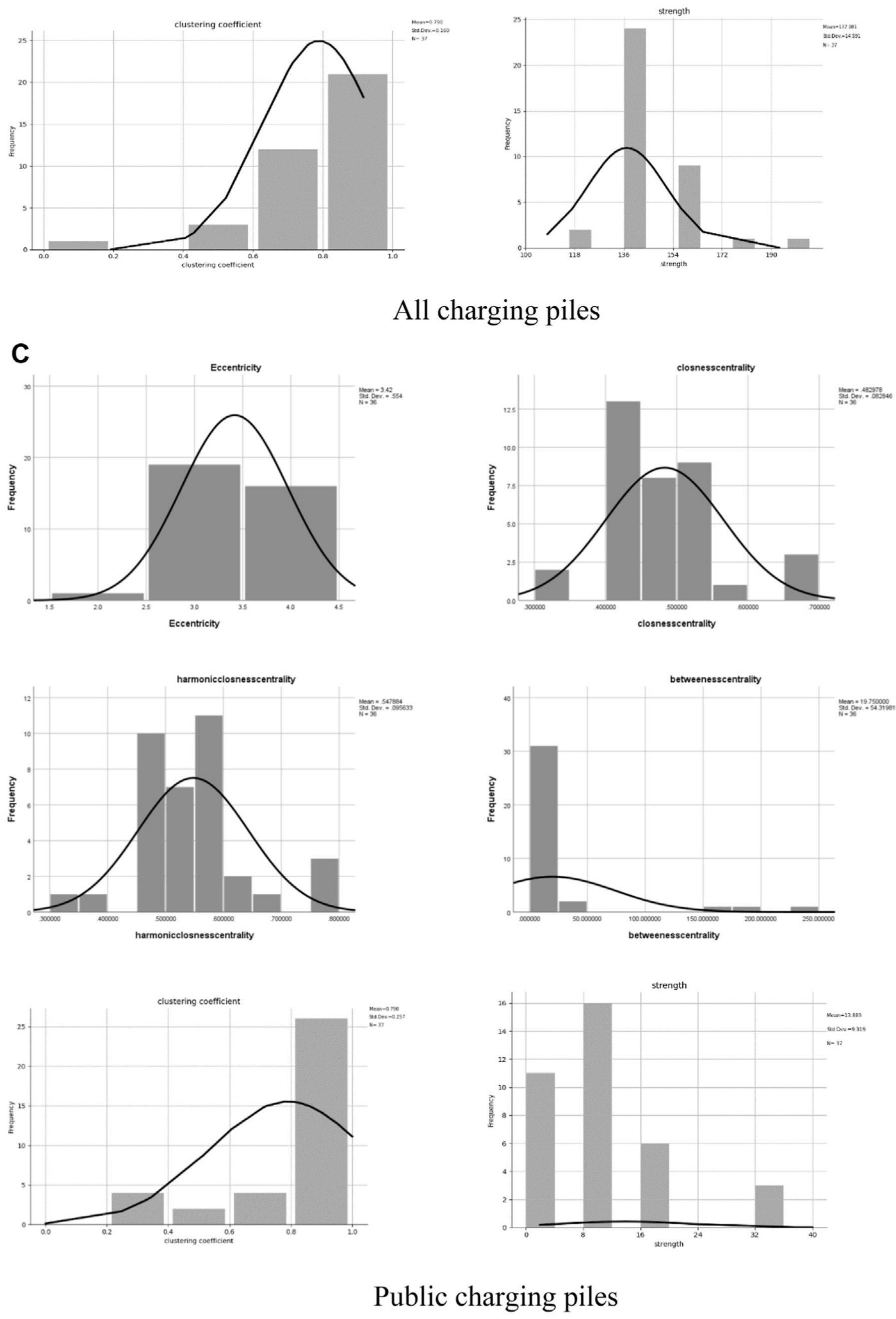


FIGURE 4 | Histogram of the frequency distribution of the centrality of **(A)** new-energy piles, **(B)** all charging piles, and **(C)** public charging piles. One presents the eccentricity, 2 presents the closeness centrality, 3 presents the harmonic closeness centrality, four presents the betweenness centrality, 5 presents the clustering coefficient, and 6 presents the strength.



All charging piles

Public charging piles

FIGURE 4 | (Continued)

According to Sanjiv and Purohit [22], in network *N* with *n* nodes, the betweenness centrality $B_c(n)$ of node *n* is calculated as follows. First, all the shortest paths of each node pair (*p*, *q*) are calculated, it is then evaluated whether node *n* is on the shortest path of each node pair (*p*, *q*), the results are finally cumulated. This process can be simplified as

$$B_c(n) = \sum\nolimits_{p \neq q \neq n \in N} \frac{\beta_{pq}(n)}{\beta_{pq}}, \tag{6}$$

where β_{pq} is the number of shortest paths between nodes *p* and *q* and β_{pq} *(n)* is the number of shortest paths passing through node *n*. Considering the scale of the network, it can divide the number of pairs without node *n* to normalization. It is $(k - 1)$ $(k - 2)$ for the directed graph and $(k - 1)$ $(k - 2)/2$ for the undirected graph.

The computations of the betweenness and closeness centrality are based on the computations of the shortest distance. In the search for the shortest path for each node pair, the modified Floyd–Warshall algorithm has complexity of $\Theta(W^3)$. On a sparse

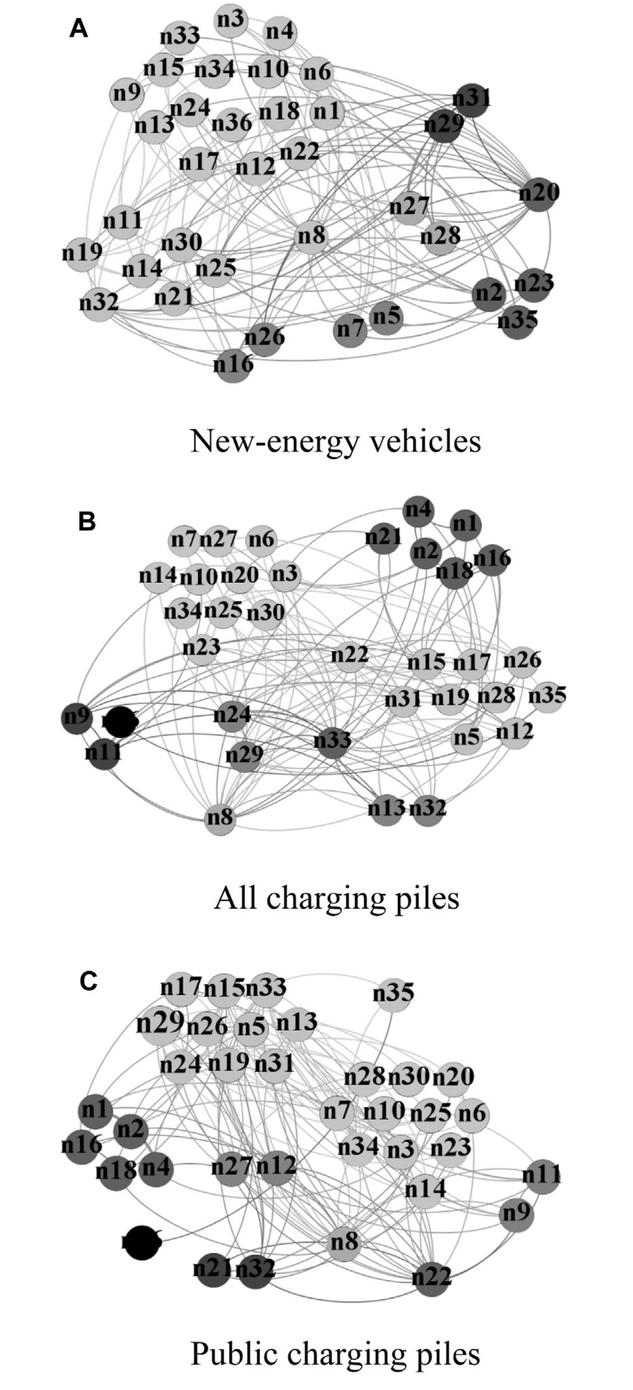


FIGURE 5 | Clustering of the three networks (where n1 to n36 represent individual months from May 2016 to April 2019).

graph, the efficiency of the Johnson algorithm exceeds $O(W2logW + WE)$. O(WE) is the efficiency of calculating the betweenness centrality on the weighted graph using the Brandes algorithm.

On an undirected graph, weighted edges should not be considered in the calculation of the betweenness and closeness centrality of nodes. More importantly, the norm of graph processing is not to use rings or weighted edges to render relationships simple. In these circumstances, adopting the Brandes algorithm will halve the ultimate centrality owing to the double calculation of the shortest path.

Closeness Centrality

According to graph theory, closeness is a measure of the complex centrality of a node. Shallower nodes (i.e., nodes having shorter geodesic distances) have higher values of closeness. The nodes that are more central have higher closeness values, and the closeness thus represents the minimum path length in the network. Additionally, closeness is often related to other measurements. The closeness centrality is the average geodesic distance (e.g., shortest path) from node n to other accessible nodes [22]:

$$C_C = \frac{\sum_{q\in N\setminus n} d_{W\,(n,q)}}{k-1}, \tag{7}$$

where $k \geq 2$ is the distance of access section N from node n in the network. The closeness centrality is a measure of the time that it takes for a given node to propagate information to other reachable nodes in a network. The closeness centrality $C_C(n)$ of node n is defined as the reciprocal of the sum of the geodesic distances to all other nodes N [22]:

$$C_c = \frac{1}{\sum_{q\in N\setminus n} d_W(n,q)}\ . \tag{8}$$

Closeness can be obtained using different methods and algorithms. Dangalchev [23] modified the definition of closeness so that it can be applied to a non-connected graph and is easier to calculate:

$$C_c = \sum_{q\in N\setminus n} 2^{-d_W\,(n,q)}. \tag{9}$$

Eigenvector Centrality

Most hub nodes are found in line with the integral structure of the network, and the eigenvector centrality is then measured. The dimensions of the distance between nodes are acquired through factor analysis. Each node in a network has a relative index value based on the principle that the contribution of a high-index node connecting to a node is more than that of a low-index node [24].

Let p_i be the (exponential) value of node i and $A_{i,j}$ be the adjacency matrix of the network. When node i is the neighboring node of node j, $A_{i,j} = 1$, or conversely, $A_{i,j} = 0$. Generally, like the case of a random matrix, each term of A can be a real number representing the connection strength. For node i, the centrality is proportional to the index sum of the nodes connecting with it. Thus,

$$p_{i=}\frac{1}{\lambda}\sum_{j\in M(i)} p_j = \frac{1}{\lambda}\sum_{j=1}^{N} A_{i,j}p_j, \tag{10}$$

where $M(i)$ is the set of nodes connected to node i, N is the number of nodes, and λ is a constant. The matrix form is $P = \frac{1}{\lambda}AP$, whereas the characteristic equation is $AP = \lambda P$.

TABLE 4 | Parameters of power law distributions.

	New energy vehicle	All charging piles	Public charging piles
SSE	1.98	9.562	11.5
R-square	0.794	0.8509	0.7799
RMSE	0.3518	0.7984	0.8755

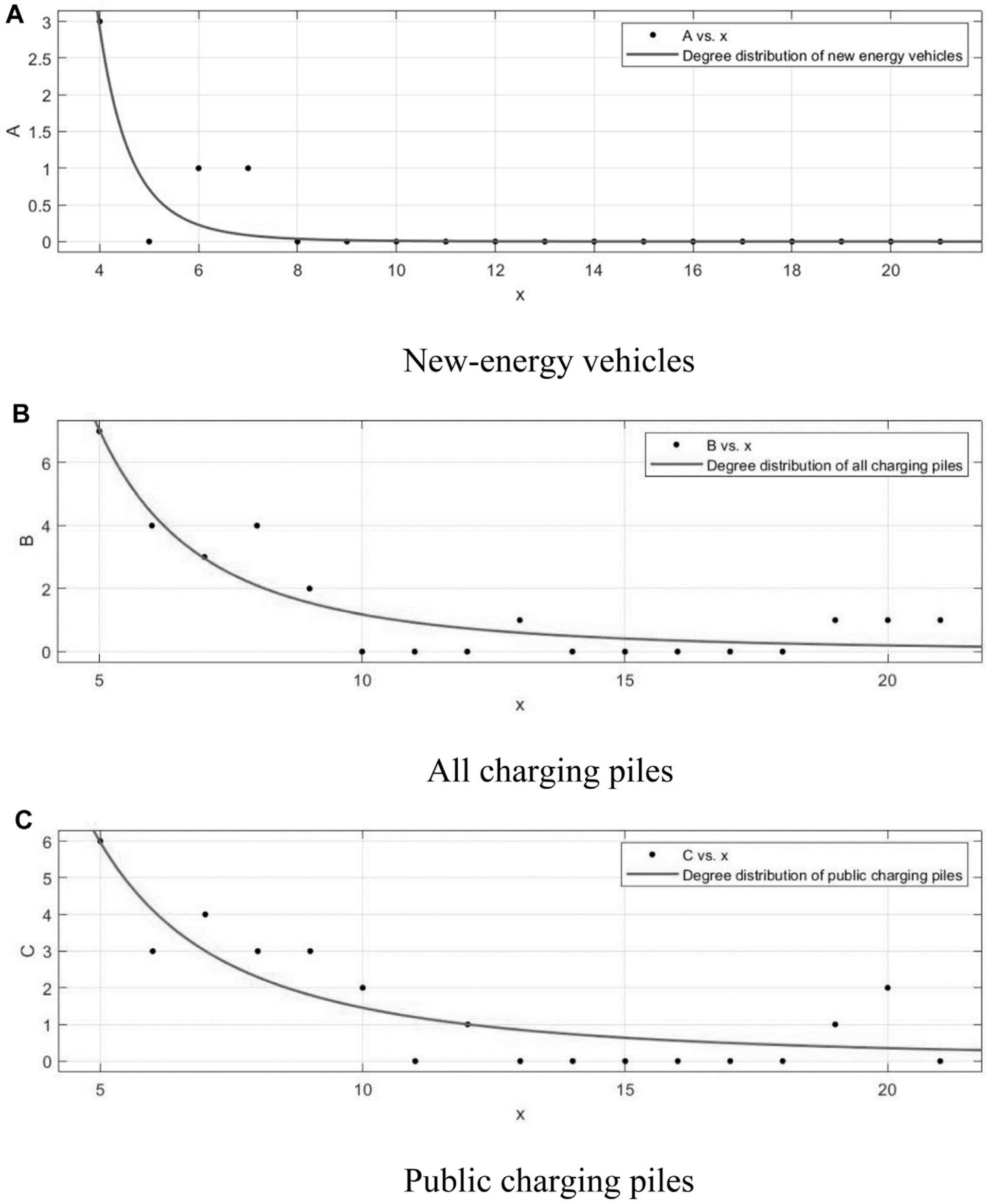


FIGURE 6 | Degree distributions of new-energy vehicles, all charging piles, and public charging piles.

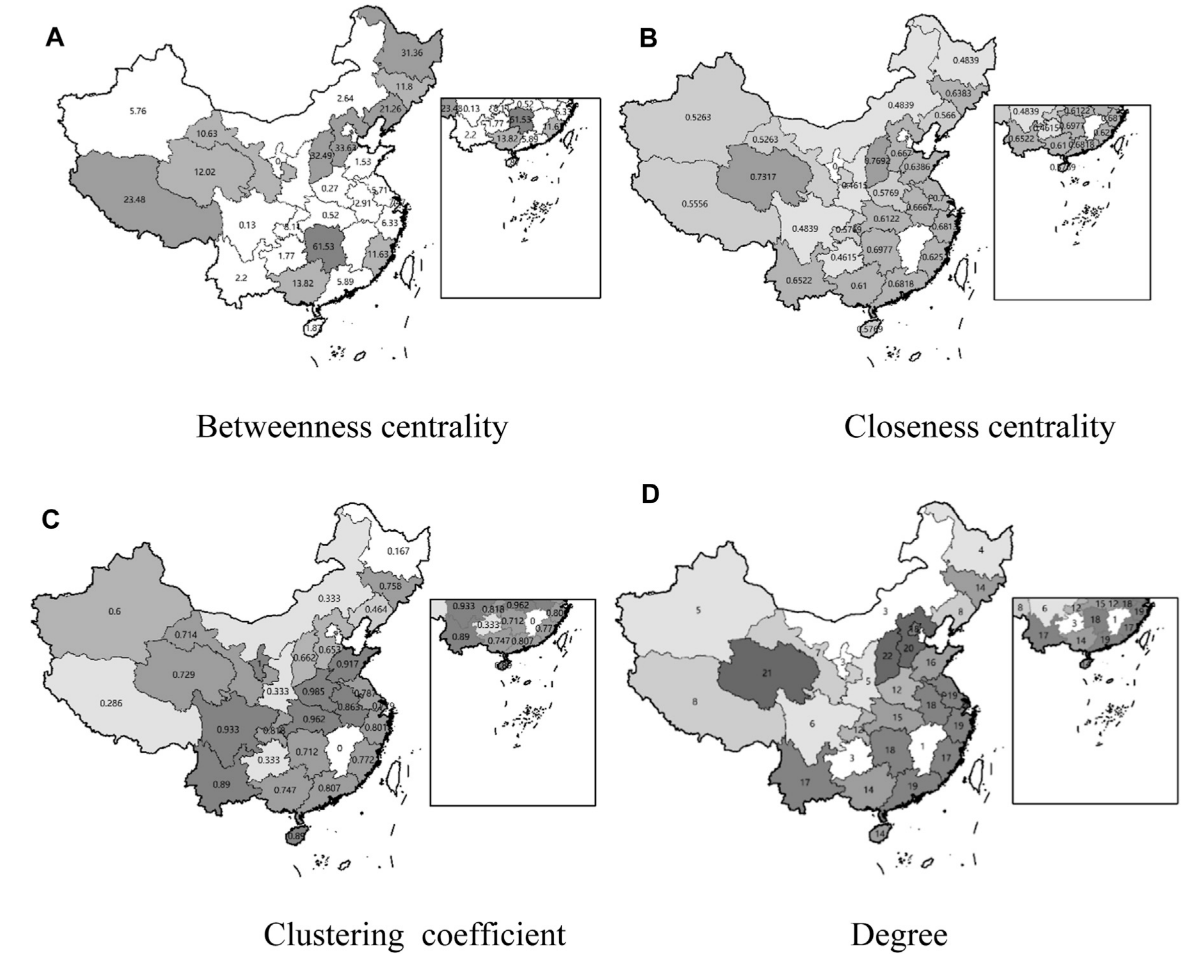


FIGURE 7 | (A) Betweenness centrality, **(B)** closeness centrality, **(C)** clustering, and **(D)** degree of centrality for all provinces in China.

Strength

In a directed and weighted network, the strength denotes the total weights of the edges connecting to one node [25]. In this paper, the strength is a measure of the number of EVs in different provinces. The strength is calculated as

$$S_i = \sum\nolimits_{j \le N_i} w_{i,j}, \tag{11}$$

where N_i is the set of nodes connected to node i and $w_{i,j}$ is the weight of the edge from node j to node i.

Clustering Coefficient

The clustering coefficient describes the characteristics of the graph (or network). A graph G consists of a number of vertices V and a number of lines (called edges) E between vertices. Two adjacent vertices are called adjacent points. The clustering coefficient of a network is defined as [25]:

$$C(N) = \frac{3 \times number\,of\,triangles\,in\,the\,network}{number\,of\,connected\,triples\,of\,vertices}. \tag{12}$$

Clustering

Applying complex network theory to the primal data, the relations within the data are represented on a graph as nodes and edges. In this way, there is a great advantage over the traditional static method in that we can capture the dynamic features and community structures. Furthermore, the nodes and edges can be clustered into different groups. The clustering is an analyzable phenomenon in that the correlation between the set of nodes in a subgroup is higher than that outside the group. In an attempt to obtain an effective clustering result, we adopt a widely used principle proposed by Newman [26]. The modularity Q is formularized as

$$Q = \sum_{i=1}^{K}\left[\frac{l_i^{in}}{L} - \left(\frac{d_i}{2L}\right)^2\right] = 1 - \frac{L_{\text{inter}}}{L} - \frac{1}{K} - \frac{1}{K}\sum_{j-2}^{K}\sum_{k=1}^{j-1}\left(\frac{d_j - d_k}{2L}\right)', \tag{13}$$

where K is the number of subgroups, L is the number of edges, and l_i^{in} and $d_i = l_i^{in} = l_i^{\text{inter}}$ are respectively the number of edges in the corresponding subgroup and the total number of sides of the cluster i. $L_{\text{int}er}$ on behalf of the total number of intergroup edges. The modularity Q is calculated as follows. We first calculate the

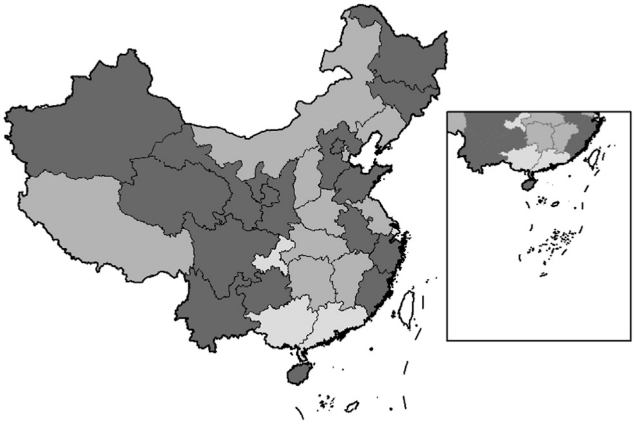


FIGURE 8 | Clustering results of public charging piles shown by province.

number of inner edges in the subgroup minus the expectation of the same number of edges falling in the random network without the clustering structure. The value of *Q* indicates the quality of the clustering. The obviousness of the clustering structure is positively dependent on the value of Q.

RESULTS AND DISCUSSION

Analysis of Network Properties

Analysis of Centrality

Following the analysis of the degree of centrality of the complex networks, three network centricities are calculated and six indicators are presented in frequency diagrams; these are the eccentricity, closeness centrality, harmonic closeness centrality, betweenness centrality, strength, and clustering coefficient.

A comparison of the three eccentricities reveals that in the complex network of new-energy vehicles, the eccentricity is low and the distribution is relatively uniform. The eccentricity of the network nodes of charging piles is mainly around 3 or 4, and the distribution of all charging piles is especially concentrated. This reflects the rapid manufacture of new-energy vehicles, whereas the rate of manufacture of charging piles is relatively stable.

The three networks have similar distributions of closeness centrality and betweenness centrality, and it appears that they all have the right-hand bias. In the field of topology and related mathematics, closeness is an elementary concept of the topological space. Intuitively, when two sets are arbitrarily close, they are said to be tight. This concept is easy to adopt in a metric space that defines the distance between elements within a space, but it is difficult to extend to a topological space without a specific metric distance. In network analysis, closeness represents the minimum path length, which means that in the development of the three networks over the 3 years, there is a high possibility of there being extreme quantitative values. Additionally, the strengths and clustering coefficients of the three networks are similar, with medium values having the highest frequency, which indicates that the development of the network of EV charging piles is steady and provinces in China are well connected and coordinated in the advance of EV infrastructure. The results match those of the analyses in the first part above.

Figure 4 shows that the centrality distributions of new-energy vehicles and charging piles are somewhat similar and that they are in the process of coordinated development. An increase in the penetration rate of new-energy vehicles requires a foundation of a sufficient number of public charging piles.

Small World

As explained earlier, we use a fast modular method to cluster nodes in the networks. The results are shown in **Figure 5**. The densities of the three networks and the number of subgroups are similar.

Although the numbers are largely similar, we find that there are more subgroups in the network of new-energy vehicles than in the network of charging piles. This is because when we conduct the clustering, if data are relatively flat with a lower peak value, the network distance increases over time, such that the clustering results have more subgroups.

Figure 5 shows that, in the three clustering networks, N8 is distinct, which is consistent with the peak in December 2016 for the underlying trend. At this time, the government issued a notice on accelerating the construction of charging piles and supporting facilities for EVs in residential areas.

Overall, however, the results of clustering in these three networks reveal that the development of the network of EVs and that of the network of the piles are in fact inconsistent. The nodes in each subgroup are largely different, and there is therefore still much to do for the pace of manufacture of piles to catch that of EVs.

Distribution of the Degree of Centrality

The power law distribution is a common statistical phenomenon. Fitting parameters for power law distributions are given in **Table 4**.

Figure 6 and **Table 4** show that the distributions of the degree of centrality of new-energy vehicles, total charging piles, and public charging piles follow power laws, with more time nodes and fewer connected edges, and the number of nodes decreases with an increase in the degree of centrality. This clearly indicates that the networks of new-energy vehicles and charging piles are small-world networks and that the manufacture of new-energy vehicles and charging piles will be greatly affected by external factors at critical moments.

The sales of fuel-powered cars were in the middle of a slump in 2018, but China's new-energy automobile market grew in 2018 relative to sales in 2016 and 2017. This contrast is closely related to a number of new-energy vehicle subsidies (e.g., a tax exemption for the purchase of new-energy automobile vehicles, tariff cuts of reversed transmission enterprise technology upgrades, and double integral policy). This correspondence shows the importance of promoting government policy.

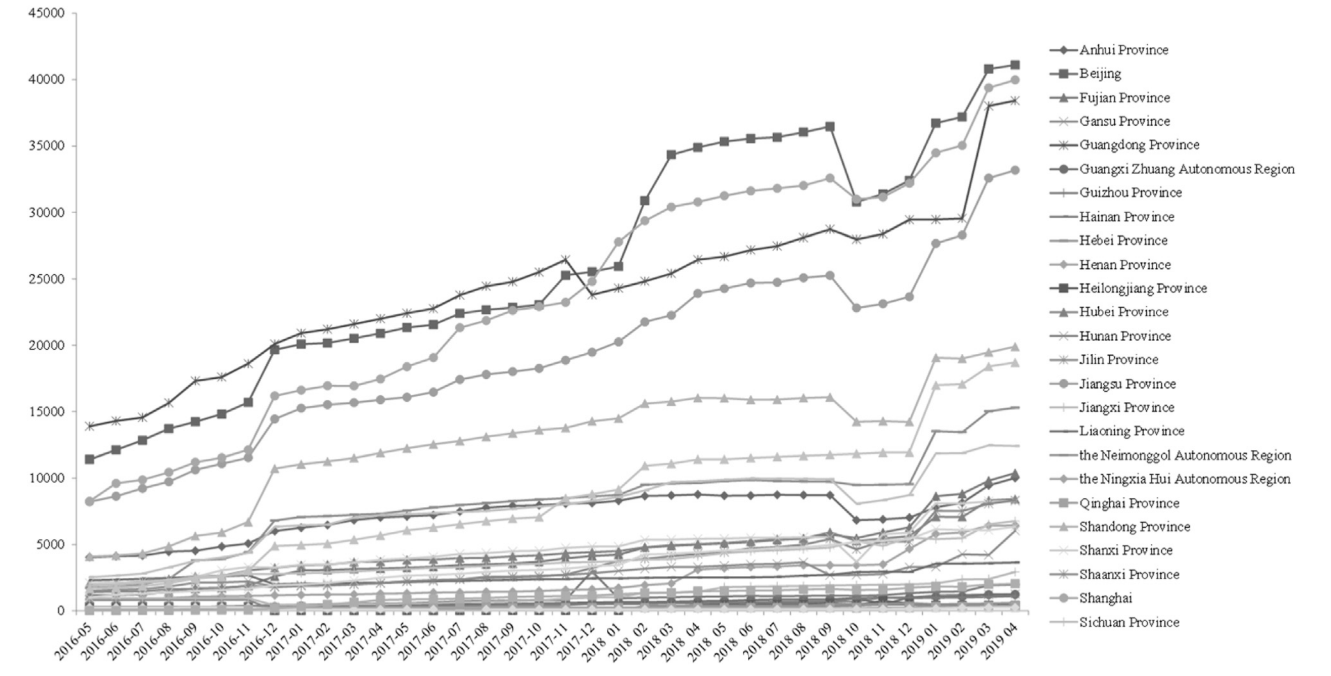


FIGURE 9 | Monthly increases in the number of public charging piles in all provinces.

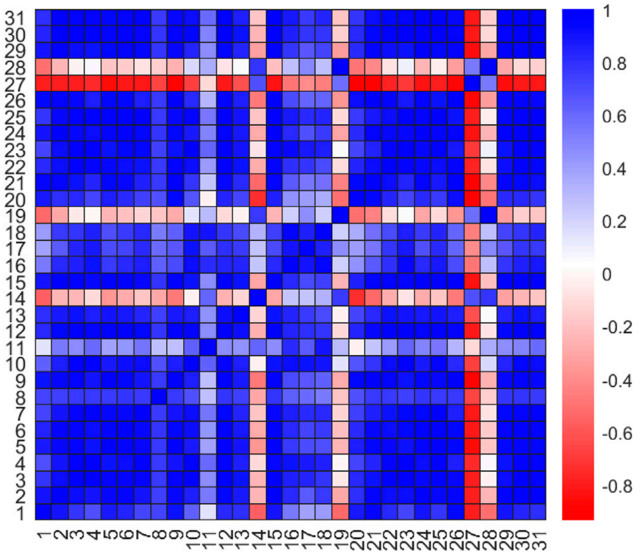


FIGURE 10 | Heat map of the similarity of provinces.

Complex Network Clustering of Provinces

Figure 7 shows that in the development of charging pile networks, Hunan Province had the highest betweenness centrality, Jiangxi Province had the lowest closeness centrality, clustering, and degree, and Shanxi Province and Qinghai Province had the highest degree of centrality.

The developmental trends for 2016 to 2019 can be divided into four stages, which are basically the four natural years. The figure shows that the manufacturing of new-energy vehicles and charging piles in China is accelerating year by year.

The visualization of the monthly increase in the number of public charging piles for China's new-energy vehicles in **Figure 8** shows that the clustering results for China's provinces can be divided into three categories.

The first category includes Anhui Province, Beijing, Fujian Province, Gansu Province, Guangdong Province, Hainan Province, Hebei Province, Henan Province, Hubei Province, Jiangsu Province, Qinghai Province, Shandong Province, Shanxi Province, Shanghai, Tianjin, Yunnan Province, Zhejiang Province, and Chongqing. The results show that the closeness centrality and degree of centrality of these provinces, which are accelerating development areas in *Guidance on the Development of Electric Vehicle Charging Infrastructure 2015–2020* issued by the National Development and Reform Commission of China in 2015, are relatively high. These provinces have a good foundation for the development of EVs in that they have a large population base and a high population density and require intensive haze control. The local governments of these provinces have formulated and implemented relevant policies earlier and more frequently to guide the development of new-energy vehicles and charging piles. In addition, Beijing, Tianjin and Hebei, and the Yangtze River Delta and the Pearl River Delta are three key areas for haze prevention and control. In particular, Beijing, Shanghai, Jiangsu, and Guangdong have formulated many policies of promoting new-energy vehicles and charging piles for different application scenarios.

In **Figure 9**, Beijing is at the top of the number of public charging piles. In April 2019, the number of public charging piles in Beijing reached 930,000, in particular because of the serious air pollution and the urgent need for governance. Local government in Beijing has issued a series of policies related to car purchase subsidies and welfare for new-energy vehicles. As an example, there is a lottery for the purchase of new-energy vehicles, which drives the use of new-energy vehicles. Additionally, Beijing considered the construction and use of charging piles earlier and more thoroughly than other provinces. In Shunyi District of Beijing, construction units of public charging facilities that meet the requirements of the state and municipality may apply for government subsidies, and new-energy vehicles using public charging piles are given a charging service fee subsidy. Overall, Beijing's new-energy development is policy driven.

Meanwhile, the use of new-energy vehicles and charging piles in Guangdong Province, which ranks the second, is technology driven. Guangdong Province is home to many high-tech new-energy car manufacturers, such as China's leading new-energy car company, BYD, which is headquartered in Shenzhen. The local technological atmosphere supports the development of new-energy cars and charging piles in the province.

Qinghai Province, Gansu Province, and Yunnan Province have weak industrial foundations, insufficient research and development capacities, insufficient promotion policies, lagging infrastructure, and unimproved market environments. However, they all have a place in the industrial chain of energy resources and new-energy vehicles and charging piles. These provinces are energy-driven. Although the development of charging piles in Qinghai Province started relatively late, the province's clean-energy resources have broad application prospects in the province's new-energy vehicle charging service business, and there are thus similarities between Qinghai Province and areas of accelerating development in terms of the overall development trend. Additionally, Yunnan Province has unique advantages because of its energy resources. The local government of Yunnan Province attaches importance to the active development of smart services, closely follows the pace of development in the region, formulates and implements development plans, and is committed to combining the tourism resources of the province with the development of new-energy vehicles. For instance, at tourist distribution centers and key scenic spots, tourist buses, shared cars, and self-driving camps (bases) will be built, and regional charging networks will be created to realize intelligent travel, "a mobile phone to travel in Yunnan". Therefore, although there is a gap between the number of charging piles and the provinces in the eastern region, the trend of development is fast.

The second category includes the Xinjiang Uygur Autonomous Region, Tibet Autonomous Region, Inner Mongolia Autonomous Region, Shaanxi Province, Liaoning Province, Jilin Province, and Heilongjiang Province. These areas are mainly northwestern and northeastern provinces in which the development of the charging pile network started relatively late, mostly from 2017 to 2019. We take the northeastern provinces as an example. Many automobile industry bases were set up in northeastern China when new China was first founded. However, the technologies of fuel energy are now somewhat backward, and the existing industrial base in northeastern China has resulted in a slow conversion from old to new ways of generating kinetic energy. In addition, as established industries are important to local employment, the local government pays little attention to new industries, and the development of new-energy vehicles and charging piles has been slow. The northwestern provinces and regions, such as the Xinjiang Uygur Autonomous Region, Tibet Autonomous Region, and Inner Mongolia Autonomous Region, are characterized by a vast areas of land and sparse populations. Moreover, there are many ethnic minorities and strong ethnic traditions. The new-energy vehicle market space is small and the costs of constructing charging piles are high in these regions. The cities have weak development potential except for the provincial capitals and some larger cities.

In contrast with accelerating the construction of charging piles in developing regions, the main purpose of constructing public charging piles in the regions of the second category is to further improve the convenience of transportation, thus strengthening connectivity, accelerating regional development, and gradually building a national inter-city fast-charging network based on expressways.

The third category includes Sichuan Province, Guizhou Province, and the Guangxi Zhuang Autonomous Region. The development of the new-energy vehicle charging pile network began reasonably early, around 2016, in each of these three provinces. However, none of the provinces has advantages in the industrial chain, and the automobile industry is weak in these provinces. At the same time, owing to the renewal of new-energy vehicles in the eastern regions, old fuel-based vehicles have been eliminated, which have emission specifications superior to the existing ones in the western region. These old vehicles are therefore flowing into the western market because their second-hand transaction prices are lower, squeezing the already insufficient space for EVs in the car market. In addition, the geographical conditions of these provinces are a major disadvantage to the adoption of new-energy vehicles and charging piles. Rugged terrain accelerates the power consumption of new-energy vehicles, and the planning and construction efficiency of charging piles is limited. There is thus little motivation to purchase new-energy vehicles, and the overall development of the EV network is slow relative to the development of the economic base.

CONCLUSION

This paper used time series data for May 2016 to April 2019 to build a complex network and used characteristic data for the network to obtain supplementary information, so as to establish a new effective connection between the time series and the complex network. The results show that the overall speeds of development of networks of new-energy vehicles and charging piles in China are similar, but the speed of development of the charging pile network is relatively slow. Moreover, both the networks of new-energy vehicles and charging piles are greatly affected by special events, such as policy implementations, and it is thus crucial to formulate policies that can be effectively implemented.

China is in the background of "New Infrastructure". We carried out cluster analysis on provincial data of public charging piles after time-series visualization, considering that relevant industrial development policies are mostly formulated by provincial governments. The results of the research are summarized as follows. 1) Regional factors play a dominant role in the development of networks of new-energy vehicles and charging piles. A basic regional characteristic of China is that the eastern provinces are more developed than the western provinces, which is obviously reflected in the clustering results. The eastern developed provinces, with a high degree of urbanization, high population densities, and superior economic foundations, have good application conditions for the development of networks of new-energy vehicles and charging piles in that they have a broad market space and rapid socioeconomic development. 2) Whether a province occupies a place in the industrial chain of new-energy vehicles and charging piles and how important it is in the industrial chain strongly affect the situation of local construction. In the case of the upstream provinces, the reserves of energy resources and the difficulty of collection are important. In the case of the downstream provinces, the ability to conduct research and development and the technical level of relevant enterprises are important. 3) National and local industrial policies play an important role in the development of networks of new-energy vehicles and charging piles. In 2018, various ministries and commissions in China issued a series of policies that promoted the rapid development of networks of new-energy vehicles and charging piles. As an example, the Ministry of Industry and Information Technology issued the Notice on Strengthening the Administration of The Catalogue of New-energy Vehicles Exempted from Vehicle purchase Tax (Draft for Comments). Additionally, the Ministry of Finance, Ministry of Science and Technology, and Development and Reform Commission issued "On the adjustment to perfect the new energy automobile application finance subsidy policy notice", a national department would raise technical threshold requirements, perfect the subsidy standard requirements, and adjust the operating range to the new-energy vehicles subsidies.

Overall, the outlook of the domestic new-energy vehicle market in China remains good, and the development potential is extremely large. With the replacement of social energy and on the basis of the good development prospects of China's new-energy vehicles, charging piles will inevitably be adopted broadly as the supplemental energy infrastructure of new-energy vehicles. Provinces that are developing rapidly need to further improve the efficiency of charging pile construction. The construction of charging piles of new-energy vehicles and the development of new-energy vehicles promote and restrict each other. To further develop the network of new-energy vehicles, the premise must be to reduce the ratio of the vehicle pile. Provinces that are developing slowly need to upgrade their industrial structures in light of local conditions and enhance the conditions for new-energy applications. **Figure 10** shows the similarity of EV development between provinces in China.

We suggest that government have an in-depth understanding of the local application basis, geographical factors, cultural factors, and other application conditions before making future policies. The government should set reasonable development goals, actively implement a subsidy policy for the construction of new-energy vehicle charging piles, and scientifically guide the construction of EV charging infrastructure. We suggest that in the future construction of charging piles, enterprises consider reasonable construction that balances supply and demand and combines with new intelligent infrastructure and the Internet of things. As an example, the sharing mode can be combined with the operation of charging piles. Through win–win cooperation among the government, enterprises, and users, it will be possible to promote the rapid development of the EV industry and create a better air environment.

The present study adopted a single index to analyze China's use of new-energy vehicles and charging piles owing to the limitations of the data breadth, scale, and accuracy. In future studies, we will further collect data of relevant indicators and use complex networks in coupled analysis to explore in detail the reasons for variations in development across provinces and cities.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

MW conceived the research. FW designed the analyses and compiled the data. JC conducted the analyses. MW, FW and JC wrote the paper. All authors read and approved the final manuscript.

REFERENCES

1. Huo H, Zhang Q, Wang MQ, Streets DG, and He K. Environmental Implication of Electric Vehicles in China. *Environ Sci Technol* (2010) 44(13):4856–61. doi:10.1021/es100520c
2. Gong H, Wang MQ, and Wang H. New Energy Vehicles in China: Policies, Demonstration, and Progress. *Mitig Adapt Strateg Glob Change* (2013) 18(2): 207–28. doi:10.1007/s11027-012-9358-6
3. IEA. *Stock of Fast Public Electric Light Duty Vehicles Chargers, 2015–2020*. Paris: IEA (2020). Available at: https://www.iea.org/data-and-statistics/charts/stock-of-fast-public-electric-light-duty-vehicles-chargers-2015-2020 (Accessed July 20, 2021).
4. Hardman S, Jenn A, Tal G, Axsen J, Beard G, Daina N, et al. A Review of Consumer Preferences of and Interactions with Electric Vehicle Charging Infrastructure. *Transportation Res D: Transport Environ* (2018) 62:508–23. doi:10.1016/j.trd.2018.04.002
5. Faria MV, Baptista PC, and Farias TL. Electric Vehicle Parking in European and American Context: Economic, Energy and Environmental Analysis. *Transportation Res A: Pol Pract* (2014) 64:110–21. doi:10.1016/j.tra.2014.03.011
6. Schuitema G, Anable J, Skippon S, and Kinnear N. The Role of Instrumental, Hedonic and Symbolic Attributes in the Intention to Adopt Electric Vehicles. *Transportation Res Part A: Pol Pract* (2013) 48:39–49. doi:10.1016/j.tra.2012.10.004

7. Larson PD, Viáfara J, Parsons RV, and Elias A. Consumer Attitudes about Electric Cars: Pricing Analysis and Policy Implications. *Transportation Res Part A: Pol Pract* (2014) 69:299–314. doi:10.1016/j.tra.2014.09.002
8. He J, Zhang L, Hu R, and Xie Y(2020). "Optimal Site Selection Planning of EV Charging Pile Based on Genetic Algorithm," in: 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC); June 12–14, 2020; Chongqing, China. 1799–803.
9. Zhang X, and Bai X. Incentive Policies from 2006 to 2016 and New Energy Vehicle Adoption in 2010-2020 in China. *Renew Sustain Energ Rev* (2017) 70: 24–43. doi:10.1016/j.rser.2016.11.211
10. Lee DH, Park SY, Kim JW, and Lee SK. Analysis on the Feedback Effect for the Diffusion of Innovative Technologies Focusing on the green Car. *Technol Forecast Soc Change* (2013) 80(3):498–509. doi:10.1016/j.techfore.2012.08.009
11. Sierzchula W, Bakker S, Maat K, and van Wee B. The Influence of Financial Incentives and Other Socio-Economic Factors on Electric Vehicle Adoption. *Energy Policy* (2014) 68:183–94. doi:10.1016/j.enpol.2014.01.043
12. Coad A, de Haan P, and Woersdorfer JS. Consumer Support for Environmental Policies: An Application to Purchases of green Cars. *Ecol Econ* (2009) 68(7):2078–86. doi:10.1016/j.ecolecon.2009.01.015
13. Zhang X, Wang K, Hao Y, Fan J-L, and Wei Y-M. The Impact of Government Policy on Preference for NEVs: The Evidence from China. *Energy Policy* (2013) 61:382–93. doi:10.1016/j.enpol.2013.06.114
14. Li W, Long R, and Chen H. Consumers' Evaluation of National New Energy Vehicle Policy in China: An Analysis Based on a Four Paradigm Model. *Energy Policy* (2016) 99:33–41. doi:10.1016/j.enpol.2016.09.050
15. She Z-Y, Sun Q, Ma J-J, and Xie B-C. What Are the Barriers to Widespread Adoption of Battery Electric Vehicles? A Survey of Public Perception in Tianjin, China. *Transport Policy* (2017) 56:29–40. doi:10.1016/j.tranpol.2017.03.001
16. White LV, and Sintov ND. You Are what You Drive: Environmentalist and Social Innovator Symbolism Drives Electric Vehicle Adoption Intentions. *Transportation Res Part A: Pol Pract* (2017) 99:94–113. doi:10.1016/j.tra.2017.03.008
17. Hu J, Xia C, Li H, Zhu P, and Xiong W. (2020). Properties and structural analyses of USA's regional electricity market: A visibility graph network approach. *Applied Mathematics and Computation* 385, 125434. doi:10.1016/j.amc.2020.125434
18. Cui X, Hu J, Ma Y, Wu P, Zhu P, and Li H-J. (2021). Investigation of stock price network based on time series analysis and complex network. *International Journal of Modern Physics B* 35(13):2150171. doi:10.1142/S021797922150171X
19. Qiao H.-H, Deng Z.-H, Li H.-J, Jin N-D, Hu J, Song Q, and Gao L. (2021). Research on historical phase division of terrorism: An analysis method by time series complex network. *Neurocomputing* 420, 246–265. doi:10.1016/j.neucom.2020.07.125
20. Lacasa L, Luque B, Ballesteros F, Luque J, and Nuño JC. From Time Series to Complex Networks: The Visibility Graph. *Proc Natl Acad Sci* (2008) 105(13): 4972–5. doi:10.1073/pnas.0709247105
21. Gao Z-K, Fang P-C, Ding M-S, and Jin N-D. Multivariate Weighted Complex Network Analysis for Characterizing Nonlinear Dynamic Behavior in Two-Phase Flow. *Exp Therm Fluid Sci* (2015) 60:157–64. doi:10.1016/j.expthermflusci.2014.09.008
22. Sanjiv S, and Purohit GN. A New Centrality Measure for Tracking Online Community in Social Network. *Int J Inf Technol Comput Sci* (2012) 4:47–53. doi:10.5815/ijitcs.2012.04.07
23. Dangalchev C. Residual Closeness in Networks. *Physica A: Stat Mech its Appl* (2006) 365(2):556–64. doi:10.1016/j.physa.2005.12.020
24. Ruhnau B. Eigenvector-centrality - a Node-Centrality? *Social Networks* (2000) 22(4):357–65. doi:10.1016/S0378-8733(00)00031-9
25. Costa LDF, Rodrigues FA, Travieso G, and Villas Boas PR. Characterization of Complex Networks: A Survey of Measurements. *Adv Phys* (2007) 56(1): 167–242. doi:10.1080/00018730601170527
26. Newman MEJ. Equivalence between Modularity Optimization and Maximum Likelihood Methods for Community Detection. *Phys Rev E* (2016) 94(5): 052315. doi:10.1103/PhysRevE.94.052315

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

ORIGINAL RESEARCH
published: 24 November 2021
doi: 10.3389/fphy.2021.785885

Risk Analysis of the Transmission Route for the African Swine Fever Virus in Mainland China

Jiang-Hong Hu[1,2], *Xin Pei*[3], *Gui-Quan Sun*[4,1] *and Zhen Jin*[1,2]*

[1]Complex Systems Research Center, Shanxi University, Taiyuan, China, [2]Shanxi Key Laboratory of Mathematical Techniques and Big Data Analysis on Disease Control and Prevention, Shanxi University, Taiyuan, China, [3]School of Mathematics, Taiyuan University of Technology, Taiyuan, China, [4]Department of Mathematics, North University of China, Taiyuan, China

African swine fever first broke out in mainland China in August 2018 and has caused a substantial loss to China's pig industry. Numerous investigations have confirmed that trades and movements of infected pigs and pork products, feeding pigs with contaminative swills, employees, and vehicles carrying the virus are the main transmission routes of the African swine fever virus (ASFV) in mainland China. However, which transmission route is more risky and what is the specific transmission map are still not clear enough. In this study, we crawl the data related to pig farms and slaughterhouses from Baidu Map by writing the Python language and then construct the pig transport network. Following this, we establish an ASFV transmission model over the network based on probabilistic discrete-time Markov chains. Furthermore, we propose spatiotemporal backward detection and forward transmission algorithms in semi-directed weighted networks. Through the simulation and calculation, the risk of transmission routes is analyzed, and the results reveal that the infection risk for employees and vehicles with the virus is the highest, followed by contaminative swills, and the transportation of pigs and pork products is the lowest; the most likely transmission map is deduced, and it is found that ASFV spreads from northeast China to southwest China and then to west; in addition, the infection risk in each province at different times is assessed, which can provide effective suggestions for the prevention and control of ASFV.

Keywords: African swine fever virus, transmission route, pig transport network, dynamic model, assessing the infection risk

OPEN ACCESS

Edited by:
Peican Zhu,
Northwestern Polytechnical University, China

Reviewed by:
Sanling Yuan,
University of Shanghai for Science and Technology, China
Hui-Jia Li,
Beijing University of Posts and Telecommunications (BUPT), China

****Correspondence:***
Zhen Jin
jinzhn@263.net

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *29 September 2021*
Accepted: *18 October 2021*
Published: *24 November 2021*

Citation:
Hu J-H, Pei X, Sun G-Q and Jin Z (2021) Risk Analysis of the Transmission Route for the African Swine Fever Virus in Mainland China. Front. Phys. 9:785885. doi: 10.3389/fphy.2021.785885

1 INTRODUCTION

African swine fever (ASF) is a highly infectious and fatal disease of wild boars and domestic pigs caused by the African swine fever virus (ASFV) [1]. It is characterized by featuring a short course of onset, a mortality rate of up to 100% in the most acute and acute infections; clinical manifestations of fever; cyanosis of the skin; and obvious bleeding in lymph nodes, kidneys, and gastrointestinal mucosa [2]. The first African swine fever outbreak in mainland China was reported in a pig farm on August, 2018, in Shenbei District, Shenyang City of Liaoning Province [3]. According to the report of the Chinese government, within 1 month after the onset of clinical symptoms, all 400 pigs on the pig farm died [4]. Since then, the disease has spread rapidly throughout China and caused the deaths of more than one million pigs [5]. According to epidemiological investigation, ASFV was transmitted in mainland China mainly by three ways. First, trades and movements of pigs themselves as well as pork

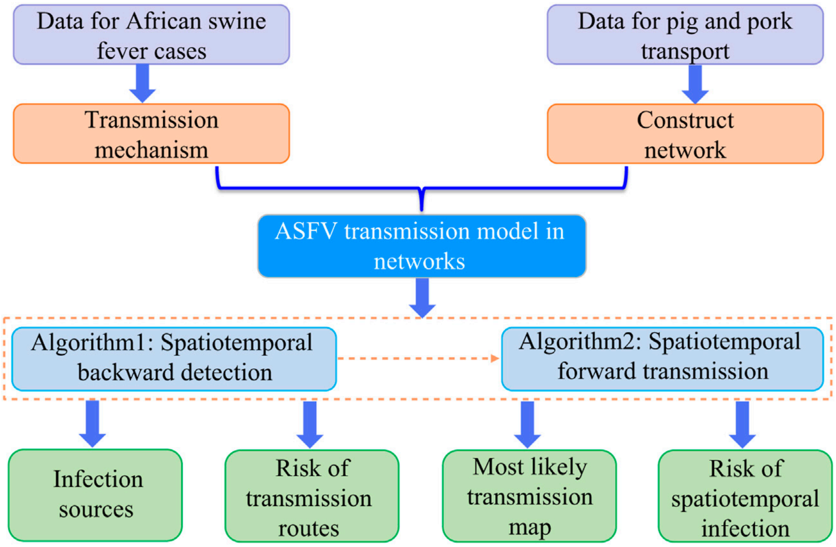

Data for African swine fever cases
Data for pig and pork transport
Transmission mechanism
Construct network
ASFV transmission model in networks
Algorithm1: Spatiotemporal backward detection
Algorithm2: Spatiotemporal forward transmission
Infection sources
Risk of transmission routes
Most likely transmission map
Risk of spatiotemporal infection

FIGURE 1 | Overview of the proposed framework in the study.

products; second, feeding pigs with contaminative swills (i.e., food residue from restaurants); and third, contaminated transport vehicles or employees without effective disinfection, that is, employees and vehicles with the virus which spreads over others [6, 7].

African swine fever broke out in China in August 2018 and has caused great economic losses for the market of pigs and pig products. Since the first spread of ASFV to China, the study about transmission regularity and control strategies has been in progress. Zhang et al. established a dynamic model to explore the impact of disinfection and fixation of employees on ASFV spread in the pig farms and presented some essential requirements for large-scale pig farms to decrease the transmission risk of ASFV [8]. Li et al. used the vulnerability index and data envelopment analysis (DEA) method to assess the regional vulnerability to ASF in mainland China from August 2018 to July 2019 and gave the severity level of African swine fever in 31 provinces in mainland China [9]. Vergne et al. evaluated the relative contribution of stable flies to the transmission of the African swine fever virus by establishing a model of the vector-borne transmission mechanism of ASFV in outdoor pig farms [10]. Ma et al. studied the distribution characteristics of African swine fever cases based on spatiotemporal clustering and the directional distribution analysis method and determined the high-risk areas of African swine fever outbreaks using the presence-only maximum entropy (MaxEnt) ecological niche model [11]. Akhmetzhanov et al. estimated the reproduction numbers, serial intervals, and transmission distances of ASF in China, according to the reconstructed ASF transmission network based on the nearest neighbor method, exponential function, equal probability, and spatiotemporal case distribution algorithms [12]. However, none of these efforts focused on the transmission process of ASFV in mainland China, and it is unclear how ASFV spreads after it has been introduced into China or how the transmission risks of the three transmission routes are. The network transmission model not only takes into account the transmission mechanism of infectious diseases but also can reflect the structure of the transmission network and has been widely used in the study of epidemic transmission [13, 14]. It will be a good choice to study the transmission process of African swine fever in mainland China.

In this study, we collect data related to pig farms and slaughterhouse locations from Baidu Map in web crawler. From this, we construct a pig transportation network and establish an ASFV transmission model on the basis of discrete-time Markov chains. Then, we propose spatiotemporal backward detection and forward transmission algorithms on the semi-directed weighted network from the constructed network and established model to analyze the risk of transmission routes, to infer the most likely transmission map of ASF in mainland China, and to assess the infection risk in different provinces at different times. **Figure 1** shows the overall framework of the study.

2 MATERIALS AND METHODS

According to the report on the official website of the Ministry of Agriculture and Rural Affairs, PRC (http://www.moa.gov.cn), ASF mainly breaks out in pig farms and some slaughterhouses. This study deals with the spread of ASFV in mainland China from August 1, 2018, to August 31, 2019, taking pig farms and slaughterhouses as the main research objects. According to the current pig breeding mode in China, the live pig industry chain from top to bottom includes pig forage, pig breeding, slaughtering, meat product processing and manufacturing, and the sale of pork and its products. We take the live pig feeding and slaughter processing industry chain which has a direct relationship with the spread of ASF as the mainline to describe the transmission of ASFV along the pig transport route

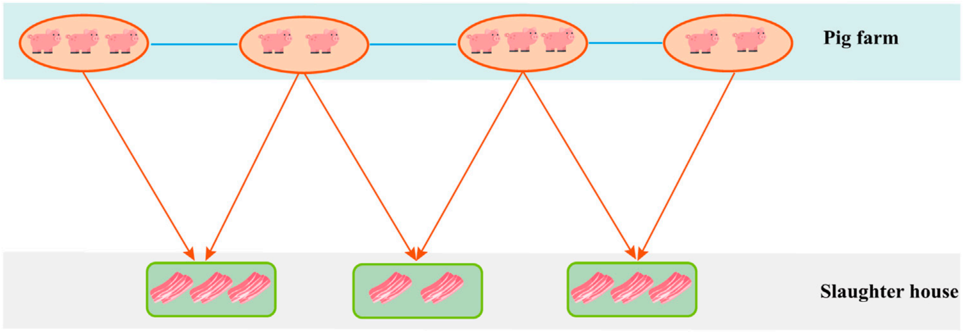


FIGURE 2 | Transmission diagram of ASFV along the live pig transport route.

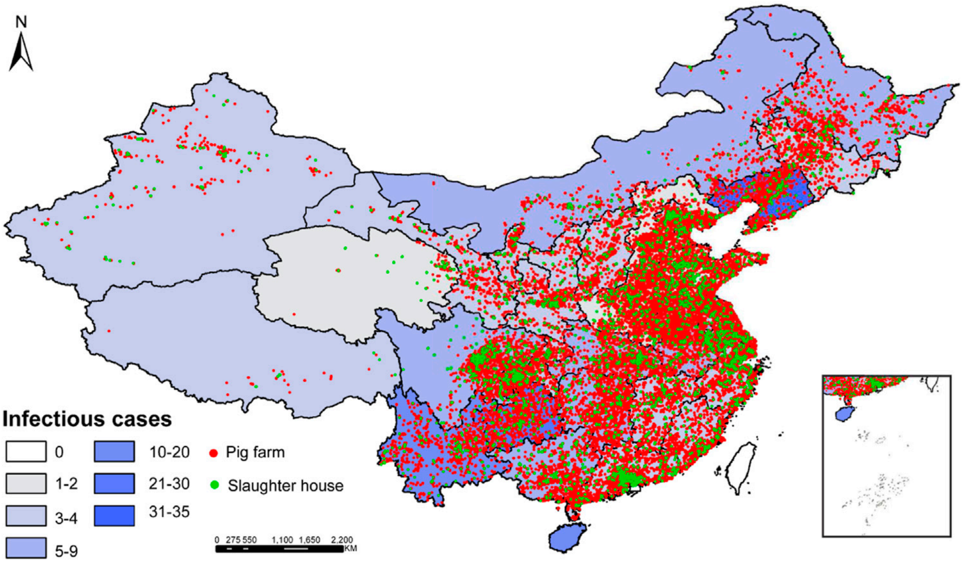


FIGURE 3 | GIS visualization map of the extracted pig farms and slaughterhouses based on the Python language.

as shown in **Figure 2**. Pig farms include small-, medium-, and large-scale farms, which are mainly engaged in the farrowing and breeding of pigs, and pig trade is carried out between these pig farms. When they are fattened to a sufficient weight, the pigs from pig farms are transported to the slaughterhouses for slaughter. The construction of the pig transportation network and the establishment of the transmission model followed are both based on this trading mode above.

2.1 Data

2.1.1 Data Collection

Data for African swine fever cases: We collected surveillance data of ASFV in mainland China from the official website of the Ministry of Agriculture and Rural Affairs of the People's Republic of China (http://www.moa.gov.cn) from August 1st, 2018, to August 31st, 2019. They include the geographical location, the type of site (pig farm or slaughterhouse), date of onset, date of report, the number of pig stocks, and whether to enable the emergency response mechanism, take blockade, and prohibit pigs or pork products to be transferred out or into the blockade area.

Data for the geographical location of the pig farm and slaughterhouse: We apply the Python language to crawl the information of pig farms and slaughterhouses in each city from Baidu Map. The search terms in the Python language include "geographic name," "pig farm, pig farmer, pig breeding cooperative," and "slaughterhouse, slaughter." The extracted results include data about the names and longitude and latitude of all sites. **Figure 3** shows the GIS visualization map of the extracted pig farms and slaughterhouses.

Data for the pig production and total population: The pig output in each province was obtained from the China Animal Husbandry and Veterinary Yearbook [15], and the total population of each province was obtained from the China Statistical Yearbook [16].

2.1.2 Data Cleaning and Preprocessing

Data cleaning: In order to make the data related to pigs and pork products accurate, we need to manually delete some sites not related to pig and pig slaughter from the above results.

Data preprocessing: We calculate the Euclidean distances between any two pig farms, each pig farm and slaughterhouse, based on their longitude and latitude using **Eq. 1**:

$$\Gamma = \sqrt{(Lat.P_1 - Lat.P_2)^2 + (Lon.P_1 - Lon.P_2)^2}. \tag{1}$$

In **Eq. 1**, *Lat.* P_1 ($Lat.P_2$) and *Lon.* P_1 ($Lon.P_2$) are the latitude and longitude of site P_1 (P_2), respectively, in Baidu Map; Γ is the Euclidean distance between site P_1 and P_2 [17].

2.2 Pig Transport Network Construction

After these above preparations, we start to construct the pig transport network. The network considered here is a semi-directed and weighted network $G = (V, E)$, where V represents the set of nodes and E denotes a set of edges [18]. As follows, we give the specific network construction project:

Nodes: The nodes can be divided into two types: pig farm node and slaughterhouse node.

Edges: We construct two types of edges according to the maximum Euclidean distance between different types of nodes:

1) Each pig farm is connected to the pig farms within 289 km [19] to establish an undirected edge.
2) Each pig farm is connected to the slaughterhouse within 285 km [19], and a directed edge from the pig farm to the slaughterhouse is established.

The maximum distance is determined according to article [19], which gives that the maximum Euclidean distance observed in the farm-to-farm movements was 289 km, while in the farm-to-abattoir movements, it was 285 km.

Time-varying: The network is temporally dynamic. Nodes and edges are deleted or added according to the information obtained from the official website of the Ministry of Agriculture and Rural Affairs, PRC (http://www.moa.gov.cn), on whether the areas with ASF outbreaks have activated the emergency response mechanism: the blockade shall be adopted to prohibit the transfer of live pigs or pork products out of or into the blocked areas.

2.3 Transmission Model

In this section, a network dynamic model based on the discrete Markov process is established in terms of the transmission process of ASFV along the pig transport route [20]. Nodes in the network comprise pig farm nodes and slaughterhouse nodes. According to the states of nodes, the pig farm nodes are divided into four states: susceptible (S_f), latent (E_f), infected (I_f), and dead (D_f), while the slaughterhouse nodes are divided into susceptible (S_h) and infected (I_h). V represents the virus carried by employees and vehicles, coming out from latent, infected, and dead (culled) pig farms, which then spreads the virus to other pig farms where they enter without thorough disinfection. W represents the contaminative swills (i.e., food residue from restaurants), which is released by pork from infected slaughterhouses and spreads the virus to others. In addition, susceptible pig farms and slaughterhouses can be infected through the trades and movements of pigs. Based on the above, the transmission process of ASF in pig farms and slaughterhouses is shown in **Figure 4**. The detailed state transitions of nodes in the network are as follows:

1) The susceptible pig farm (S_f) is infected by the ASF virus carried by employees and vehicles (V), contaminative swills (W), or latent pig farms neighbor (E_f) and infected pig farm neighbors (I_f) due to the trades and movements of pigs with probability $\lambda_{vf}(t) + \lambda_{wf}(t) + \lambda_{ff}(i, t)$ and then becomes (E_f);
2) The latent pig farm (E_f) becomes the infected farm (I_f) with probability σ;
3) The infected pig farm (I_f) is culled with probability d, which then becomes the dead pig farm (D_f);
4) The susceptible slaughterhouse (S_h) is infected by latent pig farm neighbor (E_f) and infected pig farm neighbor (I_f) due the trades of infected pigs with probability $\lambda_{E_f h}(j,t) + \lambda_{I_f h}(j,t)$, which then becomes ($I_h$).

In order to more accurately describe the status of different nodes at time t, two sub-states of the infected node (I) are introduced, which are "contagious"(C) and "maintained contagious" (M) in **Figure 4**. The contagious state (C) represents the node's newly infected state. The node in state C at the time t means that the node is susceptible at time $t - 1$, but it is infected at time t. An infected node first transits to contagious (C) at time t and then transits to being misled (M) at time $t + 1$. The node in the M state will remain infected until it dies and is removed from the network. By synthesizing the above description and flow chart in **Figure 4**, a network dynamic model of ASF based on the discrete Markov process is established. The meanings and values of variables and parameters involved in the model are shown in **Tables 1**, **2**.

Briefly, the modeling idea is illustrated by exampling the state transition of a pig farm node. At time t, a susceptible farm (S_f) node i can be infected through the following three ways and transformed into a latent pig farm node (E_f):

1) Virus carried by employees and vehicles (V): The virus from latent (E_f), infected (I_f), and dead (D_f) pig farms at time $t - 1$ attaches to the relevant transport vehicles or employees and then transmits to susceptible pig farms. The probability that a susceptible pig farm is not infected with the ASF virus carried by employees and vehicles means that none of its latent, infected, or dead pig farm neighbors spread the virus to it through vehicles or employees. Therefore, the probability that a susceptible pig farm i being infected with the ASF virus carried by employees and vehicles at time t is as follows:

$$\lambda_{vf}(t) = 1 - \prod_{i' \in \{1,2,\ldots,n\}, i' \neq i} [1 - \beta_{vf} P_E^f(i', t-1) + P_I^f(i', t-1) + P_D^f(i', t-1)]. \tag{2}$$

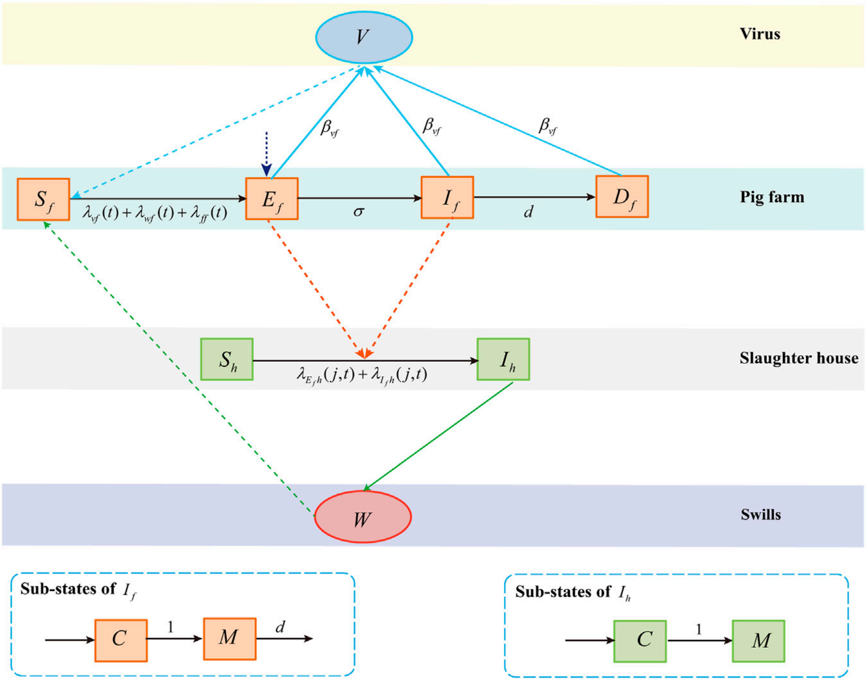


FIGURE 4 | State transition of nodes in the ASFV transmission model.

TABLE 1 | Meanings of the variables in the model.

Variable	Interpretation
∂j_{in}	The set of pig farm neighbors connected to slaughterhouse node j in the network
∂i	The set of pig farm neighbors of pig farm node j in the network
$d_{i'i}$	The distance between node i' and node i
$\eta_{i'i}$	The transmission probability from pig farm i' to pig farm i
η_{ij}	The transmission probability from pig farm node i to slaughterhouse node j
N_i^m	The number of pig outputs in the province where pig farm i is located
N_i^p	The total population of the province where pig farm i is located
$\lambda_{vf}(t)$	The probability that a susceptible pig farm being infected with the ASF virus carried by employees and vehicles V at time t
$\lambda_{wf}(t)$	The probability that a susceptible pig farm being infected by contaminative swills (W) at time t
$\lambda_{ff}\ (i,\ t)$	The probability of a susceptible pig farm i being infected by its latent or infected pig farm neighbor at time t
$P_Y^f\ (i, t)$	The probability that the arbitrary pig farm i is state Y at time t, $Y \in \{S, E, C, I, D\}$
$P_Y^h\ (j, t)$	The probability that the arbitrary slaughterhouse j is in state Y at time t, $Y \in \{S, C, I\}$

TABLE 2 | Descriptions and values of the parameters in the model.

Parameter	Interpretation	Value	Source
Σ	The transformation rate of pig farms from the latent state to the infected state	0.1	[8]
D	The culling rates of the infected pig farm	0.7	[8]
β_{ff}	The infection rate of the infected pig farm to the susceptible pig farm	0.3	Assumed
β_{fh}	The infection rate of the infected pig farm to the susceptible slaughterhouse	0.28	LHS
β_{vf}	The infection rate of latent, infected, or dead pig farms to susceptible employees and vehicles	0.49	LHS
β_{wf}	The infection rate of the infected slaughterhouse to contaminative swills	0.36	LHS

2) Contaminative swills (W): Infected slaughterhouses (I_f) at time $t-1$ sold pork products to restaurants, and then, susceptible pigs in pig farms can be infected by eating contaminated swills (i.e., food residue from restaurants) (W). The probability that a susceptible pig farm is not infected by contaminative swills means that none of its

infected slaughterhouse neighbors spread the virus to it through pork products. Therefore, the probability that a susceptible pig farm i being infected with the ASF virus carried by contaminative swills at time t is as follows:

$$\lambda_{wf}(t) = 1 - \prod_{j\in\{1,2,\ldots,n\}} [1 - \beta_{wf} P_I^h(j, t-1)]. \tag{3}$$

3) Trades of infected pigs and pork products: The susceptible pig farm at time $t - 1$ can be infected by the importation of infected pigs from latent (E_f) or infected (I_f) pig farm neighbors. The probability that a susceptible farm is not infected means that none of its latent or infected pig farm neighbors spread the virus to it through their links. Therefore, the probability that a susceptible farm node i being infected by its latent or infected pig farm neighbors at the time t is as follows:

$$\lambda_{ff}(i,t) = 1 - \prod_{i'\in\partial i} [1 - \eta_{i'i}(P_E^f(i', t-1) + P_I^f(i', t-1))], \tag{4}$$

where $\eta_{i'i}$ is the transmission probability of pig farm i' to pig farm i, and the formulation is as follows:

$$\eta_{i'i} = \beta_{ff} K_{i'i}. \tag{5}$$

In **Eq. 5**, β_{ff} is the maximum infection rate among pig farms, and $K_{i'i}$ means the transmission kernel function at pig farm node i' to i [21].

When node i' and i are the pig farms in the same province, the kernel function $K_{i'i}$ is calculated by the distance between the two farms $d_{i'i}$. The closer the distance is, the bigger the value is, and its expression is as follows:

$$K_{i'i} = \frac{k_0}{1 + \frac{d_{i'i}}{d_0}}, \tag{6}$$

where k_0 and d_0 determine the form of the kernel function, and the value refers to Ref. [21].

When nodes i' and i are pig farms from different provinces, the value $K_{i'i}$ of the kernel function not only is determined by the distance between the two farms $d_{i'i}$ but also depends on the pig output $N_{i'}^m$ and N_i^m and the total population $N_{i'}^p$ and N_i^p of the provinces. Its expression is as follows:

$$K_{i'i} = \frac{k_0}{1 + \frac{d_{i'i}}{d_0}} \cdot \frac{\frac{N_{i'}^m}{N_{i'}^p} - \frac{N_i^m}{N_i^p}}{\sum_{i'\in\partial i}\left(\frac{N_{i'}^m}{N_{i'}^p} - \frac{N_i^m}{N_i^p}\right)}. \tag{7}$$

In conclusion, the probability that an arbitrary susceptible pig farm node i at time $t - 1$ is still in the susceptible state at time t is as follows:

$$P_S^f(i,t) = [1 - \lambda_{vf}(t) - \lambda_{wf}(t) - \lambda_{ff}(t)] P_S^f(i, t-1). \tag{8}$$

That is, the pig farm is not infected by any of the above three ways.

The expression of the probability that pig farm node i is in the latent state at time t is as follows:

$$\begin{aligned} P_E^f(i,t) = {} & [\lambda_{vf}(t) + \lambda_{wf}(t) + \lambda_{ff}(i,t)] P_S^f(i, t-1) \\ & + (1-\sigma) P_E^f(i, t-1). \end{aligned} \tag{9}$$

The pig farm node i that is in the latent state at time $t - 1$ is first converted to the newly infected state C with transformation rate σ. Thus, the expression of the probability that an arbitrary pig farm node i is in the contagious state at time t is as follows:

$$P_C^f(i,t) = \sigma P_E^f(i, t-1). \tag{10}$$

Pig farm node i is in the infection state at time t, which means that it is newly infected at time t or it has been in the infection state and not culled before time t. The expression of the probability that pig farm node i is in the infection state at time t is as follows:

$$P_I^f(i,t) = P_C^f(i,t) + (1-d) P_I^f(i, t-1). \tag{11}$$

Pig farm node i is in the dead state at time t, which means that it was in the infection state at time $t - 1$ and culled at time t or it has been in the dead state before time t. The expression of the probability that pig farm node i is in the dead state at time t is as follows:

$$P_D^f(i,t) = d P_I^f(i, t-1) + P_D^f(i, t-1). \tag{12}$$

The probability that the slaughterhouse node is in each state at time t can be deduced in a similar manner. In conclusion, the network dynamics model of ASF based on the Markov process can be obtained as follows:

$$\begin{cases} P_S^f(i,t) &= [1 - \lambda_{vf}(t) - \lambda_{wf}(t) - \lambda_{ff}(i,t)] P_S^f(i, t-1), \\ P_E^f(i,t) &= [\lambda_{vf}(t) + \lambda_{wf}(t) + \lambda_{ff}(i,t)] P_S^f(i, t-1) + (1-\sigma) P_E^f(i, t-1), \\ P_C^f(i,t) &= \sigma P_E^f(i, t-1), \\ P_I^f(i,t) &= P_C^f(i,t) + (1-d) P_I^f(i, t-1), \\ P_D^f(i,t) &= d P_I^f(i, t-1) + P_D^f(i, t-1), \\ P_S^h(j,t) &= [1 - \lambda_{E_f h}(j,t) - \lambda_{I_f h}(j,t)] P_S^h(j, t-1), \\ P_C^h(j,t) &= [\lambda_{E_f h}(j,t) + \lambda_{I_f h}(j,t)] P_S^h(j, t-1), \\ P_I^h(j,t) &= P_C^h(i,t) + P_I^h(i, t-1). \end{cases} \tag{13}$$

The terms $\lambda_{vf}(t), \lambda_{wf}(t), \lambda_{ff}(i,t), \lambda_{E_f h}(j,t)$, and $\lambda_{I_f h}(j,t)$ in the model 13) can be deduced as follows:

$$\begin{aligned} \lambda_{vf}(t) = 1 - & \prod_{i'\in\{1,2,\ldots,n\}, i'\neq i} [1 - \beta_{vf}(P_E^f(i', t-1) + P_I^f(i', t-1) \\ & + P_D^f(i', t-1))], \\ \lambda_{wf}(t) = 1 - & \prod_{j\in\{1,2,\ldots,n\}} [1 - \beta_{wf} P_I^h(j, t-1)], \\ \eta_{i'i} = {} & \beta_{ff} K_{i'i}, \\ \lambda_{ff}(i,t) = {} & 1 - \prod_{i'\epsilon\partial i} [1 - \eta_{i'i} P_E^f(i', t-1) + P_E^f(i', t-1)], \\ \eta_{ij} = {} & \beta_{fh} K_{ij}, \lambda_{E_f h}(j,t) = 1 - \prod_{i\epsilon\partial j_{in}} [1 - \eta_{ij} P_E^f(i, t-1)], \\ \lambda_{I_f h}(j,t) = {} & 1 - \prod_{i\epsilon\partial j_{in}} [1 - \eta_{ij} P_I^f(i, t-1)]. \end{aligned} \tag{14}$$

Table 3 | Spatiotemporal backward detection algorithm in a semi-directed weighted network.

```
Input: Adjacency matrix of the pig transport network, a set of infected pig farm nodes
  $F = \{f_1, f_2, ..., f_n\}$ and the set of their infection times $T^f = \{t_1^f, t_2^f, ..., t_n^f\}$, a set of
  infected slaughter house nodes $H = \{h_1, h_2, ..., h_n\}$ and the set of their infection times
  $T^h = \{t_1^h, t_2^h, ..., t_n^h\}$, and a threshold $t_{max} = \max T^f \cup T^h, t_{min} = \min T^f \cup T^h$.
Initialize: $n + m$ sets of possible infection sources $U_i = \varnothing$, a set of most likely infection
  sources $U = \varnothing$.
for: ($t$ starts from $t_{max}$ to $t_{min}$) do
    for: ($o_i : o_i \in F \cup H$, infected pig farm and slaughter house nodes at time $t$) do
        if: ($o_i \in F$ that is, the node is pig farm and has not been detected) then
            Start to detect the virus from infected node $o_i$ along the undirected edge separately
            and independently at time $t$;
            Add the detected pig farm nodes into the set $U_i$.
            if: ($o_i \in H$ that is, the node is slaughter house and has not been detected) then
            Start to detect the virus from infected node $o_i$ along the directed edge separately
            and independently at time $t$, to find all pig farm nodes directly connected to it;
            Then start from the detected pig farm node to backward detect;
            Add the detected pig farm nodes into the set $U_i$.
        end
        for ($u$: any node in the set $U_i$) do
            Compute the maximum likelihood $L(u, t)$ for infected node $o_i$;
            Add the node with large maximum likelihoods in $U$.
        end
    end
end
Output: $n + m$ sets of infection sources $U_i$, and the maximum likelihood $L(u, t)$ that
  every detected node is infection source of $o_i$, a set of the most likely infection sources $U$.
```

Here, expressions **6** and **7** should be applied for kernel function $K_{i'i}$ and K_{ij} depending on whether the adjacent pig farms or slaughterhouses belong to the same province.

2.4 Algorithm

Based on the pig transport network and ASFV network dynamics model, the spatiotemporal backward detection and forward transmission algorithms on the semi-directed network are proposed in this section to detect the most probable infection source for each infected pig farm and slaughterhouse and to infer the most likely transmission route and the infection risk of nodes in the network at different times.

2.4.1 Spatiotemporal Backward Detection

In this section, the spatiotemporal backward detection algorithm on a semi-directed weighted network is proposed to detect the most likely infection source of each infected pig farm or slaughterhouse. The spatiotemporal backward detection algorithm proposed in this section is inspired by the algorithm proposed in Ref. [13]. The similarities are as follows: 1) they are both dynamic networks from the perspective of whether the network is changeable with time; 2) The detection mechanism is the same in time, and both are inverse detection. The differences are as follows: 1) The network structure is different, as the network is a directed weighted network in [13], while the algorithm proposed in this section is a semi-directed weighted network; 2) the spatial detection mechanism is different. The former is reverse detection, while in this section, apart from reverse detection, two-way detection is also applied in the algorithm; 3) the connection weights of edges are different. The connection probability of the two nodes in the former is related to the spatial distance, while in this section, in addition to the spatial distance, the number of pig productions and the total population are relative to the probability of the two nodes' connection. This algorithm is detailed as shown in **Table 3**.

The pig transport network constructed here records the spatial location and infection time of each infected pig farm or slaughterhouse. In infection source set U, the real infection source of the infected pig farm node or slaughterhouse node is more consistent with the real situation in space and time than other nodes. The key role of the algorithm proposed in the section is the disease transmission model (13), which shows the probability of any nodes which are in newly infected state C at time t. $P_C(u, t_s|o_i)$ means the probability that node u is in the newly infected state C, and the reverse detection begins from the objected node o_i in the span of t_s. According to Bayes' rule, the probability that node u transmits the virus to node o_i is in proportion to the probability that node o_i transmits the virus back to the infected node u [22]. That is, $P(o_i|u) \sim P(u|o_i)$.

$$P(u|o_i) = \prod_{o_i \in O} P_C(u, t_s|o_i). \tag{15}$$

To brief the calculation available, the maximum likelihood is derived by ln (·), which is induced by the logarithmic function, and the expression is as follows:

$$L(u,t) = \ln \prod_{o_i \in O} P_C(u, t_s|o_i). \tag{16}$$

Mathematically, among all nodes in possible infection source set U_i detected in the algorithm, the node with the largest maximum likelihood is defined as the most likely infection source of objected nodes o_i, namely,

$$(u^*, t^*) = \arg\max_{u \in U_i} L(u,t). \tag{17}$$

2.4.2 Spatiotemporal Forward Transmission

As a node may infect multiple nodes, the spatiotemporal forward propagation algorithm on a semi-directed weighted network is proposed in this section to determine the transmission influence of the infection source node in the whole network and to calculate the infection risk of nodes in the network at different times. The details of this algorithm are shown in **Table 4**.

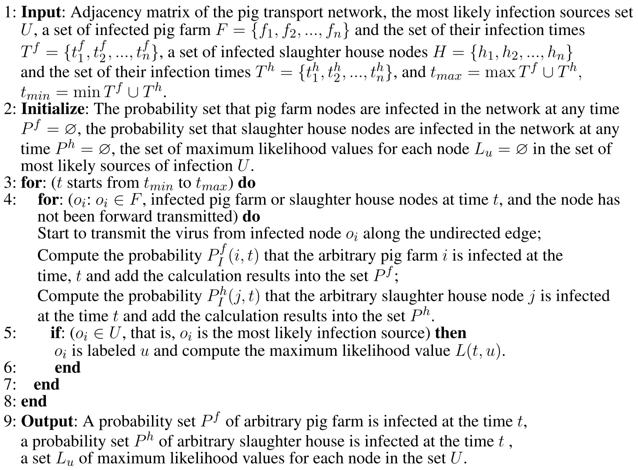
$P_C(o_i, t_s|u)$ represents the probability that node o_i is in the newly infected state C after the time span t_s, starting forward transmission from the infection source node u. $L(t, u)$ represents the maximum likelihood when the infection source node u transmits the virus to infected object nodes at time t. To brief the calculation available, the maximum likelihood is derived by logarithmic function ln (·), and the expression is as follows:

$$L(t,u) = \ln \prod_{o_i \in F} P_C(o_i, t_s|u). \tag{18}$$

In addition, **Eq. 19** can be used to estimate the infection size $I(t, u)$, which is infected by the spread of the infection source detected at time t. The validity of the algorithm proposed in this section can be verified through reviewing the accuracy of $I(t, u)$. Its expression is as follows:

$$I(t,u) = \sum_{u \in U} \sum_{o_i \in F} P_I(o_i, t|u). \tag{19}$$

TABLE 4 | Spatiotemporal forward transmission algorithm in the semi-directed weighted network.



```
Input: Adjacency matrix of the pig transport network, the most likely infection sources set
  $U$, a set of infected pig farm $F = \{f_1, f_2, ..., f_n\}$ and the set of their infection times
  $T^f = \{t_1^f, t_2^f, ..., t_n^f\}$, a set of infected slaughter house nodes $H = \{h_1, h_2, ..., h_n\}$
  and the set of their infection times $T^h = \{t_1^h, t_2^h, ..., t_n^h\}$, and $t_{max} = \max T^f \cup T^h$,
  $t_{min} = \min T^f \cup T^h$.
Initialize: The probability set that pig farm nodes are infected in the network at any time
  $P^f = \varnothing$, the probability set that slaughter house nodes are infected in the network at any
  time $P^h = \varnothing$, the set of maximum likelihood values for each node $L_u = \varnothing$ in the set of
  most likely sources of infection $U$.
for: ($t$ starts from $t_{min}$ to $t_{max}$) do
    for: ($o_i$: $o_i \in F$, infected pig farm or slaughter house nodes at time $t$, and the node has
    not been forward transmitted) do
    Start to transmit the virus from infected node $o_i$ along the undirected edge;
    Compute the probability $P_I^f(i, t)$ that the arbitrary pig farm $i$ is infected at the
    time, $t$ and add the calculation results into the set $P^f$;
    Compute the probability $P_I^h(j, t)$ that the arbitrary slaughter house node $j$ is infected
    at the time $t$ and add the calculation results into the set $P^h$.
        if: ($o_i \in U$, that is, $o_i$ is the most likely infection source) then
         $o_i$ is labeled $u$ and compute the maximum likelihood value $L(t, u)$.
         end
    end
end
Output: A probability set $P^f$ of arbitrary pig farm is infected at the time $t$,
  a probability set $P^h$ of arbitrary slaughter house is infected at the time $t$ ,
  a set $L_u$ of maximum likelihood values for each node in the set $U$.
```

3 RESULTS

In this section, based on the pig transport network and a network dynamic model of ASFV, the simulation of the spatiotemporal backward detection and forward transmission algorithm is proposed on the semi-directed weighted network, the infection source nodes are detected, the risk of three main transmission routes of ASFV is analyzed, the most likely transmission map is inferred, and the infection risk in provinces at different times is shown.

3.1 Detected Infection Sources

The most likely infection source for every infection node can be detected by the spatiotemporal backward detection algorithm. In this section, we take an infected pig farm node f_{20} in Liaoning province and an infected slaughterhouse node h_6 in Guizhou province as examples since the ASF virus first reported, and the largest number of ASFV nodes occurred in the two provinces. **Figure 5** shows all the probable infection sources of f_{20} as well as h_6 detected by the algorithm and the sequence of the maximum likelihood L (u, t). As the infection rates β_{ff}, β_{fh}, β_{vf}, β_{wf} in model 13) and the formula of maximum likelihood L $(u.t)$ are uncertain, in this section, the four parameters are sampled of a 1,000 times of random sampling with the Latin hypercube sampling method (LHS) on the assumption that the parameters are in correspondence with the normal distribution to calculate the maximum likelihood L $(u.t)$, and the simulation results are shown in the form of the violin plots, and the median, quartile range, and 95% confidence interval are shown. As shown in **Figure 5A**, for infected pig farm nodes f_{20}, six possible pig farms are detected as infection sources, among which the maximum likelihood value L (u, t) of the pig farm f_5 is greater than the other five possible infection sources, so pig farm f_5 is the most likely infection source for infected pig farm f_{20}. Similarly, as shown in **Figure 5B**, node f_{126} is the most likely infection source for infected slaughterhouse h_6.

Then, based on the most likely infection source detected, spatiotemporal forward transmission is carried out on the constructed pig transport network. **Eq. 19** is used to estimate the size I (u, t) of infected objects that were infected by the infection source at time t. The effectiveness of the algorithm is verified. **Figure 6** shows the reported accumulated infected cases as well as estimated accumulated infected cases from infection sources by forward transmission from August 2018 to August 2019. The results show that there is a small difference between the estimated accumulative cases and reported accumulative cases, so the model and algorithm proposed in this study are verified rationally.

3.2 Risk Analysis of Transmission Routes

There are three main transmission ways of ASFV in mainland China: first, trades and movements of pigs and pork products; second, feeding pigs with contaminative swills (i.e., food residue from restaurants); and third, employees and vehicles with the ASF virus which spreads over others. In this section, the risk of three transmission routes is analyzed.

First, the uncertainty and sensitivity of the parameters regarding the maximum likelihood L (u, t) of the infection source are analyzed. Uncertainty analysis and sensitivity analysis of parameters based on *LHS* and partial rank correlation coefficients (*PRCCs*) have previously been used in many infectious disease models [23, 24]. Taking infected nodes f_{20} and h_6 as examples, **Figure 7** shows the PRCCs of these four parameters to L (u, t) of all detected possible infection sources. The PRCC value of the parameter to L (u, t) is proportional to the correlation of this parameter to L (u, t). That is, the larger the PRCC value of the parameter is, the greater the influence of the parameter with regard to L (u, t) is. **Figure 7** shows that the PRCC value ($|PRCC| > 0.8$, $p < 0.05$) related to the maximum likelihood L (u, t) regarding the infection rate β_{vf} of employees and vehicles with the virus is the highest, that is, the virus transmission carried by employees and vehicles has the greatest influence on the maximum likelihood L (u, t), followed by the infection rate β_{wf} ($|PRCC| > 0.7$, $p < 0.05$) of contaminative swills; for the infection rate β_{ff}, β_{fh}, the PRCC value is lower, that is, it has little influence on the maximum likelihood L (u, t).

In addition, based on the parameters obtained in Section 4.1, it is assumed that only one of the three transmission routes, which forward-transmits the ASF virus on the constructed pig transport network, plays a role. The number of infected nodes is compared under the three transmission routes. **Figure 8** shows the newly infected nodes per month which are simulated under three assumptions. By comparison, it is found that the number of newly infected sites per month is the highest when only the virus carried by employees and vehicles is taken into account. When only contaminative swills (i.e., food residue from restaurants) are considered, the number of infected nodes is moderate; when only trade of infected pigs is considered, the number of infected nodes is the lowest. In summary, among the three main transmission routes of ASFV in mainland China, the infection risk for employees and vehicles with the virus is the highest, followed by contaminative swills, and the trade of infected pigs is the lowest.

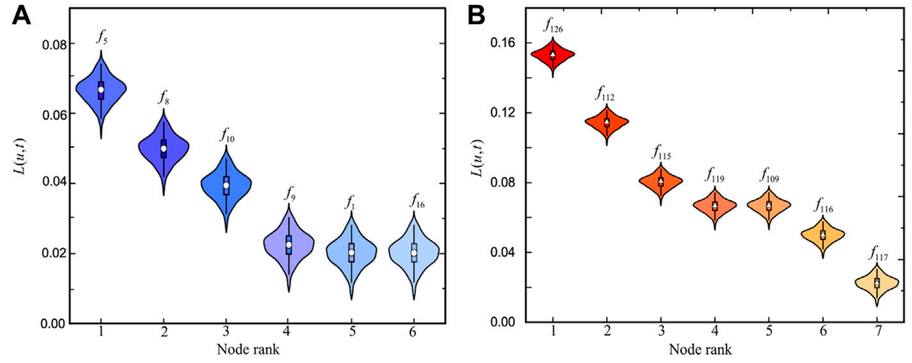


FIGURE 5 | Violin plots presenting the maximum likelihood *L* (*u*, *t*) for all possible infection sources with **(A)** infected pig farm node f_{20} and **(B)** infected slaughterhouse node h_6.

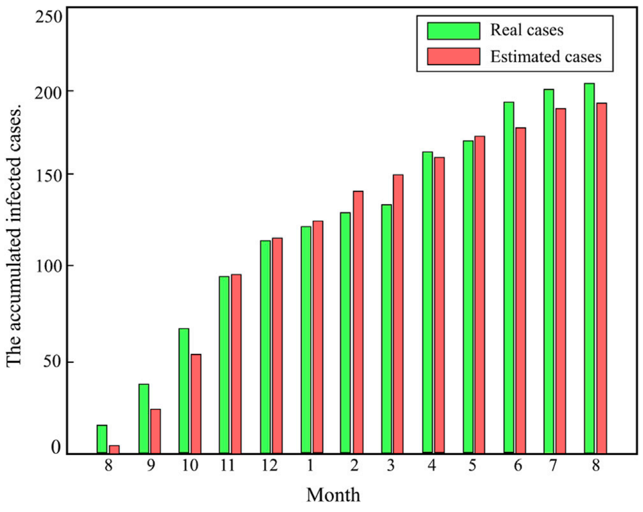


FIGURE 6 | Real accumulated infected cases and estimated accumulated infected cases through the spatiotemporal forward transmission algorithm in mainland China.

3.3 Most Likely Transmission Map

The most likely infection source for each infected node in the network can be detected by the spatiotemporal backward detection algorithm on the semi-directed network, and the transmission path can be known. Based on the transmission path of each infected node, the most likely transmission path map in provinces in mainland China from August 1, 2018, to August 31, 2019, can be deduced. **Figure 9A** shows the distribution of infected pig farms and slaughterhouses by ASFV in mainland China from August 1, 2018, to August 31, 2018, with 10 cases in Liaoning province, 3 cases in Zhejiang province, and 1 case each in Henan, Jiangsu, and Anhui provinces. **Figure 9B** presents the most likely ASFV transmission map by the spatiotemporal backward detection algorithm. The results generally show that ASFV spreads from northeast China to southwest China and then to west.

In particularly, according to the notice of "The General Office of the Ministry of Agriculture and Rural Affairs of the People's Republic of China on Typical Cases of Violation of Laws and Disciplines in the Prevention and Control of ASF" issued by the Ministry of Agriculture and Rural Affairs of the People's Republic of China (http://www.moa.gov.cn), on September 29, 2018, after the test by China Animal Health and Epidemiology Center, the source of the pigs from the slaughterhouse of Shuanghui Food Company in Zhengzhou city, Henan province, where ASF broke out on August 14, 2018, is Jiamusi, Heilongjiang province (marked with a solid blue line); on July 30, 2018, in the course of transferring pigs from Heilongjiang province, which was directed by a company in Siping city, Jilin province, ASF broke out (marked with a yellow solid line); on June 2018, piglets bought by a farmer of Shenyang, Liaoning province, from Jilin province died out of control, and the pigs sold to a farmer were confirmed to be infected with ASF on August 2. In summary, before August 1, 2018, there had been ASFV cases in Heilongjiang and Jilin provinces (marked by the red five-pointed star), and the source of pandemic occurred in Henan and Jilin provinces was most likely to be Heilongjiang. The epidemic in Liaoning province was probably spread from Jilin province.

3.4 Risk Assessment of Spatiotemporal Infection

The infection probability of each pig farm and slaughterhouse in the network at time *t* can be calculated by the spatiotemporal forward detection algorithm on the semi-directed weighted network. In reference to the whole pig transportation network in mainland China, the infection risk of each province at different times can be shown, as shown in **Figure 10**, which shows the infection risk of ASFV in provinces at four times as examples. **Figure 10A** shows the infection risk map of provinces in mainland China in October 2018. As a result, the infection risk in Liaoning province is the highest, followed by adjacent areas of the Inner Mongolia autonomous region and Jilin province. In addition, several provinces in central China and east China have a higher infection risk, while southwest China and northwest China have the lowest. **Figure 10B** shows the infection risk map in various provinces of mainland China in December 2018. The result depicts that at this time, the distribution

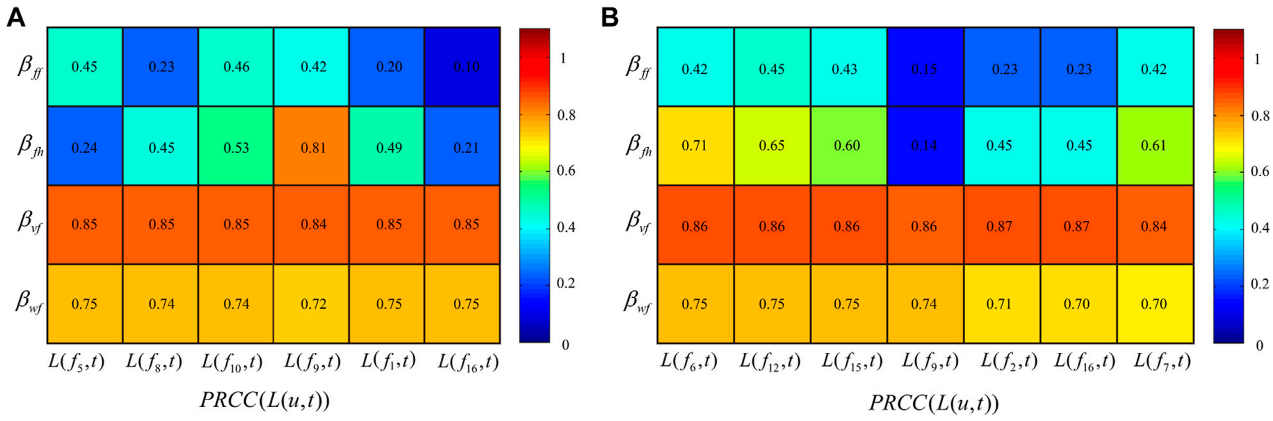


FIGURE 7 | Partial rank correlation coefficients (PRCCs) for parameters β_{ff}, β_{fh}, β_{vf}, β_{wf} with respect to the maximum likelihood $L(u, t)$ associated with all the detected possible infection sources of **(A)** infected pig farm node f_{20} and **(B)** infected slaughterhouse node h_6.

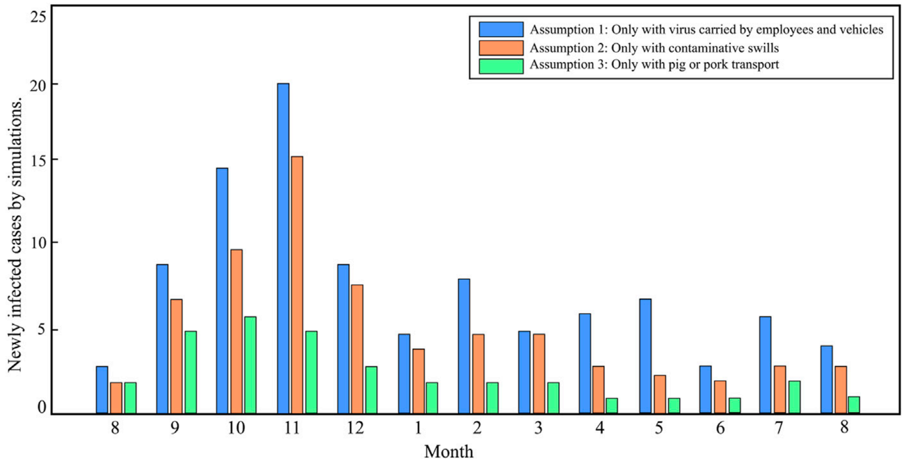


FIGURE 8 | Newly infected cases with time by simulations under the assumption that only one transmission route works.

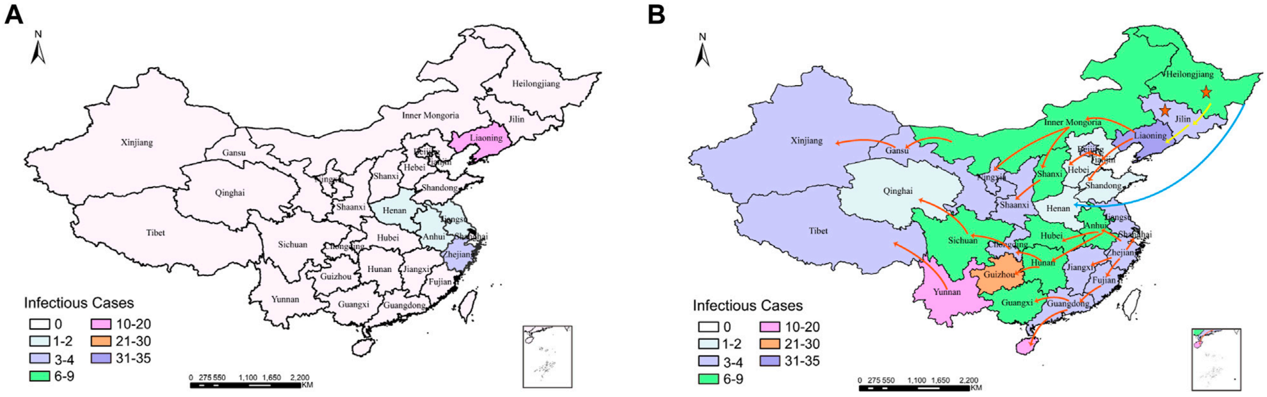


FIGURE 9 | (A) Geographic distribution of ASFV cases in mainland China from August 1st to 31st, 2018. **(B)** Geographic distribution of ASFV cases by August 31st, 2019, and inference of the most likely transmission map of ASFV based on the spatiotemporal backward detection algorithm.

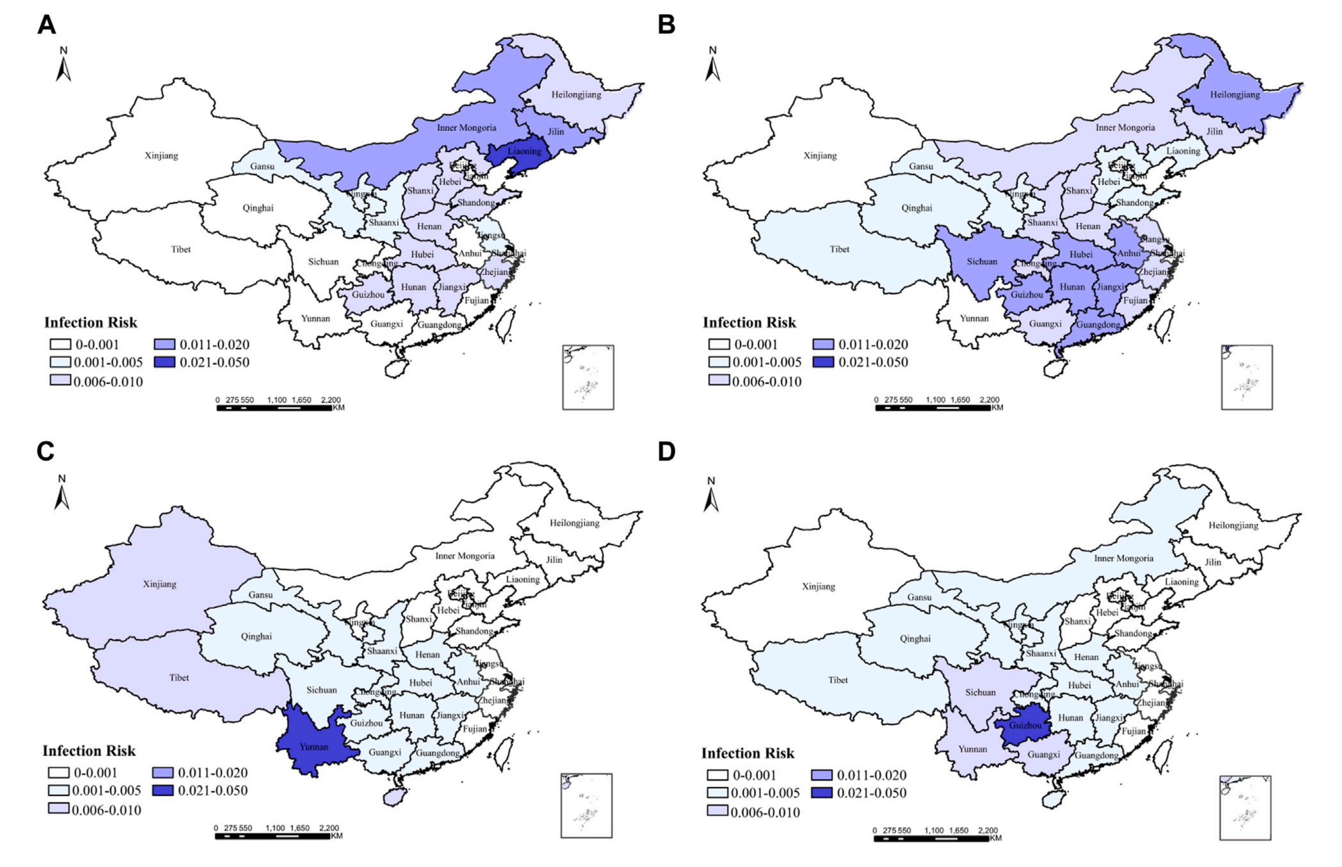


FIGURE 10 | Infection risk maps of ASFV in mainland China in **(A)** October 2018, **(B)** December 2018, **(C)** April 2019, and **(D)** June 2019.

with a high infection risk has shifted to some provinces in central China, south China, and southwest China, and some provinces in northwest China have a higher infection risk than before. **Figure 10C** shows the distribution of infection risk in various provinces of mainland China in April 2019. The result shows that the infection risk is generally low in northeast China, north China, and east China, while it is high in south, southwest, and northwest China. **Figure 10D** presents a distribution of infection risk in the provinces of mainland China in June 2019, showing that the overall infection risk has decreased at this time, with the highest risk of infection in southwest China. Summarily, with regard to the infection risk of ASFV in mainland China, the overall infection risk has been high by January 2019. As the regulatory measures such as the ban on pig transfer have been enhanced, the infection risk gradually decreases, and the high-risk areas gradually shift from northeast to southwest and northwest China.

4 CONCLUSION

In this study, the pig transmission network and the network dynamics model of ASFV based on the discrete Markov process are built based on the site's data of pig farms and slaughterhouses which are extracted from Baidu Map in the Python language. In addition, the spatiotemporal backward detection and forward algorithms on the semi-directed weighted network are proposed to detect the source of infection of pig farms and slaughterhouses infected with ASFV. Through the analysis on the transmission route risk, it is concluded that the spread risk of employees and vehicles with the virus is the highest, followed by the contaminated swill (i.e., food residue from restaurants), while the risk of pig and pork product trades is relatively lower. By tracing the source of the infected sites, we give the possible transmission path map of ASFV in mainland China. The map shows that ASFV is spread from northeast China to southwest China and then westward. By calculating the probability of each node in the network being infected at different times, we give the infection risk of each province at different times on a large spatial scale.

The innovation of this study is the establishment of a data-driven network transmission dynamics model of African swine fever, which is used to assess the transmission risk of ASFV in three transmission ways in mainland China. In practice, it is found that the transmission risk of employees and vehicles with the virus is the highest, and the infection risk in different regions at different times is shown, which can provide effective suggestions for the prevention and control of ASFV.

There are also some defects in the study. For example, we only crawled the location information of pig farms and slaughterhouses from Baidu Map and then constructed the pig transport network based on the distance between them, which may be deviated from the actual pig trade network. In the future study, we will integrate the actual traffic data between cities into the spread of infectious diseases on the network. Besides, we ignored the community structure of the real pig trade network when building the transmission model [25, 26], which is worth to be paid attention and further studied. In addition, the network transmission model we established in the study lacks theoretical analysis due to its high dimension. Therefore, it is necessary to further study the high-dimensional system dimension reduction method of the network propagation dynamics model and to analyze dynamic behavior. In general, this study depicts the risk of different transmission routes of ASFV in mainland China, which can provide effective suggestions for the prevention and control of the pandemic and is practical.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

All authors have made great contributions to the writing of the study and approved the submitted version. J-HH, G-QS, and ZJ conceived and designed the study and established the dynamical model. XP collected the data and constructed the network. J-HH and XP wrote the algorithms and simulated the results. J-HH wrote the manuscript. G-QS and ZJ provided valuable comments on the manuscript writing.

FUNDING

This work was supported by the National Natural Science Foundation of China (61873154, 11971278, 12022113, 11671241, and 12101443), the Henry Fok foundation for young teachers (171002), the Shanxi Key Laboratory (201705D111006), the Shanxi Scientific and Technology Innovation Team (201805D131012-1), the Shanxi Scholarship Council of China (HGKY2019004), and the Scientific and Technological Innovation Programs (STIP) of Higher Education Institutions in Shanxi (2019L0082).

REFERENCES

1. Galindo I, Alonso C. African Swine Fever Virus: a Review. *Viruses* (2017) 9: 103. doi:10.3390/v9050103
2. Quembo CJ, Jori F, Vosloo W, Heath L. Genetic Characterization of African Swine Fever Virus Isolates from Soft Ticks at the Wildlife/domestic Interface in mozambique and Identification of a Novel Genotype. *Transbound Emerg Dis* (2018) 65:420–31. doi:10.1111/tbed.12700
3. Li X, Tian K. African Swine Fever in china. *Vet Rec* (2018) 183:300–1. doi:10.1136/vr.k3774
4. Zhou X, Li N, Luo Y, Liu Y, Miao F, Chen T, et al. Emergence of African Swine Fever in China, 2018. *Transbound Emerg Dis* (2018) 65:1482–4. doi:10.1111/tbed.12989
5. Shen X, Pu Z, Li Y, Yu S, Guo F, Luo T, et al. Phylogeographic Patterns of the African Swine Fever Virus. *J Infect* (2019) 79:174–87. doi:10.1016/j.jinf.2019.05.004
6. Zhai S-L, Wei W-K, Sun M-F, Lv D-H, Xu Z-H. African Swine Fever Spread in china. *Vet Rec* (2019) 184:559. doi:10.1136/vr.l1954
7. Wang Y, Gao L, Li Y, Xu Q, Yang H, Shen C, et al. African Swine Fever in china: Emergence and Control. *J Biosafety Biosecur* (2019) 1:7–8. doi:10.1016/j.jobb.2019.01.006
8. Zhang X, Rong X, Li J, Fan M, Wang Y, Sun X, et al. Modeling the Outbreak and Control of African Swine Fever Virus in Large-Scale Pig Farms. *J Theor Biol* (2021) 526:110798. doi:10.1016/j.jtbi.2021.110798
9. Li J, Gao L, Huang B, Wang Y, Jin Z, Sun X, et al. Assessment of Regional Vulnerability to Africa Swine Fever in China during 2018/8-2019/7 Based on Data Envelopment Analysis Method. *Transbound Emerg Dis* (2021) 68: 2455–64. doi:10.1111/tbed.13913
10. Vergne T, Andraud M, Bonnet S, De Regge N, Desquesnes M, Fite J, et al. Mechanical Transmission of African Swine Fever Virus by Stomoxys Calcitrans : Insights from a Mechanistic Model. *Transbound Emerg Dis* (2021) 68:1541–9. doi:10.1111/tbed.13824
11. Ma J, Chen H, Gao X, Xiao J, Wang H. African Swine Fever Emerging in china: Distribution Characteristics and High-Risk Areas. *Prev Vet Med* (2020) 175: 104861. doi:10.1016/j.prevetmed.2019.104861
12. Akhmetzhanov AR, Jung S-m., Lee H, Linton N, Yang Y, Yuan B, et al. Reconstruction and Analysis of the Transmission Network of African Swine Fever in People's Republic of China, August 2018-September 2019. *bioRxiv* (2020). doi:10.1101/2020.07.12.199760
13. Pei X, Jin Z, Zhang W, Wang Y. Detection of Infection Sources for Avian Influenza A(H7N9) in Live Poultry Transport Network during the Fifth Wave in China. *IEEE Access* (2019) 7:155759–78. doi:10.1109/ACCESS.2019.2949606
14. Li H-J, Xu W, Song S, Wang W-X, Perc M. The Dynamics of Epidemic Spreading on Signed Networks. *Chaos, Solitons & Fractals* (2021) 151:111294. doi:10.1016/j.chaos.2021.111294
15. Institute of Animal Science B Chinese Academy of Agricultural Sciences, Ministry of Agriculture P. *China Animal Husbandry & Veterinary Medicine*. Beijing: China animal husbandry & veterinary medicine (2018).
16. National Bureau of Statistics P. *China Statistical Yearbook*. Beijing: China Statistics Press (2018).
17. Pamungkas CA. Aplikasi Penghitung Jarak Koordinat Berdasarkan Latitude Dan Longitude Dengan Metode Euclidean Distance Dan Metode Haversine. *Jurnal Informa: Jurnal Penelitian dan Pengabdian Masyarakat* (2019) 5:8–13. doi:10.46808/informa.v5i2.74
18. Zhu P, Zhi Q, Guo Y, Wang Z. Analysis of Epidemic Spreading Process in Adaptive Networks. *IEEE Trans Circuits Syst* (2019) 66:1252–6. doi:10.1109/TCSII.2018.2877406
19. Bigras-Poulin M, Barfod K, Mortensen S, Greiner M. Relationship of Trade Patterns of the Danish Swine Industry Animal Movements Network to Potential Disease Spread. *Prev Vet Med* (2007) 80:143–65. doi:10.1016/j.prevetmed.2007.02.004
20. Li H-J, Wang L, Bu Z, Cao J, Shi Y. Measuring the Network Vulnerability Based on Markov Criticality. *ACM Trans Knowl Discov Data* (2021) 16:1–24. doi:10.1145/3464390
21. Backer JA, Hagenaars TJ, van Roermund HJW, de Jong MCM. Modelling the Effectiveness and Risks of Vaccination Strategies to Control Classical Swine Fever Epidemics. *J R Soc Interf* (2009) 6:849–61. doi:10.1098/rsif.2008.0408
22. Lokhov AY, Mézard M, Ohta H, Zdeborová L. Inferring the Origin of an Epidemic with a Dynamic Message-Passing Algorithm. *Phys Rev E* (2014) 90: 012801. doi:10.1103/PhysRevE.90.012801
23. Blower SM, Dowlatabadi H. Sensitivity and Uncertainty Analysis of Complex Models of Disease Transmission: an Hiv Model, as an Example. *Int Stat Rev/Revue Internationale de Statistique* (1994) 62:229–43. doi:10.2307/1403510

24. Sanchez MA, Blower SM. Uncertainty and Sensitivity Analysis of the Basic Reproductive Rate: Tuberculosis as an Example. *Am J Epidemiol* (1997) 145:1127–37. doi:10.1093/oxfordjournals.aje.a009076
25. Li H-J, Bu Z, Wang Z, Cao J. Dynamical Clustering in Electronic Commerce Systems via Optimization and Leadership Expansion. *IEEE Trans Ind Inf* (2020) 16:5327–34. doi:10.1109/TII.2019.2960835
26. Li H-J, Wang L, Zhang Y, Perc M. Optimization of Identifiability for Efficient Community Detection. *New J Phys* (2020) 22:063035. doi:10.1088/1367-2630/ab8e5e

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article or claim that may be made by its manufacturer is not guaranteed or endorsed by the publisher.

PERSPECTIVE
published: 02 December 2021
doi: 10.3389/fphy.2021.789938

A Data Analytics Approach for Revealing Influencing Factors of HPV-Related Cancers From Population-Level Statistics Data

Xiaoqin Du[1] *and Qi Tan*[2]*

[1]Tianjin Central Hospital of Gynecology Obstetrics, Tianjin, China, [2]Division of Epidemiology and Biostatistics, School of Public Health, The University of Hong Kong, Hong Kong, China

Human papillomavirus (HPV) is considered as one of the major causes of multiple cancers, including cervical, anal, and vaginal cancers. Some studies analyzed the infection patterns of cancers caused by HPV using individual clinical test data, which is resource and time expensive. In order to facilitate the understanding of cancers caused by HPV, we propose to use data analytics methods to reveal the influencing factors from the population-level statistics data, which is available more easily. Particularly, we demonstrate the effectiveness of data analytics approach by introducing a predictive analytics method in studying the risk factors of cervix cancer in the United States. Besides accurate prediction of the number of infections, the predictive analytics method discovers the population statistic factors that most affect the cervical cancer infection pattern. Furthermore, we discuss the potential directions in developing more advanced data analytics approaches in studying cancers caused by HPV.

Keywords: data analytics approach, HPV-related cancer, population-level statistics data, regression model, HPV—human papillomavirus

OPEN ACCESS

Edited by:
Chao Gao,
Southwest University, China

Reviewed by:
Chijun Zhang,
Jilin University of Finance and Economics, China
Dong Li,
Shandong University, China

****Correspondence:***
Qi Tan
tanqi.g@gmail.com

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *05 October 2021*
Accepted: *01 November 2021*
Published: *02 December 2021*

Citation:
Du X and Tan Q (2021) A Data Analytics Approach for Revealing Influencing Factors of HPV-Related Cancers From Population-Level Statistics Data.
Front. Phys. 9:789938.
doi: 10.3389/fphy.2021.789938

INTRODUCTION

Human papillomavirus (HPV) is believed to cause more than 90% of anal and cervical cancers, about 70% of vaginal and vulvar cancers, and 60% of penile cancers [1, 12]. Recent studies show that HPV should be responsible for about 60–70% of cancers of the oropharynx, which traditionally have been caused by tobacco and alcohol [2]. Sexual behavior is considered as a major risk of HPV infection [13]. However, the relation between the prevalence of HPV-related cancer and the population-level demographical and economic factors remains unclear. Some studies have revealed that the rate of people getting HPV-associated cancers varies by race and ethnicity [3]. They showed that black and Hispanic women had higher rates of HPV-associated cervical cancer than women of other races and non-Hispanic women, which is of great value for further investigation into the causing mechanism of HPV-related cancers.

The previous studies rely on clinical test and evaluation, which is resource and time expensive. Though the predictive models have been used in the clinical HPV status prediction using biomarkers [14, 15], there are few studies on predicting population-level HPV-related cancer incidence. In order to facilitate the understanding of cancers caused by HPV, we propose a data analytics approach to discover influencing factors efficiently from heterogeneous data resources, such as demographical and social-economic statistic data. Since over 90% of the cervical cancers are caused by the HPV, we study the case of discovering the influencing factors of cervical cancers by analyzing the infection

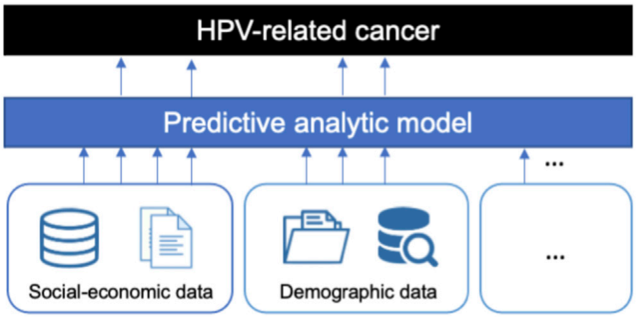


FIGURE 1 | The proposed framework. Predictive analytics model incorporates the indicators from heterogeneous data sources as features and then filters the most important indicators to explain the pattern of HPV-related cancer infection.

pattern in different states in the US. We demonstrate our proposed approach in **Figure 1**. With the predictive model, we can further predict the number of underlying HPV-related cancers, which facilitates HPV screening and vaccination by proactively deploying resources [17, 18].

MATERIALS AND METHODS

We use cervix cancer in 2018 (https://gis.cdc.gov/Cancer/USCS/#/AtAGlance/) as the target variable to analyze. We consider two types of factors: age and economic status. Specifically, we collect the population size of six age groups (children 0–18 years, adults 19–25 years, adults 26–34 years, adults 35–54 years, adults 55–64 years, adults over 65 years) and the gross domestic product (GDP) per capita income of the previous 8 years (from year 2011 to year 2018). We use these data at different states of the US as the features input into the analytics model.

We first assess the correlation between influencing factors and the target variable *via* a linear analytics model. We first normalize the features into [0, 1] for better analyzing the influences of these data. The formulation of the linear analytics model is:

$$\hat{y} = \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_D x_D,$$

where $x_d, d \in [1 \ldots D]$ is the normalized features and β_d is the corresponding coefficient. The results show that these factors account for about 43% variance of the state-wise infection pattern ($R^2 = 0.43488$).

Then to determine the most influencing factors, we learn a sparse linear model *via* Lasso method [4]. The objective of the model learning can be written as:

$$\min\|y - \hat{y}\| + \lambda|\boldsymbol{\beta}|,$$

where y is the ground-truth value of target variable, $\boldsymbol{\beta} = [\beta_1, \beta_2, \ldots, \beta_D]$ and λ is a hyperparameter. The first term is the l2 norm of the estimation error, which aims to make the analytics model better approximating the target variable, and the second term is the l1 norm the coefficient vectors, where λ controls how many influencing factors are selected in the analytics model.

RESULTS

The top five important influencing factors identified by the model are GDP per capita year 2018, GDP per capita at year 2011, age adults 26–34 years, age adults 55–64 years, and age over 65 years ($R^2 = 0.17354$). The weightings of different indicators with increasing sparsity penalty λ can be seen in **Figure 2**.

Finally, we examine the nonlinear correlation between the factors and the target variable *via* the predictivity, as discussed in [16]. We compare two models with the same input features: one linear model and a nonlinear neural network [5] with one hidden

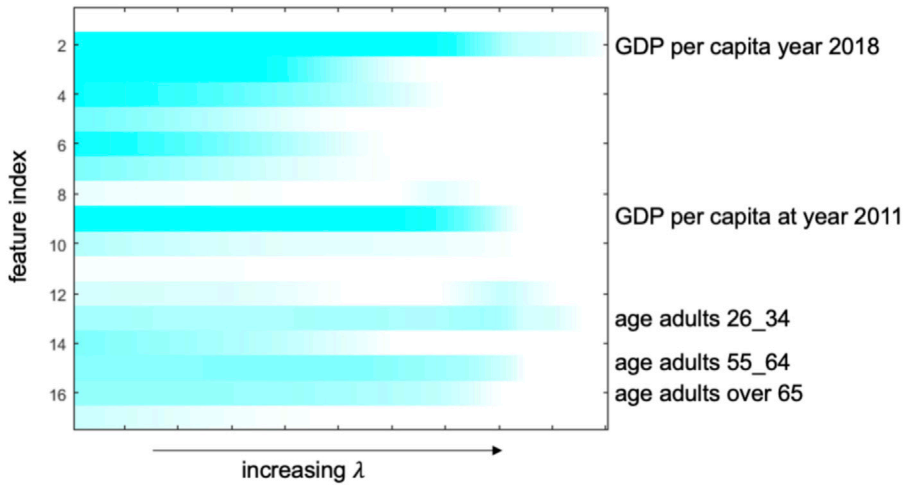


FIGURE 2 | The weightings of different indicators with increasing sparsity penalty λ.

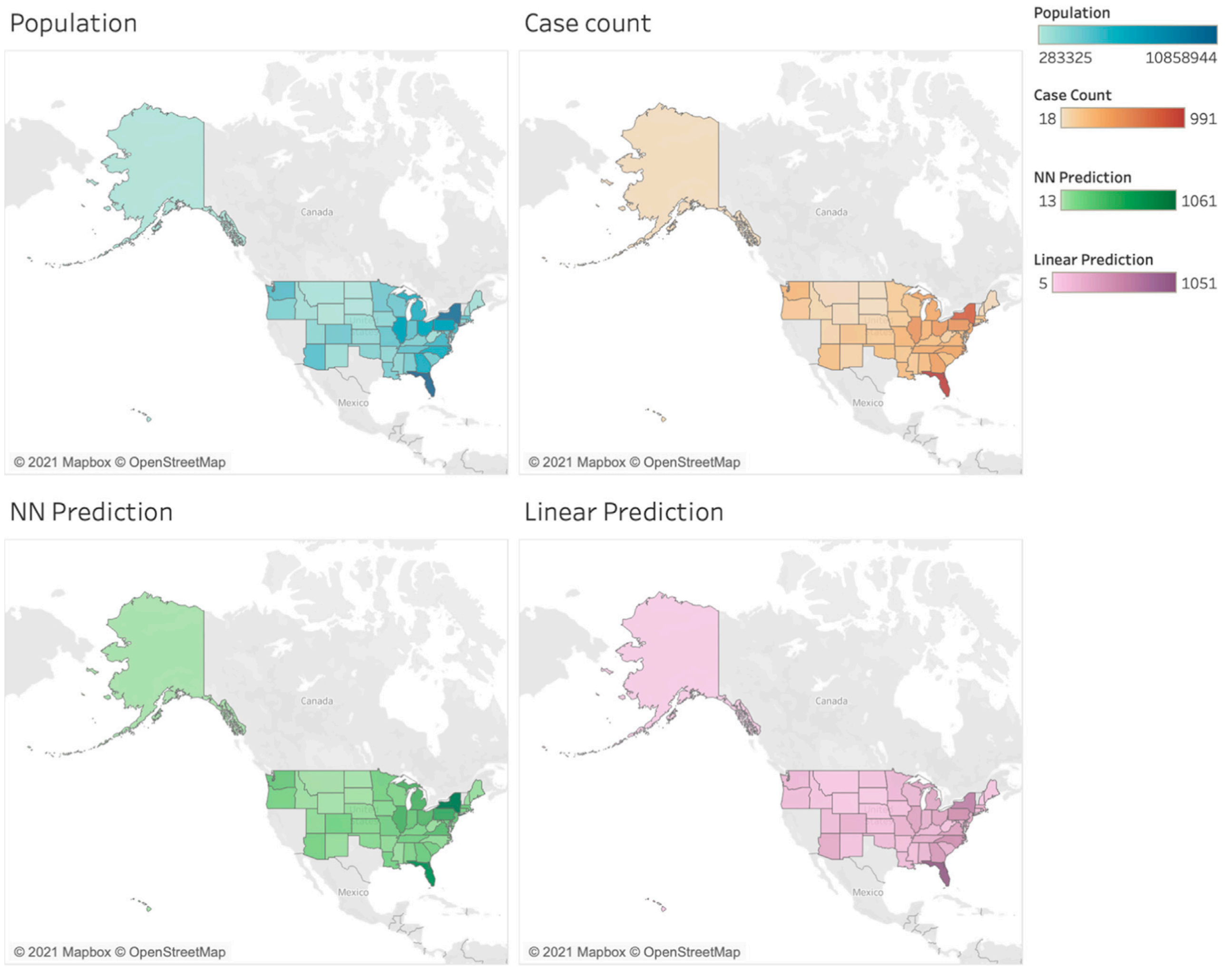


FIGURE 3 | Visualization of the predictive results. The first row shows the population and case count distribution in the US, and the second row shows the predictions of neural network (NN) model and linear model.

layer of size 16. We evaluate the predictive performance with leave-one-out strategy, i.e., train the model with all samples except one and then test the predictive performance on the one left. We use the mean absolute percentage error (MAPE) as the metrics of performance evaluation considering the variance of different target:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|\hat{y}_i - y_i|}{|y_i|}.$$

The MAPE of linear model is 0.3504, and the MAPE of neural network model is 0.3087. As the lower MAPE the better the performance, the predictive performance of neural network is much better than that of the linear model. The results show that there is nonlinear correlation among the risk factors and the incidence.

Figure 3 displays the predictive results of both models. We can see that the predictive models are able to capture the infection pattern of cervix cancers, and comparatively the neural network model produces more accurate predictions, e.g., that in New York state.

CONCLUSION

In this perspective, we proposed a data analytics approach to mining the influencing factors of HPV-related cancer from population-level statistics data. We also demonstrated the effectiveness of this approach in the case of analyzing the cervical cancers in the United States. We further examined the existence of nonlinear correlation *via* showing the superior predictivity of nonlinear model compared with the linear model. Further studies can incorporate more risk factors, such as low socioeconomic status and smoking habit [13].

Based on the current studies, further effort should be paid to analyze the complex nonlinear correlation between the influencing factors and the HPV-related incidence. For example, the advanced nonlinear models [7] and feature selection methods can be applied in the risk factor analysis. The recurrent neural network can combine the time-aggregated effect from time series data, and the attention-based model is able to directly extract the important features based on the current context. These methods can model the nonlinear correlations between the risk factors and the disease outcome. The advances in the study of model interpretability allow us to extract the key factors from the learned nonlinear models.

Moreover, causal inference methods can be incorporated to identify the causing factors reliably. There are many complex cofounding associations under the disease progression which hinder the key causing risk factors. To overcome these challenges, causal inference methods, e.g., marginal structural networks [6], can be applied to adjust the bias from the cofounding factors. A practical way to distinguish the proper factors with the nonlinear models is to identify the important features in terms of predictivity [8]. For nonlinear models, such as neural network methods, Shapley value [11] and gradient-based methods [9, 10] are commonly used for identifying the feature importance.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. These data can be found here: https://gis.cdc.gov/Cancer/USCS/#/AtAGlance/

AUTHOR CONTRIBUTIONS

XD and QT designed the method and experiments. XD collected the dataset and conducted the experiment. XD and QT analyzed the results and wrote the paper.

REFERENCES

1. Saraiya M, Unger ER, Thompson TD, Lynch CF, Hernandez BY, Lyu CW, et al.HPV Typing of Cancers Workgroup. US Assessment of HPV Types in Cancers: Implications for Current and 9-valent HPV Vaccines. *J Natl Cancer Inst* (2015) 107(6):djv086. doi:10.1093/jnci/djv086
2. Chaturvedi AK, Engels EA, Pfeiffer RM, Hernandez BY, Xiao W, Kim E, et al. Human Papillomavirus and Rising Oropharyngeal Cancer Incidence in the United States. *Jco* (2011) 29(32):4294–301. doi:10.1200/jco.2011.36.4596
3. Viens LJ, Henley SJ, Watson M, Markowitz LE, Thomas CC, Thompson TD, et al. Human Papillomavirus-Associated Cancers - United States, 2008-2012. *MMWR Morb Mortal Wkly Rep* (2016) 65(26):661–6. doi:10.15585/mmwr.mm6526a1
4. Tibshirani R. Regression Shrinkage and Selection via the Lasso. *J R Stat Soc Ser B* (1996) 58(No. 1):267–88. doi:10.1111/j.2517-6161.1996.tb02080.x
5. LeCun Y, Bengio Y, Hinton G. Deep Learning. *Nature* (2015) 521:436–44. doi:10.1038/nature14539
6. Lim B, Alaa A. Forecasting Treatment Responses over Time Using Recurrent Marginal Structural Networks. In: Proceedings of the 32nd Conference on Neural Information Processing Systems; December 2018; Montreal, Canada.
7. Gao C, Liu J. Network-based Modeling for Characterizing Human Collective Behaviors during Extreme Events. *IEEE Trans Syst Man, Cybernetics: Syst* (2017) 47(1):171–83. doi:10.1109/TSMC.2016.2608658
8. Ding M, Chen Y, Bressler SL. Granger Causality: Basic Theory and Application to Neuroscience. In: S Schelter, N Winterhalder, J Timmer, editors. *Handbook of Time Series Analysis*. Wienheim: Wiley (2006).
9. Shrikumar A, Greenside P, Shcherbina A, Kundaje A. *Not Just a Black Box: Learning Important Features through Propagating Activation differences[J]*. arXiv preprint arXiv:1605.01713 (2016).
10. Sundararajan M, Taly A, Yan Q. Axiomatic Attribution for Deep Networks. In: Proceedings of the International Conference on Machine Learning. PMLR; August 2017; Sydney, Australia.
11. Janzing D, Minorics L, Blöbaum P. Feature Relevance Quantification in Explainable AI: A Causal Problem. In: Proceedings of the International Conference on Artificial Intelligence and Statistics. PMLR; August 2020; Palermo, Sicily, Italy.
12. Lowy DR, Schiller JT. Reducing HPV-Associated Cancer Globally. *Cancer Prev Res* (2012) 5(1):18–23. doi:10.1158/1940-6207.capr-11-0542
13. Rai B, Bansal A, Singh M. Human Papillomavirus-Associated Cancers: A Growing Global Problem. *Int J App Basic Med Res* (2016) 6(2):84–9. doi:10.4103/2229-516X.179027
14. Qian G, Hu Z, Xu H, Müller S, Wang D, Zhang H, et al. A Novel Prediction Model for Human Papillomavirus-Associated Oropharyngeal Squamous Cell Carcinoma Using P16 and Subcellular β-catenin Expression. *J Oral Pathol Med* (2016) 45(6):399–408. doi:10.1111/jop.12378
15. Lang DM, Peeken JC, Combs SE, Wilkens JJ, Bartzsch S. Deep Learning Based HPV Status Prediction for Oropharyngeal Cancer Patients. *Cancers* (2021) 13:786. doi:10.3390/cancers13040786
16. Gao C, Liu J. Uncovering Spatiotemporal Characteristics of Human Online Behaviors during Extreme Events. *PLOS ONE* (2015) 10(10):e0138673. doi:10.1371/journal.pone.0138673
17. Du Z, Nugent C, Galvani AP, Krug RM, Meyers LA. Modeling Mitigation of Influenza Epidemics by Baloxavir. *Nat Commun* (2020) 11:2750. doi:10.1038/s41467-020-16585-y
18. Bai Y, Yang B, Lin L, Herrera JL, Du Z, Holme P. Optimizing sentinel Surveillance in Temporal Network Epidemiology. *Sci Rep* (2017) 7:4804. doi:10.1038/s41598-017-03868-6

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

PERSPECTIVE
published: 17 December 2021
doi: 10.3389/fphy.2021.795641

Geographically Explicit Network Analysis of Urban Living and Working Interaction Pattern in Shenzhen City, South China

Zhilu Yuan[1], *Haojia Lin*[1,2]*, *Shengjun Tang*[1] *and Renzhong Guo*[1,2]

[1]*Research Institute for Smart Cities, School of Architecture and Urban Planning, Shenzhen University, Shenzhen, China,* [2]*School of Resource and Environmental Sciences, Wuhan University, Wuhan, China*

Human daily mobility plays an important role in urban research. Commuting of urban residents is an important part of urban daily mobility, especially in working days. However, the characteristic of the mobility network formed by the commuting of urban residents and its impact on the internal structure of the city are still an important work that needs to be explored further. Aiming to study the living–working interaction pattern of meta-populations over urban divisions within cities, a fine-grained dataset of living–working tracking of Shenzhen is curated and used to construct an urban living–working mobility network, and the living–working interaction pattern is analyzed through the community structures of the network. The results show that human daily mobility plays an important role in understanding the formation of urban structure, the administrative divisions of the city affect human daily mobility, and human daily mobility reacts on the formation of urban structure.

Keywords: human mobility, community structures, network analysis, Shenzhen, commuting pattern

OPEN ACCESS

Edited by:
Chao Gao,
Southwest University, China

Reviewed by:
Hongzhi Miao,
Dalian Maritime University, China
Hongfei Jia,
Jilin University, China
Yongxing Li,
Beijing University of Technology, China

****Correspondence:***
Haojia Lin
linhaojia@whu.edu.com

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *15 October 2021*
Accepted: *17 November 2021*
Published: *17 December 2021*

Citation:
Yuan Z, Lin H, Tang S and Guo R (2021) Geographically Explicit Network Analysis of Urban Living and Working Interaction Pattern in Shenzhen City, South China. Front. Phys. 9:795641. doi: 10.3389/fphy.2021.795641

INTRODUCTION

Large-scale demographic census enables measurements of human living–working traces, which have become popular and served as essential reasons of motivation for human mobility [1]. The living–working interactions of meta-populations over urban divisions within cities have been extensively studied in a recent work (e.g., urban activities [1–4] and urban balance [5–8]). To study human living–working movements, especially within cities, living–working networks provide a useful way to characterize living–working styles among people in different sites. Although urban transportation patterns between locations change over time, many studies of human mobility assume they are representative [9], neglecting the limitation of transportation. For example, in Shenzhen, on average, people travel by subway for distance longer than 27 km while by bus for 9 km [10]. This is, arguably, due to the lack of fine-grained public datasets that could describe the living–working dynamics within cities. There are some open-access datasets covering small geographical locations considering the time ordering of location tracking, such as networks of mobile-phone users within a city [11] and between cities [12], which can infer people's living and working locations potentially. However, fine-grained living–working datasets covering large geographical regions within a city with large populations are still missing from the open-access datasets.

In this paper, we curate and amass a fine-grained dataset of living–working mobility to study urban interactions. We capture living and working position tracking of millions of workers from an open-data program in Shenzhen. Each location in our dataset represents a group of workers in an official administrative division. Directed movement of each individual from a source living location to a destination working location denotes a change of location for the corresponding individual. The overall directed mobility network of locations is finally compiled by sequentially processing the directed movements for all individuals. In the network, a node represents a location. A weighted edge represents the total number of workers' movements from living to working locations.

The dataset contains movements of nearly 6 million anonymized cellular phone users among 71 subdivisions (henceforth locations), covering 10 geographically adjacent districts, investigated during the year 2015. This total geographic area, located in Shenzhen as a major city in the Guangdong Province of China, covers more than 20 square kilometers and, in 2017, had a population of nearly 12.9 million. This city has become one of the four largest and wealthiest cities in China [13]. Thus, it is a fine-grained living–working dataset covering large geographical regions within a city with large populations. However, the characteristic of the mobility network formed by the commuting of urban residents and its impact on the internal structure of the city are still an important work that needs to be explored further. To study the living–working interaction pattern of meta-populations over urban divisions within cities, we process the above raw dataset to extract a dynamic and directed mobility network of the city. Then, based on the constructed urban living–working mobility network, the human living–working mobility pattern and the community structure of the city are analyzed. We found that there is an interaction between the human daily mobility and the formation of urban structure, the administrative divisions of the city affect human daily mobility, and human daily mobility reacts on the urban structure.

MATERIALS AND METHODS

Data Extraction and Processing

A largest urban living–working tracking dataset using demographic census logs in the year 2015 in Shenzhen is used in this study. The dataset consists of living–working location records of 5.6 million anonymized workers, and these locations include 71 geographically neighboring subdistricts over 10 districts. An individual worker is investigated by the Shenzhen Municipal Human Resources and Social Security Bureau (http://hrss.sz.gov.cn/) from the social security systems in Shenzhen used by workers to buy the social security and health insurance for every working individual.

The reliability of location and time information of workers' living and working locations in the network data largely depends on the reliability of the underlying source data. We verify the consistency via the geographically explicit distribution of locations. Although this dataset is fine-grained, it has several limitations. First, even though the dataset covers a cohort of millions of workers, it is only for 2015. Second, the individual's working position is the last known recorded location of this individual in the year 2015. This recording might cause bias. The individual might leave Shenzhen after 2015.

Construction of Living–Working Network

To simulate human mobility within Shenzhen, we construct the living–working network (a directed mobility network) based on the dataset by taking each place (a city or a country) as multiple meta-populations in different locations. Each location is represented as a node in the network. Edges are directed, connecting nodes where users move from origin (living locations) to the destination (working locations) and weighted by the total number of workers in this scenario. An individual worker–directed movement from the living location i to the working location j denotes that in a user's living–working record. We reinterpret the flow matrices F as adjacency matrices that describe the daily living–working mobility network, the vertices are the 71 locations, and the edge between a given vertex pair, i and j, is weighted by the flow N_{ij} (the number of workers from location i to location j). As shown in **Table 1**, the dataset is processed into two data tables: Network table and Location table. In the Network table, each row represents the total number of daily movements by workers from locations i to j, and there are three columns ordered by the living location, the working location, and the corresponding directional weight. In the Location table, there are two columns ordered by the location identifier and the corresponding charactered name.

Mobility Analysis of Living–Working Network

In the urban living–working mobility network, nodes are defined as locations, and edges are weighted by the population flows from living locations to working locations. To analyze the intensity of human mobility, the network centrality of individual locations is calculated, and the consistency of the network with the degree distribution is verified. We consider the network defined by the daily living–working mobility matrix F, and the in-degree and out-degree of the location i are then given by $D_{in}(i) = \sum_j N_{ji}$ and $D_{out}(i) = \sum_j N_{ij}$, respectively. On the contrary, to analyze the geographical range of human mobility, the commuting distance of workers is calculated and the distance distribution is compared to that of a mobility network by taxi trips in Shenzhen. We defined the commuting distance as the length of the shortest path in the driving mode between the center of the living location and the working location.

Structure Analysis of Living–Working Network

Additionally, to explore the characteristic of the mobility network formed by the commuting of urban residents and what impact does it have on the internal structure of the city, we analyze the community structure of the urban living–working mobility

TABLE 1 | Data description of the living–working network.

Data table	Field	Description
Network	Living	Numerical administrative division code for each living location
	Working	Numerical identification for each working location
	Weight	Total number of movements from the living location to the working location
Location	Location	Numerical administrative division code for each location
	Name	Charactered name of the corresponding location

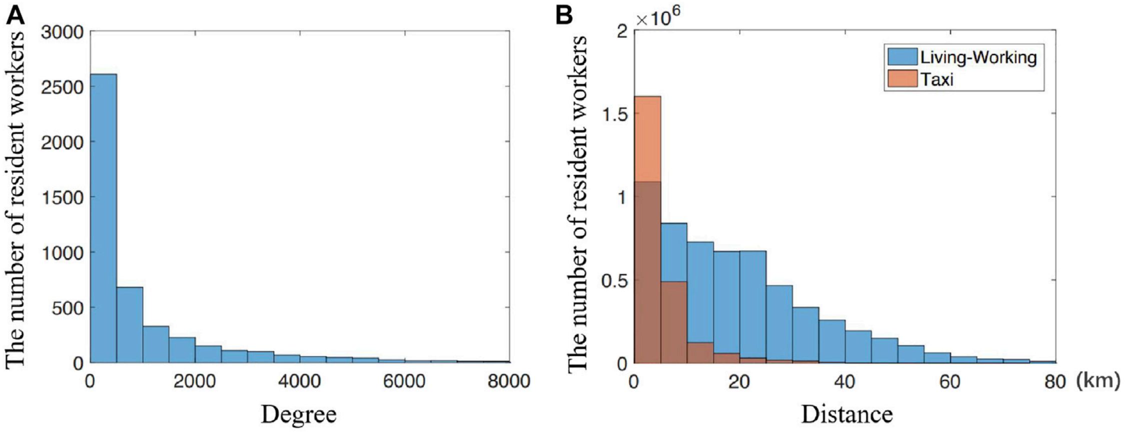


FIGURE 1 | Distribution plots. **(A)** Empirical degree distribution. The *x*-axis denotes the network degree of locations. The *y*-axis denotes the number of workers from the living to the working location. **(B)** Empirical distance distribution with this living–working network. The *x*-axis denotes the commuting distance of workers, and the unit is kilometer. The *y*-axis denotes the number of associated individuals. In contrast, we show the distance distribution of a static mobility network with zones as nodes and passenger flows as edges, aggregating 2 million taxi trips in Shenzhen from April 18, 2011, to April 26, 2011, over 1,634 zones [18]. The median distance is 16 km for our published dataset, in contrast with 3 km for those taxi trips.

network using the Louvain community detection algorithm [14–17]. It identifies disjoint subsets of locations such that their intra-connectivity far exceeds their inter-connectivity. All the locations in the network are divided into seven community-groups, in which the meta-populations in locations are highly intra-connected within the group but only loosely inter-connected across the group [18–20]. Then, the detected community structure is used to map the Shenzhen administrative divisions in 2017 and 2005 to analyze whether human daily mobility is infected by the administrative division, and whether there is interaction between the human daily mobility and the formation of urban structure.

RESULTS

Mobility Pattern of Living–Working Network

In the analysis of the intensity and range of human mobility, **Figure 1A** shows the distribution of the directed network's degrees and the number of resident workers in living and working locations. The *x*-axis denotes the network degree of locations, and the *y*-axis denotes the number of resident workers in those locations. The degree of a node denotes the total number of living–working movements passing through the corresponding node. We can see that, with the increase of network degree, the number of resident workers decreases rapidly, that is to say, locations with more active external relations have fewer resident populations. **Figure 1B** shows the distance distribution as compared to that of a mobility network by taxi trips in Shenzhen [18]. We can observe that living–working pairs with less than 80 km account for over 99.96% of the total, and the median living–working distance is 16 km, in contrast with a shorter median traveling distance of 3 km by taxi trips.

Community Structure of Living–Working Network

In the analysis of the relationship between the human daily mobility and the formation of urban structure, **Figure 2** shows the community structures for each day with colors denoting different detected communities. After mapping communities to the administrative divisions, we can observe locations within a division tend to be in the same community. Interestingly, the administrative divisions in 2005 are larger and can group locations within the same community better, especially in the Bao'an division.

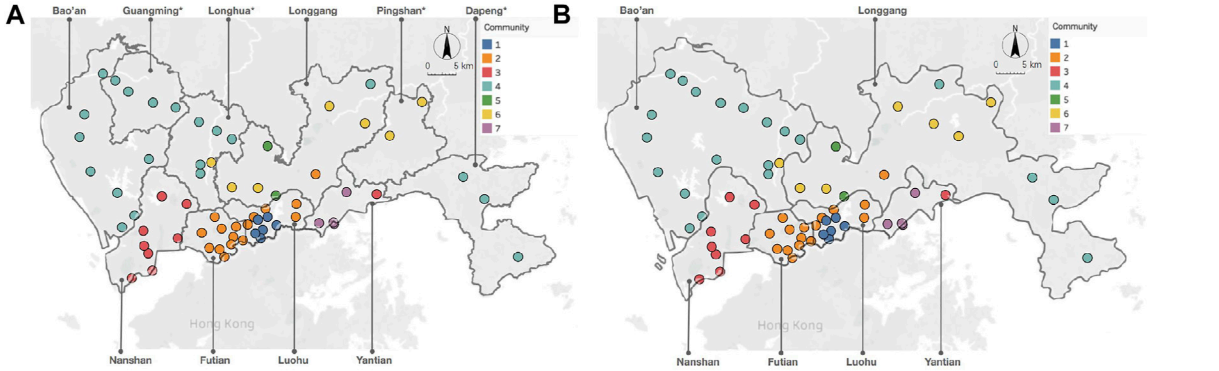


FIGURE 2 | Community structures of the Shenzhen living–working network. We construct the directed network via aggregating all workers' living and working records. The Louvain community detection algorithm serves to probe community structures based on this network. We map this community structure with colors denoting different communities to Shenzhen administrative divisions in 2017 **(A)** and 2005 **(B)**. The newly established divisions are marked by asterisks in **(A)**, but not in **(B)**. The spatial map was created using the OpenStreetMap online platform (http://www.openstreetmap.org/) under the license of CC BY-SA (http://www.openstreetmap.org/copyright). More details of the license can be found at http://creativecommons.org/licenses/by-sa/2.0/. Line graphs were drawn using Tableau Software for Desktop version 9.2.15 (https://www.tableau.com/zh-cn/support/releases/9.2.15). The layouts were modified with Keynote version 6.6.2 (http://www.apple.com/keynote/).

DISCUSSION AND CONCLUSION

Urban living and working tracking can provide fine-grained traveling data within cities on daily scales, giving us a feasible way to explore human daily mobility, especially in working days. Although there are different choices of transportation patterns between locations within the city, neglecting the limitation of transportation, origin–destination is representative in some study of human mobility. Many urban problems are related to the commuting of urban residents, such as traffic congestion in the morning and evening rush hours, jobs-housing balance in the urban structure, and the fairness of urban facilities. Moreover, human daily mobility plays an important role in understanding the formation of urban structure. On the one hand, there is an interaction between the human daily mobility and the formation of urban structure, and the daily life of urban residents is usually within a certain range, within which there are more internal connections and less connections with other external areas. In other words, human daily mobility shapes the urban structure, and in each structure, intra-connectivity far exceeds inter-connectivity. On the other hand, the administrative divisions of the city affect human daily mobility, and human daily mobility reacts on the urban structure. The administrative division will affect the scope of daily activities of urban residents because some things will be more convenient in the same administrative division. To some extent, most of residents' social relationships are within the region. This study reinforces the importance of the living–working interaction pattern of meta-populations over urban divisions within cities in urban management. Human mobility is an important research direction in urban research, and human living–working traces served as essential reasons of motivation for human daily mobility.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Material, and further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

ZY and HL conceived the study, designed statistical and modeling methods, conducted analyses, and wrote the article. ZY, HL, ST, and RG interpreted the results and revised the article.

FUNDING

We acknowledge the financial support by the National Natural Science Foundation of China (Project Nos. 71901147, 41801392, 41901329, 41971354, and 41971341), the Research Program of Shenzhen S&T Innovation Committee (Project Nos. JCYJ20210324093600002 and JCYJ20210324093012033), the Science Foundation of Guangdong Province (Project Nos. 2121A1515012574 and 2019A1515010748), the Open Fund of Key Laboratory of Urban Land Resources Monitoring and Simulation, MNR, China (No. KF-2019-04-014), Natural Science Foundation of Jilin Provincial Science and Technology Department (No. 20180101332JC), the Science and Technology Project of the Jilin Provincial Education Department (No. JJKH20210135KJ), and Jilin Science and technology development plan project (No.20190303124SF).

REFERENCES

1. Du Z, Yang B, Liu J. *Understanding the Spatial and Temporal Activity Patterns of Subway Mobility Flows* (2017).
2. Gao C, Fan Y, Jiang S, Deng Y, Liu J, Li XJITOITS. *Dynamic Robustness Analysis of a Two-Layer Rail Transit Network Model* (2021).
3. Long Y, Han H, Tu Y, Shu X. Evaluating the Effectiveness of Urban Growth Boundaries Using Human Mobility and Activity Records. *Cities* (2015) 46: 76–84. doi:10.1016/j.cities.2015.05.001
4. Zhang D, He T, Zhang F, Xu C. Urban-Scale Human Mobility Modeling with Multi-Source Urban Network Data. *Ieee/acm Trans Networking* (2018) 26: 671–84. doi:10.1109/tnet.2018.2801598
5. Bagrow JP, Lin Y-R. Mesoscopic Structure and Social Aspects of Human Mobility. *PLoS ONE* (2012) 7:e37676. doi:10.1371/journal.pone.0037676
6. Barbosa H, Barthelemy M, Ghoshal G, James CR, Lenormand M, Louail T, et al.Human Mobility: Models and Applications. *Phys Rep* (2018) 734:1–74. doi:10.1016/j.physrep.2018.01.001
7. Deng Y, Wang J, Gao C, Li X, Wang Z, Li X, Applications Assessing Temporal-Spatial Characteristics of Urban Travel Behaviors from Multiday Smart-Card Data. *Physica A: Stat Mech its Appl* (2021) 576:126058. doi:10.1016/j.physa.2021.126058
8. Leana CR, Meuris J. Living to Work and Working to Live: Income as a Driver of Organizational Behavior. *Annals* (2015) 9:55–95. doi:10.5465/19416520.2015.1007654
9. Yan XY, Wang WX, Gao ZY, Lai YC. Universal Model of Individual and Population Mobility on Diverse Spatial Scales. *Nat Commun* (2017) 8:1639–9. doi:10.1038/s41467-017-01892-8
10. Tan J, Huang Y, Li Z, Wang L, Guo W, Li H. *The Characteristics on Commuting Travel Mode Split and Origin Destination (OD) Distribution in Shenzhen, China, 2017 6th Data Driven Control and Learning Systems (DDCLS)*. IEEE (2017). p. 716–20.
11. Du Z, Yang Y, Gao C, Huang L, Huang Q, Bai Y. The Temporal Network of mobile Phone Users in Changchun Municipality, Northeast China. *Sci Data* (2018) 5:180228–7. doi:10.1038/sdata.2018.228
12. Du Z, Yang Y, Ertem Z, Gao C, Bai YJED. *Inter-urban Mobility via Cellular Position Tracking in the Southeast Songliao Basin, Northeast China* (2019). p. 6.
13. Province SBOG. *Guangdong Statistical Yearbook of* (20172018).
14. Blondel VD, Guillaume J-L, Lambiotte R, Lefebvre E, experiment. Fast Unfolding of Communities in Large Networks. *J Stat Mech* (2008) 2008: P10008. doi:10.1088/1742-5468/2008/10/p10008
15. De Meo P, Ferrara E, Fiumara G, Provetti A. *Generalized Louvain Method for Community Detection in Large Networks, 2011 11th International Conference on Intelligent Systems Design and Applications*. IEEE (2011). p. 88–93.
16. Que X, Checconi F, Petrini F, Gunnels JA. *Scalable Community Detection with the Louvain Algorithm*. In: 2015 IEEE International Parallel and Distributed Processing Symposium. IEEE (2015). p. 28–37.
17. Fazlali M, Moradi E, Tabatabaee Malazi H. Adaptive Parallel Louvain Community Detection on a Multicore Platform. *Microprocessors and Microsystems* (2017) 54:26–34. doi:10.1016/j.micpro.2017.08.002
18. Yan X-Y, Zhao C, Fan Y, Di Z, Wang W-X. Universal Predictability of Mobility Patterns in Cities. *J R Soc Interf* (2014) 11:20140834. doi:10.1098/rsif.2014.0834
19. Zhang Z, Pu P, Han D, Tang M, Applications Self-Adaptive Louvain Algorithm: Fast and Stable Community Detection Algorithm Based on the Principle of Small Probability Event. *Physica A: Stat Mech its Appl* (2018) 506: 975–86. doi:10.1016/j.physa.2018.04.036
20. Traag VA, Waltman L, Van Eck NJ. From Louvain to Leiden: Guaranteeing Well-Connected Communities. *Sci Rep* (2019) 9:5233–12. doi:10.1038/s41598-019-41695-z

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

ORIGINAL RESEARCH
published: 14 January 2022
doi: 10.3389/fphy.2021.822581

Modeling Complex Networks Based on Deep Reinforcement Learning

Wenbo Song[1], *Wei Sheng*[1], *Dong Li*[1]*, *Chong Wu*[2] *and Jun Ma*[3]

[1]School of Mechanical, Electrical and Information Engineering, Shandong University, Weihai, China, [2]School of Management, Harbin Institute of Technology, Harbin, China, [3]School of Computer Science and Technology, Shandong University, Qingdao, China

The network topology of complex networks evolves dynamically with time. How to model the internal mechanism driving the dynamic change of network structure is the key problem in the field of complex networks. The models represented by WS, NW, BA usually assume that the evolution of network structure is driven by nodes' passive behaviors based on some restrictive rules. However, in fact, network nodes are intelligent individuals, which actively update their relations based on experience and environment. To overcome this limitation, we attempt to construct a network model based on deep reinforcement learning, named as NMDRL. In the new model, each node in complex networks is regarded as an intelligent agent, which reacts with the agents around it for refreshing its relationships at every moment. Extensive experiments show that our model not only can generate networks owing the properties of scale-free and small-world, but also reveal how community structures emerge and evolve. The proposed NMDRL model is helpful to study propagation, game, and cooperation behaviors in networks.

Keywords: complex network, deep reinforcement learning, scale-free, small world, community evolution

OPEN ACCESS

Edited by:
Chao Gao,
Southwest University, China

Reviewed by:
Xin Sun,
Harbin Institute of Technology, China
Fusang Zhang,
Institute of Software (CAS), China

****Correspondence:***
Dong Li
dongli@sdu.edu.cn

Specialty section:
This article was submitted to
Social Physics,
a section of the journal
Frontiers in Physics

Received: *26 November 2021*
Accepted: *22 December 2021*
Published: *14 January 2022*

Citation:
Song W, Sheng W, Li D, Wu C and Ma J (2022) Modeling Complex Networks Based on Deep Reinforcement Learning. Front. Phys. 9:822581. doi: 10.3389/fphy.2021.822581

1 INTRODUCTION

There are many complex systems in nature and human society, and most of which can be abstractly modeled as complex networks composed of nodes and links between nodes. The common social network, internet network, urban transportation network and gene regulatory network are typical complex networks. The rise of complex network theory provides a new perspective for the research of complex system science, and is of great significance to understanding the natural phenomena and the law of social operation.

Complex networks are dynamic systems that change continuously with time. How to construct network evolution models is the core problem of the complex network research. On the one hand, the network model helps to reveal the internal mechanism driving the dynamic evolution of network topology. On the other hand, as the network structure supports the physical behavior on the network, the network model also plays an important role in the study of network behaviors. Therefore, the network model is the basis of understanding network structure and function, their related studies have been widely applied in social computing [1], information retrieval, intelligent transportation, bioinformatics and other fields.

The studies of network evolution models started from 1959, the classical random network model was developed by Erdos and Renyi [2], named as ER model. This model assumes that links between nodes are generated with a random probability. With the development of data processing technology, researchers found that many real complex networks are not random networks. Aiming at this limitation, Watts et al. [3], Newman et al. [4] and Kleinberg [5] proposed small world models, which are able to explain some structure characteristics (i.e., short average distance and large clustering coefficient) of real complex networks. At the same time, the scale-free network models were developed, which leverage the mechanism of preferential connection, redirection or copy to interpret the power-law property of the node degree distribution. In recent years, following the above typical models, new mechanisms represented by weight [6–9], local world [10, 11], nonlinear growth [11–14], location information [15–18], popularity and homophily [19–21], and triangle closure [22–26] are used to construct network evolution models.

The construction idea of the existing network models are as following: based on data analysis and experimental observation, the corresponding increase and decrease rules of nodes and relationships are formulated, and nodes passively perform edge deletion and addition behaviors according to these rules. Although the existing models can generate networks that meet some characteristics of real networks, they all ignore the fact that each node in the network is an autonomous intelligent individual, and it should actively update its social relations based on their own experience and surrounding environment. The autonomous behavior of each node eventually leads to the dynamic evolution of the whole network.

Deep reinforcement learning is a subject of decision optimization, and the dynamic evolution of network is the results of node decisions. Based on the above facts, leveraging deep reinforcement learning to build a network model should be a meaningful attempt. In this paper, we propose a network modeling method based on deep reinforcement learning called the NMDRL model. In the NMDRL model, each node in the network is regarded as an agent, which continuously gains experience by taking actions to change its current state and obtaining corresponding rewards. At the same time, the agent learns to optimize its own action selection strategy by using a large amount of experience to make its autonomous decision more intelligent. After a large number of experiments, it has been shown that the NMDRL model evolves to generate a network that conforms well to the power-law distribution and the small-world characteristics of the real network. In addition, the model reproduces the emergence, growth, fusion, split and disappear of community in the evolutionary network.

The remainder of this article is structured in the following manner. In **section 2**, we detail the process of our model building, and a more detailed training process of the NMDRL model algorithm is included. The results of our experiments are presented in **section 3**. Finally, in **section 4** we present the conclusion and discussion.

2 MODEL

2.1 NMDRL Model Overview

Different from existing network models simulating the network dynamic process under some rules, we propose the deep reinforcement learning-based model without any limitation rule, referred to as NMDRL model. As the network topology structure data suffers from high dimensionality and low efficiency, our NMDRL model is developed in a low-dimensional latent space in which each node is represented as a vector. At the same time, we also construct the transforming mechanism between network low-dimensional vector representation and high dimensional topology representation.

In the NMDRL model, the dynamic evolution process of complex network is modeled as a Markovian decision process, and each network node is considered as an intelligent agent. Every agent interacts with the environment and accumulates experience through continuous exploration attempts. At the same time, the agent also can use the accumulated experience to update its parameters, enhance its intelligence and help itself to make better autonomous decisions. Following deep reinforcement learning, each agent in our model owns four basic attributes:

- States: S represents all the states in the environment, and the agent can make certain responses by sensing the states of the environment. In the NMDRL model, a two-dimensional space is decomposed into some grids with the same size, and each grid is resumed as a state. We can change the number of states by adjusting parameters (i.e., row number and column number).
- Actions: A represents all the actions of the agent. In the low-dimensional space, the agent has six actions in total: up, down, left, right, stay and random. "stay" means the agent does not move its position, and "random" means that the agent randomly selects one from all states.
- Reward: R represents the reward or penalty of an agent after taking an action. The reward is returned from the environment use to evaluate the action performed by an agent. When an agent moves from one state to another state, the reward obtained by this agent is the number of all nodes located in the new state. For example, an agent i moves from *state 1* to *state 2*, and there are 10 agents in the *state 2*. In this situation, the reward obtained by agent i is 10.
- Policy: Q is the behavior function of an agent. It determines how the agent chooses its next action. In this paper, the Q function is fitted by a neural network with the parameters Θ. There are two popular strategies for action selection through the behavior function. 1) The action corresponding to the biggest value of Q can be chosen, this strategy is called the greedy policy. 2) The ϵ-greedy policy, which is a strategy including both random and greedy policies. The formula of the ϵ-greedy policy is as following:

$$a = \begin{cases} \text{random action}, & \text{with probability } \epsilon \\ \arg\max_a Q(s, a), & \text{otherwise} \end{cases} \tag{1}$$

In order to ensure that the agent continues to explore the environment and fully make use of its existing experience, our NMDRL model adopts the ϵ-greedy policy.

2.2 NMDRL Model Training

In the NMDRL model, each node in the network is considered as an agent. This section introduces how to train the NMDRL

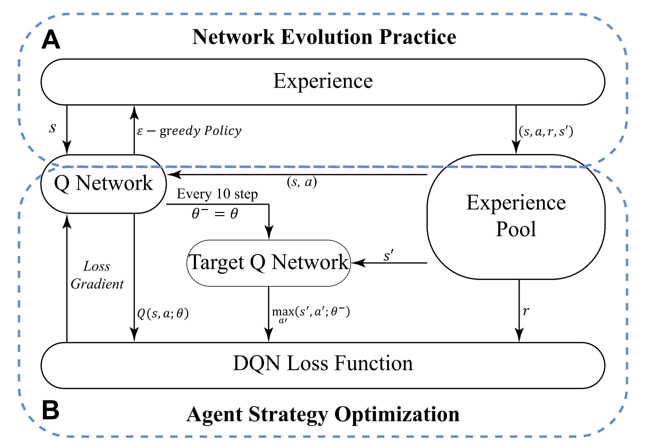


FIGURE 1 | NMDRL training flow. **(A)** is the network evolution practice part of the algorithm, **(B)** is the agent strategy optimization part of the algorithm.

model, and makes each agent have the ability to make autonomous decisions in the process of network evolution. As shown in **Figure 1**, the NMDRL model training mainly includes network evolution practice and agent strategy optimization.

Specifically, **Figure 1A** shows the details of the **network evolution practice**, where the Q-network predicts the q-value of all the actions generated by the agent with state *s*, and then selects the action *a* according to the ϵ-greedy policy. Agent action will lead the network evolution. In this paper, the Q-network is implemented by a multilayer neural network presented in **Figure 2**. The neural network consists of an input layer, two hidden layers and an output layer, where the two hidden layers contain 128 and 64 neurons, respectively. The input of the neural network is the vector representation of the agent's state, and the final output is the estimation *Q* of the agent action after the calculation of the two hidden layers. A quadruplet (*s*, *a*, *r*, *s*′) can be extracted from the above network evolution practice, and denoted as a piece of experience. The experience pool $\{(s_1, a_1, r_1, s_2)\ldots(s_t, a_t, r_t, s_{t+1})\ldots\}$ is a collection of many experience data stored, and is used to train Q-network.

In order to optimize the agent evolution strategy, we first need to define the loss function of the Q-network that is to minimize the reward error between the predicted and true values. According to the Bellman equation, the loss function of the Q-network is defined as given in **Eq. 2**, which ensures that the Q-network performs more effective learning.

$$L_Q = \left(r_t + \gamma \max_a \hat{Q}(s_{t+1}, a) - Q(s_t, a_t)\right)^2 \tag{2}$$

$\hat{Q}(s_{t+1}, a)$ is the predicted value of the target Q-network, which is introduced to ensure more stable training and is a copy of the Q-network parameters Θ at fixed time intervals. $Q(s_t, a_t)$ is the output of the Q-network in the current state s_t with behavior a_t. γ is a discount for future rewards.

The process of **agent policy optimization** part is shown in **Figure 1B**, where the agent will sample the experience randomly from the experience pool, put (*s*, *a*) into the Q-network to obtain the predicted Q value, use the maximum value obtained from *s*′ in the target Q-network as the target value. The parameters of the Q-network are updated using the loss function given in **Eq. 2**. At the same time, the target Q-network is updated once for every 10 updates to the Q-network. After continuously collecting and learning from the network evolution experience, the network evolution strategy of each agent is continuously updated. Finally, all the agents work together to model the network according to their own network evolution strategies.

Algorithm 1. Training Algorithm of the NMDRL.

```
Input: State row number $N_r$, State column number $N_c$, Number of nodes(agents) in each state at the
    beginning $n$, max episode number $T$, max number of experience pool $E$, exploration probability $\epsilon$, and
    target Q-network copy frequency $F$, ability to value at future reward $\gamma$
Output: All agent parameters $\Theta$
1: Total number of nodes $N = N_r \times N_c \times n$
2: for agent from 1 to $N$ do
3:     Initialize experience pool $P$ with max size $E$
4:     Initialize agent parameters $\Theta$
5:     Initialize target Q-network parameters $\hat{\Theta} = \Theta$
6: end for
7: for $episode$ = 1 to $T$ do
8:     for agent from 1 to $N$ do
9:         Select $a = \begin{cases} \text{random action}, & \text{with probability } \epsilon \\ \arg\max_a Q(s,a), & \text{otherwise} \end{cases}$
10:        Make action and move to the corresponding state $s'$ and get reward r
11:        Insert $(s, a, r, s')$ into P
12:        Sample batch experience from $P$ randomly
13:        Update parameters $\Theta$ using Gradient Descent
14:        Update target Q-network parameters $\hat{\Theta} = \Theta$ every $F$ episodes
15:    end for
16:    Update the low-dimensional coordinates
17:    Update all nodes in the state
18:    Update agent Neighbors in a same state
19:    Convert low-dimensional data into high-dimensional network data
20: end for
```

In summary, the detailed procedure of training the NMDRL network model is described in Algorithm 1. First, the experience pool, Q-network and target Q-network of all agents are initialized in lines 1–6. In lines 9–15, NMDRL training is performed for all agents, including the practice of network evolution and policy optimization. In lines 16–18, the global information obtained from the network evolution is updated. Finally, in line 19, the low-dimensional space data is transformed to the high-dimensional space data.

2.3 The Network Conversion Mechanism From Low-Dimension Representation to High-Dimension Representation

In the NMDRL model, the interactions among nodes and move behaviors of nodes are designed in a low-dimensional space. Therefore, a mechanism of converting a network from low-dimensional vector representation to high-dimensional topology representation is necessary.

The aids of the conversion mechanism are to add/delete edges and assign weights for these edges based on the network low-dimensional vector representation. In this paper, we make use of the threshold and attenuation rules to design this conversion mechanism. We assume that if two nodes are in a same state, an edge between them will be created. The weight of this edge will be weakened or enhanced over time. When the value of the weight is smaller than a threshold, this edge will be deleted. The details of the conversion mechanism are as following:

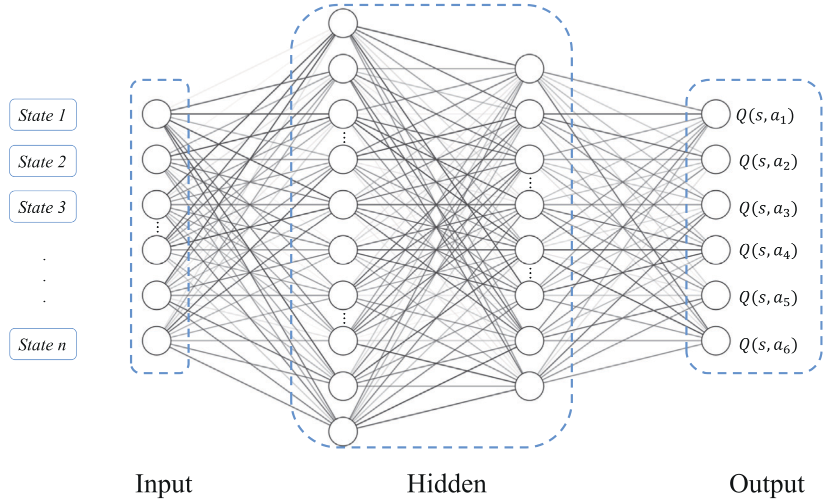


FIGURE 2 | Structure of the NMDRL multilayer neural network.

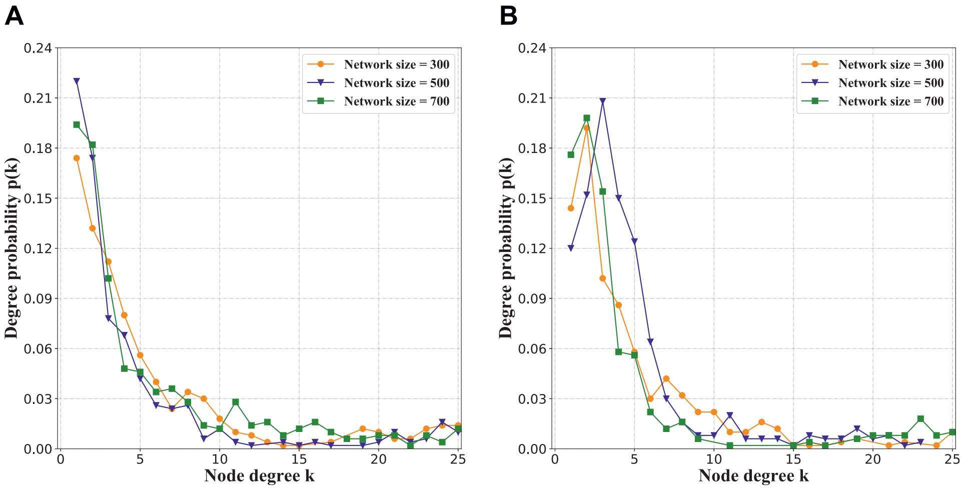


FIGURE 3 | (A,B) are the degree distributions of the networks generated by NMDRL model.

- If two nodes with no edge are in a same state, we create a new edge for these two nodes, and assign an initial weight $w^{initial}$ for this edge.
- If two nodes with an edge are not in a same state any more, the weight of the edge will be weaken. We denote the weight of the existing edge as w^{last} at the last time step. The new weight $w^{current}$ at current time step is w^{last}/a, where a is the attenuation index. If two nodes owing one edge are not in a same state for several time steps, the edge weight will be continuously weakened. Once the new weight $w^{current}$ is smaller than the threshold t, we believe that the relationship strength between the two nodes is too weak and this edge should no longer exist. This edge will be deleted.
- If two nodes with an edge are in a same state again, the weight of the edge will be strengthened. The new weight $w^{current}$ at current time step is $w^{last}/a+ w^{initial}$, where the former part is the attenuation of time to the past weight, and the second part is the enhancement of the weight when two nodes are in a same state.

3 RESULTS

In this section, we conduct comprehensive experiments to validate the effectiveness of the NMDRL model from the following aspects: node degree distribution, network clustering coefficient, network average distance, community

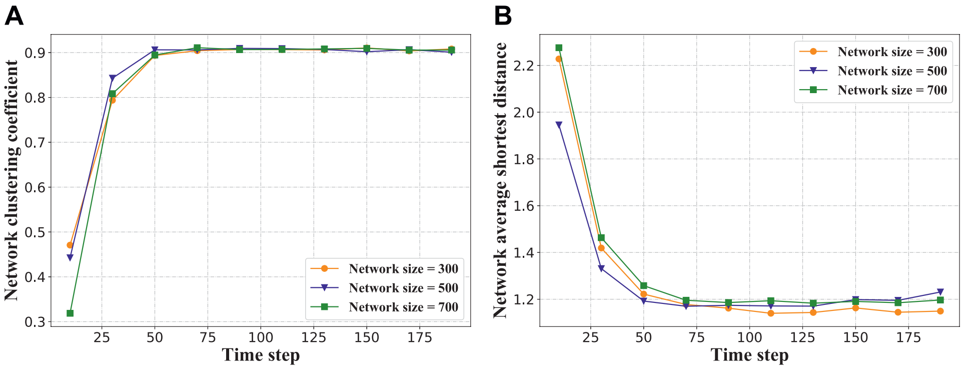


FIGURE 4 | (A) is variation of network clustering coefficient with time step, **(B)** is variation of network average distance with time step.

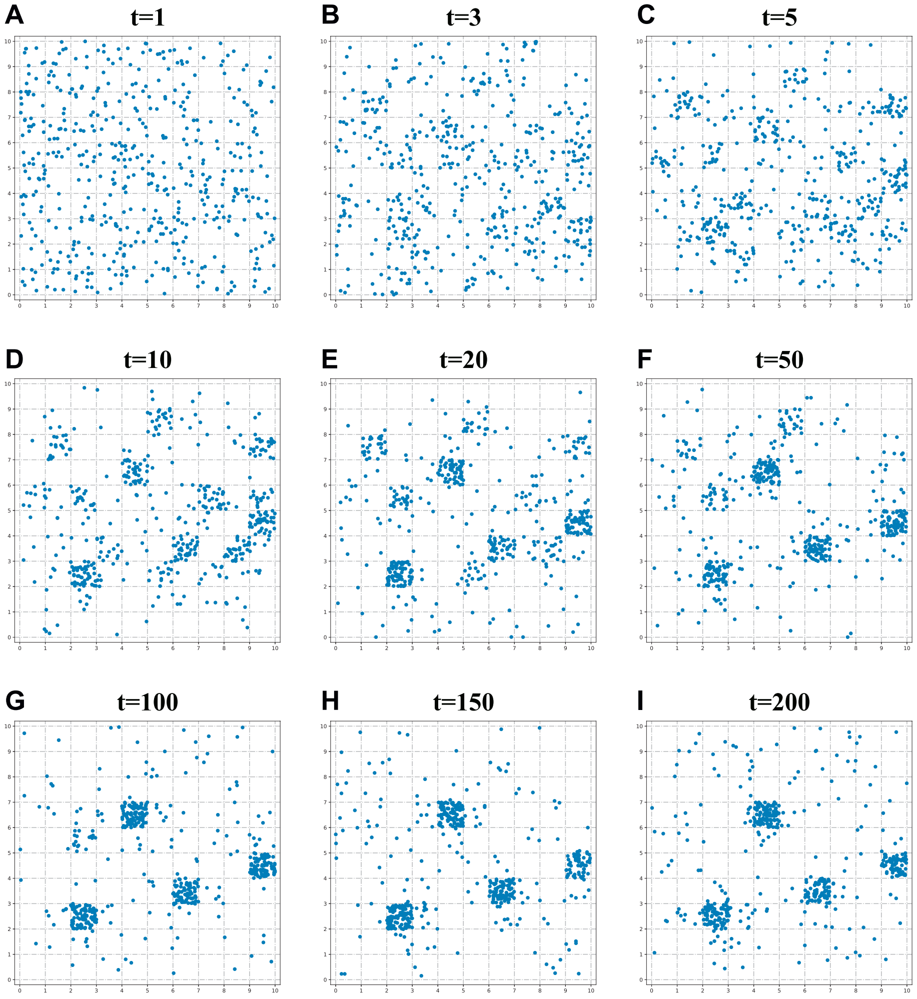


FIGURE 5 | Demonstration of network evolution process in the low dimensional vector space with parameters $\gamma = 0.2$, $\epsilon = 0.2$, network size = 500, **(A)** t = 1, **(B)** t = 3, **(C)** t = 5, **(D)** t = 10, **(E)** t = 20, **(F)** t = 50, **(G)** t = 100, **(H)** t = 150, **(I)** t = 200.

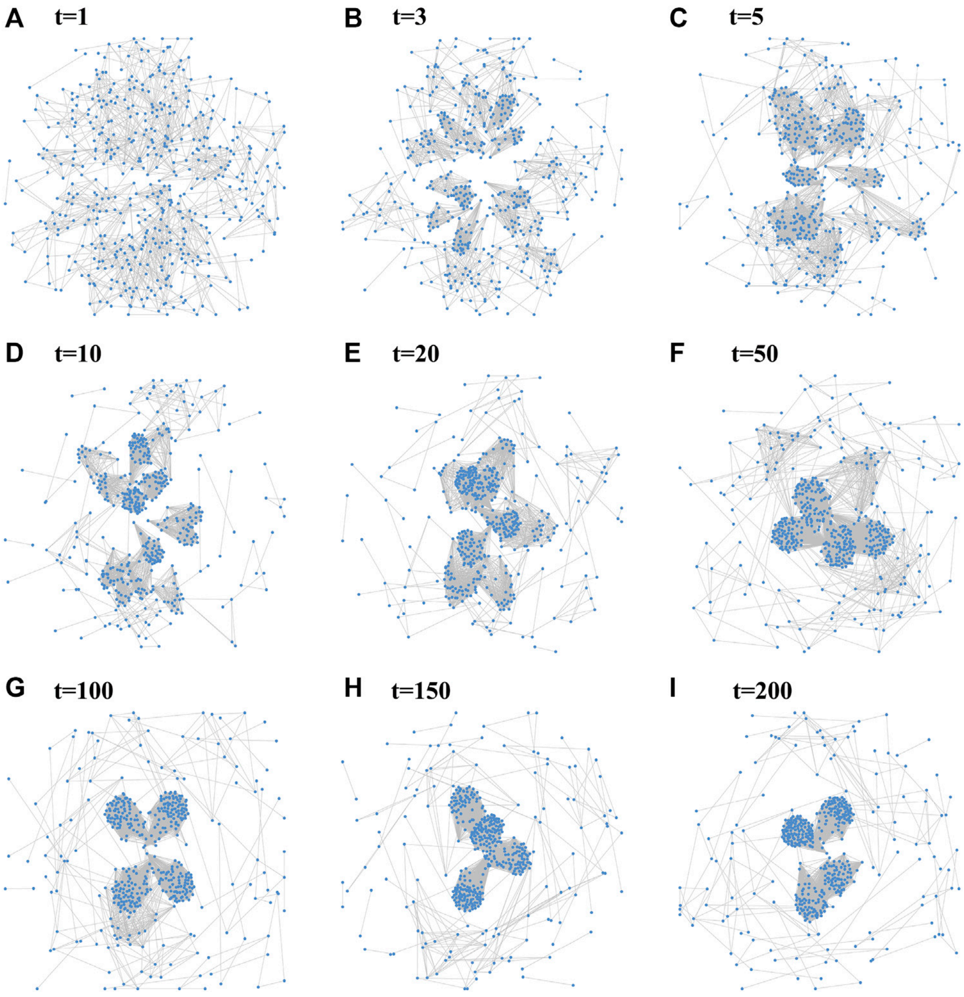


FIGURE 6 | Demonstration of network evolution process in the high dimensional vector space with parameters $\gamma = 0.2$, $\epsilon = 0.2$, network size = 500, **(A)** t = 1, **(B)** t = 3, **(C)** t = 5, **(D)** t = 10, **(E)** t = 20, **(F)** t = 50, **(G)** t = 100, **(H)** t = 150, **(I)** t = 200.

formation and evolution. In the experiments, we set row number and column number of two-dimensional space to be 10, so there are 100 states in total.

3.1 Degree Distribution

In a network, the degree of a node is the number of connections this node owns. The degree distribution $P(k)$ of a network is defined to be the fraction of nodes in the network with degree k [27], which is an important index in studying complex networks. Here, we try to analyze the degree distributions of the networks generated by the NMDRL model.

Figure 3A shows the degree distributions of the networks generated by the NMDRL model, where orange, blue and green curves represent three networks containing 300, 500 and 700 nodes respectively. Other parameters γ and ϵ are set to be 0.8 and 0.6 respectively. It can be seen from **Figure 3A** that there are a small number of nodes with high degree and a large number of nodes with a low degree in the generated networks. This means that the degree distribution of the networks created by the NMDRL model follows the power-law distribution property. The exponent of the power-law distribution is between (2, 3), and the generated networks are scale-free networks.

Although power-law distribution is the most common degree distribution, not all real networks own this kind of degree distribution. Researchers find that some real networks obey the subnormal distribution which is between the normal distribution and the power-law distribution [28]. Here, we adjust the parameter ϵ from 0.6 to 0.7, and other parameters remain unchanged. **Figure 3B** presents the degree distributions of the generated networks with three sizes. We can see that the degree distributions of these networks follow the subnormal distribution form. All of the above results indicate that our NMDRL model is able to generate networks with power-law or subnormal distribution that exist in real networks.

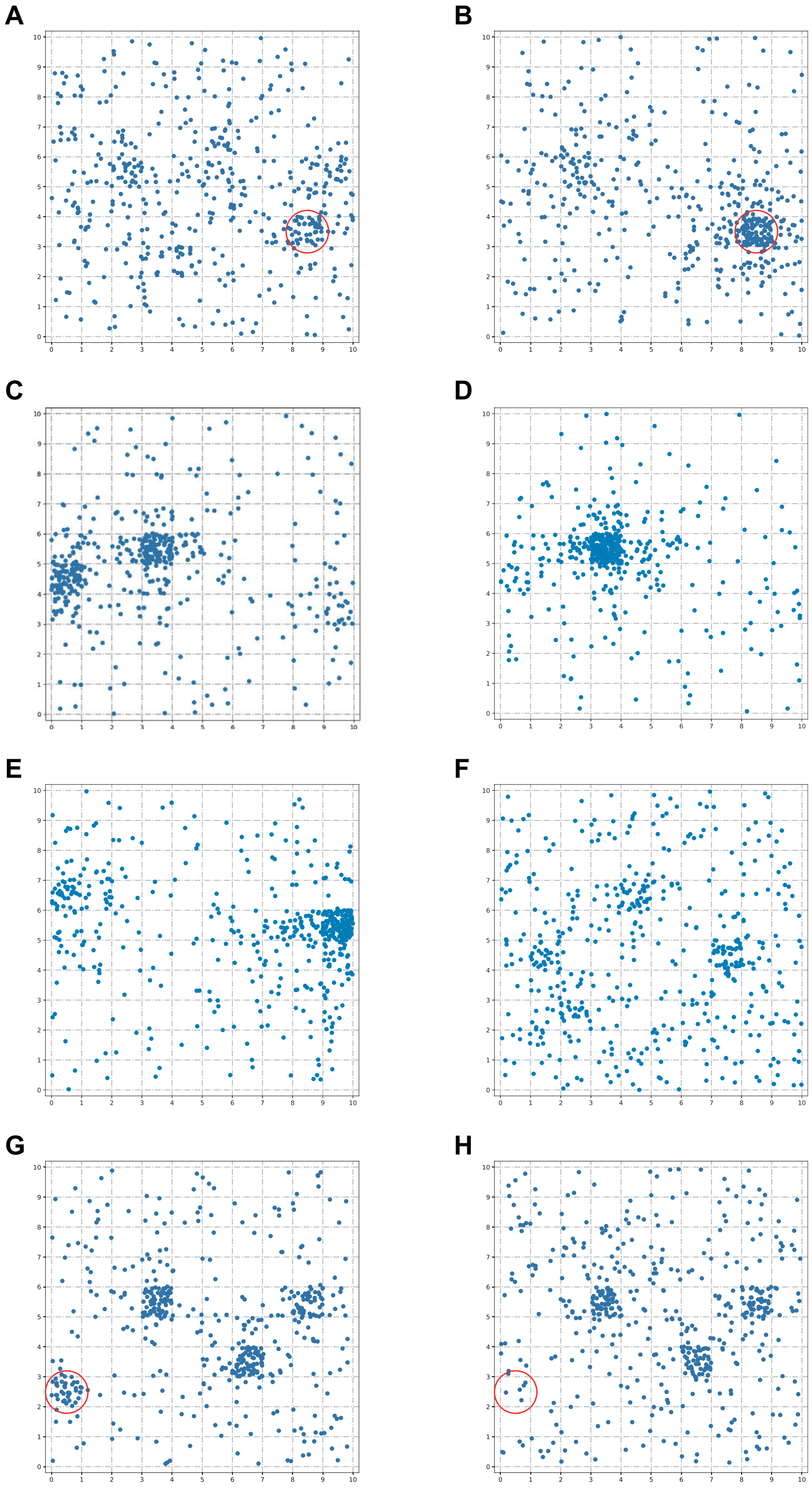


FIGURE 7 | Evolutionary behaviors of communities. **(A,B)** presents community growth behavior, **(C,D)** presents community fusion behavior, **(E,F)** presents community split behavior, and **(G,H)** presents community disappear behavior.

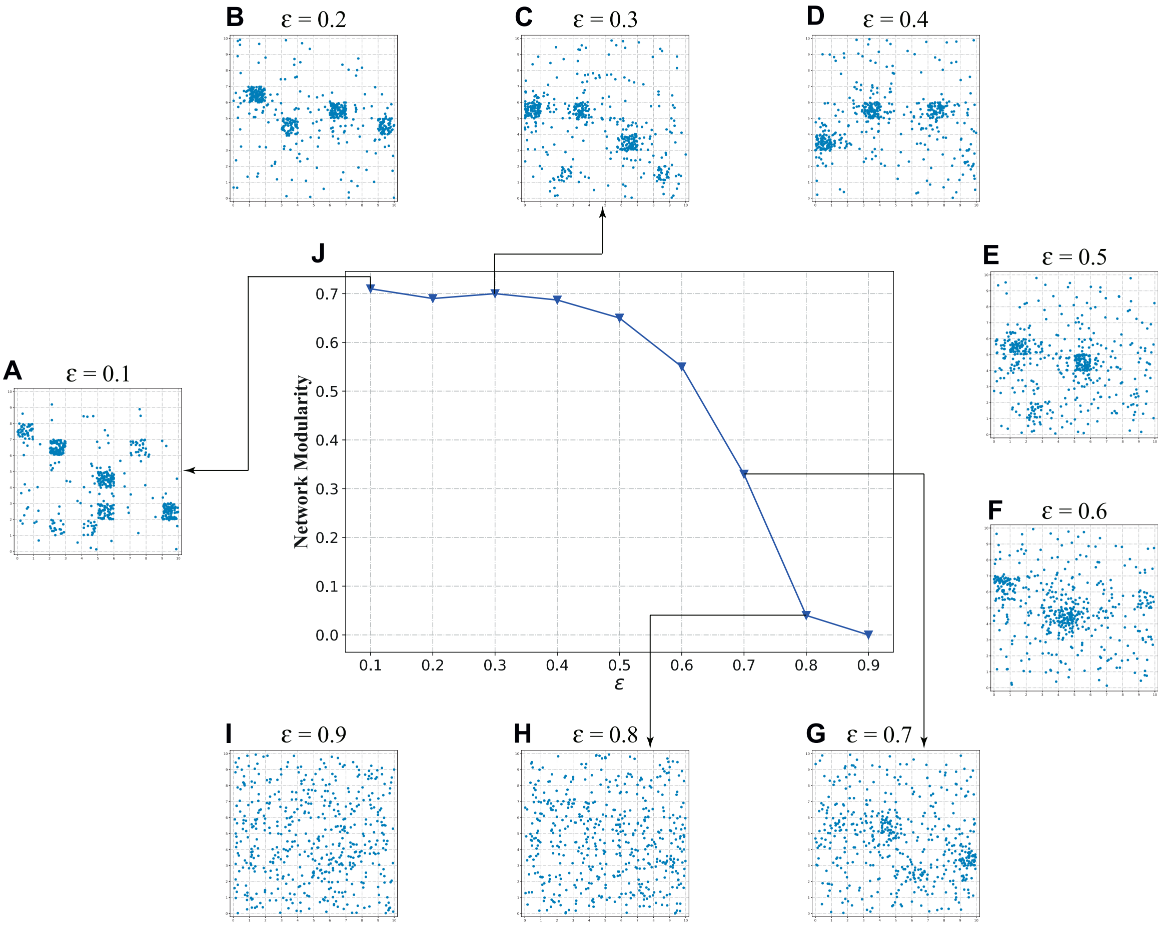


FIGURE 8 | Impact of ϵ on the network community evolution process.**(A–I)** are the representations of ϵ in (0.1–0.9) in the low dimensional vector space, respectively, and **(J)** is the variation curve of network modularity with increasing ϵ.

3.2 Clustering Coefficient and Average Path Length

Clustering coefficient and average path length are also two classic metrics of complex networks. For a network, if its clustering coefficient is large and average path length is small, this network can be called a small-world network. Here, we try to explore whether the networks generated by the NMDRL model own the small-world property.

Figure 4A plots the change of clustering coefficient over 200-time steps for three network sizes. x-axis represents the time step, and y-axis represents the network clustering coefficient. Orange, blue and green curves are corresponding to three networks owing 300, 500 and 700 nodes respectively. It can be seen from **Figure 4A** that the value of the clustering coefficient is small in the early stage of network evolution. This is because that each grid in the two-dimensional space owns the same number of nodes at the initial time, therefore the initial network is a regular network in fact whose clustering coefficient is small. The clustering coefficient of the dynamic network increases from time 1 to 70, and changes a little from 71 to 200. This means that the dynamic network reaches the stable state with a high clustering coefficient.

Figure 4B plots the change of average path length over time for different network sizes. At the initial time step, the average path length of the network is relatively large. This value decreases from time 1 to 70, and changes a little from 71 to 200. The phenomenon observed from **Figure 4A** and **Figure 4B** indicates that our proposed model is able to drive a regular network to evolve into a network with a big clustering coefficient and small average path length (i.e., small-world network).

3.3 Community Emergence and Evolution

In this subsection, we try to analyze the community [29, 30] formation and evolution capability of the NMDRL model. Since our model is constructed in a low-dimensional space, and a network conversion mechanism from low-dimensional representation to high-dimensional representation is developed at the same time. This design makes us analyze and verify the model effect from both low and high dimensional levels.

Specifically, **Figure 5** and **Figure 6** show the network dynamic evolution process from time 1 to 200 in low-dimensional and high-dimensional spaces respectively. Both of these two figures clearly present the formation process of community structures in the network, and the communities in high and low dimensional spaces match each other very well. These observed results illustrate the effectiveness of our NMDRL model in terms of explaining the mechanism of community emergence on the one hand, and also support the rationality of our transformation mechanism on the other hand.

Besides community formation, we also find that our NMDRL model is able to reproduce several common evolutionary behaviors. 1) *Growth*. The size of the circular region in **Figure 7A** and **Figure 7B** shows a significant growth over time. 2) *Fusion*. Some smaller communities in **Figure 7C** move to the larger communities. They fuse together after some time steps and form some larger communities in **Figure 7D**. 3) *Split*. An extremely large community in **Figure 7E** splits into multiple smaller communities in **Figure 7F**. 4) *Disappear*. The community in the circular region in **Figure 7G** disappears in **Figure 7H**.

In the NMDRL model, parameter ϵ is used to balance the ability of agent to explore and exploit. Here, we attempt to analyze the impact of ϵ on community emergence. **Figure 8J** presents the change of network modularity under different ϵ values. When ϵ is in the range of 0.1–0.5, network modularity is smoothly maintained at a high level. Start with $\epsilon = 0.5$, network modularity sharply decreases and enters into a very low level. It is also observed that when $\epsilon > 0.7$, the network will no longer have associations when the data in the low dimensional vector space shown in **Figures 8A–I**. This indicates that the ϵ is a key parameter of determining whether the community appears.

4 CONCLUSION AND DISCUSSION

In view of the limitation of nodes passively updating relationships in the existing models, we try to leverage deep reinforcement learning to develop a network evolution model. In the model, each node considered as an agent interacts with its neighbors and makes strategic choices based on its utility at every moment. A large number of simulation results validate that the generated networks by our model have three most important structure characters of real networks: scale-free, small-world and community.

Some challenges remain. How to learn model parameters based on real networks and apply the learned model in some typical tasks (*i.e.* link prediction) is one of our future directions. To be more relevant, the impact of multiple agents with different intelligence on the evolving network will also be another direction of our future research.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

WS: methodology, software, data analysis and writing-original draft. WS: data analysis, writing-review and editing. DL: conceptualization, methodology, writing–original draft and review. CW and JM: supervision, writing-review and editing.

FUNDING

This work is funded by the National Natural Science Foundation of China (Nos. 62076149, 61702138, 61602128, 61672322, 61672185, 72131005 and 71771066), and the Shandong Province Natural Science Foundation of China (No. ZR2016FQ13), and the China Postdoctoral Science Foundation (Nos. 2020T130368, 2019M662360, 2017M621275 and 2018T110301), and Young Scholars Program of Shandong University, China, Weihai (No. 1050501318006), and Science and Technology Development Plan of Weihai City, China (No. 1050413421912).

REFERENCES

1. Pei H, Yang B, Liu J, Chang K. Active Surveillance via Group Sparse Bayesian Learning. *IEEE Trans Pattern Anal Mach Intell* (2020) Online ahead of print. doi:10.1109/TPAMI.2020.3023092
2. Erdos P, Rényi A. On the Evolution of Random Graphs. *Publ Math Inst Hung Acad Sci* (1960) 5(1):17–60.
3. Watts DJ, Strogatz SH. Collective Dynamics of 'Small-World' Networks. *Nature* (1998) 393(6684):440–2. doi:10.1038/30918
4. Newman ME, Watts DJ. Renormalization Group Analysis of the Small-World Network Model. *Phys Lett A* (1999) 263(4-6):341–6. doi:10.1016/s0375-9601(99)00757-4
5. Kleinberg JM. Navigation in a Small World. *Nature* (2000) 406(6798):845. doi:10.1038/35022643
6. Yook SH, Jeong H, Barabási A-L, Tu Y. Weighted Evolving Networks. *Phys Rev Lett* (2001) 86(25):5835–8. doi:10.1103/physrevlett.86.5835
7. Zhou Y-B, Cai S-M, Wang W-X, Zhou P-L. Age-Based Model for Weighted Network with General Assortative Mixing. *Physica A: Stat Mech its Appl* (2009) 388(6):999–1006. doi:10.1016/j.physa.2008.11.042
8. Barrat A, Barthelemy M, Pastor-Satorras R, Vespignani A. The Architecture of Complex Weighted Networks. *Proc Natl Acad Sci* (2004) 101(11):3747–52. doi:10.1073/pnas.0400087101
9. Li P, Yu J, Liu J, Zhou D, Cao B. Generating Weighted Social Networks Using Multigraph. *Physica A: Stat Mech its Appl* (2020) 539:122894. doi:10.1016/j.physa.2019.122894
10. Li X, Chen G. A Local-World Evolving Network Model. *Physica A: Stat Mech its Appl* (2003) 328(1-2):274–86. doi:10.1016/s0378-4371(03)00604-6
11. Feng S, Xin M, Lv T, Hu B. A Novel Evolving Model of Urban Rail Transit Networks Based on the Local-World Theory. *Physica A: Stat Mech its Appl* (2019) 535:122227. doi:10.1016/j.physa.2019.122227
12. Feng M, Deng L, Kurths J. Evolving Networks Based on Birth and Death Process Regarding the Scale Stationarity. *Chaos* (2018) 28(8):083118. doi:10.1063/1.5038382

13. Liu J, Li J, Chen Y, Chen X, Zhou Z, Yang Z, et al. Modeling Complex Networks with Accelerating Growth and Aging Effect. *Phys Lett A* (2019) 383(13): 1396–400. doi:10.1016/j.physleta.2019.02.004
14. Li J, Zhou S, Li X, Li X. An Insertion-Deletion-Compensation Model with Poisson Process for Scale-Free Networks. *Future Generation Comput Syst* (2018) 83:425–30. doi:10.1016/j.future.2017.04.011
15. Hristova D, Williams MJ, Musolesi M, Panzarasa P, Mascolo C. Measuring Urban Social Diversity Using Interconnected Geo-Social Networks. In: Proceedings of the 25th international conference on world wide web; 2016 May 11–15; Montreal, Canada (2016). p. 21–30. doi:10.1145/2872427.2883065
16. Noulas A, Shaw B, Lambiotte R, Mascolo C. Topological Properties and Temporal Dynamics of Place Networks in Urban Environments. In: Proceedings of the 24th International Conference on World Wide Web; 2015 May 18–22; Florence, Italy (2015). p. 431–41. doi:10.1145/2740908.2745402
17. Zhou L, Zhang Y, Pang J, Li C-T. Modeling City Locations as Complex Networks: An Initial Study. In: International Workshop on Complex Networks and their Applications; 2016 November 30–December 02; Milan, Italy. Springer (2016). p. 735–47. doi:10.1007/978-3-319-50901-3_58
18. Ding Y, Li X, Tian Y-C, Ledwich G, Mishra Y, Zhou C. Generating Scale-free Topology for Wireless Neighborhood Area Networks in Smart Grid. *IEEE Trans Smart Grid* (2018) 10(4):4245–52. doi:10.1109/TSG.2018.2854645
19. Muscoloni A, Cannistraci CV. A Nonuniform Popularity-Similarity Optimization (Npso) Model to Efficiently Generate Realistic Complex Networks with Communities. *New J Phys* (2018) 20(5):052002. doi:10.1088/1367-2630/aac06f
20. Papadopoulos F, Kitsak M, Serrano MÁ, Boguñá M, Krioukov D. Popularity versus Similarity in Growing Networks. *Nature* (2012) 489(7417):537–40. doi:10.1038/nature11459
21. Liu Y, Li L, Wang H, Sun C, Chen X, He J, et al. The Competition of Homophily and Popularity in Growing and Evolving Social Networks. *Scientific Rep* (2018) 8(1):1–15. doi:10.1038/s41598-018-33409-8
22. Holme P, Kim BJ. Growing Scale-free Networks with Tunable Clustering. *Phys Rev E Stat Nonlin Soft Matter Phys* (2002) 65(2):026107. doi:10.1103/PhysRevE.65.026107
23. Li G, Li B, Jiang Y, Jiao W, Lan H, Zhu C. A New Method for Automatically Modelling Brain Functional Networks. *Biomed Signal Process Control* (2018) 45:70–9. doi:10.1016/j.bspc.2018.05.024
24. Tang T, Hu G. An Evolving Network Model Based on a Triangular Connecting Mechanism for the Internet Topology. . In: International Conference on Artificial Intelligence and Security; 2019 July 26–28; New York, NY. Springer (2019). p. 510–9. doi:10.1007/978-3-030-24268-8_47
25. Overgoor J, Benson A, Ugander J. Choosing to Grow a Graph: Modeling Network Formation as Discrete Choice. In: The World Wide Web Conference; 2019 May 13–17; San Francisco, CA (2019). p. 1409–20.
26. Jia D, Yin B, Huang X, Ning Y, Christakis NA, Jia J. Association Analysis of Private Information in Distributed Social Networks Based on Big Data. *Wireless Commun Mobile Comput* (2021) 2021(1):1–12. doi:10.1155/2021/1181129
27. Barabási AL, Albert R. Emergence of Scaling in Random Networks. *Science* (1999) 286(5439):509–12. doi:10.1126/science.286.5439.509
28. Feng M, Qu H, Yi Z, Kurths J. Subnormal Distribution Derived from Evolving Networks with Variable Elements. *IEEE Trans Cybern* (2017) 48(9):2556–68. doi:10.1109/TCYB.2017.2751073
29. Zhang F, Liu H, Leung Y-W, Chu X, Jin B. Cbs: Community-Based Bus System as Routing Backbone for Vehicular Ad Hoc Networks. *IEEE Trans Mobile Comput* (2016) 16(8):2132–46. doi:10.1109/TMC.2016.2613869
30. Zhang F, Zhang D, Xiong J, Wang H, Niu K, Jin B, et al. From Fresnel Diffraction Model to fine-grained Human Respiration Sensing with Commodity Wi-Fi Devices. *Proc ACM Interact Mob Wearable Ubiquitous Technol* (2018) 2(1):1–23. doi:10.1145/3191785

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

ORIGINAL RESEARCH
published: 21 January 2022
doi: 10.3389/fphy.2022.834094

Identifying Critical Meteorological Elements for Vegetation Coverage Change in China

Huimin Bai[1], *Li Li*[2], *Yongping Wu*[3], *Guolin Feng*[3,4], *Zhiqiang Gong*[4] *and Guiquan Sun*[1,5]*

[1]Complex Systems Research Center, Shanxi University, Taiyuan, China, [2]School of Computer and Information Technology, Shanxi University, Taiyuan, China, [3]College of Physics Science and Technology, Yangzhou University, Yangzhou, China, [4]Laboratory for Climate Studies, National Climate Center, China Meteorological Administration, Beijing, China, [5]Department of Mathematics, North University of China, Taiyuan, China

Intensifying global climate change has a significant influence on the vegetation, which is the basis of most of Earth's ecosystems. It is urgent to identify the critical meteorological elements of vegetation coverage changes to address the problems induced by climate change. Many studies, ranging from theoretical advances to data-driven analyses, have been devoted to investigating meteorological elements' roles in changing vegetation coverage. However, little has been considered in the aspect of the meteorological elements' seasonal scale in data-driven studies. Herein, taking China as an example, we collected satellite-derived vegetation coverage data from 2000 to 2020. We then analyzed the meteorological elements, on a seasonal scale, that affect the vegetation coverage change in terms of temperature, precipitation, and solar radiation. We revealed that the critical meteorological elements facilitating vegetation coverage area change differ in both time and space and gave a detailed analysis in line with such findings. Moreover, an apparent seasonal delay effect of meteorological elements on the vegetation coverage change is uncovered.

Keywords: vegetation coverage, temperature, precipitation, solar radiation, China, Lasso regression

OPEN ACCESS

Edited by:
Peican Zhu,
Northwestern Polytechnical University, China

Reviewed by:
Fei Ji,
Lanzhou University, China
Chaoqun Xu,
Jiangsu University, China

****Correspondence:***
Guiquan Sun
gquansun@126.com

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *13 December 2021*
Accepted: *03 January 2022*
Published: *21 January 2022*

Citation:
Bai H, Li L, Wu Y, Feng G, Gong Z and Sun G (2022) Identifying Critical Meteorological Elements for Vegetation Coverage Change in China. Front. Phys. 10:834094. doi: 10.3389/fphy.2022.834094

1 INTRODUCTION

Changes in ecosystem structure and function are caused by climate change, topography, and human activities, such as afforestation [1, 2]. Vegetation is the medium of land–air interaction, and it purifies the air and provides food [3–7]. Vegetation growth requires three processes: photosynthesis is a process in which green plants absorb water and CO_2 through leaf stomata, produce organic matter, and release O_2 under the action of visible light and enzyme catalysis; transpiration is a process in which plants absorb water from roots, only 1%–5% of which is used for photosynthesis, and the remaining is emitted into the air through leaf stomata; respiration is the process of oxidative decomposition of plants to release energy, water, and CO_2 [8, 9]. The vegetation mainly interacts with the outside world through leaf stomata; the leaf stomatal state and conductance has remarkable difference under different climatic conditions [10, 11]. Most vegetation only opens stomata under light and interacts with the external environment; stomatal conductance is determined by temperature, moisture, humidity, and CO_2 concentration [12–15]. Thus, temperature, precipitation, humidity, solar radiation, and CO_2 concentration will affect vegetation growth [16–20].

The scale has always been a research hot spot in the field of ecology [1]. In recent years, most scholars have been carrying out the influence of climate change on vegetation in different temporal and spatial scales based on remote sensing data [21–25]. On the global scale, it is proposed that the growing season vegetation change in the high latitudes of the northern hemisphere is governed by temperature, the arid and semi-arid areas are dominated by precipitation, and Amazon and South and East Asia are dominated by solar radiation [16, 17]. Chen et al. [26] studied the impact of different climatic periods on vegetation change in the Northern Hemisphere from 1982 to 2013 and found that the impact of temperature on vegetation gradually decreased from spring to autumn. Conversely, the impact of solar radiation on vegetation increased. On the regional scale, Qu et al. [3] studied the key meteorological factors affecting vegetation growth during the growing season in China, indicating that northern China is mainly affected by precipitation, and other regions are affected by temperature. Piao et al. [21] studied the correlation between NDVI and climate variables for temperate grassland in China from 1982 to 1999, suggesting that the trend change of the NDVI caused by climate change is different between different vegetation types and seasons. Zhou et al. [27] analyzed the relationship between climate variability and NDVI in eastern China through correlation analysis, which showed that the NDVI in the arid area was negatively correlated with temperature and positively correlated with precipitation. The dominant factor of vegetation change is the temperature in southern China [23].

We know that the critical meteorological factors affecting vegetation change are different among different regions, decades, seasons, and vegetation types. The previous research mainly studies how temperature and precipitation affect vegetation change through correlation and trend analyses. Still, little is known about how vegetation growth responds to solar radiation change and quantifies the impact of seasonal meteorological factors on vegetation change in China. In addition, studies have shown that the impact of meteorological elements on vegetation growth has a delayed effect [28–30], Saatchi et al. [31] investigated that low-frequency drought events in the Amazon cause continuous change of forest canopy. Vegetation growth is driven by the current climate conditions and depends on early climate conditions. Therefore, the delay effect must be considered when studying the impact of meteorological elements on vegetation. The seasonal effect of meteorological elements on vegetation coverage is unclear in China. Understanding how vegetation change responds to meteorological elements is conducive to predicting and evaluating future vegetation changes.

This study aims to identify the critical meteorological element (temperature, precipitation, and solar radiation) periods that influence vegetation growth in different regions of China. Using reanalysis of meteorological elements and satellite-derived vegetation coverage data in China during 2000–2020, we examined the relationship between meteorological elements and vegetation coverage change. We explored the delay effects of meteorological elements on vegetation coverage change on a seasonal scale. More importantly, we analyzed how meteorological elements in different seasons (winter, spring, summer, and autumn) have influenced vegetation coverage in the growing season using least absolute shrinkage and selection operator (Lasso) regression analysis.

2 MATERIALS AND METHODS

2.1 Data and Preprocessing

Vegetation coverage (VC) is a critical indicator of vegetation growth, which is often used in research fields of ecology, climate, hydrology, and so on [32]. The monthly VC data are calculated from the NDVI data in the 1-km monthly synthetic product of the moderate-resolution imaging spectrometer (MODIS) according to **Eq. 1**.

$$VC = \frac{NDVI - NDVI_s}{NDVI_v - NDVI_s}, \tag{1}$$

where $NDVI_s$ represents the pixel value without vegetation coverage, and $NDVI_v$ represents the pixel value of complete vegetation coverage. The monthly VC at a spatial resolution of $0.01^\circ \times 0.01^\circ$ was observed during 2000–2020. Vegetation coverage less than 0.05 is considered as non-vegetated areas, which are not considered in the study [33]. VC images of the growing season (April–October) [33–37], spring (April–May), summer (June–August), and autumn (September–October) were obtained by calculating the mean of the corresponding months [21]. The meteorological elements (0.25° monthly 2m temperature (TEM), total precipitation (TPR), and surface net solar radiation (SSR)) for the period 1999–2020 are obtained from ERA Interim Data from the European Centre for Medium Range Weather Forecast (ECMWF).

We resample the VC data to ensure the same resolution of the meteorological element data, and the linear trend of the VC and meteorological element time series was removed before statistical analysis [26, 38].

2.2 Methodology

2.2.1 Partial Correlation Analysis

We analyze the data for partial correlation to explore the relationship between VC and single meteorological elements (TEM, TPR, and SSR) after controlling the influence of other variables. We calculate the partial correlation coefficient between meteorological elements and VC. Strong partial correlation means that meteorological element exerts strongly impact VC change in the regions. The critical value of the partial correlation coefficient at the 5% significance level is 0.455. We can significantly correlate VC and meteorological elements if the calculated absolute values are more significant than the critical value. The partial correlation coefficient between vegetation coverage and SSR after controlling the two variables (TEM and TPR) is

$$r_{SSR,VC,TEM,TPR} = \frac{r_{SSR,VC,TEM} - r_{SSR,TEM,TPR} \cdot r_{VC,TEM,TPR}}{\sqrt{1 - r^2_{SSR,TEM,TPR}}\sqrt{1 - r^2_{VC,TEM,TPR}}}, \tag{2}$$

where

$$r_{SSR,VC,TEM} = \frac{r_{SSR,VC} - r_{SSR,TEM} \cdot r_{VC,TEM}}{\sqrt{1 - r^2_{SSR,TEM}}\sqrt{1 - r^2_{VC,TEM}}}, \tag{3}$$

where $r_{SSR,VC,TEM}$ is Pearson's correlation coefficient. Similarly, the partial correlation coefficient between VC and TEM and TPR can be obtained. The influence of meteorological elements on VC

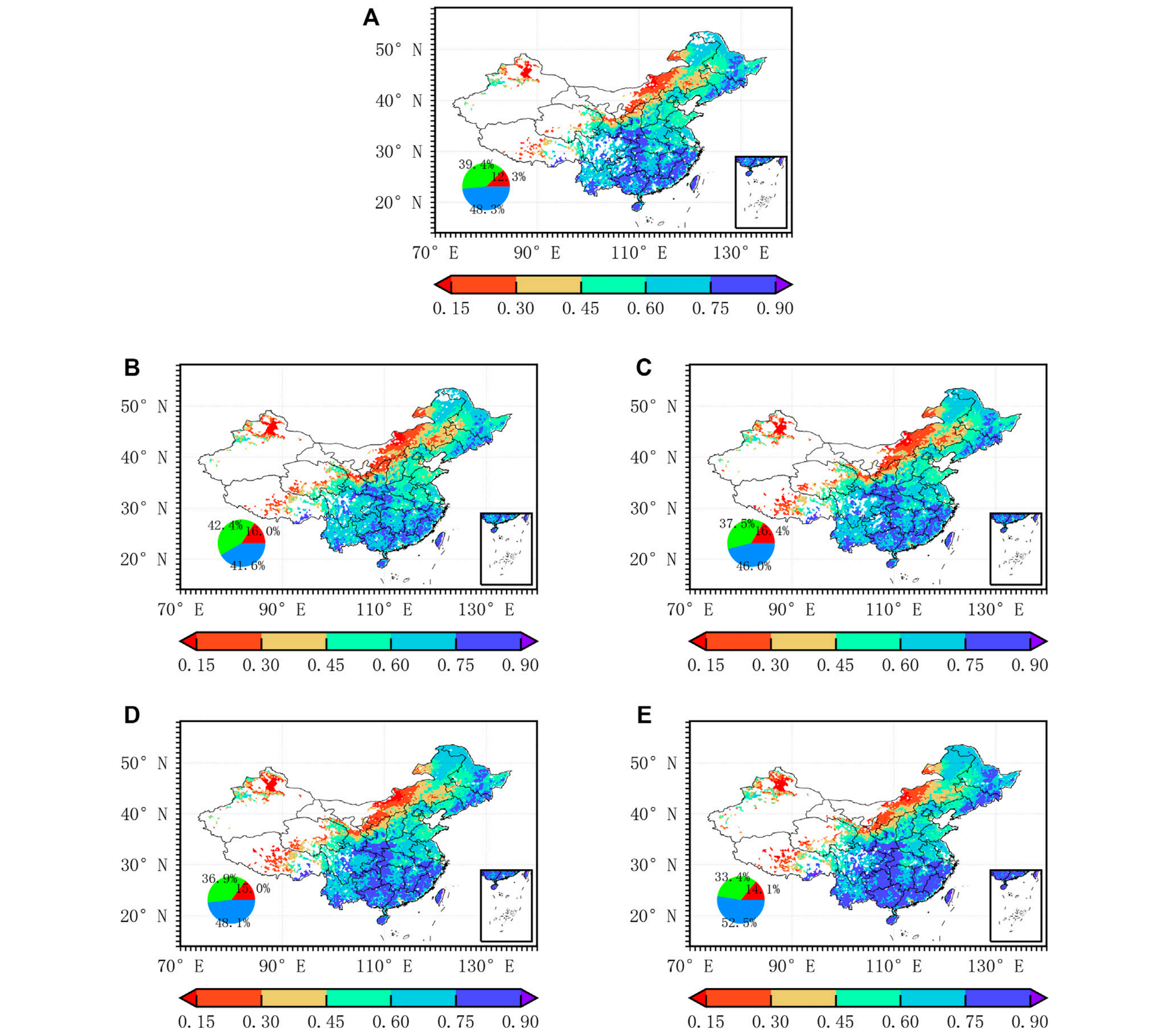


FIGURE 1 | Mean value of vegetation coverage in the growing season for different periods; areas with mean value < 0.05 are blank. **(A)** 2000–2020, **(B)** 2000–2005, **(C)** 2006–2010, **(D)** 2011–2015, and **(E)** 2016–2020. The pie chart shows the proportion of high, medium, and low vegetation coverage, where blue, green, and red represent high vegetation coverage (0.6–1.0), medium vegetation coverage (0.3–0.6), and low vegetation coverage (0.05–0.3), respectively.

in different seasons is determined by partial correlation analysis. Similarly, partial correlation analysis also explains the seasonal delay effect of meteorological elements on VC.

2.2.2 Lasso Regression

To determine the critical meteorological element periods affecting growing season VC change in China, Lasso regression was used to study the relationship between seasonal meteorological elements and VC change in the growing season [39–41]. Lasso is a classical variable selection model which reduces the regression coefficient of insignificant variables to 0, retains only a few significant variables, and vastly reduces the influence of multicollinearity between variables. Lasso regression is suitable for defining the critical climate stages affecting VC change in the growing season in China. The regression coefficient of Lasso output can effectively interpret the influence of meteorological elements in different seasons on VC in the growing season. The mean vegetation coverage in the growing season is used as the index of annual vegetation growth condition; the meteorological elements (TEM, TPR, and SSR) in four seasons (previous year winter, spring, summer, and autumn) are used as independent variables, and the annual vegetation growth condition is used as dependent variables to construct the regression model. In order to eliminate the influence of different variable dimensions on the vegetation model, the data of the input are standardized at the time.

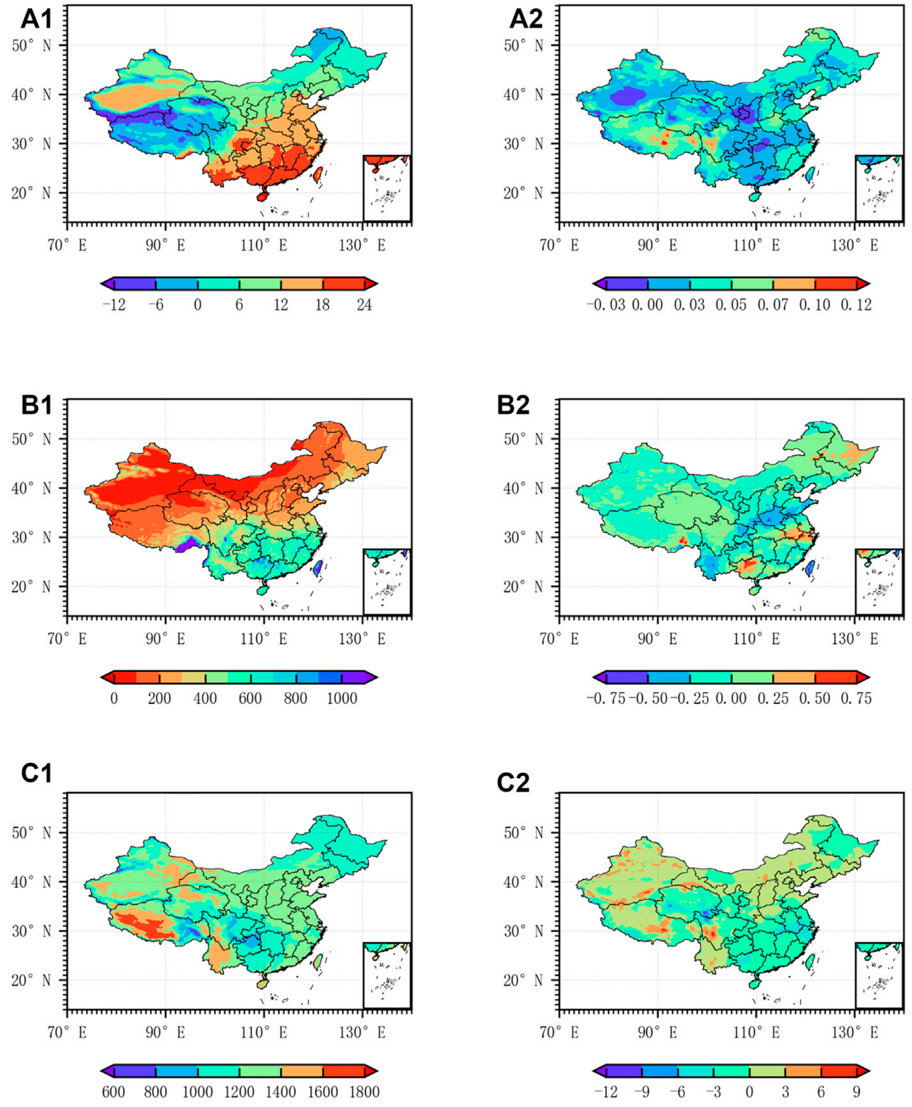


FIGURE 2 | (A1–C1) Mean value of interannual TEM (°C), TPR (mm) and SSR ($\times 10^4 J/m^2$) during 2000–2020 in China. **(A2–C2)** Linear trend of interannual TEM (°C/ Year), TPR (mm / Year), and SSR ($\times 10^4 J/m^2 \times Year$) during 2000–2020 in China.

In addition, to study the changes of meteorological elements affecting vegetation growth in different decades, we divided the data from 2000 to 2020 into two stages: 2000–2010 and 2011–2020.

3 RESULT

3.1 Basic Characteristics of Vegetation and Meteorological Elements

The VC gradually increases from northwest to southeast from 2000 to 2020 (**Figure 1**). Growing season average VC is calculated for the periods 2000–2005, 2006–2010, 2011–2015, and 2016–2020. During 2000–2005, the high vegetation coverage (0.6–1.0) area accounts for 41.6%, the low vegetation coverage (0.05–0.3) area accounts for 16.0%, and the medium vegetation coverage (0.3–0.6) area accounts for about 42.4% in China. The proportion of high vegetation coverage area increased to 52.5% during 2016–2020, and the proportion of low and medium coverage area decreased to 14.1% and 33.4% respectively. The high-coverage areas in China gradually increased, and the medium-coverage and low-coverage areas gradually decreased from 2000 to 2020. Low-coverage areas are mainly distributed in Xinjiang, central and western Inner Mongolia, and Tibet. High-coverage areas are mainly distributed in southern China and the northeast, indicating that vegetation coverage has obvious differences in different regions of China.

The annual average spatial distribution of meteorological elements, TEM and TPR, gradually increase from the northwest to southeast (**Figures 2A1–C1**). The spatial distribution of TEM and TPR is consistent with the VC characteristics; that is, the area with appropriate TEM and

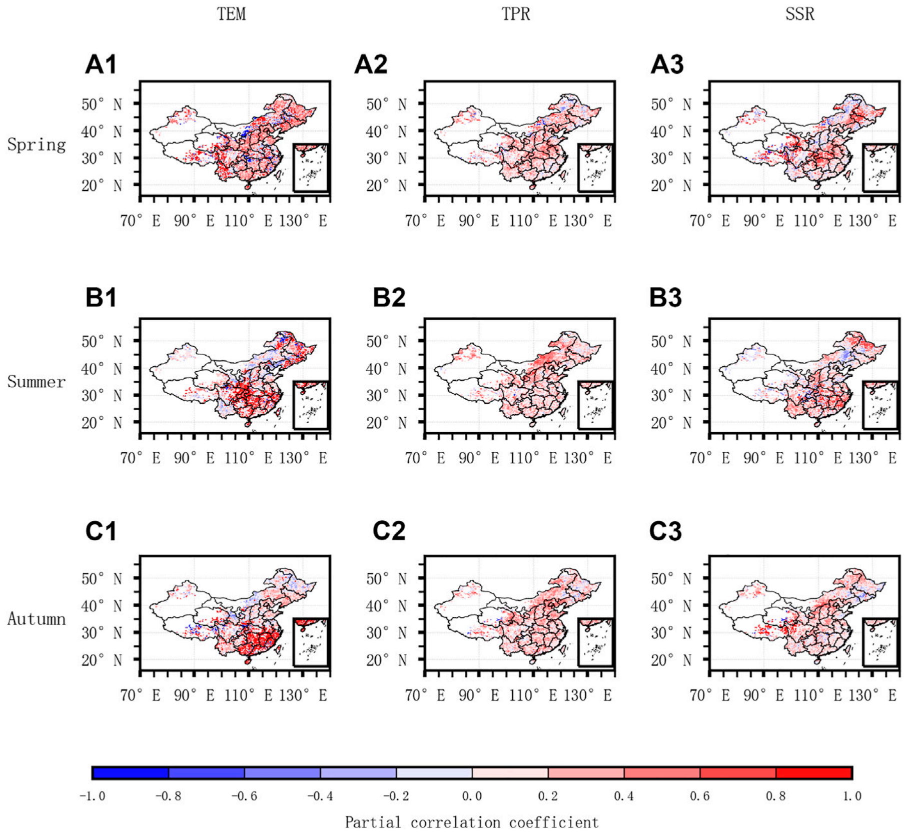


FIGURE 3 | Partial correlation coefficient between vegetation coverage and meteorological elements (TEM, TPR, and SSR) in different seasons (spring, summer, and autumn), the absolute value of the partial correlation coefficient ≥0.455; it passes the 95% significance test.

sufficient TPR has higher vegetation coverage. The annual average TEM in China is about −12–24°C, the low-temperature area is mainly distributed in southern Xinjiang and Tibet, and the high-temperature area is distributed in South China. In northern China, the mean annual TPR is 0–200 mm, which belongs to arid and semi-arid area [42]. In southern China, the mean annual TPR is about 600 mm, which pertains to humid areas. The SSR remains in the range of (1000–1400) × $10^4 J/m^2$ in other areas, except for Xizang and Yunnan during 2000–2020. The TEM shows a warming trend, but the Tarim Basin in Xinjiang tends to get colder during 2000–2020. Except for northeast, Qinghai, and Sichuan, the TPR of other provinces generally shows a downward trend. The SSR increased in the northwest and northeast and decreased in the middle (**Figures 2A2–C2**).

3.2 Change in Meteorological Elements Constrains Vegetation Coverage Between Seasons

Figure 3 and **Table 1** show the partial correlation coefficients between VC change and meteorological elements (TEM, TPR, and SSR) in spring, summer, and autumn. Meteorological elements have obvious temporal and spatial heterogeneity on VC change [43, 44]. During spring, significant correlations between VC change and TEM (TPR and SSR) were observed across 40.29% (33.72% and 28.48%) of the total vegetation regions of China. The significant positive effects of TEM on vegetation are concentrated in northeast, Tibet, and Yunnan. The significant positive effects of TPR on vegetation are mainly distributed in arid and semi-arid areas, such as northwest and North China Plain. The significant positive effects of SSR on vegetation are mainly distributed in Chongqing, Hubei. During summer, the significantly positively affected areas by TEM decreased from 17.02% to 7.16%, especially in southern and northeast China. In arid and semi-arid areas, such as Inner Mongolia and North China Plain, VC change is negatively correlated with TEM and positively correlated with TPR. Warming in summer can enhance the activity of photosynthetic enzymes and postpone the date of frost in autumn. However, in arid and semi-arid areas, TPR increase can promote vegetation growth and warming will aggravate water loss and inhibit vegetation growth [45, 46]. The effect of SSR on VC is basically consistent with TEM. During autumn, significant positive correlations between VC and TEM were observed in southern China; SSR has little effect on VC. No matter which season, VC is sensitive to TPR in northern China, such as Inner Mongolia, Xinjiang, and North China. In the humid area of southern China, vegetation growth is susceptible to TEM. A similar result was also obtained in climate–vegetation studies [23].

Early climate change will affect the current vegetation growth, so meteorological factors have delayed effect on vegetation

TABLE 1 | Proportion of vegetated areas with vegetation coverage significantly related to meteorological elements in China.

Meteorological Elements	Spring		Summer		Autumn	
	Positive	Negative	Positive	Negative	Positive	Negative
TEM	17.02	23.27	7.16	7.53	14.40	13.86
TPR	15.09	18.63	7.29	2.21	10.45	12.66
SSR	12.66	14.82	6.52	3.78	7.36	5.71

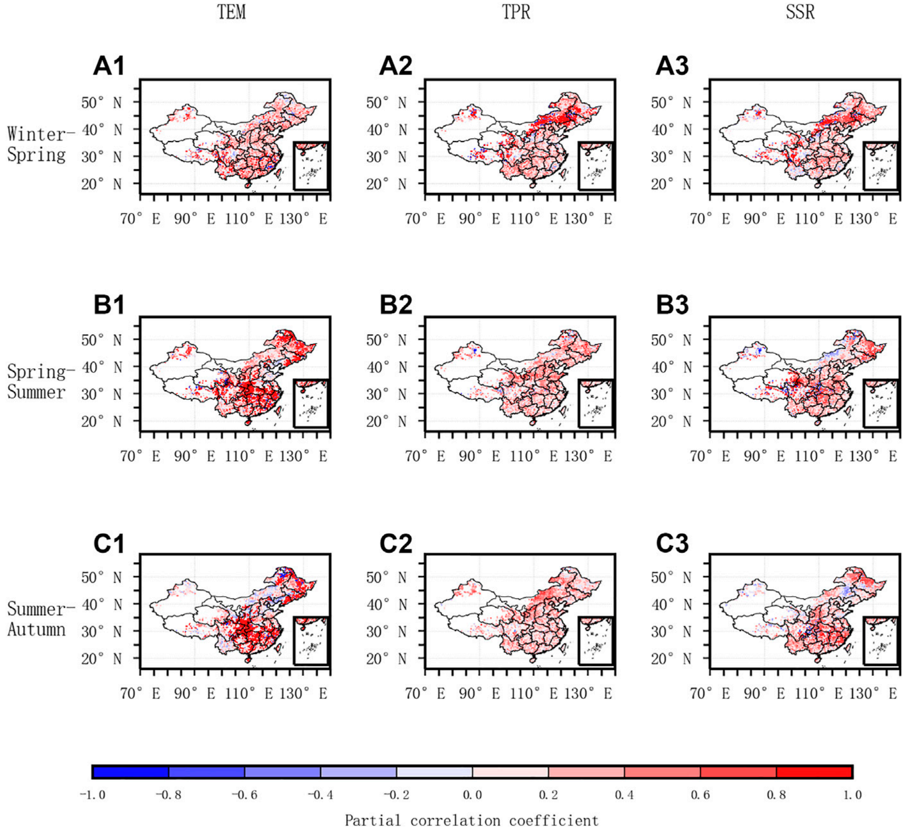


FIGURE 4 | Partial correlation coefficient between vegetation coverage and meteorological elements (TEM, TPR, and SSR) of the previous seasons, the absolute value of the partial correlation coefficient ≥0.455; it passes the 95% significance test.

growth. The partial correlations between VC and previous-season TEM, TPR, and SSR are provided in **Figure 4**. The strong relationship between spring VC and previous-year winter TPR and SSR was observed in Inner Mongolia. A strong positive relationship between spring TEM and summer VC change was observed in northeast and southern China. In North China, summer VC change is negatively correlated with spring TEM and positively correlated with spring TPR; spring SSR has little effect on summer VC. The autumn VC change in central and southern China is positively correlated with summer TEM and negatively correlated in other regions; summer TPR has a significant positive correlation with autumn VC. The correlation between summer SSR and TEM on vegetation change is basically the same. Suggesting that the meteorological elements have a strong seasonal effect on the VC change, significant correlations between spring VC change and previous-winter TEM (TPR and SSR) were observed across 19.62% (25.06% and 23.28%) of the total vegetation regions of China (**Table 2**). It shows that the seasonal effect of previous-winter TPR and SSR on spring VC change is stronger than that of TEM. However, the influence of TEM and SSR in spring and summer on VC change in summer and autumn is stronger than that of TPR.

3.3 Drivers of Growing Season Vegetation Cover Change

Meteorological element change is thought to affect growing season VC change, including TEM, TPR, and SSR in four seasons (previous winter, spring, summer, and autumn). We study the key drivers of vegetation coverage change in different regions of China and efforts to quantitatively

TABLE 2 | Proportion of vegetation areas with vegetation coverage significantly related to previous-season meteorological elements in China.

Meteorological Elements	Winter-Spring		Spring-Summer		Summer-Autumn	
	Positive	Negative	Positive	Negative	Positive	Negative
TEM	7.72	11.9	19.8	17.85	15.09	18.63
TPR	12.25	12.81	7.98	5.98	7.29	2.21
SSR	11.37	11.91	12.41	16.04	10.45	12.66

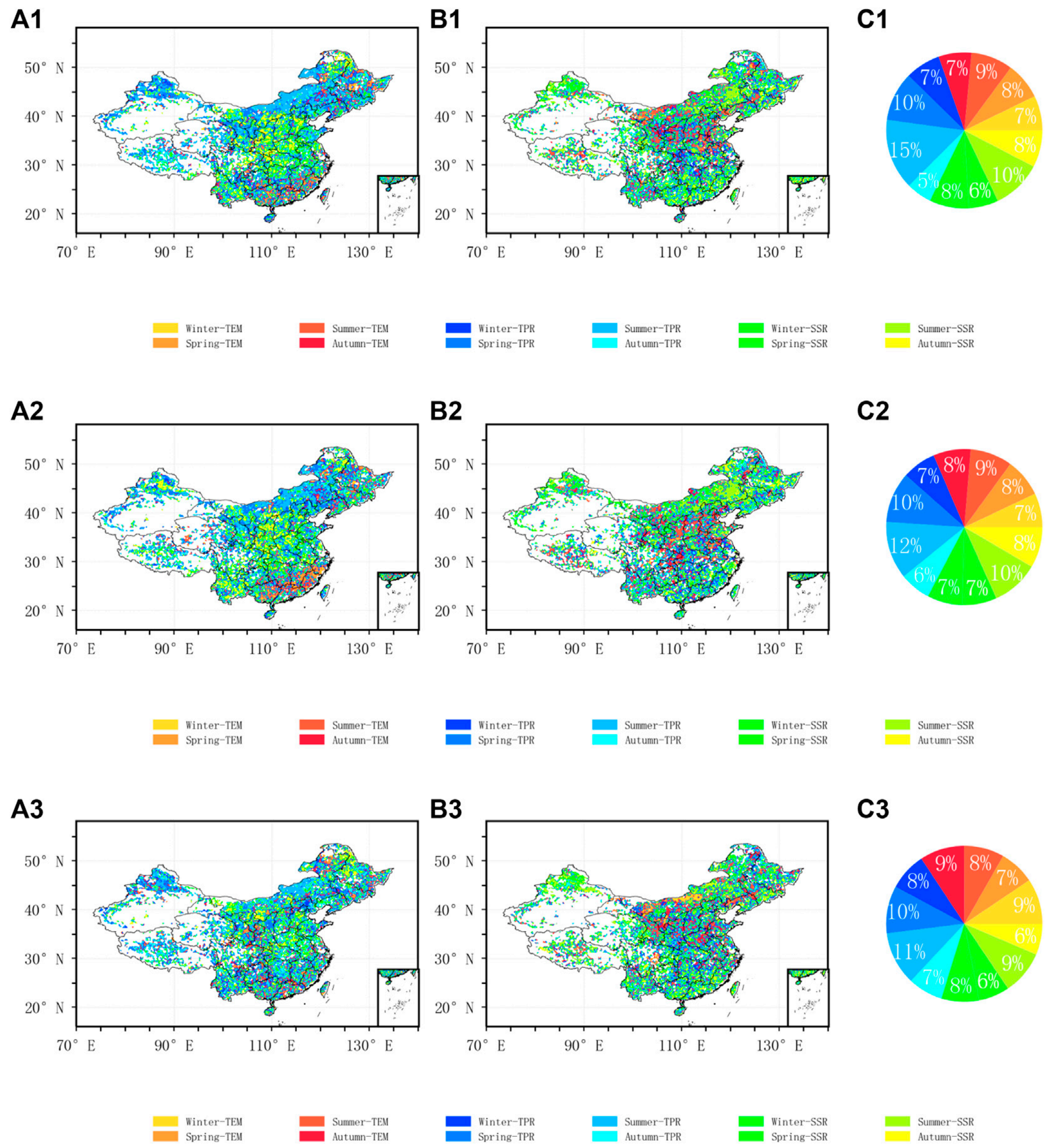


FIGURE 5 | Regression coefficient: **(A1–A3)** Positive effect of meteorological elements on vegetation coverage change. **(B1–B3)** Negative effect of meteorological elements on vegetation coverage change. **(C1–C3)** Proportion of vegetation area in China governed by meteorological elements in different seasons, where **(A1–C1)** indicate 2000–2020, **(A2–C2)** indicate 2000s, and **(A3–C3)** indicate 2010s.

analyze the contribution of each variable to vegetation coverage increase in a different decade. Precipitation-driven VC increase is most evident in northwest China and Inner Mongolia. The regions belong to arid and semi-arid areas and lack water. Therefore, the increase of precipitation can promote VC increase. SSR is a necessary condition for photosynthesis, and the increase of solar radiation in an appropriate range is conducive to vegetation growth. As the SSR in North China and central China is low, the increase of solar radiation is conducive to vegetation growth. In Yunnan, the SSR positively impacted VC change in the 2000s, but the impact of SSR gradually decreases and the impact of precipitation gradually increases in the 2010s. In the coastal area of South China, it is mainly affected by temperature in the 2000s, but the influence of temperature on it gradually weakens and precipitation gradually increases in the 2010s. From the 2000s to 2010s, the impact of precipitation on the change of vegetation coverage in China gradually increased. The effects of radiation and temperature gradually weakened. It is also found that the vital meteorological factors promoting vegetation growth are different in different decade periods. Among all meteorological elements, the driving factors of vegetation change in the growing season in China are mainly precipitation in spring and summer, followed by radiation in spring. In 2000s, 35% of Chinese vegetation coverage was dominated by precipitation, mainly concentrated in northern China. From the 2000s to 2010s, the area dominated by temperature increased from 32% to 36%, but the radiation decreased from 32% to 29% (**Figure 5**). There are apparent inter-decadal changes in temperature and radiation.

4 DISCUSSION AND CONCLUSION

This article analyzes the relationship between meteorological elements and vegetation on a seasonal scale in China. The influence of meteorological elements on growing season vegetation change has obvious seasonal shift and inter-decadal difference. The conclusions are as follows:

In China, the high-coverage areas gradually increased and the medium- and low-coverage areas gradually decreased from 2000–2005 to 2016–2020. Low-coverage areas are mainly distributed in Xinjiang and Inner Mongolia as well as Tibet, and high-coverage areas are in southern and northeast China; TEM and TPR gradually increase from the northwest to southeast (**Figures 2A1–C1**). The spatial distribution of TEM and TPR is basically consistent with the VC characteristics.

The results show that the key meteorological factors affecting vegetation change are different in different regions and seasons. In arid and semi-arid area, especially Inner Mongolia and Xinjiang [47, 48], the VC change mainly depends on TEM, whereas the relationship between VC change and TPR and SSR is weak during spring. There is a strong positive correlation between VC change and TPR during summer and autumn, but the relationship between VC change and TEM is negatively correlated (**Figure 3**). TPR increase can promote vegetation growth and warming will aggravate water loss and inhibit vegetation growth [45, 46]. These findings are consistent with the response of vegetation to climate in arid and semi-arid areas [3, 16, 17]. The vegetation growth in spring mainly depends on the previous-winter TPR and SSR, and the vegetation growth in autumn depends on the summer TPR (**Figure 4**). In addition, our study also shows that the increase of TPR in spring and summer is conducive to the increase of annual vegetation coverage (**Figures 5C1–C3**). In northeast China, spring TEM and summer VC change has a strong relationship. In the North China Plain, the VC change during spring is significantly positively correlated with TEM and TPR. During summer and autumn, it is significantly positively correlated with TPR, but the correlation with TEM is weak. Vegetation growth in summer mainly depends on spring TPR; spring SSR has little effect on summer VC. This is consistent with previous studies [3, 22, 49]. The increase of autumn SSR and winter and spring TPR is conducive to the increase of VC. In humid areas, such as South China and East China, the VC change mainly depends on the change of TEM in every season, and there is a statistically significant correlation between VC change and SSR (**Figure 3**). These findings are consistent with those of previous studies [3, 23]. TEM has a strong seasonal delay effect on VC change, but TPR is relatively weak (**Figure 4**).

Through the research, we can know the critical meteorological elements affecting vegetation change in different regions of China. Critical meteorological element modeling can more accurately predict future vegetation coverage. In the course of this study, the scale of vegetation and climate data does not match, and interpolation may cause errors in the data set. On the other hand, the study ignored the impact of human activities, such as ecological engineering and crop irrigation.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Material; further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

HB is responsible for data processing, drawing, and article writing. GS, LL, and YW are responsible for raising scientific issues. ZG and GF provided data.

FUNDING

The project is funded by the National Key Research and Development Program of China (Grant no. 2018YFE0109600), the National Natural Science Foundation of China under Grant no. 42 075 029, the Program for the Outstanding Innovative Teams (OITs) of Higher Learning Institutions of Shanxi, and the Outstanding Young Talents Support Plan of Shanxi province.

REFERENCES

1. Wu J, Hobbs R. Key Issues and Research Priorities in Landscape Ecology: An Idiosyncratic Synthesis. *Landscape Ecol* (2002) 17:355–65. doi:10.1023/A:1020561630963
2. Wang F, Wang X, Zhao Y, Yang Z. Temporal Variations of Ndvi and Correlations between Ndvi and Hydro-Climatological Variables at lake Baiyangdian, china. *Int J Biometeorol* (2014) 58:1531–43. doi:10.1007/s00484-013-0758-4
3. Qu B, Zhu W, Jia S, Lv A. Spatio-temporal Changes in Vegetation Activity and its Driving Factors during the Growing Season in china from 1982 to 2011. *Remote Sensing* (2015) 7:13729–52. doi:10.3390/rs71013729
4. Wang W, Anderson BT, Phillips N, Kaufmann RK, Potter C, Myneni RB. Feedbacks of Vegetation on Summertime Climate Variability over the north American Grasslands. Part I: Statistical Analysis. *Earth Interactions* (2006) 10: 1–27. doi:10.1175/EI196.1
5. Wang X, Piao S, Ciais P, Li J, Friedlingstein P, Koven C, et al. Spring Temperature Change and its Implication in the Change of Vegetation Growth in north america from 1982 to 2006. *Proc Natl Acad Sci* (2011) 108:1240–5. doi:10.1073/pnas.1014425108
6. Bonan GB, Pollard D, Thompson SL. Effects of Boreal forest Vegetation on Global Climate. *Nature* (1992) 359:716–8. doi:10.1038/359716a0
7. Piao S, Wang X, Park T, Chen C, Lian X, He Y, et al. Characteristics, Drivers and Feedbacks of Global Greening. *Nat Rev Earth Environ* (2020) 1:14–27. doi:10.1038/s43017-019-0001-x
8. Dusenge ME, Duarte AG, Way DA. Plant Carbon Metabolism and Climate Change: Elevated Co2 and Temperature Impacts on Photosynthesis, Photorespiration and Respiration. *New Phytol* (2019) 221:32–49. doi:10.1111/nph.15283
9. Smith NG, Dukes JS. Plant Respiration and Photosynthesis in Global-Scale Models: Incorporating Acclimation to Temperature and Co2. *Glob Change Biol* (2013) 19:45–63. doi:10.1111/j.1365-2486.2012.02797.x
10. Monje O, Bugbee B. Radiometric Method for Determining Canopy Stomatal Conductance in Controlled Environments. *Agronomy* (2019) 9:114. doi:10.3390/agronomy9030114
11. Yu Q, Xie X, Sun S. Andances in Simulation of Plant Photosynthetic Productivity and Canopy Evapotranspiration. *Acta Ecologica sinica (in Chinese)* (1999) 19:10.
12. Ma X, Song L, Yu W, Hu Y, Liu Y, Wu J, et al. Growth, Physiological, and Biochemical Responses of Camptotheca Acuminata Seedlings to Different Light Environments. *Front Plant Sci* (2015) 6:321. doi:10.3389/fpls.2015.00321
13. Si J, Chang Z, Su Y, Xi H, Feng Q. Stomatal Conductance Characteristics of Populus Euphrat Ica Leaves and Response to Environmental Factors in the Extreme Arid Region. *Acta Botanica Boreali-Occidentalia Sinica* (2008) 28: 125–30. doi:10.3321/j.issn:1000-4025.2008.01.023
14. von Caemmerer S, Farquhar GD. Some Relationships between the Biochemistry of Photosynthesis and the Gas Exchange of Leaves. *Planta* (1981) 153:376–87. doi:10.1007/BF00384257
15. Buckley TN, Mott KA. Modelling Stomatal Conductance in Response to Environmental Factors. *Plant Cel Environ* (2013) 36:1691–9. doi:10.1111/pce.12140
16. Nemani RR, Keeling CD, Hashimoto H, Jolly WM, Piper SC, Tucker CJ, et al. Climate-driven Increases in Global Terrestrial Net Primary Production from 1982 to 1999. *science* (2003) 300:1560–3. doi:10.1126/science.1082750
17. Zhao L, Dai A, Dong B. Changes in Global Vegetation Activity and its Driving Factors during 1982-2013. *Agric For Meteorology* (2018) 249:198–209. doi:10.1016/j.agrformet.2017.11.013
18. Hou M, Zhao H, Wang Z. Vegetation Responses to Climate Change by Using the Satellite-Derive Normalized Difference Vegetation index: A Review. *Climatic Environ Res (in Chinese)* (2013) 18:353–64. doi:10.3878/j.issn.1006-9585.2012.11137
19. Sun G-Q, Zhang H-T, Wang J-S, Li J, Wang Y, Li L, et al. Mathematical Modeling and Mechanisms of Pattern Formation in Ecological Systems: a Review. *Nonlinear Dyn* (2021) 104:1677–96. doi:10.1007/s11071-021-06314-5
20. Xue Q, Liu C, Li L, Sun G-Q, Wang Z. Interactions of Diffusion and Nonlocal Delay Give Rise to Vegetation Patterns in Semi-arid Environments. *Appl Maths Comput* (2021) 399:126038. doi:10.1016/j.amc.2021.126038
21. Piao S, Mohammat A, Fang J, Cai Q, Feng J. Ndvi-based Increase in Growth of Temperate Grasslands and its Responses to Climate Changes in china. *Glob Environ Change* (2006) 16:340–8. doi:10.1016/j.gloenvcha.2006.02.002
22. Duo A, Zhao W, Qu X, Jing R, Xiong K. Spatio-temporal Variation of Vegetation Coverage and its Response to Climate Change in north china plain in the Last 33 Years. *Int J Appl Earth Observation Geoinformation* (2016) 53:103–17. doi:10.1016/j.jag.2016.08.008
23. Gao J, Jiao K, Wu S, Ma D, Zhao D, Yin Y, et al. Past and Future Influence of Climate Change on Spatially Heterogeneous Vegetation Activity in china. *Earth Syst Dyn Discuss* (2017) 2017:1–20. doi:10.5194/esd-2017-13
24. Piao S, Fang J, Zhou L, Guo Q, Henderson M, Ji W, et al. Interannual Variations of Monthly and Seasonal Normalized Difference Vegetation index (Ndvi) in china from 1982 to 1999. *J Geophys Res* (2003) 108:4401. doi:10.1029/2002JD002848
25. Zhou L, Kaufmann RK, Tian Y, Myneni RB, Tucker CJ. Relation between Interannual Variations in Satellite Measures of Northern forest Greenness and Climate between 1982 and 1999. *J Geophys Res* (2003) 108:4004. ACL 3–1–ACL 3–16. doi:10.1029/2002JD002510
26. Chen C, He B, Guo L, Zhang Y, Xie X, Chen Z. Identifying Critical Climate Periods for Vegetation Growth in the Northern Hemisphere. *J Geophys Res Biogeosci* (2018) 123:2541–52. doi:10.1029/2018JG004443
27. Zhou C, Shi R, Zhang C, Liu C, Gao W. Spatio-temporal Distribution of Ndvi and its Correlation with Climatic Factors in Eastern china during 1998-2008. In: W Gao, NB Chang, J Wang, editors. Remote Sensing And Modeling Of Ecosystems For Sustainability XI *(SPIE)*. San Diego, CA: SPIE Optical Engineering + Applications (2014). p. 96–105. doi:10.1117/12.2060768
28. Davis MB. Lags in Vegetation Response to Greenhouse Warming. *Climatic Change* (1989) 15:75–82. doi:10.1007/BF0013884610.1007/bf00138846
29. Wu D, Zhao X, Liang S, Zhou T, Huang K, Tang B, et al. Time-lag Effects of Global Vegetation Responses to Climate Change. *Glob Change Biol* (2015) 21: 3520–31. doi:10.1111/gcb.12945
30. Kuzyakov Y, Gavrichkova O. Review: Time Lag between Photosynthesis and Carbon Dioxide Efflux from Soil: a Review of Mechanisms and Controls. *Glob Change Biol* (2010) 16:3386–406. doi:10.1111/j.1365-2486.2010.02179.x
31. Saatchi S, Asefi-Najafabady S, Malhi Y, Aragao LEOC, Anderson LO, Myneni RB, et al. Persistent Effects of a Severe Drought on Amazonian forest Canopy. *Proc Natl Acad Sci* (2013) 110:565–70. doi:10.1073/pnas.1204651110
32. Gong Z, Zhao S, Gu J. Correlation Analysis between Vegetation Coverage and Climate Drought Conditions in north china during 2001-2013. *J Geogr Sci* (2017) 27:143–60. doi:10.1007/s11442-017-1369-5
33. He B, Chen A, Jiang W, Chen Z. The Response of Vegetation Growth to Shifts in Trend of Temperature in china. *J Geogr Sci* (2017) 27:801–16. doi:10.1007/s11442-017-1407-3
34. Peng S, Chen A, Xu L, Cao C, Fang J, Myneni RB, et al. Recent Change of Vegetation Growth Trend in china. *Environ Res Lett* (2011) 6:044027. doi:10.1088/1748-9326/6/4/044027
35. Piao S, Cui M, Chen A, Wang X, Ciais P, Liu J, et al. Altitude and Temperature Dependence of Change in the spring Vegetation green-up Date from 1982 to 2006 in the Qinghai-Xizang Plateau. *Agric For Meteorology* (2011) 151: 1599–608. doi:10.1016/j.agrformet.2011.06.016
36. Tucker CJ, Pinzon JE, Brown ME, Slayback DA, Pak EW, Mahoney R, et al. An Extended AVHRR 8-km NDVI Dataset Compatible with MODIS and SPOT Vegetation NDVI Data. *Int J Remote Sensing* (2005) 26:4485–98. doi:10.1080/01431160500168686
37. Zhou L, Tucker CJ, Kaufmann RK, Slayback D, Shabanov NV, Myneni RB. Variations in Northern Vegetation Activity Inferred from Satellite Data of Vegetation index during 1981 to 1999. *J Geophys Res* (2001) 106:20069–83. doi:10.1029/2000JD000115
38. Chen C, He B, Yuan W, Guo L, Zhang Y. Increasing Interannual Variability of Global Vegetation Greenness. *Environ Res Lett* (2019) 14:124005. doi:10.1088/1748-9326/ab4ffc
39. Tibshirani R. Regression Shrinkage and Selection via the Lasso. *J R Stat Soc Ser B (Methodological)* (1996) 58:267–88. doi:10.1111/j.2517-6161.1996.tb02080.x
40. Adak A, Murray SC, Božinović S, Lindsey R, Nakasagga S, Chatterjee S, et al. Temporal Vegetation Indices and Plant Height from Remotely Sensed Imagery Can Predict Grain Yield and Flowering Time Breeding Value in maize via Machine Learning Regression. *Remote Sensing* (2021) 13:2141. doi:10.3390/rs13112141

41. Wang S, Liu Q, Huang C. Vegetation Change and its Response to Climate Extremes in the Arid Region of Northwest china. *Remote Sensing* (2021) 13: 1230. doi:10.3390/rs13071230
42. Huang J, Ma J, Guan X, Li Y, He Y. Progress in Semi-arid Climate Change Studies in china. *Adv Atmos Sci* (2019) 36:922–37. doi:10.1007/s00376-018-8200-9
43. Jiao K-W, Gao J-B, Liu Z-H, Wu S-H, Fletcher TL. Revealing Climatic Impacts on the Temporal and Spatial Variation in Vegetation Activity across china: Sensitivity and Contribution. *Adv Clim Change Res* (2021) 12:409–20. doi:10.1016/j.accre.2021.04.006
44. Liu H, Zhang M, Lin Z, Xu X. Spatial Heterogeneity of the Relationship between Vegetation Dynamics and Climate Change and Their Driving Forces at Multiple Time Scales in Southwest china. *Agric For Meteorology* (2018) 256-257:10–21. doi:10.1016/j.agrformet.2018.02.015
45. Shen M, Piao S, Jeong S-J, Zhou L, Zeng Z, Ciais P, et al. Evaporative Cooling over the Tibetan Plateau Induced by Vegetation Growth. *Proc Natl Acad Sci USA* (2015) 112:9299–304. doi:10.1073/pnas.1504418112
46. Fracheboud Y, Luquez V, Björke´n L, Sjö̈din A, Tuominen H, Jansson S. The Control of Autumn Senescence in European Aspen. *Plant Physiol* (2009) 149: 1982–91. doi:10.1104/pp.108.133249
47. Zhuang Q, Wu S, Feng X, Niu Y. Analysis and Prediction of Vegetation Dynamics under the Background of Climate Change in Xinjiang, china. *PeerJ* (2020) 8:e8282. doi:10.7717/peerj.8282
48. Chuai XW, Huang XJ, Wang WJ, Bao G. NDVI, Temperature and Precipitation Changes and Their Relationships with Different Vegetation Types during 1998-2007 in Inner Mongolia, China. *Int J Climatol* (2013) 33:1696–706. doi:10.1002/joc.3543
49. Bai H, Gong Z, Sun G, Li L, Zhou L. Influence of Meteorological Elements on Summer Vegetation Coverage in north china. *Chin J Atmos Sci (in Chinese)* (2022) 46:1–13. doi:10.3878/j.issn.1006-9895.2102.20233

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

PERSPECTIVE
published: 24 January 2022
doi: 10.3389/fphy.2022.835992

RiskEstim: A Software Package to Quantify COVID-19 Importation Risk

Mingda Xu[1,2,3†], *Zhanwei Du*[1,2†], *Songwei Shan*[1,2], *Xiaoke Xu*[3], *Yuan Bai*[1,2], *Peng Wu*[1,2], *Eric H. Y. Lau*[1,2] *and Benjamin J. Cowling*[1,2]*

[1]*WHO Collaborating Centre for Infectious Disease Epidemiology and Control, School of Public Health, Li Ka Shing Faculty of Medicine, The University of Hong Kong, Hong Kong, Hong Kong SAR, China,* [2]*Laboratory of Data Discovery for Health, Hong Kong Science and Technology Park, Hong Kong, Hong Kong SAR, China,* [3]*College of Information and Communication Engineering, Dalian Minzu University, Dalian, China*

We present an R package developed to quantify coronavirus disease 2019 (COVID-19) importation risk. Quantifying and visualizing the importation risk of COVID-19 from inbound travelers is urgent and imperative to trigger public health responses, especially in the early stages of the COVID-19 pandemic and emergence of new SARS-CoV-2 variants. We provide a general modeling framework to estimate COVID-19 importation risk using estimated pre-symptomatic prevalence of infection and air traffic data from the multi-origin places. We use Hong Kong as a case study to illustrate how our modeling framework can estimate the COVID-19 importation risk into Hong Kong from cities in Mainland China in real time. This R package can be used as a complementary component of the pandemic surveillance system to monitor spread in the next pandemic.

Keywords: COVID-19, SARS-CoV-2, importation risk, R package, introduction risk

OPEN ACCESS

Edited by:
Zhen Wang,
Northwestern Polytechnical University, China

Reviewed by:
Wenzhuo Song,
Jilin University, China
Xi Wang,
The Chinese University of Hong Kong, Hong Kong SAR, China
Lu Zhong,
Rensselaer Polytechnic Institute, United States

***Correspondence:**
Benjamin J. Cowling
bcowling@hku.hk

†*These authors have contributed equally to this work and share first authorship*

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *15 December 2021*
Accepted: *04 January 2022*
Published: *24 January 2022*

Citation:
Xu M, Du Z, Shan S, Xu X, Bai Y, Wu P, Lau EHY and Cowling BJ (2022) RiskEstim: A Software Package to Quantify COVID-19 Importation Risk. Front. Phys. 10:835992. doi: 10.3389/fphy.2022.835992

INTRODUCTION

The ongoing global pandemic of COVID-19 caused by severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) has caused incredible global disruption and challenges, in addition to the substantial health impact [1]. As of December 12, 2021, more than 269 million confirmed cases and 5.3 million deaths were reported worldwide [2].

Local outbreaks were often associated with the importation of infections. Quantifying and visualizing the importation risk of COVID-19 from inbound travelers is important for public health responses, especially in the early stage of an epidemic wave [3, 4]. For example, some studies showed that border control measures, such as flight restrictions and quarantine for inbound travelers from high-risk places (e.g., based on the number of new daily cases [5]), might have delayed epidemics in the destination countries [6–8]. In addition, assessment of the COVID-19 importation risk is needed for places where a high level of population immunity to COVID-19 has not been achieved in the target populations [9] or the government is considering relaxing border control measures [10–12]. Here, we present the R package *RiskEstim*, the latest codebase version developed to quantify COVID-19 importation risk. First, we outline the general modeling framework of the R package to estimate COVID-19 importation risk using daily pre-symptomatic prevalence data from multi-origin locations and air traffic data.

Hong Kong started the alert against COVID-19 and screening of the travellers from mainland China at the very early beginning of the pandemic [13]. Due to the sound public health infrastructure and the in-time response, the information for the reported cases in Hong Kong was highly reliable, and most of the imported cases at that time originated from mainland China [14]. From the above considerations, we used Hong Kong as a case study to illustrate how our modeling framework can estimate the multi-origin COVID-19 importation risk in real time.

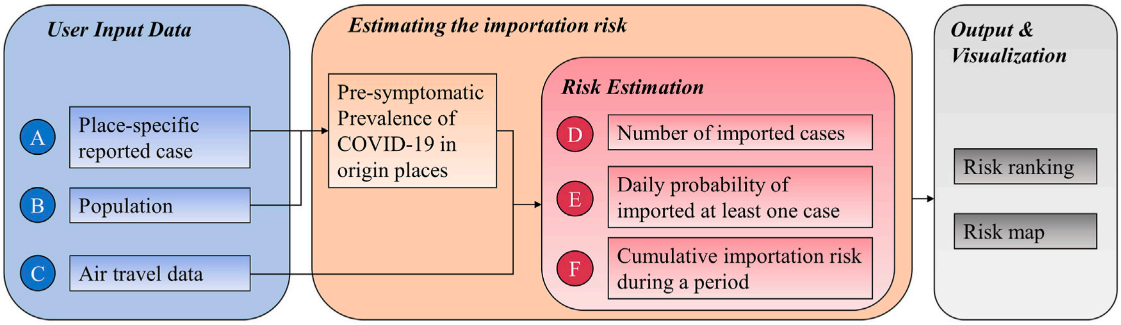

User Input Data
A
Place-specific reported case
B
Population
C
Air travel data
Estimating the importation risk
Pre-symptomatic Prevalence of COVID-19 in origin places
Risk Estimation
D
Number of imported cases
E
Daily probability of imported at least one case
F
Cumulative importation risk during a period
Output & Visualization
Risk ranking
Risk map

FIGURE 1 | An illustration of the proposed framework to estimate the importation risk. This modeling framework includes three main modules [1]: the module of user input data is used to store data submitted by the users, such as daily reported cases and air travel flow [2]; the module of estimating the importation risk is used to estimate the importation risk of the target place based on the input data [3]; the visualization module is used to visualize output, such as the risk maps of the origin places, which could bring the importation risk to the destination.

METHODS

The Modeling Framework of Estimating Importation Risk in the RiskEstim

To quantify the importation risk of COVID-19 from the place of origin to the destination, we first estimated the daily pre-symptomatic prevalence of the COVID-19 in each origin place. Then we calculated the number of potential imported cases using the estimated daily pre-symptomatic prevalence of origin places and the daily origin-destination air traffic data. Next, we estimated the probability of importing at least one case as the indicator of importation risk to rank the origin places and visualize the risk maps. The modeling framework is shown in **Figure 1**.

User Input Data in the RiskEstim

Using Hong Kong as an example, we applied the R package to estimate the importation risk from 15 high-risk cities in Mainland China into Hong Kong in early 2020 [15]. Daily confirmed COVID-19 cases reported by the Chinese Center for Disease Control and Prevention (China CDC) from January 1, 2020, to February 29, 2020 were obtained for the analysis, [16–18]. Because Hubei Province changed the definition of cases on February 12, 2020, which yielded a dramatic increase in the number of cases on February 12, 2020 and February 13, 2020 (14840 on February 12, 2020 and 4823 on February 13, 2020) [19]. To reduce the reporting bias due to different case definitions for COVID-19 during the study period [20], we assumed the number of reported cases in Wuhan on February 12 was the same as those on February 11, and that for February 13 were the same as that on February 14.

Estimating Pre-Symptomatic Prevalence of COVID-19 in Origin Places

The daily prevalence of pre-symptomatic infections could be estimated with the Package based on the daily reported cases in the origin place(s) input by the user. Let ω_t^o be the number of reported cases in the origin place O on day t. Then on average the cases reported on day t developed symptoms on day $t - T_{rep}$ and were infected on day $t - T_{rep} - T_{inc}$, where T_{rep} and T_{inc} are the mean reporting delay and the median incubation period in days. Using this forward method, we estimated the daily numbers of infected individuals in the origin places.

In our case study of Hong Kong, we calculated the daily prevalence of pre-symptomatic COVID-19 in multiple origin places including cities from Hubei province and other provinces, and the estimates were consistent with the imported cases from these places during the early stage of the epidemic in Hong Kong [14]. Let y_t^o denote the place-specific pre-symptomatic prevalence of the place O on the day t, and H denote the cities in Hubei province. We used the median incubation period to denote the period where transmission would occur from infected cases. The place-specific pre-symptomatic prevalence is given by:

$$y_t^o = \begin{cases} \mu \Sigma_{d=t-T_{inc}+1}^{t} I_d^o, o \epsilon H \\ \Sigma_{d=t-T_{inc}+1}^{t} I_d^o, \mathit{otherwise} \end{cases}$$

where μ is the ascertainment rate ratio, representing the ascertainment rate of symptomatic cases in all non-Hubei provinces relative to Hubei province, which reflects the probability ratio of non-Hubei Provinces reporting a symptomatic case to Hubei Province [18]. I_d^o denotes the incidence of SARS-CoV-2 infection in an origin place on day d. The parameters are summarized in **Table 1**.

Estimating the Importation Risk

The place-specific importation risk of the destination was estimated based on [1]: daily pre-symptomatic prevalence of COVID-19 in origin places [2]; data on air passenger movements by place of origin and destination. Let $\Gamma_t^{o,d}$ be the imported cases from the origin place O to the destination d on day t:

$$\Gamma_t^{o,d} = \alpha * y_t^o * M_t^{o,d}$$

TABLE 1 | Model parameters in the modeling framework.

Parameters	Description	Values	References
T_{rep}	Mean reporting delay	7 days	[21]
T_{inc}	Median incubation period	5 days	[22]
μ	Ascertainment rate ratio of symptomatic cases in all non-Hubei provinces relative to that of Hubei province	5.14	[18]
α	Scaling factor that relates the importation to scaled reported cases in high-surveillance places	5.05	[18]

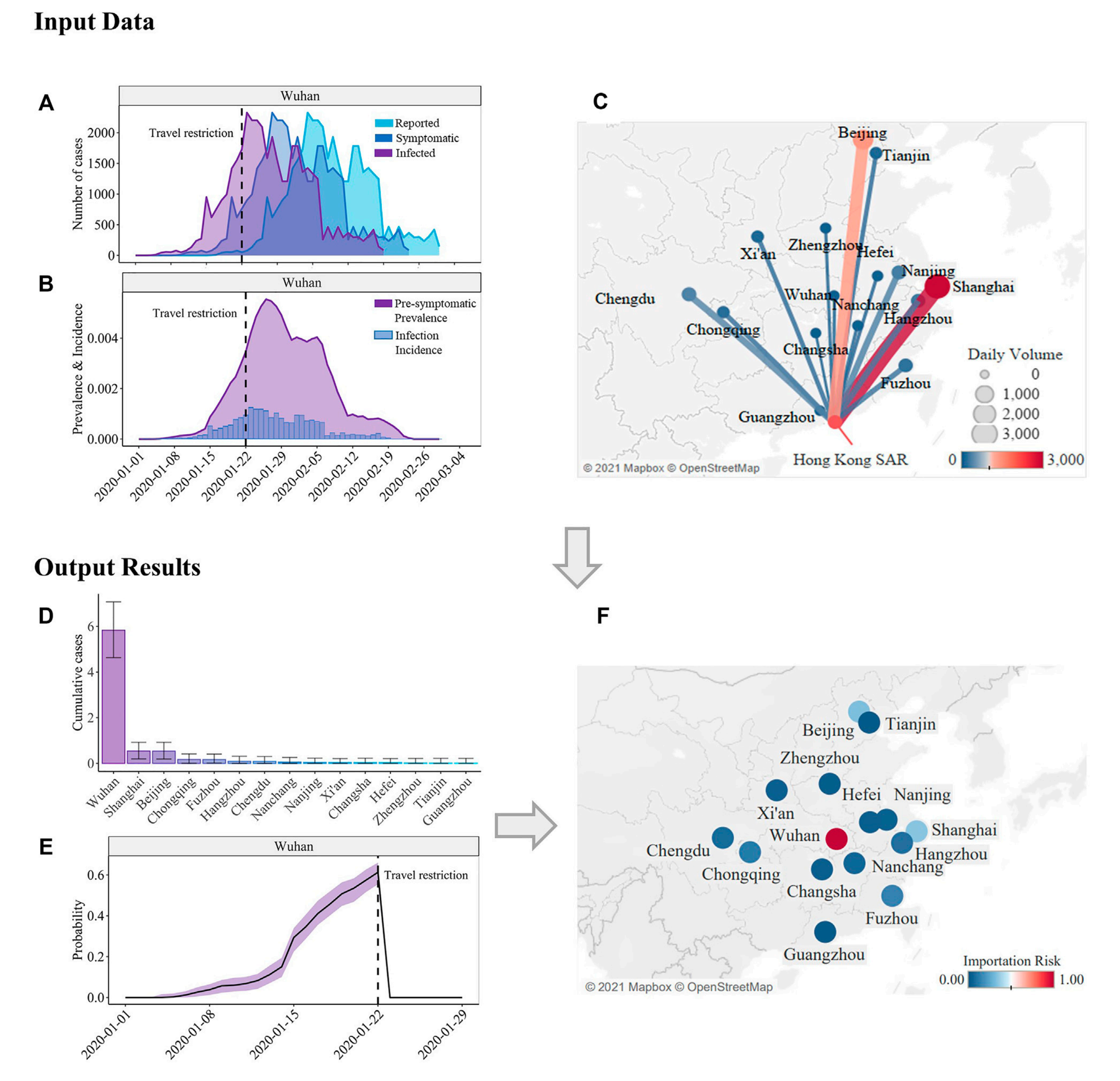


FIGURE 2 | The results of estimating the importation risk in the case study of Hong Kong. **(A)** Daily reported cases, estimated daily symptomatic cases based on daily reported cases, and estimated daily infected cases in Wuhan. **(B)** Daily pre-symptomatic prevalence and infection incidence during the study period in Wuhan. **(C)** Air travel flows on January 22, 2020. **(D)** Cumulative cases imported from the 15 cities in Mainland China to Hong Kong. **(E)** The probability of importing at least one case from Wuhan to Hong Kong during the study period. **(F)** Cumulative importation risk from 15 cities in Mainland China to Hong Kong. The map was created using Tableau Software for Desktop version 2021.2.5 (https://www.tableau.com/support/releases/desktop/2021.2.5).

where $M_t^{o,d}$ represents the number of air passengers from origin place O to destination d on the day t, and α is the scaling factor adjusting for the impact on the force of importation from varied surveillance efficiency on COVID-19 in different places [18]. With the assumption that the number of imported cases per day followed the Poisson distribution, we evaluated the 95% confidence interval (CI) of the imported cases based on 100 simulations. Following the study of estimating the probability of cases imported [23, 24], we estimated the cumulative importation risk $\Phi_t^{o,d}$, which denotes the cumulative probability of importing at least one case from the origin place O to the destination d during the period T between t_a and t_b, given by:

$$\Phi_T^{o,d} = 1 - exp\left(- \int_{t=t_a}^{t_b} \Gamma_t^{o,d} dt \right)$$

RESULTS

In our case study, we used daily reported cases of COVID-19 from 15 Mainland China cities, which were previously identified by Lai et al. [15] as high-risk cities COVID-19 imports during January 2020, to estimate the daily pre-symptomatic prevalence of these cities (**Figures 2A,B**). Based on the daily pre-symptomatic prevalence of these cities and the data on air travel flows between these 15 higher-risk Mainland China cities and Hong Kong (**Figure 2C**), we estimated the importation risk of Hong Kong (**Figures 2D–F**). The estimated number of imported cases from our model was 7.6 (95% CI: 5.0–12.1) from 15 higher-risk Mainland China cities into Hong Kong which was consistent with the reported 7 cases originating from Mainland China in Hong Kong before the Wuhan travel ban (January 23, 2020) [14, 25]. The estimated probability of importation of at least one case indicated that Wuhan exported the highest number of cases (5.8, 95% CI: 4.6–7.1) into Hong Kong, followed by Shanghai (0.5, 95% CI: 0.2–0.9) and Beijing (0.5, 95% CI: 0.2–0.9), during the study period.

DISCUSSION AND CONCLUSION

This study aims to provide a general modeling framework to estimate COVID-19 importation risk. We illustrate the feasibility and reliability of the proposed framework with a case study which estimates the importation risk of COVID-19 to Hong Kong from multi-origin places using pre-symptomatic prevalence of infection and air traffic data. Notably, the method accommodates origin places where multiple variants circulate by estimating the importation risk of each variant separately then aggregating them in the destination places, given the availability of prevalence data and human movement data. The method implemented in this study is from a previous study [18] and the reliability of it is demonstrated in the case study of Hong Kong, while proposing a technically innovative method with competitive accuracy is not our major focus. At the current time, only a main method is supported in our modeling framework, while it can be extended by other well-designed and fine-calibrated methods in the future, such as [24, 26–29]. These analyses of the correlation between importation risk and population movement data, preparedness, and vulnerability at the destination, will be further complemented.

This R package *RiskEstim* provides a general modeling framework to estimate the importation risk of infectious disease based on epidemiological and human movement data during an epidemic. The R package can be used as a complementary approach to the pandemic surveillance system to improve response to emerging SARS-CoV-2 variants and the next pandemic. In addition, the R package provides a modifiable codebase that can be extended to estimate the importation risk of other respiratory infectious diseases, such as influenza.

DATA AVAILABILITY STATEMENT

Our software package is developed in R, called RiskEstim. All code to perform the analyses and generate the figures in this study are available from the corresponding author upon reasonable request. Publicly available datasets were analyzed in this study. This data can be found here: https://doi.org/10.5281/zenodo.4266642.

AUTHOR CONTRIBUTIONS

BC, EL, XX, and ZD were involved in the conceptualization and design of the study. MX and ZD designed the statistical methods, conducted analyses, wrote the manuscript, and MX developed the R package. EL, ZD, SS, YB, and PW reviewed and edited the draft.

FUNDING

This work was supported by AIR@InnoHK administered by the Innovation and Technology Commission. We acknowledge the financial support from an initiative funded by the COVID-19 Therapeutics Accelerator and convened by Health Data Research UK, Seed Fund for Basic Research for New Staff of the University of Hong Kong (grant no. 202009185062), and National Natural Science Foundation of China (grant nos. 72104208, 61773091 and 62173065).

REFERENCES

1. Chakraborty I, Maity P. COVID-19 Outbreak: Migration, Effects on Society, Global Environment and Prevention. *Sci Total Environ* (2020) 728:138882. doi:10.1016/j.scitotenv.2020.138882
2. Weekly Epidemiological Update on COVID-19 - 14 December 2021 (2021). Available from: https://www.who.int/publications/m/item/weekly-epidemiological-update-on-covid-19—14-december-2021 (Accessed December 14, 2021).
3. Wu JT, Leung K, Leung GM. Nowcasting and Forecasting the Potential Domestic and International Spread of the 2019-nCoV Outbreak Originating in Wuhan, China: a Modelling Study. *The Lancet* (2020) 395: 689–97. doi:10.1016/s0140-6736(20)30260-9
4. Du Z, Wang L, Cauchemez S, Xu X, Wang X, Cowling BJ, et al. Risk for Transportation of Coronavirus Disease from Wuhan to Other Cities in China. *Emerg Infect Dis* (2020) 26:1049–52. doi:10.3201/eid2605.200146
5. CDC. How CDC Determines the Level for COVID-19 Travel Health Notices (2021). Available from: https://www.cdc.gov/coronavirus/2019-ncov/travelers/how-level-is-determined.html (Accessed December 06, 2021).
6. Nakamura H, Managi S. Airport Risk of Importation and Exportation of the COVID-19 Pandemic. *Transport Policy* (2020) 96:40–7. doi:10.1016/j.tranpol.2020.06.018
7. The Global Health Security Index (2019). Available from: https://www.ghsindex.org/ (Accessed December 14, 2021).
8. Grépin KA. Evidence of the Effectiveness of Travel-Related Measures during the Early Phase of the COVID-19 Pandemic: a Rapid Systematic Review. *BMJ Glob Health* (2021) 6, e004537. doi:10.1136/bmjgh-2020-004537
9. Leung K, Wu JT, Leung GM. Effects of Adjusting Public Health, Travel, and Social Measures during the Roll-Out of COVID-19 Vaccination: a Modelling Study. *The Lancet Public Health* (2021) 6:e674–e682. doi:10.1016/s2468-2667(21)00167-5
10. Devi S. COVID-19 Resurgence in Iran. *The Lancet* (2020) 395:1896. doi:10.1016/s0140-6736(20)31407-0
11. Lai S. Assessing the Effect of Global Travel and Contact Reductions to Mitigate the COVID-19 Pandemic and Resurgence. *medRxiv* (2020) 7:914–923. doi:10.1101/2020.06.17.20133843
12. Ruktanonchai NW, Floyd JR, Lai S, Ruktanonchai CW, Sadilek A, Rente-Lourenco P, et al. Assessing the Impact of Coordinated COVID-19 Exit Strategies across Europe. *Science* (2020) 369:1465–70. doi:10.1126/science.abc5096
13. Government Launches Preparedness and Response Plan for Novel Infectious Disease of Public Health Significance. Available from: https://www.info.gov.hk/gia/general/202001/04/P2020010400179.htm (Accessed December 14, 2021).
14. Lai CKC, Ng RWY, Wong MCS, Chong KC, Yeoh YK, Chen Z, et al. Epidemiological Characteristics of the First 100 Cases of Coronavirus Disease 2019 (COVID-19) in Hong Kong Special Administrative Region, China, a City with a Stringent Containment Policy. *Int J Epidemiol* (2020) 49: 1096–105. doi:10.1093/ije/dyaa106
15. Lai S, Bogoch II, Ruktanonchai NW, Watts A, Lu X, Yang W, et al. Assessing Spread Risk of Wuhan Novel Coronavirus within and beyond China, January-April 2020: a Travel Network-Based Modelling Study. *medRxiv* (2020). doi:10.1101/2020.02.04.20020479
16. Verity R, Okell LC, Dorigatti I, Winskill P, Whittaker C, Imai N, et al. Estimates of the Severity of Coronavirus Disease 2019: a Model-Based Analysis. *Lancet Infect Dis* (2020) 20:669–77. doi:10.1016/s1473-3099(20)30243-7
17. Emery JC, Russell TW, Liu Y, Hellewell J, Pearson CA. The Contribution of Asymptomatic SARS-CoV-2 Infections to Transmission on the Diamond Princess Cruise Ship. *Elife* (2020) 9:e58699. doi:10.7554/eLife.58699
18. Menkir TF, Chin T, Hay JA, Surface ED, De Salazar PM, Buckee CO, et al. Estimating Internationally Imported Cases during the Early COVID-19 Pandemic. *Nat Commun* (2021) 12:311. doi:10.1038/s41467-020-20219-8
19. Tsang TK, Wu P, Lin Y, Lau EHY, Leung GM, Cowling BJ. Effect of Changing Case Definitions for COVID-19 on the Epidemic Curve and Transmission Parameters in mainland China: a Modelling Study. *The Lancet Public Health* (2020) 5:e289–e296. doi:10.1016/s2468-2667(20)30089-x
20. Update on COVID-19 as of 24:00 on 12 February. Available from: http://www.nhc.gov.cn/xcs/yqtb/202002/26fb16805f024382bff1de80c918368f.shtml (Accessed December 14, 2021) (2021).
21. Zhang J, Litvinova M, Wang W, Wang Y, Deng X, Chen X, et al. Evolving Epidemiology and Transmission Dynamics of Coronavirus Disease 2019 outside Hubei Province, China: a Descriptive and Modelling Study. *Lancet Infect Dis* (2020) 20:793–802. doi:10.1016/s1473-3099(20)30230-9
22. Lauer SA, Grantz KH, Bi Q, Jones FK, Zheng Q, Meredith HR, et al. The Incubation Period of Coronavirus Disease 2019 (COVID-19) from Publicly Reported Confirmed Cases: Estimation and Application. *Ann Intern Med* (2020) 172:577–82. doi:10.7326/m20-0504
23. Du Z, Wang L, Yang B, Ali ST, Tsang TK, Shan S, et al. Risk for International Importations of Variant SARS-CoV-2 Originating in the United Kingdom. *Emerg Infect Dis* (2021) 27:1527–9. doi:10.3201/eid2705.210050
24. Yang B, Tsang TK, Wong JY, He Y, Gao H, Ho F, et al. The Differential Importation Risks of COVID-19 from Inbound Travellers and the Feasibility of Targeted Travel Controls: A Case Study in Hong Kong. *Lancet Reg Health - West Pac* (2021) 13:100184. doi:10.1016/j.lanwpc.2021.100184
25. Latest Situation of Novel Coronavirus Infection in Hong Kong. 2022. Available at: https://chp-dashboard.geodata.gov.hk/covid-19/en.html (Accessed December 14, 2021).
26. Jia JS, Lu X, Yuan Y, Xu G, Jia J, Christakis NA. Population Flow Drives Spatio-Temporal Distribution of COVID-19 in China. *Nature* (2020) 582:389–94. doi:10.1038/s41586-020-2284-y
27. Gilbert M, Pullano G, Pinotti F, Valdano E, Poletto C, Boëlle P-Y, et al. Preparedness and Vulnerability of African Countries against Importations of COVID-19: a Modelling Study. *The Lancet* (2020) 395:871–7. doi:10.1016/s0140-6736(20)30411-6
28. Li X, Chen M, Nie F, Wang Q. Locality Adaptive Discriminant Analysis. in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17)* (2017):2021–2027. doi:10.24963/ijcai.2017/306
29. Li X, Chen M, Nie F, Wang Q. A Multiview-Based Parameter Free Framework for Group Detection. In: Thirty-First AAAI Conference on Artificial Intelligence (2017).

Conflict of Interest: BC reports honoraria from AstraZeneca, GSK, Moderna, Pfizer, Roche, and Sanofi Pasteur.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Fast Popularity Value Calculation of Virtual Cryptocurrency Trading Stage Based on Machine Learning

Tong Zhu[1,2]*, *Chenyang Liao*[2], *Ziyang Zhou*[2], *Xinyu Li*[2] *and Qingfu Zhang*[2]

[1] *Tongji University College of Electronic and Information Engineering, Shanghai, China,* [2] *The Third Research Institute of the Ministry of Public Security, Shanghai, China*

This paper proposes a new definition method of currency, which further divides the current hot digital currency according to its legitimacy, encryption, centralization, and other characteristics. Among these, we are mainly interested in virtual cryptocurrencies. Virtual cryptocurrency is one of the application directions of blockchain technology. Its essence is a distributed shared ledger database, which generally has the characteristics of decentralization and non-tampering. The technologies supporting the practical application of virtual cryptocurrencies involve multiple scientific and technological fields such as mathematical algorithms, cryptography, Internet communication, and computer software. Since the launch of the first virtual cryptocurrency bitcoin in 2009, it has developed rapidly worldwide. As of August 1, 2021, more than 11,570 virtual cryptocurrencies have been publicly issued and traded globally, with a total value of over $1.68 trillion. This paper proposes the virtual cryptocurrency trading popularity value system as a standardized index for quantitative analysis of virtual cryptocurrency trading, and the virtual cryptocurrency trading index system as a barometer of the virtual cryptocurrency trading market. It has contributed schemes to the analysis of the market rules of virtual cryptocurrency transactions and the realization and early warning of abnormal virtual cryptocurrency transactions, which are the two main hot research directions of virtual cryptocurrency. To be specific, the popularity value of virtual cryptocurrency transactions provides parameters for analyzing individual virtual cryptocurrencies, and the popularity index of virtual cryptocurrency transactions provides parameters for analyzing the virtual cryptocurrency trading market, so as to prevent major risks of virtual cryptocurrency transactions.

Keywords: blockchain, virtual cryptocurrencies, popularity value, data-driven, social physics

OPEN ACCESS

Edited by:
Chao Gao,
Southwest University, China

Reviewed by:
Jun Wu,
Shanghai Jiao Tong University, China
Zhuo Sun,
Beijing University of Posts and Telecommunications (BUPT), China

****Correspondence:***
Tong Zhu
zhutong.gass@gmail.com

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *02 October 2021*
Accepted: *02 December 2021*
Published: *25 January 2022*

Citation:
Zhu T, Liao C, Zhou Z, Li X and Zhang Q (2022) Fast Popularity Value Calculation of Virtual Cryptocurrency Trading Stage Based on Machine Learning. Front. Phys. 9:788508. doi: 10.3389/fphy.2021.788508

INTRODUCTION

In October 2008, Satoshi Nakamoto published "*Bitcoin: A Peer-to-Peer Electronic Cash System*," introducing people to a digital currency called bitcoin. On January 12, 2009, bitcoin made its first transaction. Since then, digital finance, especially virtual cryptocurrency, has gradually become one of the most important application scenarios of Blockchain Technology (BT) worldwide. At the same time, the volume of virtual cryptocurrency issuance is growing rapidly around the world. As of midnight on August 1, 2021, more than 689,000 bitcoin blocks have been mined, with a total circulating market value of about $750 billion. According to the public data of Coinbase (the first bitcoin exchange with formal license in the USA, listed on NASDAQ) and other exchange websites, as of midnight on August 1, 2021, there are 625 public trading platforms in the world, and more than 11,570 virtual cryptocurrencies are publicly issued and traded. Among them, 5,408 are still actively traded, and the total market value of publicly issued and traded virtual cryptocurrencies exceeds $1.68 trillion.

The continuous innovation and development of digital information technology has accelerated the social process and profoundly influenced the trend of human civilization. Since the advent of virtual cryptocurrencies, combined with the characteristics of blockchain, the transaction mode itself has the characteristics of decentralization, anonymity, multiple currencies, large amount, volatility, difficult to regulate, and so on, resulting in a variety of abnormal transactions. Moreover, there are also abnormal transactions caused by security vulnerabilities such as block bifurcation and theft of numbers and coins.

The main contributions of this paper are briefly summarized as follows:

1) This paper reconstructs the theoretical system of currency definition, reclassifies digital currency, and puts forward a set of definitions mainly applicable to digital currency.
2) This paper will put forward a popularity value system and its algorithm that are actually applied to virtual cryptocurrency, which will help in the field of social public security and effectively enrich the existing virtual cryptocurrency theory.
3) Based on the market data of virtual cryptocurrency as the support, this paper applies the latest research results in the field of artificial intelligence technology to realize the abnormal warning algorithm model of virtual cryptocurrency.

The remainder of the paper is organized as follows: *Theoretical System of Abnormal Transaction of Virtual Cryptocurrency* section introduces the theoretical system of abnormal transaction of virtual cryptocurrency proposed by us. In *Analysis of Abnormal Transaction Situation Based on Virtual Cryptocurrency Popularity Value System* section, based on the popular value system of virtual cryptocurrency, this paper designs two algorithms to realize the situation analysis of abnormal transactions of virtual cryptocurrency, and verifies the virtual cryptocurrency market data from 0:00 to 24:00 on August 3, 2021. *Conclusion* section presents our conclusions and avenues of future research.

THEORETICAL SYSTEM OF ABNORMAL TRANSACTION OF VIRTUAL CRYPTOCURRENCY

The Concept and Definition of Digital Currency, Electronic Currency, and Virtual Currency

There is still no clear authoritative academic definition of the concept of virtual cryptocurrency at home and abroad. Digital Currency, Virtual Currency, Cryptocurrency, Electronic Money, etc., are mostly used to refer to bitcoin-like things. It is urgent to clarify the concepts of digital currency, virtual currency, cryptocurrency, electronic currency, and so on for the same research object by referring to various literature materials. For example, literature [1] puts forward research on the exploration of gold standard credit currency and digital currency, and uses the concept of digital currency to explore the current monetary system and the diversified monetary system of combined gold standard system. Literature [2] proposed that the COMMODITY Futures Trading Commission of the United States issued a consultation on the underwriting scheme of virtual currency, and used the concept of virtual currency to give hints and warnings on how to avoid the potential trading risks of fraud. Literature [3] proposed a study on the fluctuation of cryptocurrency transaction price and investor attention, and used the concept of cryptocurrency to analyze the relationship between investor attention and transaction volatility based on a large data set of about 25 million users. Literature [4] proposed empirical evidence from Indonesia to study the influence of quality and price on the loyalty of Electronic Money users, and the concept of Electronic Money was studied based on the sample of 400 people and model variables on the influence of server reliability and security on the final benefit.

The same concept, at present, also needs to clearly define its scope. For example, before Bitcoin, network game currency is also known as a virtual currency, and at the same time there is a virtual game currency trading at home and abroad research, such as the literature [5] the study of online game virtual currency trading revenue recognition and document [6] the study of digital game virtual currency trading, that also use the concept of virtual currency.

Based on a large number of literature materials, many foreign literatures refer to Bitcoin cryptocurrencies from a technical perspective. From a practical point of view, many domestic literatures call Bitcoin virtual currencies to distinguish it from legal tender. Based on the dual attributes of practicality and technology, this paper calls Bitcoin and other virtual cryptocurrencies. Based on the full absorption of current domestic research results and the development status of cryptocurrencies at home and abroad, the following definitions are proposed: 1) Commodity Currency is a commodity that has value and physical form, which can be traded as an exchange for equivalent value. 2) Digital Currency (DC) is a currency that shows its value attribute in the digital form and corresponds to the commodity currency. It generally refers to all currencies that exist in digital form and can be used as means of payment. It is a general term for electronic money and virtual currency. 3) Electronic Money (Official Digital Currency) is official digital money. Electronic money is a digitized form of fiat money, equivalent to fiat money, for example, electronic money stored in the form of magnetic cards, central bank digital money, and so on. 4) Virtual Currency (Unofficial Digital Currency) is unofficial digital currency. Virtual cryptocurrency is a digital currency issued by a non-statutory authority. It is generally used as a means of payment in a specific virtual space on the Internet, but it does not have the status and value of legal tender, such as QQ

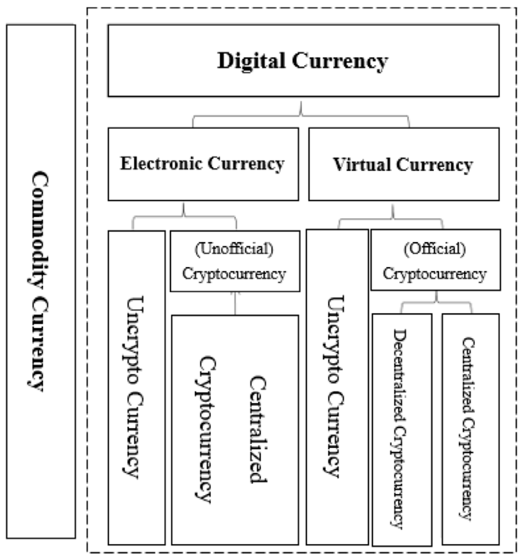

Digital Currency
Commodity Currency
Electronic Currency
Virtual Currency
Uncrypto Currency
(Unofficial) Cryptocurrency
Centralized Cryptocurrency
Uncrypto Currency
(Official) Cryptocurrency
Decentralized Cryptocurrency
Centralized Cryptocurrency

FIGURE 1 | Concept and definition of currencies.

coins, online game coins, and so on. 5) Cryptocurrency is a digital currency generated based on cryptography and generally encrypted by cryptography in transactions, storage, and payments, such as Bitcoin and central bank digital currency. 6) Virtual Cryptocurrency is cryptocurrency that does not have legal status among digital currencies, including decentralized cryptocurrencies such as Bitcoin and Ethereum, and centralized cryptocurrencies such as USDT.

After giving each currency qualitative, we classify according to its connotation, generation mechanism, and operation principle according to certain classification standards, delimit its boundaries, and get the relationship between each currency, as **Figure 1**.

The subsequent research of this paper will mainly focus on virtual cryptocurrency.

Development Status of Digital Cryptocurrency

Digital currencies are closely related to blockchain technology [7]. Blockchain is a technology that securely stores transaction records on peer-to-peer networks, rather than storing them at a single site. Blockchain is run by a network of independent servers, called nodes, scattered around the world. The application of blockchain technology has been extended to digital finance, Internet of Things, intelligent manufacturing, supply chain management, digital asset trading, and other fields. At present, major countries around the world are speeding up the layout of blockchain technology development.

Digital currency is the first successful case of blockchain. The characteristics of its decentralization, anonymity, and safety for the user do not depend on banks and other intermediaries, and direct point-to-point trading may provide the biggest advantage to enhance the autonomous control ability of the end user—this in financial history is also a very big change. However, although bitcoin and other digital currencies are also known as money, due to their lack of value connotation and sharp price fluctuations, it is difficult to play the basic functions of money, such as the function of value scale, which makes the current digital currency closer to a financial asset in essence.

With the rapid development of network technology and digital economy, the public's demand for convenience, security, universality, and privacy of retail payment is increasing day by day. According to the "*White paper on the development of China's Central bank digital currencies*" [8] released by the People's Bank of China in 2021, central banks or monetary authorities in many countries and regions closely follow the development achievements of fintech [9] and actively explore the digital form of legal tender. Legal digital currency is moving from theory to reality.

In the process of the stable development of digital virtual cryptocurrency, non-statutory virtual currency has become a forefront, has a lot of traffic, and has been an extensive concern by speculators, including many speculators through the hype of virtual cryptocurrency to obtain huge profits; the musk is that there is no lack of such capital tycoon and some social celebrities involved, and the platform for them has had a profound impact on the development of virtual cryptocurrencies. The source of virtual cryptocurrency is Bitcoin, also known as virtual cryptocurrency, launched by Satoshi Nakamoto in 2008. Later, on the basis of the currency and virtual encrypted monetary growth development, the etheric fang for the platform of the second generation of virtual cryptocurrency was developed through intelligent core application contract implementation, and now, although only in a few countries, the government expressed support for virtual cryptocurrency, while most of the national governments are in opposition to or are on the sidelines, However, the third generation of virtual cryptocurrencies is still budding. At present, according to incomplete statistics, since the advent of Bitcoin, the private sector has launched a variety of virtual cryptocurrencies, and the online public circulation and issuance of virtual cryptocurrencies has reached more than 10,000, with a total market value of more than $130 million, and is still increasing at an extremely fast speed.

Research Status of Virtual Cryptocurrency

Virtual cryptocurrencies such as the blockchain technology, P2P technology, and encryption technology, such as hot technology, are declared as "decentralized" and "completely anonymous," but the lack of value support, price volatility, trading defects such as low efficiency, and large energy consumption limit its hard currency function in our daily life, and at the same time, the virtual cryptocurrency is used for speculation, There are potential risks that threaten financial security and social stability, and it has become a payment tool for illegal economic activities such as money laundering. Virtual cryptocurrency has major defects, and some institutions attempt to launch the so-called "stable currency" by trying to anchor it with sovereign currency to stabilize the currency or related assets, and business plans for global stability of the currency; it will give the international monetary system, the payment and settlement system, and

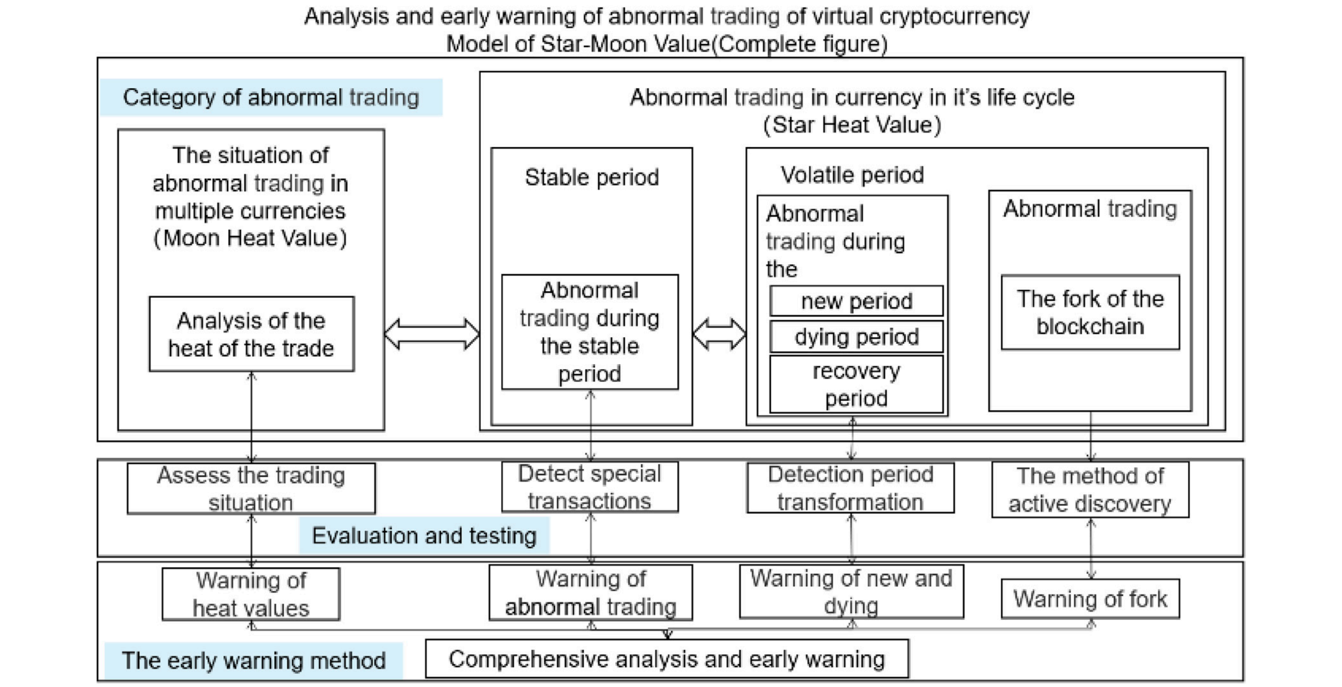

Analysis and early warning of abnormal trading of virtual cryptocurrency
Model of Star-Moon Value(Complete figure)
Category of abnormal trading
The situation of abnormal trading in multiple currencies (Moon Heat Value)
Analysis of the heat of the trade
Abnormal trading in currency in it's life cycle (Star Heat Value)
Stable period
Abnormal trading during the stable period
Volatile period
Abnormal trading during the
new period
dying period
recovery period
Abnormal trading
The fork of the blockchain
Assess the trading situation
Detect special transactions
Detection period transformation
The method of active discovery
Evaluation and testing
Warning of heat values
Warning of abnormal trading
Warning of new and dying
Warning of fork
The early warning method
Comprehensive analysis and early warning

FIGURE 2 | Stellaluna value model for anomaly analysis and warning of virtual cryptocurrency.

monetary policy, such as cross-border capital flow management, many risks and challenges [10].

At present, many countries in the world are promoting research on virtual cryptocurrency transaction behavior and warning, and analyzing its integration layout with existing economic applications from multiple aspects such as technology and law. For example, literature [11] proposes the investigation and research of the EU's virtual cryptocurrency regulations on virtual cryptocurrency developers and transaction users; literature [12] studies the impact of Facebook's launch of Libra virtual cryptocurrency project on financial infrastructure and regulation; literature [13] analyzes the problems and challenges of the cryptocurrency exchange market in India from the perspective of triggering financial risks; literature [14] studies the risks and challenges of virtual cryptocurrency transactions in different jurisdictions and SWIFT transactions; literature [15] studies the virtual cryptocurrencies and gold assets, such as Bitcoins and the diversified investment transactions such as the stock market; literature [16] studies the digital financial assets and the monetary impact on the Russian business transactions in current laws and regulations; literature [17] studies the current stage of the European Union blockchain and virtual cryptocurrency trading regulation, and so on.

At present, the abnormal encryption based on virtual currency trading and warning image data to carry out the relevant analysis is one of the main directions of research hot spots; the researchers are mainly focused on virtual cryptocurrency data exchange through time, frequency, capital flows, the rules of combination of multiple currencies in a variety of ways, and so on, whether virtual cryptocurrency trading volatility and abnormal trading to forecast early warning.

In the analysis based on time, some researchers put forward methods that can improve the accuracy of prediction and warning. Literature [18] is proposed using the Markov switching model window effect of abnormal to virtual cryptocurrency trading to predict warning, by comprehensive research samples within the coefficient and the outside influence on the sample, using the window effect of Markov switching models to predict early warning, and verified in some specific window on the tail that can better realize the precision of forecasting warning. Literature [19] proposed a weighted and pay attention to the memory channel convolution neural network to predict abnormal virtual cryptocurrency trading early warning method, based on the strong correlation between different virtual cryptocurrencies and using the technology of deep learning implementation with a weighted and pay attention to the memory channel convolution neural network model to forecast daily virtual cryptocurrency trading.

In terms of frequency-based analysis methods, some researchers have carried out analysis and prediction of currency transactions of multiple virtual cryptocurrencies. In 2021, Kim et al. proposed to use GARCH and SV random volatility to predict and warn the volatility of abnormal transactions of virtual cryptocurrencies. By studying nine major virtual cryptocurrencies such as Bitcoin, Ethereum, and Bitcoin Cash, the Bayesian Stochastic Volatility (SV) model and GARCH model are implemented to detect trading volatility [20]. Literature [21] proposed a random approximation algorithm and sequence learning method based on volatility dynamics, and proposed a random volatility model with jump return volatility to analyze and warn abnormal transactions of virtual

TABLE 1 | Rules for compiling series index of virtual cryptocurrency popularity value system.

Indices of virtual cryptocurrency	Definition
VC7	The Virtual Cryptocurrency 7 Index tracks the seven largest virtual cryptocurrencies in the world by market capitalization. An index based on a square root ratio represents the change in the value of the seven major virtual cryptocurrencies. The sample market value of the virtual cryptocurrency 7 index accounts for more than 75% of the entire virtual cryptocurrency, and its ups and downs represent the capital flow of large capital groups
VC20	The Cryptocurrency 20 Index tracks the price movements of the world's top 20 virtual currencies by market capitalization, and is a price trend index for virtual currencies that monitors the structure of their component prices over time. The sample of the top 20 virtual cryptocurrencies in circulation market value of major digital asset exchanges aims to reflect the overall performance of the mainstream virtual cryptocurrencies in the current digital currency market
VC100	The Cryptocurrency 100 Index tracks the price movements of the world's top 100 virtual cryptocurrencies by market capitalization. Equivalent to the whole market currency, accounting for more than 95%, the overall rise and fall reflect the market capital, and whether there is incremental capital entry
VC7X	The virtual cryptocurrency 7X index excludes bitcoin, which has the largest market capitalization, from the virtual cryptocurrency 7 index, and represents the changes in the value of the six major virtual currencies besides Bitcoin. The overall influence is huge, and it is also the first choice of big funds. It also has the bloodsucking effect, reflecting the overall trend of the positive market
VC20X	The Virtual Cryptocurrency 20X Index excludes the top 7 virtual currencies with the largest market capitalization based on the Virtual Cryptocurrency 20 Index and is used to represent the value of 13 major virtual currencies other than the virtual cryptocurrency 7 index relative to Bitcoin. Reflects the momentum of new virtual cryptocurrency trends
VC100X	The Cryptocurrency 100X Index excludes the top 20 virtual currencies with the largest market capitalization based on the Cryptocurrency 100 Index and is used to represent the value of 80 major virtual currencies other than the Cryptocurrency 20 index relative to Bitcoin. The market value is relatively small, but the hype value is high, and the risk is great

cryptocurrencies. The results proved that these virtual cryptocurrency transactions showed abnormal return fluctuation relationship. Salim Lahmiri et al. proposed to use the method of high frequency trading deep learning to predict and give early warning to virtual cryptocurrency bitcoin trading, and applied the deep forward neural network (DFFNN) to analyze and predict the high frequency trading data of virtual cryptocurrency Bitcoin. Based on this, the influence of standard numerical training algorithm on the accuracy obtained by DFFNN is also studied [22].

In terms of the analysis method based on capital flow, research is carried out on the correlation between capital and abnormal transaction of virtual cryptocurrency. Rahmani et al. drew on the relevant experience of deep learning architecture in financial market prediction and proposed to use deep learning algorithm to predict and give early warning of abnormal capital flow in the direction of virtual cryptocurrency transaction [23]. By constructing LSTM model to predict the daily closing direction of bitcoin and USDT in virtual cryptocurrencies, the researchers also analyzed the accuracy of the model and the risk of trading gains and losses based on the model, and evaluated the impact of MACD index and input matrix dimension on the prediction and warning accuracy.

In terms of analysis methods based on multiple currencies and multiple rule combinations, Kakinaka Shinji et al., based on the asymmetric relationship between price and volatility, is a significant feature of the time series of financial markets. Therefore, they proposed the method of multifractal cross-correlation for virtual cryptocurrency trading and prediction and early warning. Studying these relationships between up-market (bull market) and down-market (bear market) mechanisms in a dynamic way provides a new method for predicting and warning abnormal trading of virtual cryptocurrency [24].

Although some researches on virtual cryptocurrency trading do not directly provide the method of abnormal trading warning, additional variables such as daily return rate, standard deviation, value at risk, conditional value at risk, trading volume, and other dimensions that are very important for the analysis of virtual cryptocurrency trading are introduced. In addition, many innovative methods are proposed for reference in terms of how to analyze the data of virtual cryptocurrency transactions and how to select variables. In literature [25], for example, this paper proposes a virtual encryption based on PROMETHEE II currency trading portfolio selection criterion method, and the model of sample performance with five other kinds commonly used the optimal portfolio model; according to the index of all observed, the proposed model is better than all the other models, where the odds ratio is from 50 to 94%. The literature also suggests the benefits of adopting more currency standards and an appropriate multi-parameter approach in the analysis and selection process of virtual cryptocurrency transaction data.

Some researchers also provide abnormal transaction warning methods of virtual cryptocurrency from other useful perspectives by choosing different data dimensions. For example, in 2021, Kądziołka Kinga proposed a multi-criteria evaluation method for virtual cryptocurrency exchange based on PROMETHEE II and taxonomy with the analysis of the available data published on the Internet website by hierarchical clustering with k-means algorithm [26]. This method can also be used as a supplement to multi-dimensional evaluation of abnormal transaction warning of virtual cryptocurrency.

Based on the stock market index method, this paper innovatively proposes an abnormal movement warning

Algorithm 1 | Elite Ant Colony Algorithm Based on Mixed Parameters

```
Input: Virtual cryptocurrency quotation data
Output: Virtual cryptocurrency real-time trading popularity value
Procedure:
Step 1. Data pre-processing
Step 2. Set the initial parameters
Step 3. Iterative search
Step 3.1. Randomly determine the initial position and set a forbidden area
Step 3.2. Calculate the transfer probability
Step 3.3. Assign individual weight interval
Step 3.4. Search the weight interval of other weight factors by traversing in turn
Step 4. Tag updates
```

algorithm based on the popularity value system of virtual cryptocurrency. By referring to the sequence similarity comparison algorithm in the field of speech recognition and biological information, the sequence similarity comparison algorithm is improved as an abnormal detection algorithm of virtual cryptocurrency. It provides a new idea for the study of abnormal warning of virtual cryptocurrency.

The Overall Model of Abnormal Transaction of Virtual Cryptocurrency

Virtual cryptocurrency anomaly detection and early-warning star-moon value model is shown in **Figure 2**. Based on current research results at home and abroad [27–29], this model organically combines the definition, research, and early warning of virtual cryptocurrency anomalies into an overall model.

Normally, in the whole currency virtual cryptocurrencies trading environment, we can think of the influence of external factors on the overall virtual cryptocurrency in a fair level, then we can pass the current transaction data and compare them with the historical transaction data analysis, and calculate the current external factors' impact on the virtual cryptocurrency trading situation; this is the "virtual cryptocurrency popularity value". Based on the change of this popularity value, abnormal transactions of virtual cryptocurrency can be further detected and warned. In the case of a single virtual cryptocurrency trading to multi-currency virtual cryptocurrency "trading popularity value" as a "background noise," a single virtual cryptocurrency

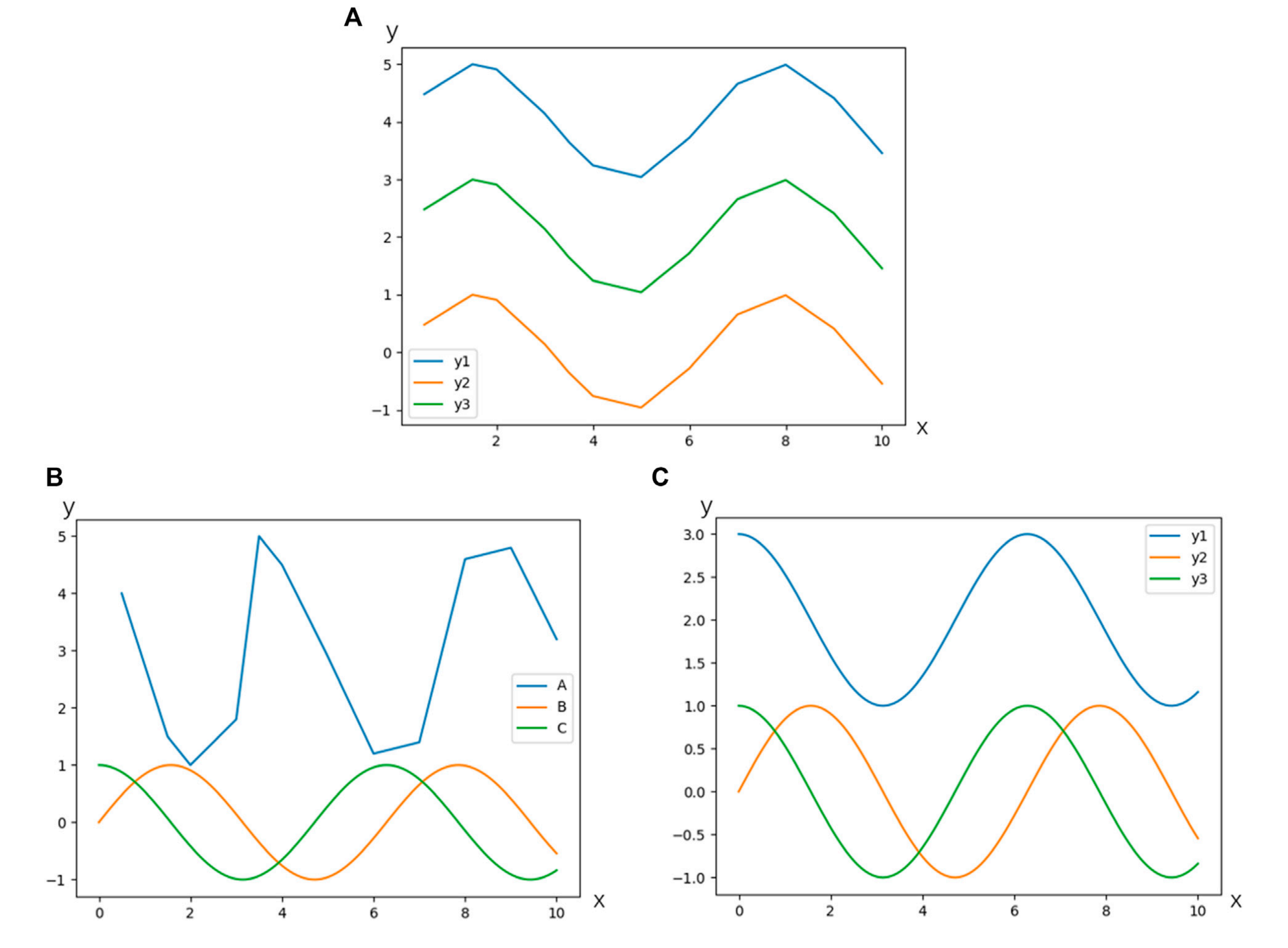
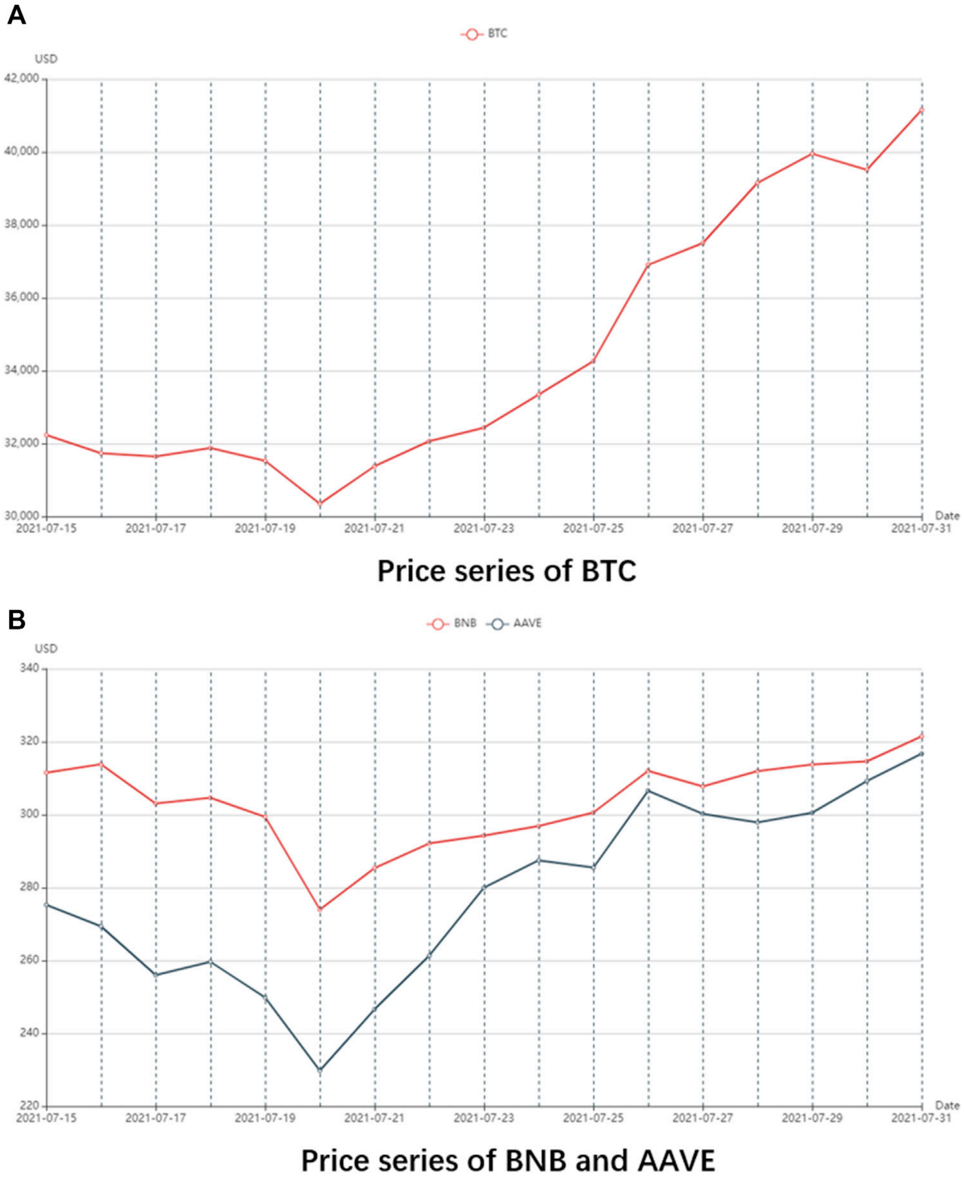


FIGURE 3 | Definition of similarity.



Price series of BTC

Price series of BNB and AAVE

FIGURE 4 | Price series from July 15 to July 31. **(A)** Price series of BTC. **(B)** Price series of BNV and AAVE.

is determined based on the elimination of "background noise" transaction in which life cycle stage integrated different life cycle phase virtual cryptocurrency, which, according to the general characteristics of virtual cryptocurrency trading, abnormal transaction behaviors are more likely to occur in the emerging period, extinction period, and recovery period. Therefore, these three periods and the specific event period of the outbreak of blockchain fork are collectively referred to as the volatile period. The abnormal transaction risk of a single virtual cryptocurrency can be warned by detecting the volatile period and cycle transformation. For a single currency in a stable period, on the basis of protecting the privacy of a single transaction, detect and warn whether the virtual cryptocurrency transaction is affected by security factors and leads to abnormal transactions. To be specific, we can design an algorithm to calculate the transaction popularity value of multiple currencies as a benchmark, and each single currency will conduct qualitative detection according to the popularity value. If abnormal transaction characteristics are met, quantitative detection will be carried out further. The model also includes detection, warning, and disposal of abnormal transactions of virtual cryptocurrencies caused by security factors such as blockchain bifurcation.

In general, the model covers the definition, analysis, and early warning of virtual cryptocurrency anomalies to achieve comprehensive early warning on four different levels of abnormal transaction situation of multi-currency virtual cryptocurrency, abnormal transaction cycle risk of a virtual

TABLE 2 | Shape distance schema table.

K_{i+1} Pattern K_i	$K_{i+1} < -th$			$-\text{th} < K_{i+1} < th$	$K_{i+1} > th$		
	$\Delta k < 0$	$\Delta k = 0$	$\Delta k > 0$				
$K_i < -th$				0	3		
	−3	−2	−1				
$-\text{th} < K_i < th$	−3			0	3		
					$\Delta k < 0$	$\Delta k = 0$	$\Delta k > 0$
$K_i > th$	−3			0			
					1	2	3

cryptocurrency, an abnormal transaction of virtual cryptocurrency caused by security factors based on privacy protection, and abnormal transaction of virtual cryptocurrency bifurcation.

The main contributions of this paper are briefly summarized as follows: 1) This paper reconstructs the theoretical system of currency definition, reclassifies existing definition of currency, and puts forward a set of definitions mainly applicable to digital currency. 2) This paper will put forward a popularity value system and its algorithm that are actually applied to virtual cryptocurrency, which will help in the field of social public security and effectively enrich the existing virtual cryptocurrency theory. 3) Based on the market data of virtual cryptocurrency as the support, this paper applies the latest research results in the field of artificial intelligence technology to realize the abnormal warning algorithm model of virtual cryptocurrency.

ANALYSIS OF ABNORMAL TRANSACTION SITUATION BASED ON VIRTUAL CRYPTOCURRENCY POPULARITY VALUE SYSTEM

Virtual Cryptocurrency Popularity Value System

The Concept of Virtual Cryptocurrency Popularity Value

As of midnight on August 1, 2021, more than 11,570 virtual cryptocurrencies have been publicly issued worldwide, among which 5,408 are still actively traded. Unlike stocks, futures, and precious metals, the current “head” effect of virtual cryptocurrency trading is very obvious, with the top 7 accounting for more than 75% of the total cryptocurrency trading, and the top 100 accounting for about 95% of the total cryptocurrency trading. The market capitalization of the top 1,000 accounts for about 99.7% of the entire virtual cryptocurrency trade.

The overheated hype and the legend of overnight wealth have caused “herd behavior” [30–35] in the virtual cryptocurrency market, which has caused many individual investors to blindly enter [36], and also provided opportunities for some criminals to take advantage of blind individual investors’ ignorance of virtual cryptocurrency. The launch of “unstable value virtual cryptocurrencies” has ruined the fortunes of many people who want to get rich quick. The existence of herd behavior and the “thunderbolt” of many virtual cryptocurrencies make people eager for an early warning method to avoid risk.

Based on the actual situation of virtual cryptocurrency trading, we innovatively proposed virtual cryptocurrency trading popularity value. The data of the top 1,000 virtual cryptocurrency transactions in each large virtual cryptocurrency trading market are used as samples to calculate the popularity value of virtual cryptocurrency transactions; to standardize and quantify virtual cryptocurrency, it is intended to serve as a benchmark reference value for abnormal transactions in virtual cryptocurrencies.

Virtual Cryptocurrency Popularity Value System Series Index

Based on the popularity value of each virtual cryptocurrency transaction, we innovatively put forward the series index of virtual cryptocurrency transaction popularity value system, ranking the top 7 (VC7), the top 20 (VC20), the top 100 (VC100), and the top 6 (VC7X) of the 7 excluding the largest market value bitcoin, 13 out of 20 stocks excluding the top 7 market capitalization (VC20X), and 80 out of 100 stocks excluding the top 20 market capitalization (VC100X). A total of 6 groups of indexes, virtual cryptocurrency trading popularity value system series indexes, aimed at reflecting the current performance of virtual cryptocurrency trading. **Table 1** shows the compiling rules of the series index of the virtual cryptocurrency trading popularity value system.

Calculation of Virtual Cryptocurrency Popularity Value Based on Hybrid Parameter Elite Ant Colony Algorithm

To scientifically evaluate the real-time popularity of virtual cryptocurrency transactions, classical quantitative analysis methods mostly use real-time price or some inherent attribute as its popularity value [37]. This algorithm has a certain scientific nature and objectively reflects its popularity directly through the market, but it also has some defects. It can reflect its popularity through price changes in a certain period of time, but it cannot reasonably reflect its real popularity in a longer period of time. For example, at two different time points, the same virtual cryptocurrency transaction has the same popularity, but because the price base of the two time points is different, under the condition of the same popularity, the final price is not the same, which has a different popularity value from the result. Because the price of virtual cryptocurrencies is not only influenced by the popularity but also by the amount of money they are issued and how they are created, the price base is different. In other cases, two virtual cryptocurrency transactions may have the same popularity for a period of time after launch, but at different prices. In general, this algorithm has some major defects, such as lack of normalization and standardization, which cannot form a comprehensive popularity value system for all currencies of

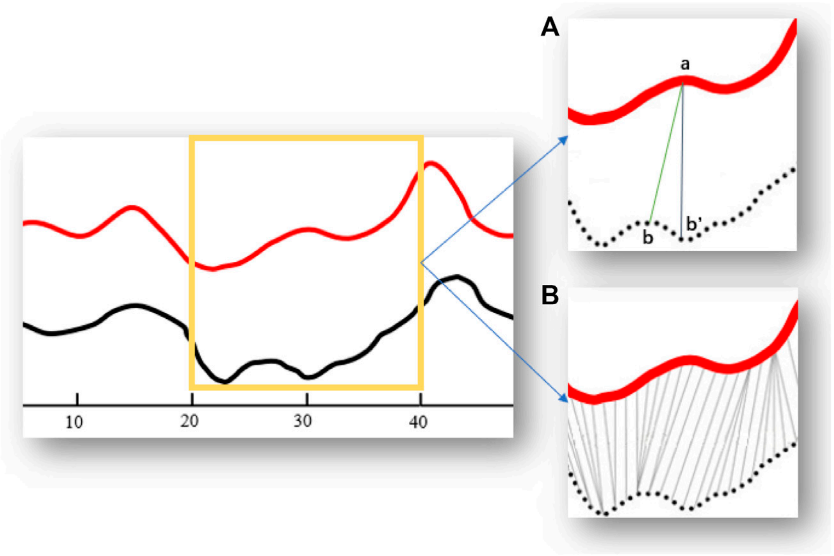


FIGURE 5 | DTW looks for waveform alignment points.

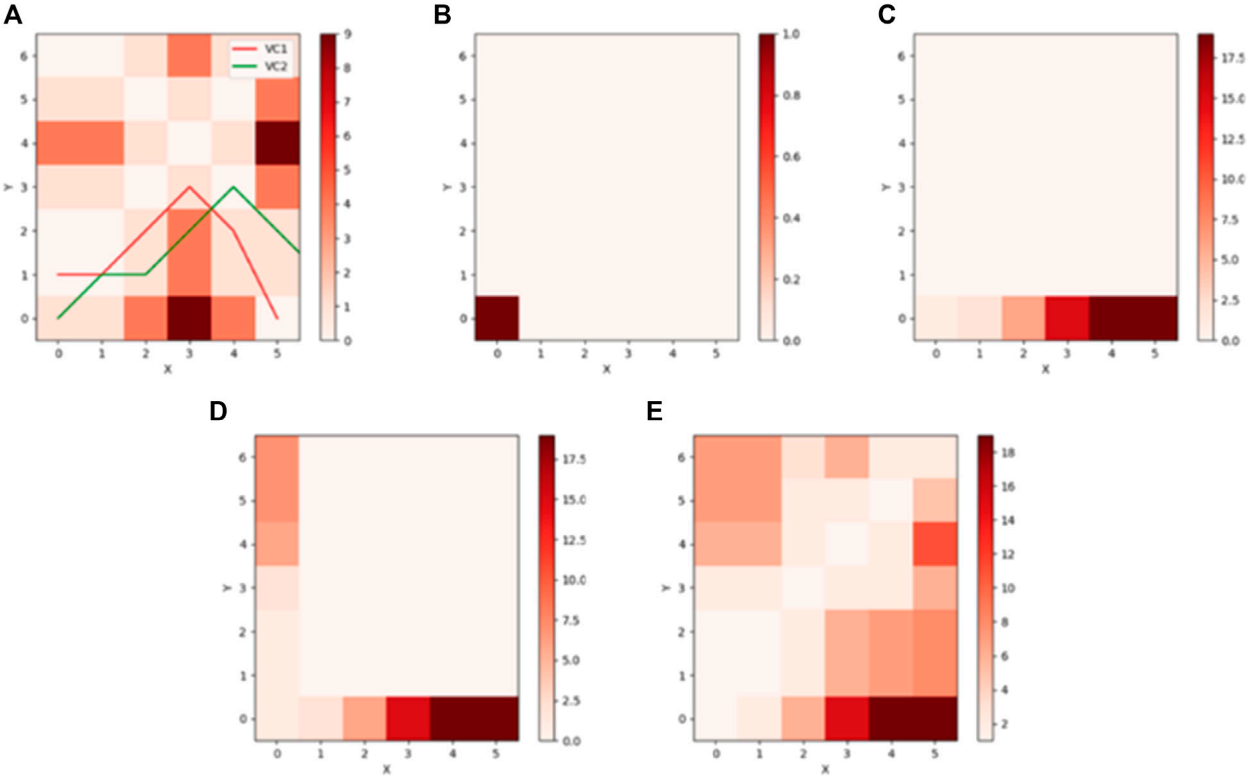


FIGURE 6 | Sequence distance matrix calculation.

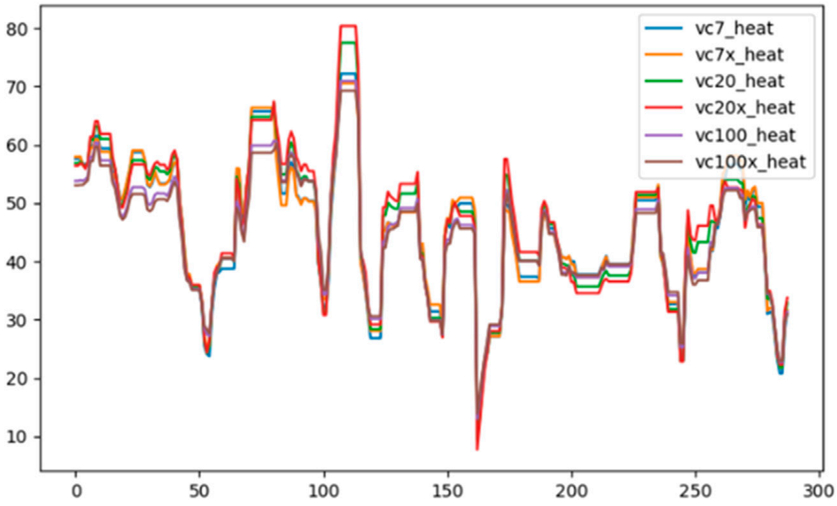


FIGURE 7 | Comparison of six trading popularity indices on August 3, 2021.

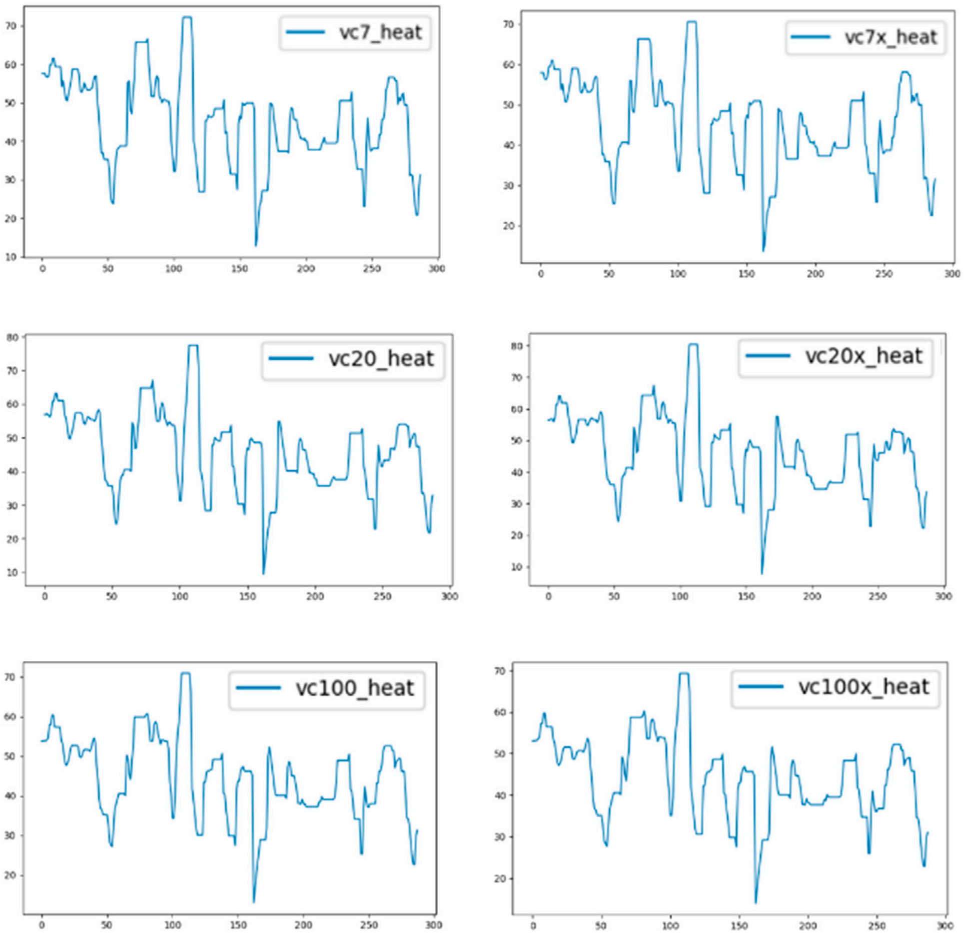


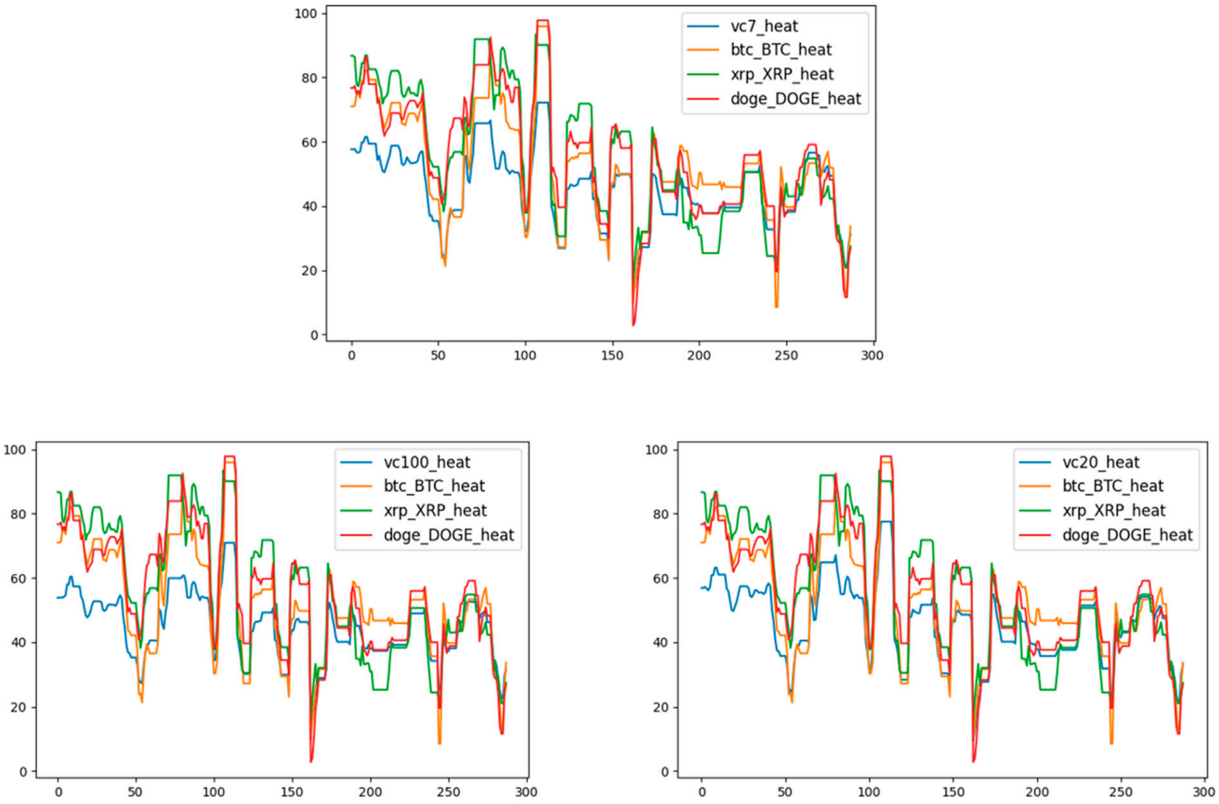
FIGURE 8 | August 3, 2021 six trading popularity indices.

virtual cryptocurrency transactions. It can only reflect the popularity changes of certain virtual cryptocurrency transactions within a period of time.

In the study of virtual cryptocurrency trading, literature [38] mentioned that to weigh the benefits and risks of bitcoin investment, an evaluation index system of virtual cryptocurrency trading activities was established, and the combination weight of comprehensive evaluation was determined by using subjective evaluation method and objective evaluation method comprehensively. However, the algorithm is unable to cope with virtual encrypted mutations situation of currency trading and poor robustness.

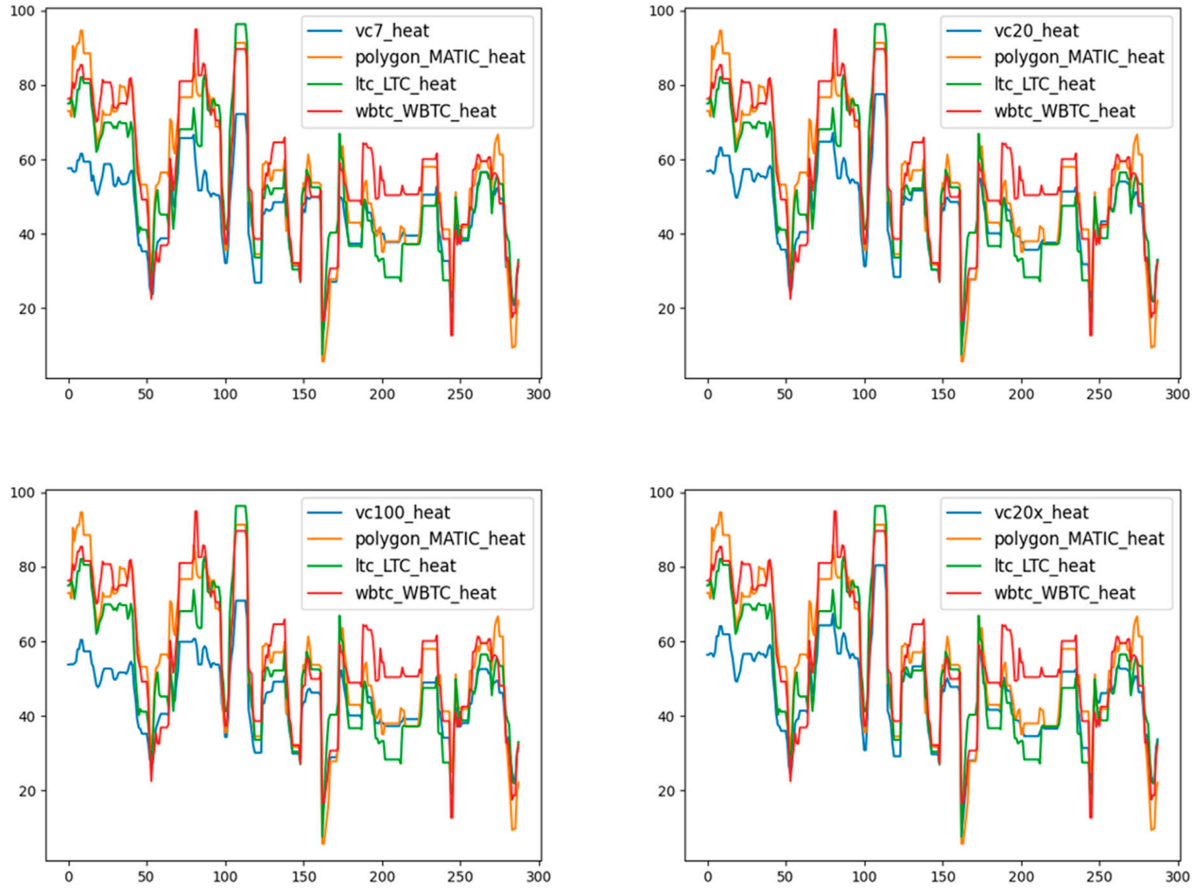
This paper proposes an elite ant colony algorithm based on mixed parameters that calculates the popularity value of virtual cryptocurrency transactions. Selection is given priority over virtual cryptocurrency trading market data, virtual cryptocurrency trading chain abnormal data, virtual cryptocurrency trading data, and so on; multi-dimensional data are complementary comprehensive data as a factor, dynamic adjustment factor, and dynamic allocation weights, respectively, combined with the normalized and standardized data processing methods, such as integrated computation virtual cryptocurrency trading phase popularity value. Compared with traditional algorithms that rely on expert experience and fixed factors and factor weights, this method is more scientific and convenient, saving a lot of human and material resources. Relying on machine learning method, the timeliness of trading popularity value system is significantly improved.

On combinatorial optimization problem and optimal solution problem, ant colony algorithm is widely used [39]; to solve the problem of multiple factor weights allocation, this article uses the elite ant colony algorithm based on hybrid parameter dynamic weighting allocation of more factors, using the advantages of the classical ant colony algorithm adaptability that is strong and adapts to the rapid changes of virtual cryptocurrency trading. In classical ant colony algorithm, the information heuristic factor α



	VC7	VC20	VC100
BTC	1604.04	149.30	541.84
XRP	126.23	301.49	628.20
DOGE	195.76	758.36	946.87

FIGURE 9 | Comparison of BTC, XRP, and DOGE popularity value with the three indexes on August 3.



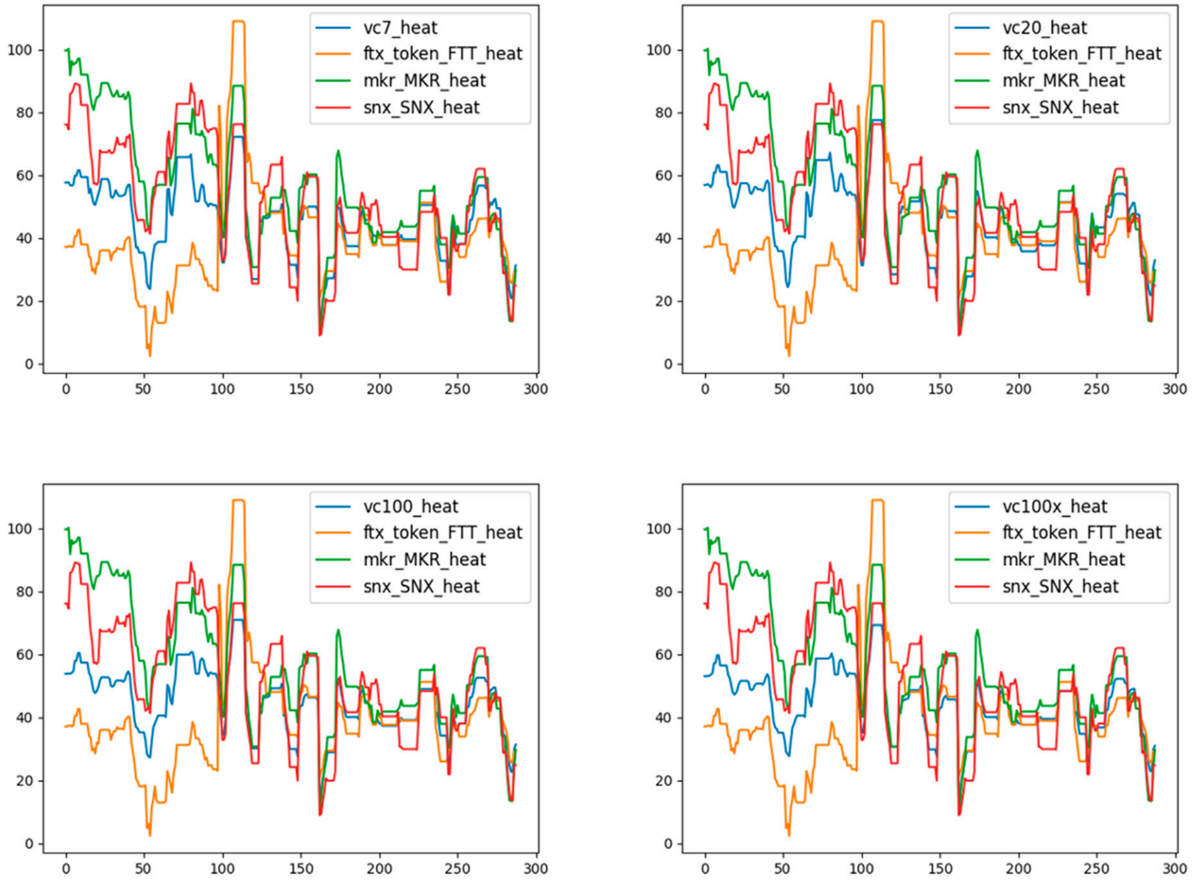
	VC7	VC20	VC100	VC20X
MITIC	466.15	217.39	237.23	233.70
LTC	892.10	4.15	320.68	162.00
WBTC	321.85	95.22	266.37	193.59

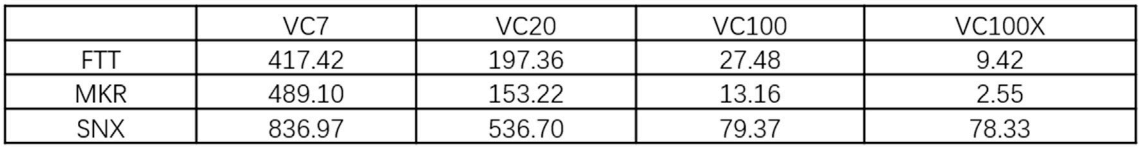
FIGURE 10 | MATIC, LTC, and WBTC popularity values compared with the four indexes on August 3.

and expectation heuristic factor β are the same for all individuals. This makes the performance of the algorithm depend sensitively on the setting of some parameters. In addition, the optimal solution may be ignored. To avoid this, classical ant colony algorithms tend to turn up the information heuristic factor α and the expectation heuristic factor β, but this leads to local minima, reducing the likelihood that another, shorter TSP path will be found later. To solve these problems, this paper optimizes the marking rules and α and β parameters of classical ant colony algorithm. We sorted TSP path lengths from small to large. Considering that the market value of currencies at the head of virtual cryptocurrency transactions accounts for too large a proportion, the improved algorithm only allowed the top 20% individuals to leave marks, and the weight was inversely proportional to the path length and multiplied by a factor that decreased linearly with the market value (i.e., the ranking factor). The first-place ranking factor is 1, and the ranking factor is reduced to 0 at 20%. To avoid the optimal TSP path being forgotten, when the first place of this iteration is worse than the optimal TSP path discovered so far, the optimal TSP path is also ranked first as an individual, and the ranking of other individuals is postponed, and marks are left according to the aforementioned rules. This ensures that the optimal TSP path is not forgotten until it is overtaken by a better TSP path.

In this paper, the steps of calculating the popularity value of virtual cryptocurrency transaction using the elite ant colony algorithm based on mixed parameters are as follows:

Step 1. Data pre-processing. Six fields in the market value of virtual cryptocurrency, total trading volume, real-time price, one-hour rise or fall, one-day rise or fall, and one-week rise or fall are taken as weight factors to calculate the popularity value of virtual cryptocurrency trading. The value range of weight is divided into (0,0.01), (0.01, 0.02), . . ., (0.99, 1.00), then each value range can be





	VC7	VC20	VC100	VC100X
FTT	417.42	197.36	27.48	9.42
MKR	489.10	153.22	13.16	2.55
SNX	836.97	536.70	79.37	78.33

FIGURE 11 | FTT, MKR, and SNX popularity values compared with the four indexes on August 3.

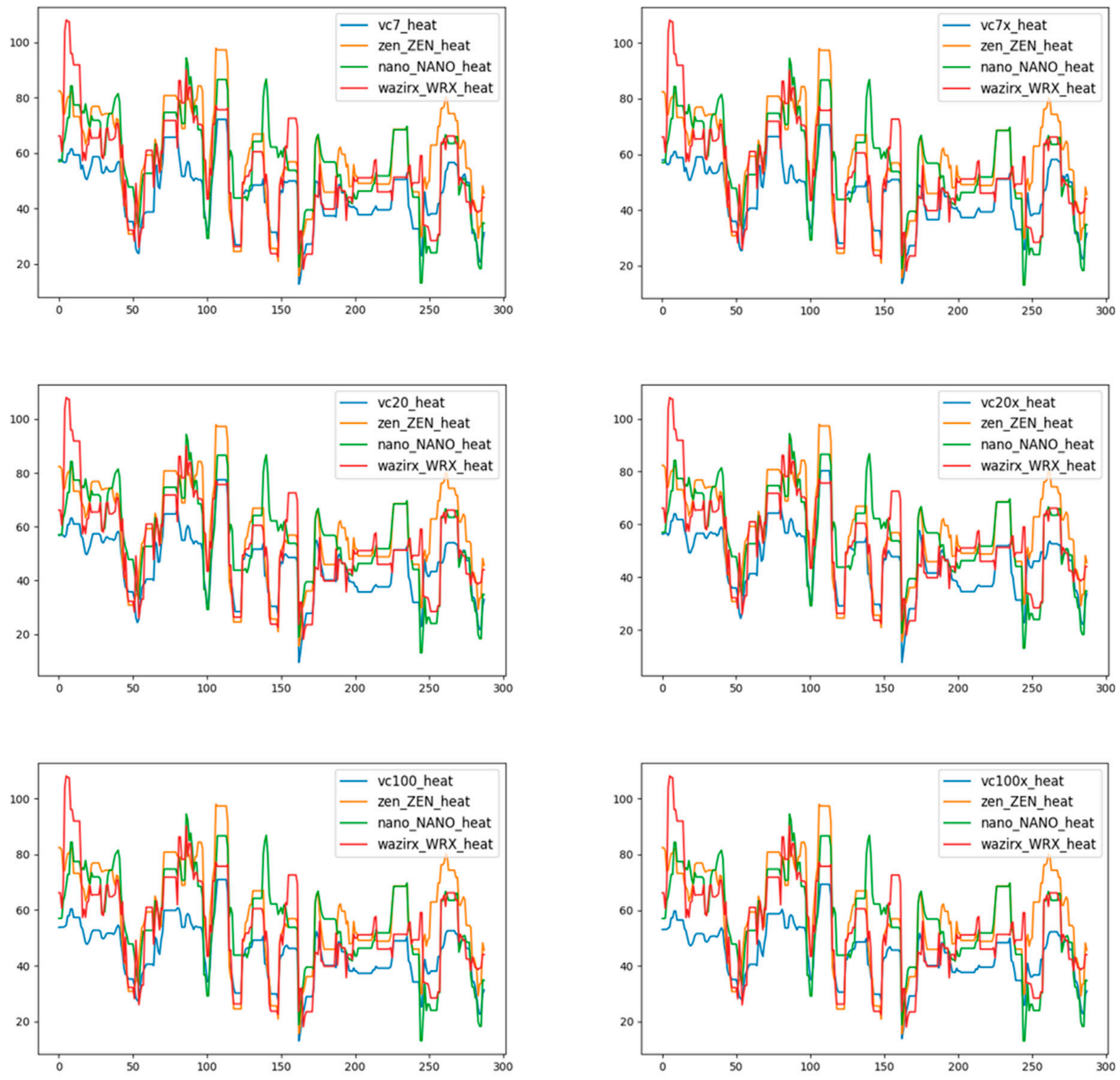
expressed as $(x_{ij}, x_{i(j+1)})$, i = 1, 2, . . . , 6; j = 0, 1, . . . , 99. At the same time, set the markers of each interval to 1.

Step 2. Set the initial parameters. We set 5 parameters: information elicitation factor—***α***, expectation heuristic factor—***β***, ant colony number—***m***, informational volatile factor—***ρ***, and ranking factor—***γ***. The larger α is, the more likely the new individual will select the region passed by the previous individual. The smaller α is, the smaller the group search range is, and it is easy to fall into local optimum. The larger β is, the more likely the population is to select the locally better interval, and the faster the iterative convergence rate is, but the local relative optimization is easy to occur. The larger m is, the more accurate the algorithm result is. However, as the algorithm approaches the convergence of the optimal solution, the effect of positive information feedback decreases, and a lot of repeated calculation work occurs. The smaller the ρ is, the larger the marker value of each interval, the larger the group search range, and the slower the convergence. The larger the ρ is, the smaller the marker value is, and it is easy to fall into local optimum. The greater the γ is, the greater the currency's influence. The first-place ranking factor is 1, and the ranking factor is reduced to 0 at 20%.

Step 3. Iterative search. The weight interval of the six factors is searched iteratively in turn, and the process is divided into four steps as follows.

Step 3.1. Randomly determine the initial position and set a forbidden area to ensure that you do not fall into a local loop.

Step 3.2. Equation 1 was used to calculate the transfer probability:



	VC7	VC20	VC100	VC7X	VC20X	VC100X
ZEN	4374.40	3567.76	2776.21	4618.42	2680.19	2631.82
NANO	3447.82	815.80	446.06	2520.22	259.92	425.64
WRX	4126.15	3581.39	3009.96	2600.98	3133.14	3245.02

FIGURE 12 | August 3 ZEN, NANO, and WRX Popularity Value vs. Six Index comparison.

$$P_j^k = \begin{cases} \dfrac{[\tau_j]^\alpha \cdot [n_j]^\beta}{\sum_{s\in J_k}[\tau_s]^\alpha}\cdot \gamma, j \in J_k \\ 0,\ others. \end{cases} \tag{1}$$

In the formula: $P_j^k(t)$ is the transition probability from individual k to weight interval j, i, j = 0, 1, . . . , 99; τ_j is the label value of the weight interval j; n_j is the information expectation heuristic parameter of weight interval j; J_k is the set of weight interval that individual k can select in the next step; α is the information elicitation factor; β is the expectation eliciting factor; γ is the ranking factor.

Step 3.3. Assign individual weight interval by random ring algorithm. The specific method is, first, sum the values of probabilities P_j (j = 0, 1, . . . , 99) in each interval of each weight factor to obtain the total probability P, and randomly generate a probability that falls between (0, P). The cycle traverses each interval and subtracts each time. The first interval with a probability less than 0 is the required interval.

Step 3.4. Search the weight interval of other weight factors by traversing in turn.

Step 4. Tag updates. In this paper, a model based on global information update is used to calculate and update the markers of each weight interval during each iteration. That is, after all the search is completed, the comprehensive evaluation index under the weight interval selected by each individual is calculated to find the optimal path. **Equation 2** is used to update the marks of the selected weight interval, and **Equation 3** is used to update the marks of other weight intervals.

$$\boldsymbol{\tau_i} = (\mathbf{1} - \boldsymbol{\rho})\boldsymbol{\tau_{i-1}} + \boldsymbol{L} \tag{2}$$

$$\boldsymbol{\tau_i} = (\mathbf{1} - \boldsymbol{\rho})\boldsymbol{\tau_{i-1}} \tag{3}$$

In the formula, $\boldsymbol{\tau_i}$ is the token increment of the weight interval of the i times; ρ is the information volatile factor; $\boldsymbol{L}$ is the comprehensive index value under the weight interval selected by the optimal individual.

Through the elite ant colony algorithm based on mixed parameters to determine the weight range of factors, the real-time trading popularity value of virtual cryptocurrency can be obtained after computing with the real-time market data of virtual cryptocurrency trading.

DTW-SSC Waveform Similarity Algorithm Based on Virtual Cryptocurrency Popularity Value System

Under the popularity value system of virtual cryptocurrency, anomalies of virtual cryptocurrency are defined in this paper as follows: when the popularity value of a currency deviates greatly from the overall popularity of the market, we believe that the currency may be abnormal, and we need to monitor it. Therefore, the anomaly detection problem of virtual cryptocurrency is then transformed into the question of whether the popularity value curve of a single virtual cryptocurrency is similar to that of the overall market index. Before measuring similarity, this paper first defines similarity.

In **Figure 3**, the paper considers that y1, y2, and y3 are similar in shape. Specifically, among the three curves, the paper considers that y2 and y3 are the two most similar (because y2 and y3 are the closest in distance).

Euclidean distance is one of the most widely used basic methods to measure the similarity of two sequences. Euclidean distance is a special case of Minkowski distance, which is used to measure the distance between numerical points and is widely used in many algorithms [40, 41]. For sequences of the same length, calculate the distance between each two points and sum them up. The smaller the distance, the more whole matching. The algorithm is shown in **Equations 4** and **5**. For sequences of different lengths, there are generally two processing methods:

Subsequence Matching

Find the part of long sequence that is most similar to short sequence. Let the sequence A be $[x_1, x_2, \ldots, x_n]$ and B be $[y_1, y_2, \ldots, y_n]$, where n > m. Scroll to calculate the distance between A and B.

$$\boldsymbol{\rho_1} = \sqrt{(\boldsymbol{x}_1 - \boldsymbol{y_1})^2 + (\boldsymbol{x}_2 - \boldsymbol{y_2})^2 + \ldots + (\boldsymbol{x_n} - \boldsymbol{y_n})^2} \tag{4}$$

$$\boldsymbol{\rho_2} = \sqrt{(\boldsymbol{x}_2 - \boldsymbol{y_1})^2 + (\boldsymbol{x}_3 - \boldsymbol{y_2})^2 + \ldots + (\boldsymbol{x_{n+1}} - \boldsymbol{y_n})^2} \tag{5}$$

Then find the minimum value of all ρ, and the index of sequence A corresponding to this distance is the part of A that is most similar to B.

Sliding Window

EK, KC et al. from Microsoft proposed that to reduce the algorithm complexity, the B sequence can be copied until it is as long as the A sequence [42]. The paper from Tianjin university points out three shortcomings of Euclidean distance in measuring time series similarity [43]: 1) it cannot distinguish shape similarity, 2) it cannot reflect the similarity of trend dynamic variation amplitude, and 3) the calculation based on point distance cannot reflect the difference of different analysis frequency, as shown in **Figure 4**.

The change trend of A and B is almost completely opposite, and the change trend of A and C is almost exactly the same. If Euclidean distance is used, then it follows that A and B are the most similar. In fact, the change is that A and C are similar. **Figure 5** is the same as mentioned previously. Normally, the closest thing we think of as y1 is y3. In fact, y3 is the result of y1 being shifted down. However, the Euclidean distance between y1 and y2 is 15,832, and the Euclidean distance between y1 and y3 is 15,876. The Euclidean distance calculated tells us that the nearest distance to y1 is y2.

Aiming at the defects of Euclidian distance, researchers began to use Pattern distance [44] to quantify similarity. First, the piecewise linear representation algorithm (PLR) was introduced to represent a sequence piecewise linear representation. The state of a sequence can be simply divided into three categories: up, down, and constant. We represent these three states as 1, −1, and 0. For the sequence in **Figure 6**, it is divided into K segments, and the slope of each segment is calculated. Positive slope means rising, negative slope means declining, and zero means unchanged. At this point, the sequence can be approximated as (−1, −1, 0, −1, 1, 0, 1, 1 . . .); combining the same adjacent patterns, we get (−1, 0, −1, 1, 0, 1 . . .) in the sequence.

As for the point segmentation of PLR algorithm, the method of bisecting is directly used. However, it can be seen from **Figure 6** that the third mode is represented as 0. In fact, the third mode is a peak that rises first and then declines, so the method of bisecting is not scientific. The bottom-up search method proposed by KEOGH E [45] solves this problem well.

Since we combine the same adjacent patterns, the pattern sequence we get must be 1, −1, 0 interval, and each pattern may span different time lengths. After merging the patterns, sequence S1 may have N patterns and S2 may have M patterns. Now we need to count them.

As shown in **Figure 4B**, we select BNB and AAVE, two virtual cryptocurrencies with similar prices and higher ranking, and express their price sequences as S1 and S2 after PLR:

$$\boldsymbol{S1} = \{(\boldsymbol{m}_{11}, \boldsymbol{t}_{11}), \ldots, (\boldsymbol{m}_{1N}, \boldsymbol{t}_{1N})\} \tag{6}$$

$$\boldsymbol{S2} = \{(\boldsymbol{m}_{21}, \boldsymbol{t}_{21}), \ldots, (\boldsymbol{m}_{2N}, \boldsymbol{t}_{2N})\} \tag{7}$$

Let S_{1i} and S_{2i} represent the ith and jth modes of S1 and S2, respectively, that is, $S_{1i} = (m_{1i}, t_{1i})$; t represents time, and no matter how the cutting is done, the final end point is the same, that is, $\boldsymbol{t}_{1n} = t_{2\mathrm{m}}$.

When the mode definition is completed, the distance can be calculated. The formula for the mode distance is

$$\boldsymbol{D} = |\boldsymbol{m}_{1i} - \boldsymbol{m}_{2i}| \tag{8}$$

Obviously, $D \in \{0, 1, 2\}$, the closer the distance is to 0, the more similar the pattern is, and the closer to 2, the less similar the pattern is. The sequence pattern distance can be obtained by adding all pattern distances:

$$\boldsymbol{D}_{S1,S2} = \sum_{i=1}^{k} |\boldsymbol{m}_{1i} - \boldsymbol{m}_{2i}| \tag{9}$$

Since each mode may span different time lengths, and the longer a mode lasts, the more information it contains in the whole sequence, we improved the aforementioned formula, and the improved mode distance formula is as follows:

$$\boldsymbol{D}_{S1,S2} = \sum_{i=1}^{k} \boldsymbol{t}_{wi} * |\boldsymbol{m}_{1i} - \boldsymbol{m}_{2i}| \tag{10}$$

where $t_{wi} = \frac{t_i}{t_N}$, t_i is the length of time that the ith mode spans, and t_N is the total length of time.

Based on the pattern distance, Dong et al. proposed the shape distance [43], which further improved the measurement effect. In simple terms, the shape distance is based on the mode distance, adding an amplitude change and resetting the mode sequence. Suppose we have obtained the equal-patterned sequence:

$$\boldsymbol{S1} = \{(\boldsymbol{1}, \boldsymbol{t}_{11}), (\boldsymbol{-1}, \boldsymbol{t}_{12}), (\boldsymbol{-1}, \boldsymbol{t}_{13}), (\boldsymbol{0}, \boldsymbol{t}_{14})\} \tag{11}$$

$$\boldsymbol{S2} = \{(\boldsymbol{1}, \boldsymbol{t}_{21}), (\boldsymbol{-1}, \boldsymbol{t}_{22}), (\boldsymbol{-1}, \boldsymbol{t}_{23}), (\boldsymbol{0}, \boldsymbol{t}_{24})\} \tag{12}$$

Set amplitude variation sequence as A, then: $A_i = y_i - y_{i-1}$, which is the difference of the sequence value corresponding to the endpoint of each split interval, then we can get

$$\boldsymbol{A1} = \{(\boldsymbol{\Delta y}_{11}, \boldsymbol{t}_1), (\boldsymbol{\Delta y}_{12}, \boldsymbol{t}_2), (\boldsymbol{\Delta y}_{13}, \boldsymbol{t}_{13}), (\boldsymbol{\Delta y}_{14}, \boldsymbol{t}_4)\} \tag{13}$$

$$\boldsymbol{A2} = \{(\boldsymbol{\Delta y}_{21}, \boldsymbol{t}_1), (\boldsymbol{\Delta y}_{22}, \boldsymbol{t}_2), (\boldsymbol{\Delta y}_{23}, \boldsymbol{t}_3), (\boldsymbol{\Delta y}_{24}, \boldsymbol{t}_4)\} \tag{14}$$

The calculation formula of shape distance is

$$\boldsymbol{D}_{S1,S2} = \sum_{i=1}^{k} \boldsymbol{t}_{wi} * |\boldsymbol{m}_{1i} - \boldsymbol{m}_{2i}| * |\boldsymbol{A}_{1i} - \boldsymbol{A}_{2i}| \tag{15}$$

In this case, t_{wi} is the time weight.

Taking virtual cryptocurrencies as an example, we believe that the price trend of virtual cryptocurrencies usually has seven states: {accelerating down, horizontal down, decelerating down, stable, decelerating up, horizontal up, accelerating up}, which is described by the model M = −3, −2, −1, 0, 1, 2, 3). A threshold value th is set to distinguish the 7 states. Let $\boldsymbol{K_i}$ represent the slope of the K segment straight line after being divided by PLR. In the mode distance, if the slope is less than 0, it represents decline, denoted as −1. If the slope is greater than 0, it means that the slope is rising, which is 1. If the slope is 0, it does not change, so let us call it 0.

Shape distance is improved on mode distance to be more complex and more realistic because more modes are introduced (7) (**Table 2**).

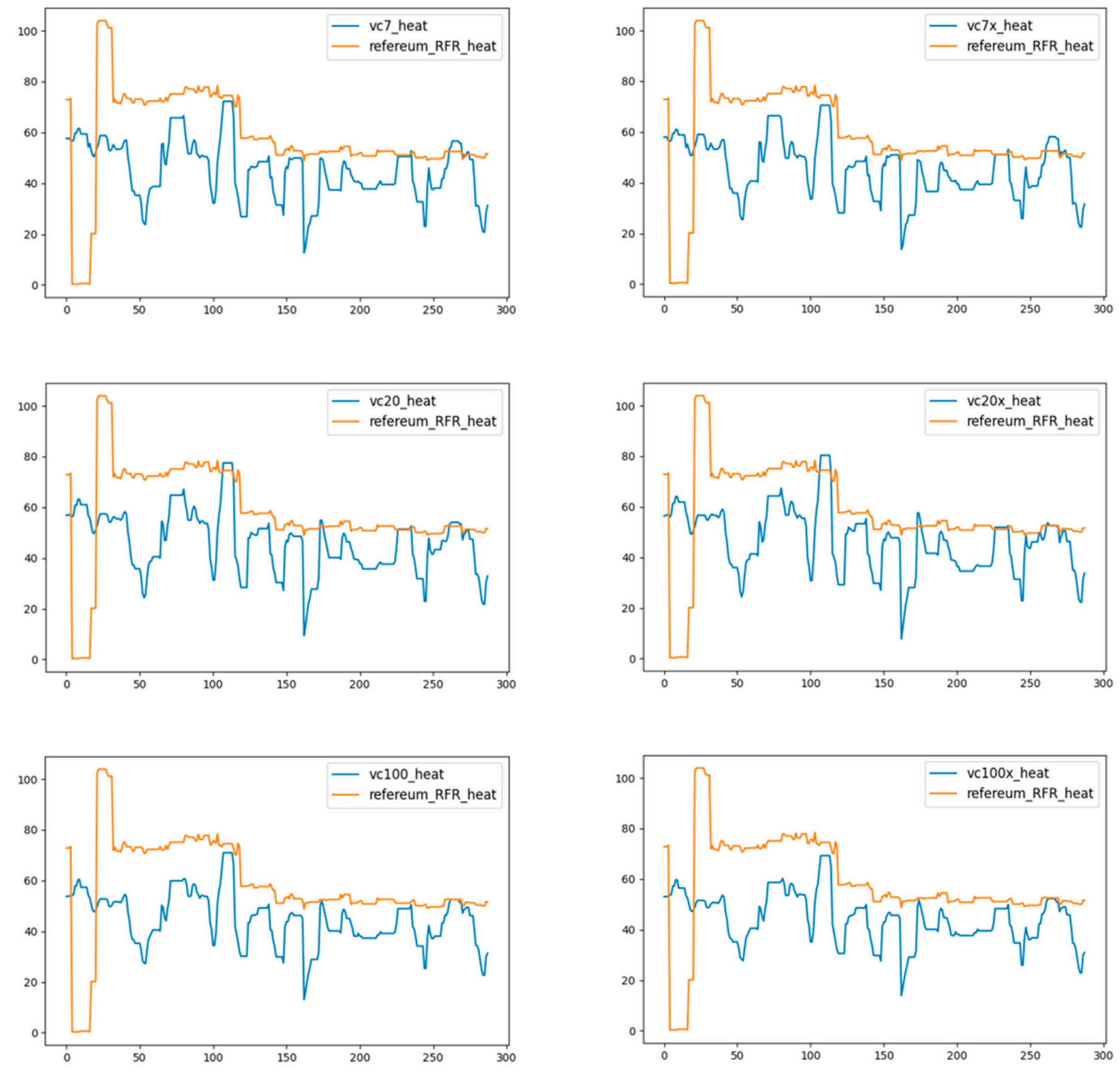
When $K_i < -th$, it belongs to one of (−3, −2, −1), and the change in slope is used to indicate whether the virtual cryptocurrency is falling at an accelerated, horizontal, or decelerated rate. If $\Delta k < 0$, it indicates that the slope is decreasing, the line is steeper, it is accelerating, set to −3 mode, if $\Delta k = 0$. If it is falling horizontally, set it to −2 mode; if $\Delta k > 0$, it is decelerating and set it to −1 mode. When $-\mathrm{th} < K_{i+1} < th$, it is considered to be approximately stable and set to 0 mode. When $K_i > -th$, and so on. The formula of shape distance also satisfies the following four distance theorems:

$$\begin{cases} \boldsymbol{D}(\boldsymbol{A}, \boldsymbol{B}) = \boldsymbol{D}(\boldsymbol{B}, \boldsymbol{A}); \\ \boldsymbol{D}(\boldsymbol{A}, \boldsymbol{B}) > \boldsymbol{0}; \\ \boldsymbol{D}(\boldsymbol{A}, \boldsymbol{A}) = \boldsymbol{0}; \\ \boldsymbol{D}(\boldsymbol{A}, \boldsymbol{B}) = \boldsymbol{0} \rightarrow \boldsymbol{A} = \boldsymbol{B}; \end{cases} \tag{16}$$

In addition, to improve the accuracy of the model and reduce the impact of absolute value differences on the whole model, it is generally necessary to standardize the original sequence. Shape distance is also more accurate because of the addition of more patterns, and the effect is better than pattern distance.

In addition to the distance method mentioned previously, there is another very intuitive indicator, correlation coefficient, which can measure not only correlation but also similarity. However, correlation coefficient is generally not used to measure similarity in time series because it cannot solve the problem of graph translation. **Figure 8** clearly illustrates this problem.

DTW (Dynamic Time Warping) [46–50] solves this problem. DTW is actually calculating Euclidean distance. As mentioned before, Euclidean distance cannot measure shape similarity well, and all points are directly corresponding according to general Euclidean distance calculation method. DTW is to find the correct corresponding points between the sequences and calculate their distance, as shown in **Figure 5A**. The solid line and the dashed line are the two speech waveforms of the same word "pen" respectively (separated on the Y-axis for observation). You can see that the overall waveform shape is very similar, but the time line is not aligned. For example, at the 20th time point, point A of the solid line waveform will correspond to point B′ of the dotted line waveform, so the traditional calculation of similarity by comparing distances is obviously unreliable because it is clear that point A on the solid line corresponds to point B on the dotted line. In **Figure 5B**, DTW can calculate their distance correctly by finding the point where the two waveforms are aligned. The fundamental task of DTW is to match points correctly. The criterion of DTW correct correspondence is that if the points of two sequences are correctly corresponding, their Euclidean distance is minimized. One point of a sequence may correspond to multiple points of another sequence. If all



	VC7	VC20	VC100	VC7X	VC20X	VC100X
RFR	5594.21	5075.03	4882.39	5180.75	4961.55	4815.84

FIGURE 13 | August 3 abnormal currency RFR popularity value and six index comparisons.

possible points are enumerated to find the most suitable point, it is an NP difficult problem in terms of time complexity. At this time, dynamic programming is introduced to optimize it to solve this problem.

Simulate the market sequence of two virtual cryptocurrencies VC1 and VC2 with different lengths, as shown in **Figure 10**.

Compute the distance matrix of two sequences. The horizontal axis represents the VC1 sequence, and the vertical axis is the VC2 sequence. The visual representation is shown in **Figure 11A**.

In the visual distance matrix, the darker the color is, the farther the distance is. Calculate a cumulative distance matrix, as shown in **Figure 11B**, and minimize the cumulative distance as the ultimate goal. Obviously row 0, column 0 of the cumulative distance matrix is equal to row 0, and column 0 of the distance matrix is equal to 1. If you keep going to the right, then the cumulative distance matrix changes as shown in **Figure 11C**. If you keep going up, then it will look as shown in **Figure 11D** because the path only moves in three directions: right, up, and slant to the upper right. If the path moves more than one square to the right or up, this indicates that one point in one sequence corresponds to multiple points in another sequence. So far, we have computed two columns of the cumulative distance matrix. For any other point, the increase in cumulative distance can come from only three directions: left, down, and slant down left. Dynamic programming is to select the direction that minimizes the current travel distance for each step forward. Therefore, for any other point, the cumulative distance can be calculated by the following **Eq. 17**:

$$\boldsymbol{AccmulatedCost}(\boldsymbol{i},\boldsymbol{j}) = \boldsymbol{Min}\{\boldsymbol{D}(\boldsymbol{i}-\mathbf{1},\boldsymbol{j}-\mathbf{1}),\boldsymbol{D}(\boldsymbol{i}-\mathbf{1},\boldsymbol{j}), \boldsymbol{D}(\boldsymbol{i},\boldsymbol{j}-\mathbf{1})\} + \boldsymbol{distance}(\boldsymbol{i},\boldsymbol{j}) \quad (17)$$

Therefore, the complete cumulative distance matrix is shown in **Figure 6E**. The best path has been clearly displayed in the cumulative distance matrix, which is the square with the lightest color in the figure. We only need to find the best path by backtracking. The best route is the [(5, 6), (4, 5), (3, 4), (2, 3), (1, 2), (1, 1), (0, 1), (0, 0)]. The DTW algorithm calculates the Euclidean distance of these points as a measure of similarity.

DTW has been widely used in the field of speech recognition due to its outstanding characteristics. Because of a syllable, it may be very long or very short; how to correctly recognize similar sounds or syllables is very important for speech recognition. By the same token, the virtual cryptocurrency in the virtual cryptocurrency market analysis is popular, and because each virtual cryptocurrency differs, the reaction time to the market changes is different also; at this point, the excellent quality of DTW (dynamic time warping) makes up for the defects, using DTW-SSC algorithm based on virtual cryptocurrency popularity value system, through extension and shortening of the time sequence. The similarity between the two popularity value sequences is calculated by using the constructed virtual cryptocurrency popularity value system. To monitor the virtual cryptocurrency market is abnormal. The DTW-SSC algorithm based on the popularity value system of virtual cryptocurrency can also actively discover the non-similar sub-sequences in the sequence. Since the characteristics of the sequence in the long sequence data such as trading quotation data or speech sequence are represented by the whole sequence, the similarity of the sequence must be investigated from the whole similarity. Even if two sequences have very similar local sub-sequences, when the overall similarity level is low, the two sequences are still considered to be dissimilar. When studying the active discovery of virtual cryptocurrency anomalies in a certain period of time, it is necessary to automatically discover some non-similar sub-sequences. The key is to identify the abnormal sub-patterns representing the same sequence as the anomaly discovery region. At this point, it is necessary not only to judge the overall sequence similarity but also to mine the abnormal sub-pattern to judge whether it conforms to the abnormal characteristics of virtual cryptocurrency. Therefore, the DTW-SSC algorithm based on the popularity value system of virtual cryptocurrency is of great significance for the early warning of abnormal virtual cryptocurrency.

This paper first calculates the popularity value of the whole currency of the public virtual cryptocurrency, generates the popularity value series of the whole currency and the index series of the virtual cryptocurrency popularity value system, and calculates the quantitative similarity S of the two sequences using the DTW-SSC algorithm based on the virtual cryptocurrency popularity value system. To judge whether the popularity value sequence of a single currency virtual cryptocurrency is similar to the corresponding exponential popularity value sequence, the smaller the quantitative similarity S is, the more similar the two sequences are; the larger the quantitative similarity S is, the more dissimilar the two sequences are. At this point, we believe that if the sequences are not similar, the virtual cryptocurrency is abnormal.

EXPERIMENTAL DESIGN AND RESULT

Experimental Design

All the experiments in this paper were run on a PC with a CPU of 3 GHz Inter Core I9-10980XE, GPU of NVIDIA GeForce RTX 3090 with 24 GB video memory, operating system of Windows10, and memory size of 32 GB. The algorithm is implemented in Python. The experimental data use the virtual cryptocurrency quotation data from 0:00 to 24:00 on August 3, 2021.

The data in this paper are from the Internet, including Coinbase, Tokenview, FeiXiaoHao, and other virtual cryptocurrency market data sites as well as Kaggle, Tushare, XBlock, and other big data science sites. The data dimension mainly includes the quotation data of virtual cryptocurrency, block data on the chain of virtual cryptocurrency, known abnormal transaction data, known abnormal transaction address, open smart contract data, public news of virtual cryptocurrency, etc. Each dimension can be subdivided again; for example, the quotation data of virtual cryptocurrency can be subdivided into price, rise and fall, trading volume, and other dimensions. The cumulative size of all experimental data exceeds 1 TB.

Experimental Result

In this article, we put forward the virtual cryptocurrency system of popularity value, and according to a study in the system with virtual cryptocurrency popularity value anomaly detection algorithm, by building the system, virtual cryptocurrency can be seen clearly in the popularity of the global, and can judge the development trend of virtual cryptocurrency, according to the system. A new abnormal research method for virtual cryptocurrency is innovatively proposed. Through the popularity value system of virtual cryptocurrency, the popularity value of each currency and the popularity value index of each virtual cryptocurrency are calculated, and the similarity comparison algorithm is used to warn abnormal virtual cryptocurrency. The index chart of the virtual cryptocurrency popularity value system on August 3 is shown in **Figure 7**.

To verify the guiding effect of the virtual cryptocurrency popularity value index on the trend of the virtual cryptocurrency popularity value, three groups of normal virtual cryptocurrency and one group of abnormal virtual cryptocurrency with different ranking locations were selected for the experiment.

According to the experimental results of virtual cryptocurrency popularity value calculation in recent 3 months, the definition of similarity degree is given through cluster analysis and error estimation of quantified similarity value. The similarity value between 0 and 1,000 is considered High Similarity. The similarity value between 1,000 and 2,000 is considered More Similarity. The similarity value between 2,000 and 3,500 is considered Similarity. The similarity value between 3,500 and 4,500 is considered Low Similarity. Also, the similarity value greater than 4,500 is considered No Similarity.

First of all, among the top seven virtual cryptocurrencies in the total market value of the virtual cryptocurrency market, three representative virtual cryptocurrencies, BTC, XRP, and DOGE, were selected as a group for the experiment, and the popularity value sequence of these three virtual cryptocurrencies was compared with three indexes VC7, VC20, and VC100. According to the DTW-SSC algorithm based on the popularity value system of virtual cryptocurrencies, the quantitative similarity S of the popularity value sequence of BTC, ETH, and DOGE and the three indexes is calculated: The corresponding trend and the quantitative similarity S are shown in **Figure 9**.

It can be clearly seen from **Figure 9** that the popularity values of BTC, XRP, and DOGE were compared with the three indices, and the overall trend of change was consistent. Then the similarity is quantitatively analyzed.

According to the analysis of quantitative Similarity results, it can be found that except the quantitative Similarity of BTC and VC7, the other eight quantitative Similarity indexes are all lower than 1,000, belonging to High Similarity, while BTC and VC7 belong to More Similarity. It indicates that the prices of BTC, XRP, and DOGE were relatively stable on August 3. BTC may have short-term fluctuations different from VC7, but it still conforms to the general fluctuation trend. By observing XRP and DOGE, it can be found that the quantitative similarity values of XRP and VC7, VC20, and VC100 increase in turn, and the similarity degree decreases in turn, which also conforms to the standard of the index.

We select MATIC, LTC, and WBTC as a group of virtual cryptocurrencies ranked from 8 to 20 in the total market value of the virtual cryptocurrency market, and compare the popularity value sequence of these three virtual cryptocurrencies with the four indices VC7, VC20, VC100, and VC20X. According to the DTW-SSC algorithm based on the popularity value system of virtual cryptocurrency, the sequence of the popularity value of MATIC, LTC, and WBTC and the quantitative similarity S of the three indexes were calculated respectively. The corresponding trend and the quantitative similarity S are shown in **Figure 10**.

It can also be clearly seen from **Figure 10** that the popularity values of MATIC, LTC, and WBTC are compared with the four indexes, and the overall change trend is consistent, and then the similarity is quantitatively analyzed.

According to the analysis of quantitative similarity results, we found that the quantitative similarity values of MATIC, LTC, and WBTC with VC7 were the highest in the comparison of the four indexes, indicating that their similarity with VC7 was low. The quantitative similarity values with VC20 were the lowest in the comparison of the four indexes, indicating that their similarity with VC20 was the highest. According to the definition of VC7 index and VC20 index, the experimental results are reasonable and all quantitative Similarity index values conform to High Similarity, indicating that the prices of MATIC, LTC, and WBTC were very stable on August 3, in line with the general fluctuation trend.

Among the virtual cryptocurrencies ranked from 21 to 100 in the total market capitalization of the virtual cryptocurrency market, we selected three representative virtual cryptocurrencies, FTT, MKR, and SNX, as a group, and compared the popularity value sequence of these three virtual cryptocurrencies with four indexes, VC7, VC20, VC100, and VC100X. According to the DTW-SSC algorithm based on the popularity value system of virtual cryptocurrencies, the quantitative similarity S of the popularity value sequence of BTC, ETH, and DOGE and the three indexes is calculated. The corresponding trend and the quantitative similarity S are shown in **Figure 11**.

It can be clearly seen from **Figure 11** that the popularity values of FTT, MKR, and SNX are compared with the four indexes, and the overall trend of change is consistent. Then, the similarity is quantitatively analyzed.

According to the analysis of quantitative Similarity results, it is found that the quantitative Similarity values of FTT, MKR, and SNX and the four indices decrease successively in the table. According to the index formulation rules, the experimental results are reasonable, and all the quantitative Similarity index values conform to High Similarity. This shows that MATIC, LTC, and WBTC prices were very stable on August 3, in line with the general trend of volatility.

We randomly selected ZEN, NANO, and WRX as a group of virtual cryptocurrencies outside the top 100 in the total market capitalization of the virtual cryptocurrency market, and compared the popularity value sequence of these three virtual cryptocurrencies with the six indexes. According to the DTW-SSC algorithm based on the popularity value system of virtual cryptocurrencies, the quantitative similarity S of ZEN, NANO, and WRX popularity value sequences and the three indexes is calculated. The corresponding trend and the quantitative similarity S are shown in **Figure 12**.

It can be clearly seen from **Figure 12** that when ZEN, NANO, and WRX popularity values are compared with the six indexes, there are some differences in the overall variation trend, but the amplitude is small and there is no significant difference. The overall trend is roughly consistent, and then the similarity is quantitatively analyzed.

According to the analysis of quantitative Similarity results, we find that the quantitative Similarity values of ZEN, NANO, and WRX and the six indexes are relatively High, among which four conform to High Similarity, all of which are from NANO. The quantitative Similarity between NANO and VC7 and VC7X indexes is Low Similarity. According to the experimental results and index compilation rules, NANO basically conforms to the general trend of fluctuation, while the experimental results of ZEN and WRX show that the quantitative Similarity between them and the six indexes is Low Similarity. It indicates that ZEN and WRX had a certain difference with the market on August 3, but the difference level is not unusual.

By observing the activities of the virtual cryptocurrency market on August 3, this paper finds out a virtual cryptocurrency that is significantly believed to be manipulated for experiment. We find RFR as an abnormal virtual cryptocurrency and compare its popularity value sequence with six indexes. According to the DTW-SSC algorithm based on the popularity value system of virtual cryptocurrency, the quantitative similarity S of RFR popularity value sequence and six indexes is calculated. The corresponding trend and the quantitative similarity S are shown in **Figure 13**.

It can be clearly seen from **Figure 13** that the overall variation trend of RFR popularity value is significantly different from that of the six indices, and then the similarity is quantitatively analyzed.

According to the analysis of quantitative Similarity results, we found that the quantitative Similarity values of abnormal virtual cryptocurrency RFR and the six indexes belong to No Similarity. Experimental results proved that the virtual cryptocurrency was abnormal on August 3.

CONCLUSION

The virtual cryptocurrency transaction popularity value system has a broad application prospect in the research of virtual cryptocurrency. At present, there is an urgent problem of abnormal transaction warning of virtual cryptocurrency in practical applications such as virtual cryptocurrency transaction, which has a great impact on the risk of virtual cryptocurrency transaction. On the basis of summarizing the characteristics and laws of current virtual cryptocurrency transactions, this paper conducts quantitative analysis on virtual cryptocurrency transactions, and studies and constructs the virtual cryptocurrency transaction popularity value system covering multiple currencies. Moreover, the paper specifically studies the elite ant colony algorithm based on mixed parameters to calculate the popularity value of virtual cryptocurrency trading and constructs the virtual cryptocurrency trading popularity value index system on this basis. Taking the popularity value of virtual cryptocurrency transaction as an indicator, combined with the DTW-SSC waveform similarity algorithm of the popularity value system of virtual cryptocurrency transaction, the paper detects whether there are external factors leading to abnormal transaction of global virtual cryptocurrency, and realizes real-time warning of abnormal transaction of global virtual cryptocurrency. Experimental results show that our proposed early-warning method has great application potential in the field of virtual cryptocurrency.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

AUTHOR CONTRIBUTIONS

TZ and CL contributed to conception and design of the study. ZZ organized the database. CL performed the statistical analysis. XL and QZ wrote sections of the article. All authors contributed to article revision, read, and approved the submitted version.

REFERENCES

1. He S. Exploration of Gold Standard, Credit Currency and Digital Currency. *Finan Mar* (2020) 5(4):316. doi:10.18686/fm.v5i4.2962
2. Rosenfeld R, Lakatos A, Beam D, Carlson J, Flax N, Niehoff P, et al. Commodity Futures Trading Commission Issues Advisory for Virtual Currency Pump-And-Dump Schemes. *J Investment Compliance* (2018) 19: 42. doi:10.1108/JOIC-04-2018-0033
3. Guindy MA. Cryptocurrency price Volatility and Investor Attention. *Int Rev Econ Finance* (2021) 76(1):556–70. doi:10.1016/j.iref.2021.06.007
4. Putra P, Jayadi R, Steven I. The Impact of Quality and Price on the Loyalty of Electronic Money Users: Empirical Evidence from Indonesia. *J Asian Finance Econ Business* (2021) 8(3):1349–59. doi:10.13106/jafeb.2021.vol8.no3.1349
5. Liu S. Recognition of Virtual Currency Sales Revenue of Online Game Companies. *Acad J Humanities Soc Sci* (2019) 2(7):29–36. doi:10.25236/AJHSS.2019.020704
6. Guo H, Hao L, Mukhopadhyay T, Sun D. Selling Virtual Currency in Digital Games: Implications for Gameplay and Social Welfare. *Inf Syst Res* (2019) 30(2):430–46. doi:10.1287/isre.2018.0812
7. Liiv I. *Data Science Techniques for Cryptocurrency Blockchains*, Vol. 9. Singapore: Springer Nature (2021). doi:10.1080/17538963.2020.1870282
8. Tong W, Jiayou C. A Study of the Economic Impact of central Bank Digital Currency under Global Competition. *China Econ J* (2021) 14(12):1–24. doi:10.1080/17538963.2020.1870282
9. Davoodalhosseini SM. Central Bank Digital Currency and Monetary Policy. *J Econ Dyn Control* (2021) 104150. doi:10.1016/j.jedc.2021.104150
10. Xu Z, Tang C. Challenges and Opportunities in the Application of China's Central Bank Digital Currency to the Payment and Settle Account System. *Fin For* (2021) 9(4):233. doi:10.18282/ff.v9i4.1553
11. Dahlberg T. What Blockchain Developers and Users Expect from Virtual Currency Regulations: A Survey Study. *IP* (2019) 24(4):453–67. doi:10.3233/ip-190145
12. Brühl V. Libra - A Differentiated View on Facebook's Virtual Currency Project. *Intereconomics* (2020) 55(1):54–61. doi:10.1007/s10272-020-0869-1
13. Derlyatka A, Fomenko O, Eck F, Khmelev E, Elliott MT. Bright Spots, Physical Activity Investments that Work: Sweatcoin: a Steps Generated Virtual Currency for Sustained Physical Activity Behaviour Change. *Br J Sports Med* (2019) 53(18):1195–6. doi:10.1136/bjsports-2018-099739
14. Yousuf Javed M, Husain R, Khan BM, Azam MK. Crypto-currency: Is the Future Dark or Bright? *J Inf Optimization Sci* (2019) 40(5):1081–95. doi:10.1080/02522667.2019.1641894
15. Guesmi K, Saadi S, Abid I, Ftiti Z. Portfolio Diversification with Virtual Currency: Evidence from Bitcoin. *Int Rev Financial Anal* (2019) 63:431–7. doi:10.1016/j.irfa.2018.03.004
16. Bashilov B, Galkina M, Berman A. Digital Financial Assets and Digital Currency: Legal Nature and Legal Regulation of Turnover. In: SHS Web of Conferences; 15–16.4.2021; Yekaterinburg, Russia, Vol. 106. EDP Sciences (2021). doi:10.1051/shsconf/202110602005
17. Miseviciute J. Blockchain and Virtual Currency Regulation in the EU. *J Investment Compliance* (2018) 19:33–38. doi:10.1108/joic-04-2018-0026
18. Wu C. Window Effect with Markov-Switching GARCH Model in Cryptocurrency Market. *Chaos, Solitons & Fractals* (2021) 146:110902. doi:10.1016/j.chaos.2021.110902
19. Zhang Z, Dai H-N, Zhou J, Mondal SK, García MM, Wang H. Forecasting Cryptocurrency Price Using Convolutional Neural Networks with Weighted and Attentive Memory Channels. *Expert Syst Appl* (2021) 183:115378. doi:10.1016/j.eswa.2021.115378
20. Kim J-M, Jun C, Lee J. Forecasting the Volatility of the Cryptocurrency Market by GARCH and Stochastic Volatility. *Mathematics* (2021) 9(14):1614. doi:10.3390/math9141614
21. Huang ZJ, James Huang Z, Xu L. Sequential Learning of Cryptocurrency Volatility Dynamics: Evidence Based on a Stochastic Volatility Model with Jumps in Returns and Volatility. *Q J Finance* (2021) 11:2150010. doi:10.1142/s2010139221500105

22. Lahmiri S, Bekiros S. Deep Learning Forecasting in Cryptocurrency High-Frequency Trading. *Cogn Comput* (2021) 13(2):485–7. doi:10.1007/s12559-021-09841-w
23. Rahmani Cherati M, Haeri A, Ghannadpour SF. Cryptocurrency Direction Forecasting Using Deep Learning Algorithms. *J Stat Comput Simulation* (2021) 91:1–15. doi:10.1080/00949655.2021.1899179
24. Kakinaka S, Umeno K. Exploring Asymmetric Multifractal Cross-Correlations of price-volatility and Asymmetric Volatility Dynamics in Cryptocurrency Markets. *Phys A Stat Mech Appl* (2021) 581:126237. doi:10.1016/j.physa.2021.126237
25. Aljinović Z, Marasović B, Šestanović T. Cryptocurrency Portfolio Selection—A Multicriteria Approach. *Mathematics* (2021) 9(14):1677. doi:10.3390/math9141677
26. Kądziołka K. The Promethee II Method in Multi-Criteria Evaluation of Cryptocurrency Exchanges. *Econ Reg Studies/Studia Ekonomiczne i Regionalne* (2021) 14(2):131–45. doi:10.22004/ag.econ.313138
27. Li S, Li Y, Han W, Du X, Guizani M, Tian Z. Malicious Mining Code Detection Based on Ensemble Learning in Cloud Computing Environment. *Simul Model Pract Theor* (2021) 113:102391. doi:10.1016/j.simpat.2021.102391
28. Li S, Zhang Q, Wu X, Han W, Tian Z. Attribution Classification Method of APT Malware in IoT Using Machine Learning Techniques. *Security Commun Netw* (2021) 2021:1–12. doi:10.1155/2021/9396141
29. Yang H, Li S, Wu X, Lu H, Han W. A Novel Solutions for Malicious Code Detection and Family Clustering Based on Machine Learning. *IEEE Access* (2019) 7:148853–60. doi:10.1109/access.2019.2946482
30. Shen X-L, Zhang KZK, Zhao SJ. Herd Behavior in Consumers' Adoption of Online Reviews. *J Assn Inf Sci Tec* (2016) 67(11):2754–65. doi:10.1002/asi.23602
31. Romero DM, Reinecke K, Robert LP. The Influence of Early Respondents: Information cascade Effects in Online Event Scheduling. In: Proceedings of the Tenth ACM International Conference on Web Search and Data Mining; 6–10.2.2017; New York, NY (2017). p. 101–10.
32. Borst I, Moser C, Ferguson J. From Friendfunding to Crowdfunding: Relevance of Relationships, Social media, and Platform Activities to Crowdfunding Performance. *New Media Soc* (2018) 20(4):1396–414. doi:10.1177/1461444817694599
33. Gao C, Liu J. Network-based Modeling for Characterizing Human Collective Behaviors during Extreme Events. *IEEE Trans Syst Man Cybernetics: Syst* (2016) 47(1):171–83. doi:10.1109/TSMC.2016.2608658
34. Kyriazis NA. Herding Behaviour in Digital Currency Markets: An Integrated Survey and Empirical Estimation. *Heliyon* (2020) 6(8):e04752. doi:10.1016/j.heliyon.2020.e04752
35. King T, Koutmos D. Herding and Feedback Trading in Cryptocurrency Markets. *Ann Oper Res* (2021) 300(1):79–96. doi:10.1007/s10479-020-03874-4
36. Akyildirim E, Aysan AF, Cepni O, Darendeli SPC. Do investor Sentiments Drive Cryptocurrency Prices? *Econ Lett* (2021) 206:109980. doi:10.1016/j.econlet.2021.109980
37. Yang H, Hu J, Huai X, Ma F, Ma Z Landslide Warning Method Based on Local Anomaly Coefficient of Multivariable Time Series. *South North Water Transfer Water Sci Techn* (2021) 19(6):1227–37. doi:10.13476/j.cnki.nsbdqk.2021.0125
38. Brière M, Oosterlinck K, Szafarz A. Virtual Currency, Tangible Return: Portfolio Diversification with Bitcoin. *J Asset Manag* (2015) 16(6):365–73. doi:10.1057/jam.2015.5
39. Liu Y, Gao C, Zhang Z, Lu Y, Chen S, Liang M, et al. Solving NP-Hard Problems with Physarum-Based Ant colony System. *IEEE/ACM Trans Comput Biol Bioinf* (2017) 14(1):108–20. doi:10.1109/tcbb.2015.2462349
40. Wang D, Gao X, Wang X. Semi-supervised Nonnegative Matrix Factorization via Constraint Propagation. *IEEE Trans Cybern* (2016) 46(1):233–44. doi:10.1109/TCYB.2015.2399533
41. Bronstein MM, Bruna J, LeCun Y, Szlam A, Vandergheynst P. Geometric Deep Learning: Going beyond Euclidean Data. *IEEE Signal Process Mag* (2017) 34(4):18–42. doi:10.1109/msp.2017.2693418
42. Keogh E, Chakrabarti K, Pazzani M, Mehrotra S. Dimensionality Reduction for Fast Similarity Search in Large Time Series Databases. *Knowledge Inf Syst* (2001) 3(3):263–86. doi:10.1007/pl00011669
43. Dong XL, Gu CK, Wang ZO. Research on Shape-Based Time Series Similarity Measure. In: 2006 International Conference on Machine Learning and Cybernetics; 13–16.8.2006; Dalian, China. IEEE (2006). p. 1253–8. doi:10.1109/icmlc.2006.258648
44. Jonoska N, Nabergall L, Saito M. Patterns and Distances in Words Related to DNA Rearrangement. *Fundam Inform* (2017) 154(1-4):225–38. doi:10.3233/fi-2017-1563
45. Keogh E. Fast Similarity Search in the Presence of Longitudinal Scaling in Time Series Databases. In: Proceedings Ninth IEEE International Conference on Tools with Artificial Intelligence; 3–8.11.1997; Newport Beach, CA. IEEE (1997). p. 578–84.
46. Kim S, Lee H, Ko H, Jeong S, Byun H, Oh K. Pattern Matching Trading System Based on the Dynamic Time Warping Algorithm. *Sustainability* (2018) 10(12):4641. doi:10.3390/su10124641
47. Shokoohi-Yekta M, Hu B, Jin H, Wang J, Keogh E. Generalizing DTW to the Multi-Dimensional Case Requires an Adaptive Approach. *Data Min Knowl Disc* (2017) 31(1):1–31. doi:10.1007/s10618-016-0455-0
48. Sharma A, Sundaram S. An Enhanced Contextual DTW Based System for Online Signature Verification Using Vector Quantization. *Pattern Recognition Lett* (2016) 84:22–8. doi:10.1016/j.patrec.2016.07.015
49. Sharma A, Sundaram S. On the Exploration of Information from the DTW Cost Matrix for Online Signature Verification. *IEEE Trans Cybern* (2018) 48(2):611–24. doi:10.1109/TCYB.2017.2647826
50. Shah M, Grabocka J, Schilling N, Wistuba M, Schmidt-Thieme L. Learning DTW-Shapelets for Time-Series Classification. In: Proceedings of the 3rd IKDD Conference on Data Science, 2016; 13.3.2016; New York, NY (2016). p. 1–8.

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers
in Physics

ORIGINAL RESEARCH
published: 04 February 2022
doi: 10.3389/fphy.2022.827929

Linking the Pattern Structures to System Robustness Based on Dynamical Models and Statistical Method

Gui-Quan Sun[1,2]*, Yizhi Pang[2], Li Li[3], Chen Liu[4]*, Yongping Wu[5], Guolin Feng[5,6], Zhen Jin[2], Bai-Lian Li[7] and Zhen Wang[8]*

[1]Department of Mathematics, North University of China, Taiyuan, China, [2]Complex Systems Research Center, Shanxi University, Taiyuan, China, [3]School of Computer and Information Technology, Shanxi University, Taiyuan, China, [4]Center for Ecology and Environmental Sciences, Northwestern Polytechnical University, Xi'an, China, [5]College of Physics Science and Technology, Yangzhou University, Yangzhou, China, [6]Laboratory for Climate Studies, National Climate Center, China Meteorological Administration, Beijing, China, [7]Ecological Complexity and Modeling Laboratory, Department of Botany and Plant Sciences, University of California, Riverside, CA, United States, [8]School of Mechanical Engineering and Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an, China

OPEN ACCESS

Edited by:
Zhanwei Du,
University of Texas at Austin,
United States

Reviewed by:
Sanling Yuan,
University of Shanghai for Science and Technology, China
Wei Duan,
Southeast University, China

****Correspondence:***
Gui-Quan Sun
gquansun@126.com
Chen Liu
liuc317@neneu.edu.cn
Zhen Wang
zhenwang0@gmail.com

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *02 December 2021*
Accepted: *07 January 2022*
Published: *04 February 2022*

Citation:
Sun G-Q, Pang Y, Li L, Liu C, Wu Y, Feng G, Jin Z, Li B-L and Wang Z (2022) Linking the Pattern Structures to System Robustness Based on Dynamical Models and Statistical Method. Front. Phys. 10:827929. doi: 10.3389/fphy.2022.827929

Pattern structures are usually used to describe the spatial and temporal distribution characteristics of individuals. However, the corresponding relationship between the pattern structure and system robustness is not well understood. In this work, we use geostatistical method–semivariogram to study system robustness for different pattern structures based on three dynamical models in different fields. The results show that the structural ratio of different pattern structures including the mixed state of spot and stripe, cold spot, stripe only, and hot spot are more than 75%, which indicated those patterns all have strong spatial dependence and heterogeneity. It was revealed that the systems corresponding to the mixed state of spot and stripe or cold spot are more robust. This article proposed a method to characterize the robustness of the system corresponding to the pattern structure and also provided a feasible approach for the study of "how structures determine their functions."

Keywords: pattern structure, system robustness, structural ratio, dynamical model, data analysis

1 INTRODUCTION

Due to some behavior mechanisms of individuals, species present heterogeneous but regular spatial distribution structures in both space and time, which is called as "pattern." These pattern structures exist widely in nature such as the clouds in the sky [1], the patterns on zebra [2], and the ripples on the water [3]. Except for these, Getzin et al. found the gap vegetation pattern—fairy circles in Western Australia [4], the regular stripes vegetation distribution on the hillside of Niger studied by Klausmeier et al. [5], and mussel beds in the intertidal zone show different scales of distribution, namely, large-scale banded distribution at the ecological level and small-scale reticular distribution at individual mussels level [6, 7]. There are also thermal convection patterns, spiral wave patterns, and hexagonal patterns observed in the experiment [8, 9].

The scientific community has a wide range of interest in the formation mechanism and structural characteristics behind the pattern. Consequently, theories of pattern dynamics have been deeply studied and form a more systematic theoretical research area [10–21]. Here, we present some typical

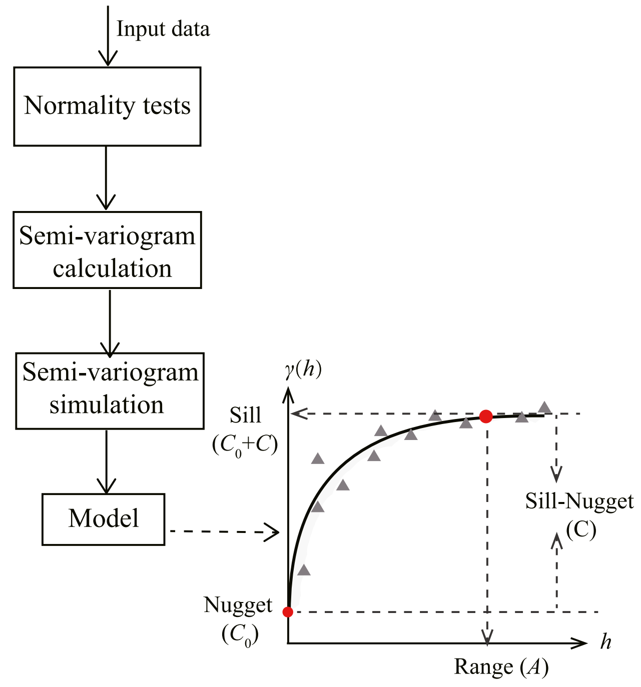


FIGURE 1 | Flowchart of quantitative system robustness index [30]. The analysis of semivariogram is mainly divided into three phases: normal test of research data, calculation of the semi-variance of variable, and simulation semivariogram to obtain the best fitted model. Structural ratio $C/(C_0 + C)$ can reflect the robustness of the system.

works on this topic. In 2001, von Hardenberg et al. pointed out that when the bare state and spot pattern state coexist, and it is also unstable, if exceeding a certain threshold, the system will be completely transformed to a bare state or desertification, that is, the spotted pattern will be used as an early warning signal of desertification [22]. In 2014, Liu et al. revealed that the interaction of self-organization behavior between different scales can improve the robustness, persistence and productivity of mussel ecosystem; in other words, the mosaic patterns with large and small scales imply the ecosystem is more robust [6]. In 2020, Bastiaansen et al. quantified the resilience of ecosystems with spatial patterns by using phase portrait [23]. However, due to the complexity of the system dynamical process and the lack of uniformity on robustness definition, many studies and conclusions are not comprehensive and have certain limitations, even lack of quantitative indicators for the robustness of each pattern structure.

In order to better answer the question “how the pattern structures determine the robustness of the system,” we obtain a series of different pattern structures based on three dynamical models in different research fields: vegetation–water coupled model [24], epidemic spatial model [25], and predator–prey model [26], and use geostatistical methods to quantitatively describe and analyze the characteristics of all different pattern structures, so as to find out which type of pattern structures are more robust for the corresponding systems.

2 CHARACTERIZATION INDEX OF SYSTEM ROBUSTNESS

In ecology, the related concepts of robustness is complicated and imprecise, but most people agree that robustness can be divided into two categories: one is the ability of the system to resist leaving (maintain) the current state after the system is subjected to external disturbance [27]; the other one is the ability of the system to return to the original stable state after suffering disturbance [28]. Therefore, in this study, we will use semi-variogram to analyze the interaction and dependence of each component within the system, and thereby give the quantitative index of the second type of system robustness.

The semivariogram is a mathematical statistical method that can reflect the randomness and structural characteristics of the variable in the spatial distribution and also is the theoretical basis of geostatistics. First, in order to avoid the proportional effect in the study, it is necessary to test the data of the studied variable (Z) and judge whether it is normal distribution or approximate normal distribution; if not, data conversion shall be carried out to make it conform to normal distribution. Afterward, the calculation of the semi-variance function value is carried out. Finally, through simulation, we obtain the best fitted semivariogram model and some important indicators, such as nugget C_0, sill $C_0 + C$, range A, and the structural ratio $C/(C_0 + C)$. **Figure 1** shows the basic process of this method and the calculation formula of the semivariogram as follows [29]:

$$\gamma(h) = \frac{1}{2N(h)} \sum_{i=1}^{N(h)} [Z(x_i) - Z(x_{i+h})]^2, \tag{1}$$

where $\gamma(h)$ represents semivariogram, h is the step length, that is the sample point spatial distance, $N(h)$ indicates the total number of data pairs when the sample point distance is h, and $Z(x_i)$ and $Z(x_{i+h})$, respectively, are the values of the variable(Z) at the spatial position x_i and x_{i+h}.

In **Figure 1**, the nugget C_0 represents the degree of random heterogeneity of variables in the region and the sill $(C_0 + C)$ refers to the maximum variation of the system induced by structural variation and random variation. Then, the structural ratio $C/(C_0 + C)$ indicates the contribution rate of structural factors to spatial heterogeneity, which can reflect the spatial dependence and the spatial heterogeneity of variables [31]. When the structural ratio is less than 25%, it indicates that the variable has weak spatial dependence; if 25% ≤ structural ratio ≤75%, it means that the variable has moderate spatial dependent; and the structural ratio >75% indicates the spatial dependence of the variable is strong [32, 33]. Generally speaking, the larger the structural ratio is, the stronger the spatial dependence and spatial heterogeneity of the variables within the system will be. For the species community, the greater the heterogeneity is, the richer the diversity will be. In other words, if the structural ratio of the pattern structure is larger, then it means that the corresponding system is more robust.

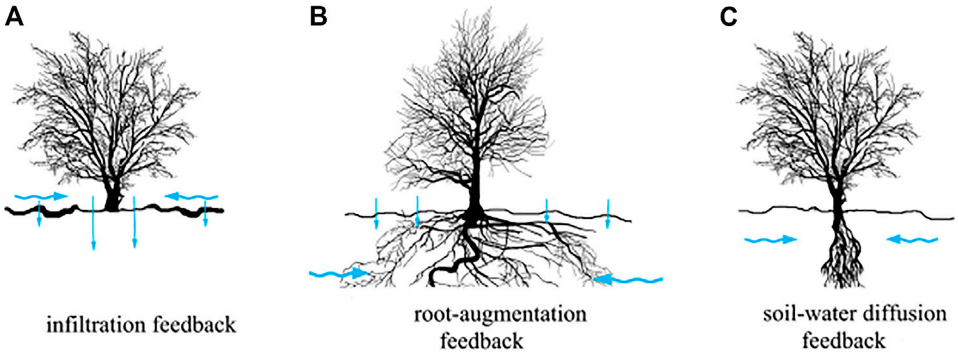


FIGURE 2 | Schematic diagram of three feedback mechanisms between vegetation and water. **(A)** Difference of infiltration rate between bare soil and vegetation covered area causes surface water flow, **(B)** vegetation roots extend laterally to absorb water, and **(C)** soil water diffusivity exceed the spread of vegetation biomass. The blue arrow indicates the direction of water flow, and the length of the straight arrow represents the size of infiltration rate.

TABLE 1 | Biological significance and value of all parameters in the model (2).

	Symbol	Biological significance	Value	Source
Parameters (dimensionless)	λ	The rate of vegetation water absorption	0.457 1	[24]
	η	Lateral extension of vegetation roots	2.8	[24]
	p	Precipitation	1.517	[24]
	v	Evaporation of soil–water	1.428 6	[24]
	δ_w	The diffusion ratio of water resources to vegetation	125	[24]
	ρ	Shading rate	[0.6, 0.94]	[24]
Variable	n	Vegetation density		
	w	Soil–water density		
	t	Time		

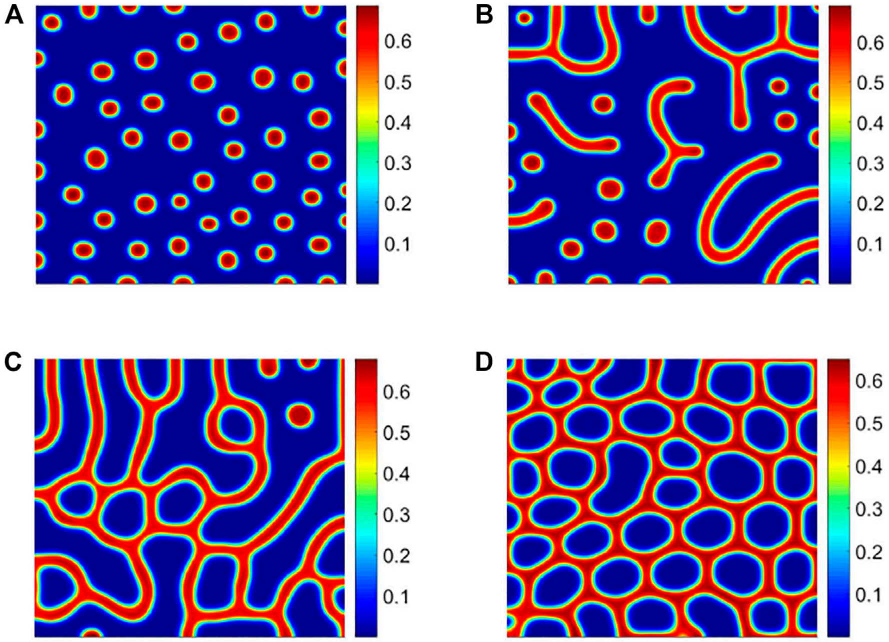


FIGURE 3 | Vegetation pattern structure under different shading rates ρ. **(A)** Hot spot pattern with $\rho = 0.6$; **(B)** spots–stripes mixed pattern with $\rho = 0.75$; **(C)** stripe pattern with $\rho = 0.9$; **(D)** cold spot pattern with $\rho = 0.94$. See **Table 1** for the other parameter values.

TABLE 2 | Fitted semivariogram models (SV) and their characteristic parameters for different vegetation pattern structures (**Figure 3**). The high value of R^2 and the low of *RSS* indicate good fitted effect.

Pattern structure	SV^a	C/(C$_0$ + C)^d	A^e	R^{2f}	Class
Hot spot	Gau^b	0.863	5.889	0.859	S^i
Spots–stripes mixed	Sph^c	0.998	8.000	0.910	S
Stripe	Gau	0.884	6.409	0.851	S
Cold spot	Gau	0.960	5.889	0.777	S

[a] *Residual (RSS) are both less than 10^{-4}.*
[b] *Gaussian model.*
[c] *Spherical model.*
[d] *C_0 – nugget, C_0+C – sill.*
[e] *Range.*
[f] *Coefficient of determination.*
[i] *Strong spatial dependence.*

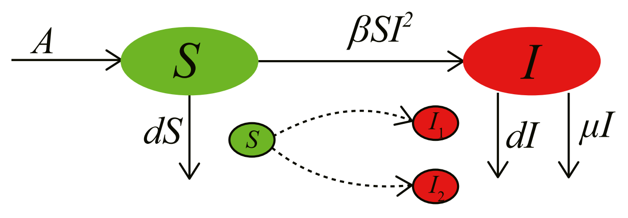


FIGURE 4 | Schematic diagram of infectious diseases transmission process. The model satisfies the following conditions: 1) the epidemic cannot be cured after being infected; 2) the susceptible is exposed to the disease repeatedly, leading to a non-linear transmission rate.

3 THE CORRESPONDING RELATIONSHIP BETWEEN PATTERN STRUCTURES AND SYSTEM ROBUSTNESS

3.1 Vegetation–Water Dynamical Model

In arid and semi-arid areas, vegetation growth is mainly limited by water resources [5]. In 2004, Gilad et al. considered three feedback mechanisms between vegetation biomass and water resources, infiltration feedback, root augmentation feedback, and soil–water diffusion feedback mechanism (**Figure 2**), and established a dynamical model for coupling a single vegetation species with water [34, 35]. On this basis, scholars made the following assumptions: 1) the lateral extension of vegetation roots is restricted and 2) the infiltration rate of bare soil and vegetation covered area is the same, and then obtained a simplified dimensionless model as follows [24]:

$$\begin{aligned}\frac{\partial n}{\partial t} &= \lambda wn(1-n)(1+\eta n)^2 - n + \Delta n,\\ \frac{\partial w}{\partial t} &= p - \nu w(1-\rho n) - \lambda wn(1+\eta n)^2 + \delta_w \Delta w.\end{aligned} \tag{2}$$

In the previous dimensionless model, the biological significances and values of all parameters are shown in **Table 1**. Next, we study the influence of the shading rate ρ on the robustness of vegetation ecosystems. Fixing other parameters, through numerical simulation, we get a series of vegetation pattern structure, as shown in **Figure 3**.

For the previous four different vegetation patterns, hot spot, mixed spots and stripes, stripe, and cold spot, we use vegetation density as a research variable and perform a semi-variogram analysis on them. According to **Table 2**, it is found that the structural ratio $C/(C_0 + C)$ of these four vegetation patterns in descending order are as follows: mixed spots and stripes > cold spot > stripe > hot spot, and the value of the aforementioned four vegetation patterns are both more than 75%, which shows that they all have strong spatial dependence and spatial heterogeneity. However, compared with the other three structures, the vegetation system with mixed spots and stripes has the strongest spatial heterogeneity, which means that the system under this pattern structure will be the most robust. For the vegetation–water dynamical model, the more robust the system is, the lower the possibility of desertification is. The spot–stripe mixed structure is conducive to vegetation diffusion in space, while the hot spot pattern shows that the vegetation presents an isolated high-density distribution, which implies the system is more susceptible to desertification. Based on experiments, Bertolini et al. also found that the vegetation system corresponding to the mixed spot–stripe pattern (labyrinthine-like pattern) is relatively robust [36], and thus, our conclusion is consistent with the previous findings.

3.2 Spatiotemporal Dynamical Model for Disease Transmission

The spread of infectious diseases will be affected by the spatial movement of the susceptible or infected, which will eventually lead to spatial patterns of the susceptible and infected individuals

TABLE 3 | Biological significance and value of all parameters in the model (3).

	Symbol	Biological significance	Value	Source
Parameters	*A*	The recruitment rate of the population	1	[25]
	d	The natural death rate of the population	1	[25]
	μ	The disease-related death rate from the infected	1.8	[25]
	D_1	The susceptible individual diffusion coefficient	6	[25]
	D_1	The infected individual diffusion coefficient	1	[25]
	β	The force of infection or the rate of transmission	[32, 42]	[25]
Variable	*S*	Susceptible individual		
	I	Infected individual		
	t	Time		

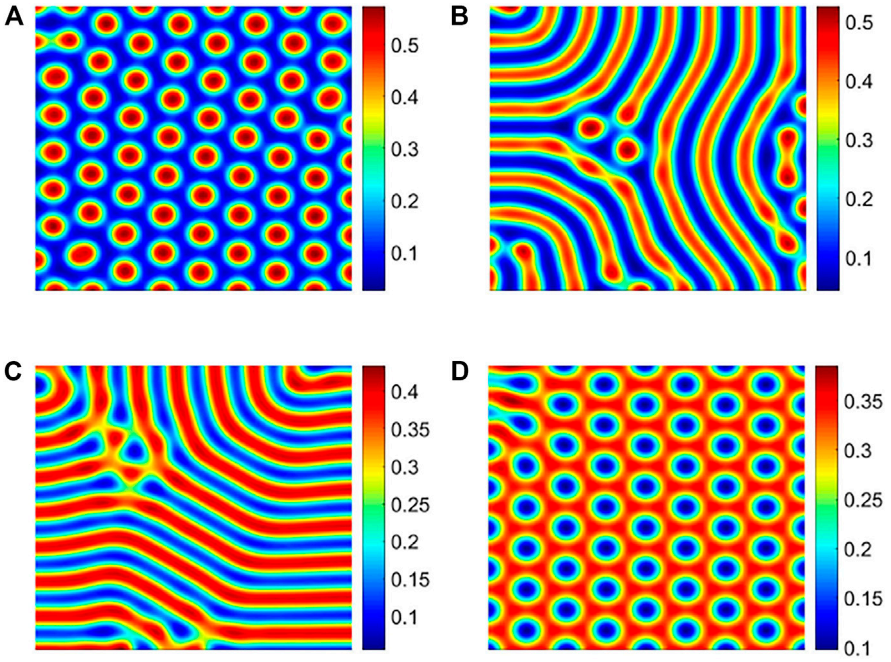


FIGURE 5 | Different patterns of infected individuals. **(A)** Hot spot pattern with $\beta = 32$; **(B)** spots and stripes mixed pattern with $\beta = 35$; **(C)** stripe pattern with $\beta = 40$; and **(D)** cold spot pattern with $\beta = 42$.

TABLE 4 | Fitted semivariogram models (SV) and its characteristic parameters for different infected pattern (**Figure 5**). The meanings of other parameters are consistent with **Table 2**.

Pattern structure	SV[a]	$C/(C_0 + C)$	A	R^2	Class
Hot spot	Gau	0.943	6.9	0.487	S
Spots–stripes mixed	Gau	0.999	5.369 4	0.533	S
Stripe	Gau	0.944	5.542 6	0.483	S
Cold spot	Gau	0.998	5.542 6	0.452	S

[a] *Residual (RSS) is less than* 10^{-4}.

[37, 38]. In view of the characteristics of acquired immunodeficiency Syndrome (AIDS), hepatitis B virus (HBV), ebola virus, and other infectious diseases that are difficult to be cured after infected, meanwhile considering the complex spatial dynamics of the susceptible and the infected, scholars proposed an epidemic spatial model with non-linear incidence rates [25]. The non-linear incidence rate is caused by twice exposures of susceptible before infection. The transmission mechanism of such infectious disease can be described in **Figure 4**. The biological significances and values of all parameters in the model are shown in **Table 3**. The specific mathematical model is as follows:

$$\begin{aligned}\frac{\partial S}{\partial t} &= A - dS - \beta S I^2 + D_1 \Delta S,\\ \frac{\partial I}{\partial t} &= \beta S I^2 - (d + \mu) I + D_2 \Delta I.\end{aligned} \tag{3}$$

Based on the spatial infectious disease model (3), we studied the impact of the spatial distribution of infected individuals on the spread of infectious diseases in the population. First, fixing other parameters, only changing the transmission rate β, and a series of numerical simulations are carried out on model (3). Finally, we get the pattern structure, as shown in **Figure 5**.

With the increase in the transmission rate, the spatial distribution of infected individuals present four different pattern structures: hot spot, mixed spots and stripes, stripe, and cold spot (**Figure 5**). From **Table 4**, we find that the

TABLE 5 | Biological significance and value of all parameters in the model (4).

	Symbol	Biological significance	Value	Source
Parameters (dimensionless)	ε	Biomass conversion rate from prey to predator	0.5	[26]
	δ	The ratio of diffusion coefficient	10	[26]
	θ	The death rate of predator	0.6	[26]
	γ	Prey consumption rate	[1.95, 2.05]	[26]
Variable	x	Prey density		
	y	Predator density		
	t	Time		

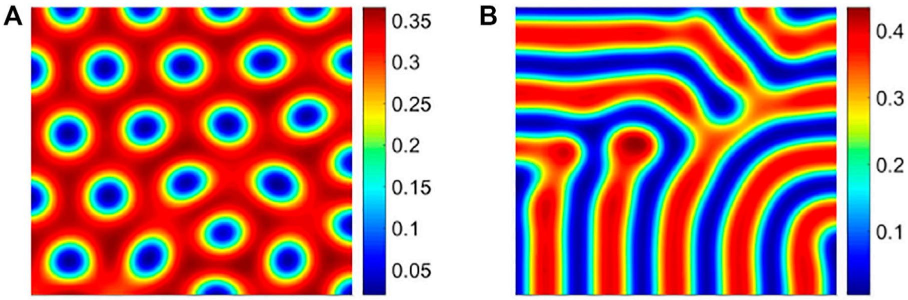


FIGURE 6 | Different pattern structures of prey population. **(A)** Cold spot pattern with γ = 1.95; **(B)** stripe pattern with γ = 2.05.

TABLE 6 | Fitted semivariogram models (SV) and its characteristic parameters for different prey pattern (**Figure 6**). The meanings of other parameters are consistent with **Table 2**.

Pattern structure	SV[a]	$C/(C_0 + C)$	A	R^2	Class
Cold spot	Gau	0.999	9.1799	0.595	S
Stripe	Gau	0.989	8.487	0.665	S

[a] Residual is less than 10^{-4}.

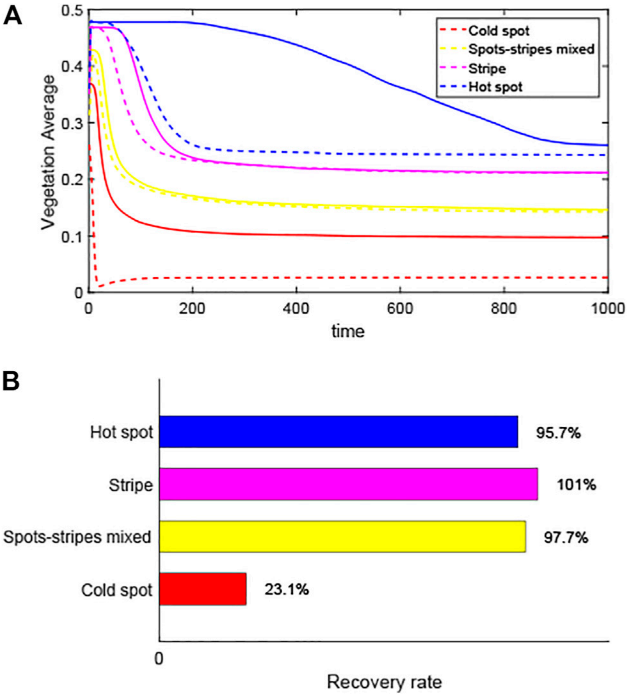


FIGURE 7 | Recovery of different pattern structures for same disturbance. **(A)** Trajectories of different patterns when they return to steady state after the 20% disturbance occurs. Solid lines: trajectories of patterns formation; dashed lines: recovery track after the equilibrium point is reduced by 20%. **(B)** Recovery rates after a disturbance.

structural ratio $C/(C_0 + C)$ of the four different spatial distributions of the infected are all greater than 75%, and their ratios are arranged in the following order: spot–stripe mixed > cold spot > stripe > and hot spot (**Table 4**). Although their structure ratio gap is small, its order is highly consistent with the order of the vegetation patterns, which demonstrates that our conclusion has certain universality.

3.3 Predator–Prey Model With Spatial Diffusion

The predator function response is an indispensable part of describing the predator–prey model. This function can reflect the influence of the competition between predators on the predation efficiency, such as the coyotes, and jackrabbits in the western wilderness of North America and the plankton experiment [39]. In particular, when the predator can capture a large amount of prey per unit time, or when saturation is not considered, a ratio-dependent functional response is obtained [40]. However, with the addition of spatial diffusion, this ratio-dependent predator–prey model will produce a heterogeneous spatial distribution structure [26]. As a result, we consider this ratio-dependent predator–prey model with diffusion terms. After introducing dimensionless variables, the model is simplified to the following equations [26]

$$\begin{aligned}\frac{\partial x}{\partial t} &= (1-x)x - \frac{\gamma x y}{x+y} + \Delta x,\\ \frac{\partial y}{\partial t} &= \varepsilon \gamma \frac{xy}{x+y} + \delta \Delta y.\end{aligned} \tag{4}$$

The dimensionless variables in model (4) and their values are shown in **Table 5**. Fixing other parameters and changing the prey consumption rate by predator γ, through numerical simulation, we obtain prey pattern structures, as shown later (**Figure 6**).

On the basis of the previous method, we conduct a semi-variogram analysis on the previous two spatial distribution structures of prey. The detailed results are shown in **Table 6**. According to the heterogeneity classification standard, we find that the structural ratio $C/(C_0 + C)$ of these two patterns is significantly higher than 75%. However, by comparison, the structural ratio of the cold spot pattern is larger than that of the stripe pattern, that is, its corresponding system is more robust.

Compared with the stripe structure, the cold spot pattern makes the prey to gather together and is not easy to be captured by the predators, and thus, the system is more robust.

The range A is an important index in geostatistics that can reflect the spatial heterogeneity or spatial dependence scale of regional variables. From the perspective of the range, we find that all the ranges obtained by analyzing the pattern structure above are larger than the sampling interval − − 1. Therefore, it shows that the sampling interval used in the study is credible for unbiased estimation of this area.

4 CONCLUSION AND DISCUSSION

System robustness and pattern structure are two important characteristics to portray the spatiotemporal complexity of systems. However, the analysis of their corresponding relationship is lack of systematic research results. Focusing on the scientific problem of "which pattern structure implies the robustness of the system, and which pattern structure means the vulnerability of the system," we combined three dynamical models from different fields, vegetation-water dynamical model [24], epidemic spatiotemporal dynamical model [25], and predator–prey spatial evolution dynamical model [26], and used geostatistical methods to analyze a series of different pattern structures produced by them. Finally, we found that the robustness of the system corresponding to different pattern structures is arranged as follows: spots and stripes mixed >, cold spot >, stripe >, and hot spot. The research results may provide some early warning signals for desertification prevention, infectious disease prevention and control, biodiversity conservation, and other related fields.

In addition, we explored the systems' resilience of different vegetation pattern structures (**Figure 7**), combined with the latest research methods [41]. We once again confirmed that the recovery rates of ecosystems with hot spot are the worst, compared with the other three pattern structures. And it also revealed that the recovery rate of the ecosystem corresponding to the spots and stripes mixed is greater than that of the cold spot. However, the only difference is that the recovery rate of system with stripe is the largest, which may be because the added disturbance increases the local density and finally affects the characteristics of the strip structure.

The evaluation of system robustness generally requires a large number of monitoring data or experimental data [42, 43]. However, our study is mainly based on the dynamical model to obtain the pattern structures, which can not only dynamically reflect the evolution of the system in time and space but also quantitatively predict the future spatiotemporal distribution structure. At the same time, the semi-variogram analysis method can comprehensively analyze and compare the spatial characteristics of each pattern structure, overcome the complexity of the previous analysis system robustness, and provide a new idea for describing the corresponding relationship between the pattern structure and the system robustness.

It is worth noting that our research ignores the scale dependence and does not combine with real data, while the pattern structure has different scales and can be completely corresponded to the real data. In this case, we can improve and verify our theoretical results based on GIS and big data analysis. Furthermore, the pattern structure can be divided into steady-state and non–steady-state structures, and this study only focuses on the former case. While for the non–steady-state pattern, its existence in the real world is more extensive, and hence, the correspondence between the unsteady structure and the robustness of the system is also an important scientific issue. We hope these questions will be systematically addressed in future research.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Material; further inquiries can be directed to the corresponding authors.

AUTHOR CONTRIBUTIONS

All authors have made great contributions to the writing of study and approved the submitted version. G-QS, YP, LL, CL, YW, and ZW established dynamical modeling. G-QS, YP, CL, and YW participated in the program design and provided valuable comments on the manuscript writing. G-QS, YP, and LL collected and processed the relevant published data. GF, ZJ, and B-LL guided and improved the manuscript.

FUNDING

The project is funded by the National Key Research and Development Program of China (Grant No. 2018YFE0109600), National Natural Science Foundation of China (Grant No. 42075029), Program for the Outstanding Innovative Teams (OIT) of Higher Learning Institutions of Shanxi, China Postdoctoral Science Foundation (Grant Nos. 2017M621110 and 2019T120199), and Outstanding Young Talents Support Plan of Shanxi province.

REFERENCES

1. Malkus JS. Cloud Patterns over Tropical Oceans: Tropical Clouds Are Arranged into Characteristic "Fingerprints" of the Weather Systems Producing Them. *Science* (1963) 141:767–78. doi:10.1126/science.141.3583.767
2. Caro T, Izzo A, Reiner RC, Walker H, Stankowich T. The Function of Zebra Stripes. *Nat Commun* (2014) 5:3535. doi:10.1038/ncomms4535
3. Lämmel M, Meiwald A, Yizhaq H, Tsoar H, Katra I, Kroy K. Aeolian Sand Sorting and Megaripple Formation. *Nat Phys* (2018) 14:759–65. doi:10.1038/s41567-018-0106-z

4. Getzin S, Yizhaq H, Bell B, Erickson TE, Meron E. Discovery of Fairy Circles in Australia Supports Self-Organization Theory. *Proc Natl Acad Sci* (2016) 113: 201522130. doi:10.1073/pnas.1522130113
5. Klausmeier CA. Regular and Irregular Patterns in Semiarid Vegetation. *Science* (1999) 284:1826–8. doi:10.1126/science.284.5421.1826
6. Liu Q-X, Herman PMJ, Mooij WM, Huisman J, Scheffer M, Olff H, et al. Pattern Formation at Multiple Spatial Scales Drives the Resilience of Mussel Bed Ecosystems. *Nat Commun* (2014) 5:5234. doi:10.1038/ncomms6234
7. Gao C, Liu C, Schenz D, Li X, Zhang Z, Jusup M, et al. Does Being Multi-Headed Make You Better at Solving Problems? A Survey of Physarum-Based Models and Computations. *Phys Life Rev* (2019) 29:1–26. doi:10.1016/j.plrev.2018.05.002
8. Yang L, Dolnik M, Zhabotinsky AM, Epstein IR. Turing Patterns beyond Hexagons and Stripes. *Chaos* (2006) 16:037114. doi:10.1063/1.2214167
9. Ou YQ. *Introduction to Nonlinear Science and Pattern Dynamics*. Beijing, China: Peking University Press (2010).
10. Turing AM. The Chemical Basis of Morphogenesis. *Bull Math Biol* (1952) 237: 37–72.
11. Xue Q, Liu C, Li L, Sun GQ, Wang Z. Interactions of Diffusion and Nonlocal Delay Give Rise to Vegetation Patterns in Semi-Arid Environments. *Appl Maths Comput* (2021) 399:126038. doi:10.1016/j.amc.2021.126038
12. van de Koppel J, Rietkerk M, Dankers N, Herman PMJ. Scale-dependent Feedback and Regular Spatial Patterns in Young Mussel Beds. *Am Nat* (2005) 165:166–77. doi:10.1086/428362
13. Liu QX, Doelman A, Rottschafer V, Jager MD, Herman P, Rietkerk M, et al. Phase Separation Explains a New Class of Self-Organized Spatial Patterns in Ecological Systems. *Proc Natl Acad Sci* (2013) 110:11905–10. doi:10.1073/pnas.1222339110
14. Sun GQ, Zhang HT, Wang JS, Li J, Wang Y, Li L, et al. Mathematical Modeling and Mechanisms of Pattern Formation in Ecological Systems: A Review. *Nonlinear Dyn* (2021) 104:1677–96. doi:10.1007/s11071-021-06314-5
15. Tian C, Ling Z, Zhang L. Nonlocal Interaction Driven Pattern Formation in a Prey-Predator Model. *Appl Maths Comput* (2017) 308:73–83. doi:10.1016/j.amc.2017.03.017
16. Ni W, Shi J, Wang M. Global Stability and Pattern Formation in a Nonlocal Diffusive Lotka-Volterra Competition Model. *J Differential Equations* (2018) 264:6891–932. doi:10.1016/j.jde.2018.02.002
17. Chakraborty B, Baek H, Bairagi N. Diffusion-Induced Regular and Chaotic Patterns in a Ratio-Dependent Predator–Prey Model with Fear Factor and Prey Refuge. *Chaos* (2021) 31:033128. doi:10.1063/5.0035130
18. Sun G-Q, Li M-T, Zhang J, Zhang W, Pei X, Jin Z. Transmission Dynamics of Brucellosis: Mathematical Modelling and Applications in China. *Comput Struct Biotechnol J* (2020) 18:3843–60. doi:10.1016/j.csbj.2020.11.014
19. Zhu P, Dai X, Li X, Gao C, Jusup M, Wang Z. Community Detection in Temporal Networks via a Spreading Process. *Europhysics Lett* (2019) 126: 48001. doi:10.1209/0295-5075/126/48001
20. Wang Z, Wang C, Li X, Gao C, Li X, Zhu J. Evolutionary Markov Dynamics for Network Community Detection. *IEEE Trans Knowledge Data Eng* (2020) 32:1. doi:10.1109/TKDE.2020.2997043
21. Gao C, Liu J. Network-Based Modeling for Characterizing Human Collective Behaviors during Extreme Events. *IEEE Trans Syst Man, Cybernetics: Syst* (2017) 47:171–83. doi:10.1109/TSMC.2016.2608658
22. von Hardenberg J, Meron E, Shachak M, Zarmi Y. Diversity of Vegetation Patterns and Desertification. *Phys Rev Lett* (2001) 87:198101. doi:10.1103/physrevlett.87.198101
23. Bastiaansen R, Doelman A, Eppinga MB, Rietkerk M. The Effect of Climate Change on the Resilience of Ecosystems with Adaptive Spatial Pattern Formation. *Ecol Lett* (2020) 23:414–29. doi:10.1111/ele.13449
24. Zelnik YR, Meron E, Bel G. Gradual Regime Shifts in Fairy Circles. *Proc Natl Acad ences* (2015) 112:12327–31. doi:10.1073/pnas.1504289112
25. Sun G-Q. Pattern Formation of an Epidemic Model with Diffusion. *Nonlinear Dyn* (2012) 69:1097–104. doi:10.1007/s11071-012-0330-5
26. Alonso D, Bartumeus F, Catalan J. Mutual Interference between Predators Can Give Rise to Turing Spatial Patterns. *Ecology* (2002) 83:28–34. doi:10.1890/0012-9658(2002)083[0028:mibpcg]2.0.co;2
27. Ogbunugafor CB, Pease JB, Turner PE. On the Possible Role of Robustness in the Evolution of Infectious Diseases. *Chaos* (2010) 20:026108. doi:10.1063/1.3455189
28. Yi CX, Jackson N. A Review of Measuring Ecosystem Resilience to Disturbance. *Environ Res Lett* (2021) 16:053008. doi:10.1088/1748-9326/abdf09
29. Usowicz B, Lipiec J. Spatial Variability of Saturated Hydraulic Conductivity and its Links with Other Soil Properties at the Regional Scale. *Scientific Rep* (2021) 11:8293. doi:10.1038/s41598-021-86862-3
30. Rahman AF, Gamon JA, Sims DA, Schmidts M. Optimum Pixel Size for Hyperspectral Studies of Ecosystem Function in Southern California Chaparral and Grassland. *Remote Sensing Environ* (2003) 84:192–207. doi:10.1016/s0034-4257(02)00107-4
31. Wang T, Kang FF, Han HR, Cheng XQ, Bai YC. Factors influencing spatial heterogeneity of soil moisture content in small catchment of Mount Taiyue, Shanxi Province. *J Ecol* (2017) 37:3902–11. doi:10.5846/stxb201604170709
32. Wang Z, Wang Q, Cheng Q. Spatial Heterogeneity of Soil Nutrients in Old Growth Forests of Korean pine. *J For Res* (1998) 9:240–4. doi:10.1007/bf02912326
33. Cambardella CA, Moorman TB, Novak JM, Parkin TB, Karlen DL, Turco RF, et al. Field-Scale Variability of Soil Properties in Central Iowa Soils. *Soil Sci Soc America J* (1994) 58:1501–11. doi:10.2136/sssaj1994.03615995005800050033x
34. Gilad E, Hardenberg JV, Provenzale A, Shachak M, Meron E. Ecosystem Engineers: From Pattern Formation to Habitat Creation. *Phys Rev Lett* (2004) 93:981051–4. doi:10.1103/physrevlett.93.098105
35. Gilad E, von Hardenberg J, Provenzale A, Shachak M, Meron E. A Mathematical Model of Plants as Ecosystem Engineers. *J Theor Biol* (2007) 244:680–91. doi:10.1016/j.jtbi.2006.08.006
36. Bertolini C, Cornelissen B, Capelle J, Koppel J, Bouma TJ. Putting Self-Organization to the Test: Labyrinthine Patterns as Optimal Solution for Persistence. *Oikos* (2019) 128:1805–15. doi:10.1111/oik.06373
37. Zhu P, Zhi Q, Guo Y, Wang Z. Analysis of Epidemic Spreading Process in Adaptive Networks. *IEEE Trans Circuits Syst* (2019) 66:1252–6. doi:10.1109/tcsii.2018.2877406
38. Gao C, Su Z, Liu J, Kurths J. Even central Users Do Not Always Drive Information Diffusion. *Commun ACM* (2019) 62:61–7. doi:10.1145/3224203
39. Beddington JR. Mutual Interference between Parasites or Predators and its Effect on Searching Efficiency. *J Anim Ecol* (1975) 44:331–40. doi:10.2307/3866
40. Arditi R, Ginzburg LR. Coupling in Predator-Prey Dynamics: Ratio-Dependence. *J Theor Biol* (1989) 139:311–26. doi:10.1016/s0022-5193(89)80211-5
41. Zhao LX, Zhang K, Siteur K, Li XZ, Koppel J. Fairy Circles Reveal the Resilience of Self-Organized Salt Marshes. *Sci Adv* (2021) 7:eabe1100. doi:10.1126/sciadv.abe1100
42. Gowda K, Chen Y, Iams S, Silber M. Assessing the Robustness of Spatial Pattern Sequences in a Dryland Vegetation Model. *Proc R Soc A* (2016) 472: 20150893. doi:10.1098/rspa.2015.0893
43. Gao C, Fan Y, Jiang S, Deng Y, Liu J, Li X. Dynamic Robustness Analysis of a Two-Layer Rail Transit Network Model. *IEEE Trans Intell Transportation Syst* (2021) 22:1–16. doi:10.1109/tits.2021.3058185

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

ORIGINAL RESEARCH
published: 14 February 2022
doi: 10.3389/fphy.2022.830774

Data-Driven State Fragility Index Measurement Through Classification Methods

Xin Li, Alexandre Vidmer, Hao Liao and Kezhong Lu**

National Engineering Laboratory on Big Data Application on Improving Government Governance Capabilities, Guangdong Province Key Laboratory of Popular High Performance Computers, College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China

As environmental changes cause a series of complex issues and unstable situation, exploring the impact of environmental changes is essential for national stability, which is helpful for early warning and provides guidance solutions for a country. The existing mainstream metric of national stability is the Fragile States Index, which includes many indicators such as abstract concepts and qualitative indicators by experts. In addition, these indicators may have preferences and bias because some data sources come from unreliable platforms; it may not reflect the real situation for the current status of countries. In this article, we propose a method based on ensemble learning, named CR, which can be obtained by quantifiable indicators to reflect national stability. Compared with the current mainstream methods, our proposed CR method highlights quantitative factors and reduces qualitative factors, which is an advantage of simplicity and interoperability. The extensive experimental results show a significant improvement over the SOTA methods (7.13% improvement in accuracy, 2.02% improvement in correlation).

Keywords: data-driven, quantifiable, ranking, state for fragility, classification

OPEN ACCESS

Edited by:
Peican Zhu,
Northwestern Polytechnical University, China

Reviewed by:
Yanqing Hu,
Southern University of Science and Technology, China
Lin Jianhong,
University of Zurich, Switzerland

****Correspondence:***
Hao Liao
haoliao@szu.edu.cn
Kezhong Lu
kzlu@szu.edu.cn

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *07 December 2021*
Accepted: *05 January 2022*
Published: *14 February 2022*

Citation:
Li X, Vidmer A, Liao H and Lu K (2022) Data-Driven State Fragility Index Measurement Through Classification Methods. Front. Phys. 10:830774. doi: 10.3389/fphy.2022.830774

1 INTRODUCTION

In recent years, tremendous scientific outcomes have proved that environmental changes cause a considerable impact on the development of human society, and it shows different degrees of influence in different regions. As early as the end of the 20th century, many countries have begun to pay attention to research on the impact of environmental changes on the country's economic and political stability, take measures to contain the speed of environmental changes, and respond to subsequent effects. In addition, at the beginning of the 21st century, the U.S. Department of Defense identified unstable factors related to the environment as a primary strategic consideration. Evidence shows that environmental pressure is an essential factor in contemporary conflicts [1]. It also shows that the conflicts caused by the environment are inspired by dynamic, complex, and interactive processes, not just a simple and deterministic relationship. Therefore, it is necessary to establish an analytical framework to understand the antecedents and consequences [35].

Environmental changes may lead to fatal results. Drastic changes in the environment will cause a series of humanitarian disasters, such as political violence, which makes an already fragile country worse off [2–4]. The urgent need and competition for limited resources such as energy, food, and water may rise to national military confrontation, thus causing a series of humanitarian disasters and political violence [5]. Environmental pressures caused by environmental changes are generally combined with weak governance and social divisions, thereby exacerbating national stability [6–9].

Therefore, it is insightful to mine association rules from existing data and evaluate the impact of environmental changes on national stability [10].

Besides, many metrics could evaluate national stability such as the Country Indicators for Foreign Policy (CIFP) [11] published by Carleton University and The Peace and Conflict Instability Ledger (PCIL) [12] compiled by the University of Maryland. The most widely used and well-performing metric is the Fragile States Index (FSI) [13] published annually by the American Bimonthly Foreign Policy and The Fund for Peace. The Fragile States Index consists of many indicators such as economic, military, politics, and society. And each indicator is obtained through the process of quantitative resource and qualitative evaluation by experts. However, the design of the Fragile State Index is complicated, and the concept used in evaluating the value of the indicator is abstract, which is not convenient for research and analysis. Specifically, in content analysis, there is a data preference for countries with sound volume, which may not reflect the real situation of some fragile countries. In the process of qualitative review by experts, there may also be subjective tendencies or stereotypes, making the value of the indicator unable to express accurately.

Ensemble learning is a learning algorithm. It utilizes multiple classifiers to combine to handle a more powerful performance [14,15]. Multiple classifiers through ensembles will generally perform better than a single classifier. Taking our daily life as an example, the social division of labor cooperation is a powerful way to improve work efficiency. By giving full care to individual strengths, individuals with different strengths are brought together to make them play a greater role. This is where the idea of ensemble learning comes from.

In this work, we study how to utilize quantitative ways to pursue reasonable ranking to measure national stability, which has certain similarity and high correlation with the FSI ranking. Our proposed CR method mainly involves label by stage division, machine learning methods for experiments, and ranking aggregation under the best method at each stage. The main contributions of our work are as follows: 1) We propose a new method based on the idea of ensemble learning, which applies different classifiers to aggregate the optimal classifier under different stages to achieve better prediction performance. 2) Combining the advantages of multiple machine learning methods with the idea of ensemble learning enables our method to obtain more valuable and insightful information, which enables our CR method universality. 3) Through a large number of experiments, our results show that ranking obtained by our proposed CR method is in line with the actual situation of the current national stability; it pays more attention to the quantification of the original data and reduces the impact of abstract indicators on the prediction performance, which makes it more quantifiable and interpretable.

2 MATERIALS AND METHODS

In this section, we present our datasets and briefly introduce the classification methods. The classification methods could be divided into traditional machine learning and neural network methods.

2.1 Data

Our data resources are obtained from the World Bank,[1] Germanwatch,[2] The Fund for Peace,[3] the International Trade Organization,[4] and the Belt and Road Initiative.[5] The descriptions of the data resources are as follows:

- Improved water, carbon emissions, and cultivated land: It is obtained from the World Bank and Germanwatch. Specifically, records per capita improved water of 236 countries from 2000 to 2015, the per capita carbon emissions (metric tons per capita) of 242 countries from 2000 to 2014, and the per capita cultivated land (hectares per capita) of 178 countries from 2007 to 2015. Improved water resources are not only related to the national investment but also to the harshness of the climate. (The percentage of per capita improved water in developed countries is generally higher.) And carbon emissions are a useful indicator for demonstrating a country's industrial development. Cultivated land is a basis for agricultural production, and it is also an essential indicator for the country's productivity. The changes in these indicators will have a great impact on the politics, economy, and society of the region.
- Climate Risk Index: It is collected from The Fund for Peace, which records the annual Global Climate Risk Index (CRI [16]) of 187 countries from 2007 to 2014. The CRI aims to analyze the extent to which countries and regions are affected by extreme weather loss events (storms, floods, heat waves, etc.). Regarding future climate change, the CRI may become a warning signal. Thus, it is considered a factor affecting environmental changes.
- International Trade Network: We obtained international trade network data from the International Trade Organization, which records the export trade status of 786 commodities in 192 countries/regions from 2001 to 2015. The international trade network data can well reflect the economic vitality and productivity of the region. The fitness and complexity algorithm was proposed by the authors of the study mentioned in reference [17], which aims to measure the economic health of a country and is a correct and simple way to measure the competitiveness of a country. Because of its outstanding performance in national economic forecasts, it was adopted by the World Bank in 2017. In the study, we examined the international trade network to apply the algorithm to calculate a fitness value to reflect the economic vitality and productivity of the country.

[1] https://data.worldbank.org

[2] https://germanwatch.org

[3] https://fundforpeace.org

[4] https://www.wto.org

[5] http://belt.china.org.cn/

- The Belt and Road (B&R): We obtained data from the Belt and Road Initiative. Politics is also an important factor for affecting national stability; we set up a new feature to indicate whether a country belongs to the Belt and Road (B&R) initiative. B&R aims to actively develop economic and cultural exchanges with countries along the route [18]. At present, there are 65 countries and regions along with the Belt and Road Initiative, including 10 ASEAN countries, 8 South Asian countries, 5 Central Asian countries, 18 west Asian countries, 7 Commonwealth of Independent States countries, and 16 central and eastern European countries. Most of them are still developing countries.
- Continent: The continent is geographic data. Geography is important to a country's development. It is the reason for great differences in economy, politics, society, and culture among different continents, especially in economic development.

2.2 Benchmark and Baseline Methods

2.2.1 Benchmark

The Fragile States Index (FSI, [13]) is a well-known ranking for evaluating national stability, which is published annually by the American Bimonthly Foreign Policy and The Fund for Peace. The FSI consists of 12 indicators in four aspects: economics, military, politics, and society. The score of each index ranges from 0 to 10, where 0 represents the most stable and 10 represents the least stable, thus forming a score spanning the scale of 0–120. The scores of each indicator are obtained through the process of content analysis (CAST) [19], quantitative resource, and qualitative evaluation by experts. Therefore, the most significant value of FSI is to rank and classify different countries and give different degrees of suggestions to different countries, so that they can better prepare for emergencies [11,20]. In this study, we use FSI ranking as our proposed CR method's benchmark.

2.2.2 Baseline Methods

To evaluate our proposed CR method, we compare with two groups of baselines including traditional machine learning and neural network methods.

1) Traditional Machine Learning
   - Support vector machine (SVM, [21]) is a powerful method for nonlinear problems. Its decision boundary is the maximum margin hyperplane to be solved for the learning sample.
   - Decision tree: Decision tree (DT, [22]) is a widely used non-parametric supervised algorithm, which can be used for classification and regression problems. The decision tree simulates people's decision-making process and makes predictions by deriving simple decision rules from samples.
   - Random forest: Random forest (RF, [23]) is an algorithm based on ensemble learning, which is composed of decision tree and bagging. It utilized multiple decision trees to train the samples in parallel and then integrate them to form a forest to enhance the classification effect and generalization ability.
   - Gradient boost decision tree (GBDT [22]) is also an ensemble learning algorithm, which is composed of many decision trees. GBDT can deal with all kinds of data flexibly and has good prediction performance.
   - Extreme gradient boosting (XGBoost [24]) is essentially a GBDT, but it has made many improvements to the GBDT algorithm, which greatly improves the speed and efficiency of training. Compared with GBDT, XGBoost adds a regular term to the cost function, which makes the algorithm simpler. In addition, XGBoost utilizes the method of the random forest to support column sampling, which can not only alleviate the overfitting but also reduce the calculation.
   - CatBoost, proposed by [25], is a machine learning method based on gradient boosting over decision trees with the aim to deal with the category characteristics efficiently. Currently, CatBoost can be widely used in a variety of fields and problems. It does not need too many tuning parameters to get strong performance and can effectively prevent overfitting, which also makes the model robust. But it takes a lot of memory and time to process the categorical features.
   - NGBoost was proposed by [26]. It is a boosting method based on natural gradients. This method can directly get the full probability distribution in output space, which can be used for probability prediction to quantify uncertainty. It aims to solve the problem of general probabilistic prediction which is difficult to be handled by existing gradient promotion methods. Currently, NGBoost prediction is much more competitive than other boosting methods.
2) Neural Network
   - Multi-layer perceptron (MLP, [27]), which is composed of an input layer, a hidden layer, and an output layer, is a simple neural network and the basis of other neural network structures.
   - Convolutional neural network (CNN, [28]) is a feedforward neural network, which is one of the representative methods of deep learning. It usually includes a convolution layer, pooling layer, and full connection layer. CNN can share convolution kernel globally and process high-dimensional data.
   - Long short-term memory (LSTM) [29] is a kind of recurrent neural network. Compared with the general neural network, it can deal with the data of sequence change. It aims to solve the problem of gradient disappearance and gradient explosion during long sequence training.
   - Gated recurrent unit (GRU), proposed by [30], is similar to the basic concept and regarded as a variant of LSTM. The GRU has a simpler structure than LSTM, and it is faster to learn and train in data matching.
   - Model-agnostic meta-learning (MAML) was proposed by [31]. It is a method based on meta-learning. MAML is used to adjust the initial parameters by one or more steps, and it achieves the goal of quickly adapting to a new task with only a small amount of data.

3 PROPOSED CR METHOD

In this section, we introduce our proposed CR method, which is based on the idea of ensemble learning. Due to each classification method is learned by data-driven which has its specific for feature selection. We choose the best classification method under different features and aggregate their predicted results to obtain better prediction results.

3.1 Stage Division

In essence, machine learning is a data-driven inductive bias method, that is, obtaining general knowledge from limited known data. Therefore, different classification methods may have different induction preferences, and the predicted results may also be different. First, we input the data into our method, which include features and labels. Next, we have the following steps: 1) The label of the input data is divided into N stages according to ranking with a certain ratio. 2) The input data are applied to various machine learning methods, and we can observe the best method performance under each stage. 3) The method and result with the best performance under each stage are chosen.

3.2 Ranking Aggregation

We chose the optimal method for each stage, and the optimal ranking of each stage can be obtained. But we cannot simply aggregate them because the problem of information interference will influence the final results. It is necessary to improve the robustness of aggregate ranking. In our method, we apply the Borda count method to aggregate the ranking. The Borda count method [32] is a traditional ranking aggregation method. Its main idea is to get a score for each element of each ranking list, which measures the gap between that element and other elements. Then, adding the scores of each element in the list produces a Borda number for each element. The ranking aggregation process is as follows:

$$R_t = \sum_{j\left(j \neq i\right)} \left(r_{ti} - r_{tj}\right), \tag{1}$$

where r_{ti} denotes the rank of element t in the i-th position. By ranking the Borda number of elements, the final ranking results are obtained.

Ranking aggregation is better than the results given by a single classification method because it can take advantage of each method and aggregate them; it is an implementation of ensemble learning for ideas. The application of the Borda count method for our method is necessary; the method can smooth the samples that predict errors in each stage during the aggregation process. Therefore, it enables our method to have strong anti-interference and universality, which improves the robustness.

4 EXPERIMENTS

To verify the effectiveness of our model, we conducted experiments on real data to evaluate national stability.

4.1 Evaluation Metrics

- Pearson correlation coefficient (PCC, [33]) is used to measure the degree of linear correlation between the two features of data α and β. According to the Cauchy–Schwarz inequality, the PCCs range from −1 to +1. Specifically, 1 represents the total positive linear correlation between the two features, 0 represents the wireless correlation, and −1 represents the total negative linear correlation between the two features. It is currently the most commonly used method to analyze the distribution trend and change trend consistency of the two sets of data and is widely used in the scientific field. The expression formula is as follows:

$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{\mathrm{E}\left[X - \mu_X\right)\left(Y - \mu_Y\right)\right]}{\sigma_X \sigma_Y}. \tag{2}$$

- Accuracy (ACC, [34]) refers to the proportion of correctly classified samples to the total samples. It does not consider whether the predicted samples are positive or negative. ACC is the most common metric. According to the results, the higher the ACC, the better will be the classifier. The expression formula is as follows:

$$\mathrm{ACC} = \frac{T_P + T_N}{T_P + T_N + F_P + F_N}, \tag{3}$$

where T_P means that positive samples are correctly predicted as positive, T_N means that negative samples are correctly predicted as negative, F_P means that negative samples are incorrectly predicted as positive, and F_N means that positive samples are incorrectly predicted as negative.

In brief, ACC represents the precision of absolute ranking, while the PCCs represent the precision of relative ranking. We choose these two metrics to evaluate the results from different facets.

4.2 Experimental Settings

Since the collected datasets cover different years and the number of countries included, to facilitate subsequent calculations, data cleaning will be performed on all sample countries and years. Therefore, we removed the countries with missing data and took the intersection of the remaining by year and country. Finally, we split the data resources from 2010 to 2014 of 122 countries, comprising a total of 610 rows, with each piece of data containing 7 features including improved water, carbon emissions, cultivated land, Climate Risk Index, International Trade Network, B&R, continent, and FSI ranking as labels. All features are shown in **Figure 1** and **Figure 2**. Then we utilize the data from 2010 to 2013 as the training set and the data from 2014 as the test set for verification. This experiment was run on a server with 1 NVIDIA Tesla V100 GPU. The operating system is Centos 7.5, and all the codes are implemented in the Python environment. The parameter settings are listed in **Supplemental Data**.

4.3 Results on Classification Task

To explore the most effective machine learning algorithm for classification, we divide the status of the country into three stages

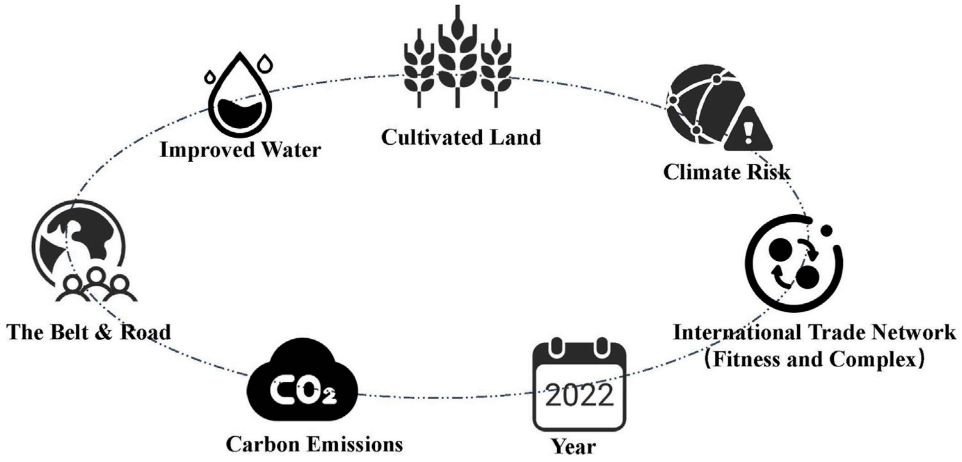


FIGURE 1 | Data description. Each piece of data contains 8 features.

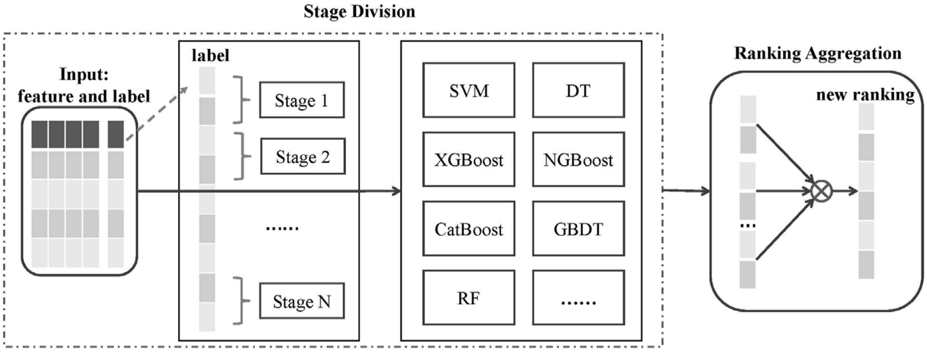


FIGURE 2 | Overview of the CR framework.

TABLE 1 | Results of status classification. The data in bold represent the best performances in these methods.

Method	ACC	PCCs
SVM	0.6762	0.7304
DT	0.8729	0.8913
RF	0.9303	0.9449
GBDT	0.8975	0.9193
XGBoost	0.9139	0.9305
CatBoost	**0.9426**	**0.9524**
NGBoost	0.8484	0.8757
MLP	0.8827	0.9019
CNN	0.8114	0.8407
LSTM	0.7950	0.8082
GRU	0.8852	0.9033
MAML	0.7981	0.8113

The bold data means the best performance in these methods.

based on the ranking of the FSI: fragile, warning, and stable, with a ratio of 3:4:3. Specifically, the countries ranked from 1 to 37 are fragile, the countries ranked from 38 to 86 are warning, and the countries ranked from 87 to 122 countries are stable. Then, we use 7 features as input, the country's status as label, and ACC and PCCs as evaluation metrics. We use traditional machine learning and neural network methods to make predictions. The results are shown in **Table 1**.

Taking the results in **Table 1**, we have the following observations. On the whole, nearly half of the methods have ACC and PCCs above 0.9. It is demonstrated that the CatBoost method performance is the best; its ACC is achieved as high as 0.9426 and PCC as high as 0.9524. In terms of methods, traditional machine learning methods perform more prominently, especially CatBoost and RF methods based on ensemble learning (two of the best performances in all evaluation metrics). In the neural network methods, only MLP and GRU reach the general level, while LSTM and MAML do not perform well. The performance of each method is shown in **Figure 3**.

4.4 Results on Ranking Task

Based on the result of status classification, we found that using classification methods can achieve good prediction performance. Thus, we propose an idea—ranking by classification. It uses classification methods to solve the ranking problem. We can

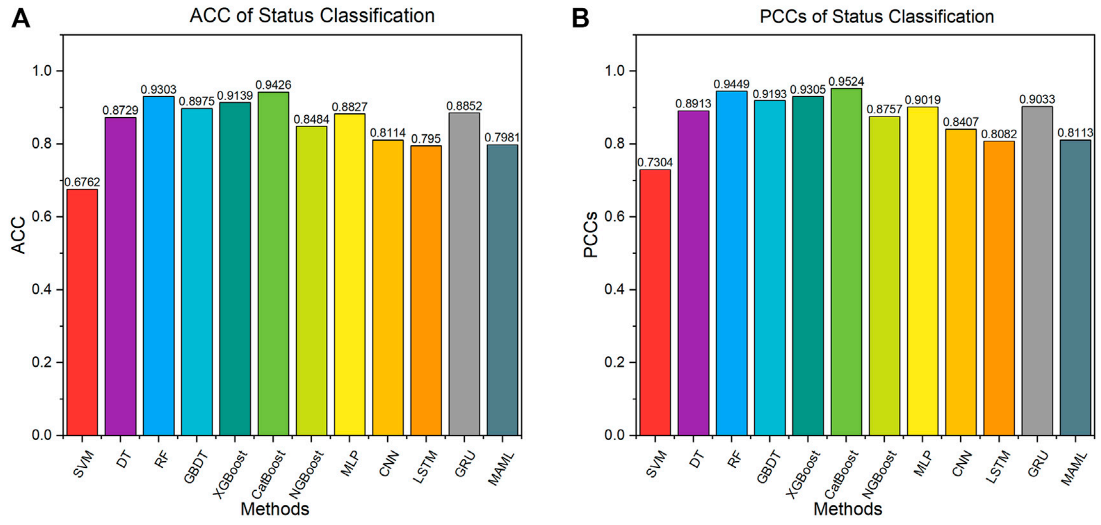


FIGURE 3 | Result of status classifications. **(A)** ACC of status classification. **(B)** PCCs of status classification

TABLE 2 | Results of methods under different stages: fragile, warning, and stable. The data in bold represent the top three performances in these methods.

Fragile			Warning			Stable		
Method	**ACC**	**PCCs**	**Method**	**ACC**	**PCCs**	**Method**	**ACC**	**PCCs**
SVM	0.2432	0.2818	SVM	0.0816	0.5229	SVM	0.2222	0.4664
DT	**0.3243**	0.5172	DT	0.1429	**0.7979**	DT	0.3333	0.7026
RF	**0.3510**	**0.7809**	RF	**0.1837**	**0.8044**	RF	**0.5000**	**0.9106**
GBDT	0.1622	0.3665	GBDT	0.1224	0.6494	GBDT	0.3056	0.4412
XGBoost	0.1892	0.5787	XGBoost	0.0816	0.4464	XGBoost	0.2222	**0.7551**
CatBoost	**0.3514**	**0.9054**	CatBoost	**0.2245**	**0.7759**	CatBoost	**0.3889**	**0.9051**
NGBoost	0.1351	0.4249	NGBoost	0.0612	0.2013	NGBoost	0.1389	0.5077
MLP	0.0811	0.1778	MLP	0.0408	0.4503	MLP	0.1389	0.6776
CNN	0.1081	0.5742	CNN	0.0816	0.5632	CNN	0.0833	0.6242
LSTM	0.1081	0.139	LSTM	0.102	0.2725	LSTM	0.1667	0.6676
GRU	0.1892	0.5456	GRU	0.102	0.4405	GRU	0.1389	0.5707
MAML	0.2432	**0.6124**	MAML	**0.1633**	0.6776	MAML	**0.3611**	0.4955

The bold data means the three best performance in these set of methods.

continue to explore the prediction accuracy of each model under the label for FSI ranking. In addition, we can also observe the pros and cons of different methods in different countries' status.

From the results given in **Table 2**, we have the following observations: In general, the ACC and PCCs of different methods at different stages are inconsistent, which means that different methods have different preferences for data fitting. For example, in the fragile stage, the ACC and PCCs of RF are not as good as CatBoost, but the RF's performance in the stable stage is indeed much higher than the performance of other methods. This shows that in our task, RF predicts the lower ranked (stable) countries more accurately. In terms of methods, CatBoost and RF are still the best performing methods. In the fragile stage, the highest performing method for ACC and PCCs is CatBoost. In the warning stage, the best performance of ACC is achieved using CatBoost, and the best performance of PCCs is with RF. In the stable stage, the best performance is achieved using RF. Besides, both ACC and PCCs of the neural network–based MAML show that the performance is better than other neural network methods. This demonstrates that in our small data sample, MAML can also have its own advantages in small sample learning. However, compared with methods based on ensemble learning, neural networks appear a serious overfitting phenomenon in our task, which may be caused by too little data. It also demonstrates that traditional machine learning methods can still perform well in small datasets. We compared the top three methods of evaluation metrics at each stage and drew scatter charts to more intuitively indicate the similarity between the rankings. The results are shown in **Figures 4A–F**; **Figure 5**.

Based on the aforementioned observation, we utilized the idea of ensemble learning to aggregate the optimal methods at each stage, which will be an effective ranking method. We choose the method with the highest evaluation metrics for each stage. According to different metrics, we can get one method based on ACC, named CR-

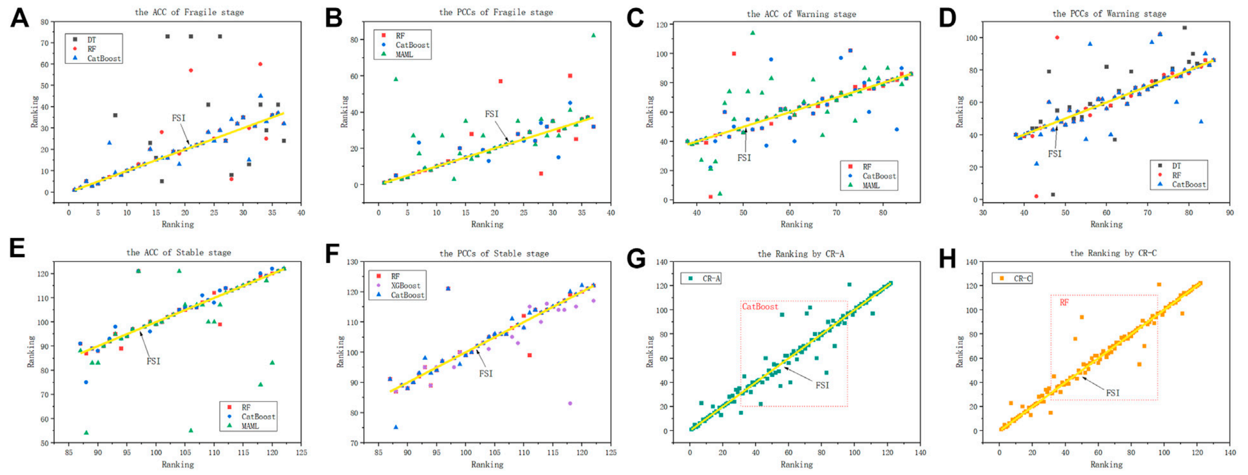


FIGURE 4 | (A–F) Ranking of different methods in different stages. **(G)** Ranking obtained by the ACC-based CR method. **(H)** Ranking obtained by the PCC-based CR method.

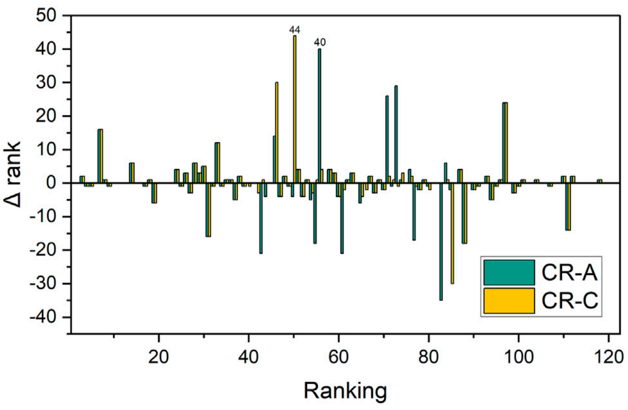


FIGURE 5 | Difference between CR ranking and FSI ranking. The Δ rank represents the error value between CR ranking and FSI ranking. The error value is obtained by subtracting the FSI ranking from the CR ranking.

TABLE 3 | Results of ranking comparison. The data in bold represent the best performances in these methods.

Method	ACC	PCCs
SVM	0.1721	0.8540
DT	0.2270	0.9469
RF	0.3213	0.9605
GBDT	0.2066	0.8860
XGBoost	0.1557	0.9305
CatBoost	0.3115	0.9614
NGBoost	0.1066	0.7954
MLP	0.0902	0.8458
CNN	0.2049	0.9104
LSTM	0.1393	0.8256
GRU	0.2131	0.9034
MAML	0.2459	0.9125
CR-A	**0.3442**	0.9730
CR-C	0.3278	**0.9808**
Improvement (%)	+7.13	+2.02

The bold data means the best performance in these methods.

A, and another method based on PCCs, named CR-C. The CR-A method is composed of CatBoost-CatBoost-RF, which is the method that achieved highest ACC in each stage, and the CR-C method is composed of CatBoost-RF-RF, which is the method that achieved the highest PCCs in each stage.

For comparison, we used the FSI ranking as label and then utilized traditional machine learning and neural network methods for modeling and comparison with the CR method. The results are given in **Table 3**.

Taking the results in **Table 3**, it is demonstrated that our method consistently outperforms baseline in terms of ACC and PCCs. Specifically, CR-A outperforms the baseline methods for ACC (7.13% improvement), while CR-C outperforms the baseline methods for PCCs (2.02% improvement). This illustrates the effectiveness of our method based on the idea of ensemble learning. It is probably because it utilizes the idea of ensemble learning to aggregate the best learning methods at each stage that can deeply fit the data, which improves the prediction accuracy and correlation coefficient.

Next, we explore the difference between CR-A and CR-C methods. In order to observe more intuitively, we use scatterplots to show the ranking obtained by two methods, and the result is shown in **Figures 4G,H**. In addition, we use **Δ** rank to represent the ranking error value. The expression formula is as follows:

$$\Delta rank_i = rank(A)_i - rank(B)_i, \tag{4}$$

where i represents an item, A and B represent different ranking methods, $rank(A)_i$ is the ranking of item i under method A, and $rank(B)_i$ is the ranking of item i under method B.

Considering the results in **Figures 4G,H**, we observe that the difference between the two methods is mainly due to the slightly larger difference in rankings in the middle segment. The CR-A method has a small error value (use Δ rank to indicate) in ranking fluctuations, but the number of error rankings is large. While the

CR-C method gets a larger Δ rank in the fluctuation of a ranking, but the number of error rankings is small.

In order to understand the difference between the two methods more clearly, we calculate the Δ rank between the CR ranking and FSI ranking. As shown in 5, the maximum Δ rank of CR-A is 40, but there are 13 error rankings with Δ rank greater than 10, and 6 error rankings with Δ rank greater than 20. The CR-C has a maximum Δ rank of 44, but there are only 9 error rankings with Δ rank greater than 10, and 4 error rankings with Δ rank greater than 20. We found that the rankings obtained by the two methods are highly convergent with FSI ranking, and they all have their own advantages. The results of ranking by CR-A and CR-C are shown in **Supplemental Data**.

5 DISCUSSION

This work mainly focuses on quantitative data to obtain a new ranking that is similar to the FSI ranking. In this study, we propose a new method—CR, based on the idea of ensemble learning. We examined stage division for labels, which utilizes some classifiers to extract complex patterns from features in each stage and aggregates the best performing ranking from each stage. Compared with the FSI, our method pays more attention to the quantification of ways, reducing the impact of abstract factors and expert qualitative indicators, which makes the new ranking more quantitative and interpretable. The experimental results show that our method is able to improve both accuracy and coefficient compared to the state-of-the-art methods.

In our experiment, we compared few traditional machine learning and neural network methods, but the results show that the prediction effect of the neural network is not as good as traditional machine learning, especially methods based on ensemble learning. The possible reason is that for the small sample data, the neural network is prone to overfitting so that it cannot perform excellent generalization performance. In addition, the experimental results also demonstrate that ensemble learning methods such as CatBoost and RF are more suitable for our work. One interesting future direction of this work is to adopt brain drain and electrical energy, which improve the method from balancing robust and accuracy.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/**Supplementary Material**; further inquiries can be directed to the corresponding authors.

AUTHOR CONTRIBUTIONS

XL and HL contributed to conceptualization, methodology, and writing—original draft preparation. XL helped with data generating and collection. XL and HL assisted with experiments and discussions. XL, AV, HL, and KL involved in writing—reviewing and editing.

FUNDING

We acknowledge the financial support from the National Natural Science Foundation of China (Grant Nos. 61 803 266 and 71 874 172), Guangdong Province Natural Science Foundation (Grant Nos. 2019A1515011173 and 2019A1515011064), and Shenzhen Fundamental Research-general project (Grant No. JCYJ20190808162601658).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fphy.2022.830774/full#supplementary-material

REFERENCES

1. Perrings C. Environmental Management in Sub-Saharan Africa In *Sustainable Development and Poverty Alleviation in Sub-Saharan Africa* (1996)29–43. doi:10.1007/978-1-349-24352-5_3
2. Busby JW, Busby J *Climate Change and National Security: An Agenda for Action*, 32. Council on Foreign Relations Press (2007).
3. Wang L, Wu JT. Characterizing the Dynamics Underlying Global Spread of Epidemics. *Nat Commun* (2018) 9:218–1. doi:10.1038/s41467-017-02344-z
4. Du Z, Wang L, Yang B, Ali ST, Tsang TK, Shan S, et al. Risk for International Importations of Variant Sars-Cov-2 Originating in the united kingdom. *Emerg Infect Dis* (2021) 27:1527–9. doi:10.3201/eid2705.210050
5. Vörösmarty CJ, Green P, Salisbury J, Lammers RB. Global Water Resources: Vulnerability from Climate Change and Population Growth. *science* (2000) 289:284–8. doi:10.1126/science.289.5477.284
6. Görg C, Brand U, Haberl H, Hummel D, Jahn T, Liehr S. Challenges for Social-Ecological Transformations: Contributions from Social and Political Ecology. *Sustainability* (2017) 9:1045. doi:10.3390/su9071045
7. Portes A, Sensenbrenner J. Embeddedness and Immigration: Notes on the Social Determinants of Economic Action. *Am J Sociol* (1993) 98:1320–50. doi:10.1086/230191
8. Hechter M. *The Modern World-System: Capitalist Agriculture and the Origins of the European World-Economy in the Sixteenth century* (1975).
9. Sassen S. *Cities in a World Economy*. Thousand Oaks, CA: Sage Publications (2018).
10. Gao C, Zhong L, Li X, Zhang Z, Shi N. Combination Methods for Identifying Influential Nodes in Networks. *Int J Mod Phys C* (2015) 26:1550067. doi:10.1142/s0129183115500679
11. Carment D, Samy Y *State Fragility: Country Indicators for Foreign Policy Assessment*, 94. Amsterdam, Netherlands: The Development (2012).
12. Backer DA, Huth PK. The Peace and Conflict Instability Ledger: Ranking States on Future Risks. In: *Peace and Conflict 2016*. Oxfordshire, England: Routledge (2016). p. 128–43. doi:10.4324/9781003076186-2
13. Carlsen L, Bruggemann R. Fragile State index: Trends and Developments. A Partial Order Data Analysis. *Soc Indic Res* (2017) 133:1–14. doi:10.1007/s11205-016-1353-y
14. Dietterich TG. Ensemble Methods in Machine Learning. In: *International Workshop on Multiple Classifier Systems*. Springer (2000). p. 1–15. doi:10.1007/3-540-45014-9_1
15. Zhu P, Lv R, Guo Y, Si S. Optimal Design of Redundant Structures by Incorporating Various Costs. *IEEE Trans Rel* (2018) 67:1084–95. doi:10.1109/tr.2018.2843181

16. Kreft S, Eckstein D, Melchior I. *Global Climate Risk index 2014. Who Suffers Most from Extreme Weather Events 1*. Bonn, Germany: Germanwatch (2013).
17. Tacchella A, Cristelli M, Caldarelli G, Gabrielli A, Pietronero L. A New Metrics for Countries' Fitness and Products' Complexity. *Sci Rep* (2012) 2:723–7. doi:10.1038/srep00723
18. Huang Y. Understanding China's Belt & Road Initiative: Motivation, Framework and Assessment. *China Econ Rev* (2016) 40:314–21. doi:10.1016/j.chieco.2016.07.007
19. Baker P. *The Conflict Assessment System Tool (Cast): An Analytical Model for Early Warning and Risk Assessment of Weak and Failing States*. Washington: The fund for Peace (2006).
20. Sekhar CSC. Fragile States. *J Developing Societies* (2010) 26:263–93. doi:10.1177/0169796x1002600301
21. Steinwart I, Christmann A. *Support Vector Machines*. Springer Science & Business Media (2008).
22. Safavian SR, Landgrebe D. A Survey of Decision Tree Classifier Methodology. *IEEE Trans Syst Man Cybern* (1991) 21:660–74. doi:10.1109/21.97458
23. Kuncheva LI. Combining Pattern Classifiers: Methods and Algorithms[M]. *John Wiley & Sons* (2014). doi:10.1198/tech.2005.s320
24. Chen T, Guestrin C. Xgboost: A Scalable Tree Boosting System. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining; August 2016 (2016). p. 785–94.
25. Prokhorenkova L, Gusev G, Vorobev A, Dorogush AV, Gulin A. Catboost: Unbiased Boosting with Categorical Features. *Adv Neural Inf Process Syst* (2018) 31:6639–49.
26. Duan T, Anand A, Ding DY, Thai KK, Basu S, Ng A, et al. Ngboost: Natural Gradient Boosting for Probabilistic Prediction. In: *International Conference on Machine Learning*. Online: PMLR (2020). p. 2690–700.
27. Mitra S, Pal SK. Fuzzy Multi-Layer Perceptron, Inferencing and Rule Generation. *IEEE Trans Neural Netw* (1995) 6:51–63. doi:10.1109/72.363450
28. Krizhevsky A, Sutskever I, Hinton GE. Imagenet Classification with Deep Convolutional Neural Networks. *Adv Neural Inf Process Syst* (2012) 25:1097–105.
29. Hochreiter S, Schmidhuber J. Long Short-Term Memory. *Neural Comput* (1997) 9:1735–80. doi:10.1162/neco.1997.9.8.1735
30. Cho K, van Merrienboer B, Gülçehre Ç, Bahdanau D, Bougares F, Schwenk H, et al. Learning Phrase Representations Using Rnn Encoder-Decoder for Statistical Machine Translation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP) (2014). doi:10.3115/v1/d14-1179
31. Finn C, Abbeel P, Levine S. Model-agnostic Meta-Learning for Fast Adaptation of Deep Networks. In: Proceeding of the International Conference on Machine Learning (PMLR); August 2017 (2017). p. 1126–35.
32. Emerson P. The Original Borda Count and Partial Voting. *Soc Choice Welf* (2013) 40:353–8. doi:10.1007/s00355-011-0603-9
33. Benesty J, Chen J, Huang Y, Cohen I. Pearson Correlation Coefficient. In: *Noise Reduction in Speech Processing*. Springer (2009). p. 1–4. doi:10.1007/978-3-642-00296-0_5
34. Diebold FX, Mariano RS. Comparing Predictive Accuracy. *J Business Econ Stat* (2002) 20:134–44. doi:10.1198/073500102753410444
35. Xing J, Huang X, Cao Y, Liao L, Liao H. Data-Driven Correlation Analysis Between Environmental Change and Regional Economic Growth[J]. *Journal of Nanjing University of Information Science and Technology(Natural Science Edition)* (2019) 11 (03):326–331. doi:10.13878/j.cnki.jnuist.2019.03.011

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

ORIGINAL RESEARCH
published: 21 February 2022
doi: 10.3389/fphy.2022.829734

A Deep Learning Framework for Video-Based Vehicle Counting

Haojia Lin[1,2], *Zhilu Yuan*[2], *Biao He*[2,3,4], *Xi Kuai*[2], *Xiaoming Li*[2] *and Renzhong Guo*[1,2]*

[1]*School of Resource and Environmental Sciences, Wuhan University, Wuhan, China,* [2]*Guangdong–Hong Kong-Macau Joint Laboratory for Smart Cities, and Shenzhen Key Laboratory of Digital Twin Technologies for Cities, and Research Institute for Smart Cities, School of Architecture and Urban Planning, Shenzhen University, Shenzhen, China,* [3]*MNR Technology Innovation Center of Territorial & Spatial Big Data, Shenzhen, China,* [4]*MNR Key Laboratory of Urban Land Resources Monitoring and Simulation, Shenzhen, China*

Traffic surveillance can be used to monitor and collect the traffic condition data of road networks, which plays an important role in a wide range of applications in intelligent transportation systems (ITSs). Accurately and rapidly detecting and counting vehicles in traffic videos is one of the main problems of traffic surveillance. Traditional video-based vehicle detection methods, such as background subtraction, frame difference, and optical flow have some limitations in accuracy or efficiency. In this paper, deep learning is applied for vehicle counting in traffic videos. First, to solve the problem of the lack of annotated data, a method for vehicle detection based on transfer learning is proposed. Then, based on vehicle detection, a vehicle counting method based on fusing the virtual detection area and vehicle tracking is proposed. Finally, due to possible situations of missing detection and false detection, a missing alarm suppression module and a false alarm suppression module are designed to improve the accuracy of vehicle counting. The results show that the proposed deep learning vehicle counting framework can achieve lane-level vehicle counting without enough annotated data, and the accuracy of vehicle counting can reach up to 99%. In terms of computational efficiency, this method has high real-time performance and can be used for real-time vehicle counting.

Keywords: intelligent transportation systems, traffic video, vehicle detection, vehicle counting, deep learning

OPEN ACCESS

Edited by:
Chao Gao,
Southwest University, China

Reviewed by:
Yu Lin,
Ningbo University, China
Junjie Pang,
Qingdao University, China
Qiuyang Huang,
Jilin University, China

****Correspondence:***
Renzhong Guo
guorz@szu.edu.cn

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *06 December 2021*
Accepted: *11 January 2022*
Published: *21 February 2022*

Citation:
Lin H, Yuan Z, He B, Kuai X, Li X and Guo R (2022) A Deep Learning Framework for Video-Based Vehicle Counting. Front. Phys. 10:829734. doi: 10.3389/fphy.2022.829734

1 INTRODUCTION

The rapid growth of the urban population and motor vehicles has led to a series of traffic problems. Intelligent transportation systems (ITSs) are considered the best tool to solve these problems. With the development of the Internet of Things (iot) technology, communications technology and computer vision, traffic surveillance has become a major technology of traffic parameter collection and plays a crucial role [1–3]. Traffic flow is an important basic parameter in ITS [4, 5], and accurately and rapidly detecting and counting vehicles based on traffic videos is a common research topic. Over the last decades, various vision approaches have been proposed to automatically count vehicles in traffic videos. Many existing vehicle counting methods rely on a vehicle detector based on the vehicle's appearance and features that are located *via* foreground detection, and vehicles are counted based on vehicle detection results [6–8]. In general, the methods for vehicle counting based on traffic videos can be divided into two subtasks: vehicle detection and vehicle counting.

In vehicle detection, the common methods include background subtraction [9, 10], frame difference [11, 12], optical flow [13, 14] and deep learning object detection [15–18]. The first three methods detect vehicles through manually extracted features, which are relatively simple, but they also have some limitations in accuracy or robustness. Instead of manually extracting features, the deep learning method simulates information processing of the human brain and enables the constructed network to perform automatic feature extraction by training on a large annotated dataset [19]. However, this method relies on a large training dataset and is difficult to apply to various traffic video scenarios. Transfer learning can be combined with deep learning to build a model of target tasks based on source tasks [17, 20–22], but combining transfer learning with deep learning in the absence of annotated data is still an important research direction to study.

For vehicle counting, the usual approaches can be mainly divided into two categories: vehicle counting based on the virtual detection area [23–25] and vehicle counting based on vehicle tracking [26–28]. The virtual detection area sets up virtual detection areas in a video to determine whether there are vehicles passing through according to the change of the area's grey value, which is highly efficient, but it easily makes mistakes due to lane changing or parallel driving. Vehicle tracking extracts the trajectory of each vehicle by matching vehicles detected in each frame of video sequences and counts the number of vehicles based on vehicle trajectories, which has a high accuracy, but the computing cost is relatively high.

With the development of deep learning object detection and the computability of hardware, such as GPUs, it is possible to build a deep learning vehicle counting model with high accuracy and efficiency, but there are still some challenges. On the one hand, it is time-consuming to build a training dataset for a specific traffic scenario. The primary problem is constructing a vehicle detection model with good performance rapidly through transfer learning in the absence of training data. On the other hand, although deep learning vehicle detection can detect vehicles more accurately, it is still inevitable that situations of missing detection or false detection occur. Avoiding errors caused by these situations is a key problem to solve to further improve the accuracy of vehicle counting when the accuracy of vehicle detection is difficult to improve.

In this paper, a deep learning framework for vehicle counting is proposed. To solve the problem of lacking training data, a method of vehicle detection model construction based on transfer learning and open datasets is proposed, which can build a vehicle detection model with high-quality performance rapidly in the absence of training data. Moreover, for the possible situations of missing detection and false detection, a vehicle counting method based on fusing the virtual detection area and vehicle tracking is proposed, which can avoid the errors caused by missing detection or false detection and further improve the accuracy of vehicle counting.

2 PROPOSED FRAMEWORK

In this section, we introduce the proposed deep learning framework for video-based vehicle counting. As shown in **Figure 1**, the framework can be divided into three stages: dataset building, vehicle detection model construction, and vehicle counting. First, to build a training dataset for deep learning vehicle detection model construction and avoid spending too much time labelling images, annotated images containing vehicles in the open dataset are extracted as training data. In addition, to further improve and evaluate the performance of the vehicle detection model, a few frame images in traffic videos are extracted and divided into supplemental training data and testing data after being labelled. Next, in the vehicle detection model construction stage, a deep learning object detection model that meets the requirements in terms of accuracy and efficiency is utilized as the basic model, and instance-based transfer learning and parameter-based transfer learning are adopted to construct a vehicle detection model. Finally, in the vehicle counting stage, vehicles in each frame are detected by the deep learning vehicle detection model and counted by the vehicle counting model based on the fusion of virtual detection area and vehicle tracking, where a missing alarm suppression module based on vehicle tracking and a false alarm suppression module based on bounding box size statistics are designed to avoid the vehicle counting error caused by missing detection or false detection.

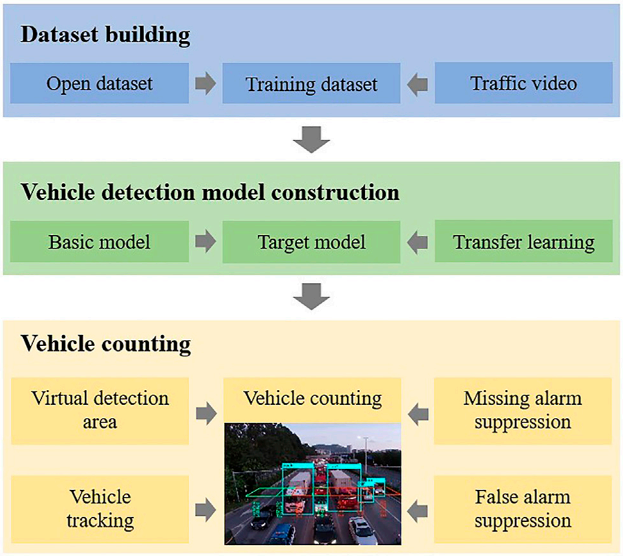


FIGURE 1 | Framework for video-based vehicle counting.

3 VEHICLE DETECTION

In the vehicle detection stage, a deep learning object detection method is adopted. Specifically, to solve the problem of lacking training data, transfer learning is used to construct a deep learning vehicle detection model. Employing a deep learning object detection model as the basic model and vehicle data in open datasets as the training data, the vehicle detection model is constructed by transfer learning.

3.1 Deep Learning Object Detection

In deep learning object detection models, a convolutional neural network (CNN) is used to extract image features, and then a classifier and regressor are used to classify and locate the extracted features. The existing models can be mainly divided into two categories: two-stage detectors and one-stage detectors. Two-stage detectors consist of a region proposal network to feed region proposals into a classifier and regressor, which have high object recognition and localisation accuracy, but the efficiency is low, such as in the R-CNN [29], Fast R-CNN [30] and Faster R-CNN [31]. One-stage detectors regard object detection as a regression problem, which takes the entire image into a classifier and regressor and predicts objects directly without a region proposal step. Thus, they run faster than two-stage detectors, such as SSD [32] and YOLO [33–35].

In vehicle counting, accuracy and efficiency are both important. Therefore, a one-stage detector seems to be a better choice for vehicle detection model construction. As a typical representative of one-stage detectors, YOLO has better performance in terms of efficiency and accuracy than many other detectors, and the trade-off between accuracy and efficiency can be made according to the requirements [35]. Thus, in this framework, YOLO is selected as the basic model to construct a vehicle detection model. The implementation of YOLO is as follows:

1) A series of convolutional layers and residual layers are used to extract the features of the input image, and finally, three feature maps with different scales are obtained.
2) Each feature map is divided into S × S grids, and B anchor boxes are set for each grid cell to detect objects.
3) If the centre of an object falls into a grid cell, the classification probabilities (including the probability of each classification) and bounding box information (central point coordinate, width, height, and detected confidence) of that object would be detected by that grid cell.
4) The classification with maximum probability and the bounding box with maximum is taken to detect confidence as the output result of each grid cell, and the product of the corresponding classification probability and the bounding box is taken to detect confidence as the final detected confidence.
5) For each feature map, the above processes are carried out, and the final result is obtained by synthesising the results of three scales.

For object detection based on YOLO, the frame images in traffic video are taken as the input, and the detection result is taken as the input of vehicle counting. $BB_j = \{bb_i, i = 1, 2, \ldots, n\}$ represents the detection result of a frame, bb_i represents one bounding box of the detected object in the detection result and consists of six attributes: detected frame number f, central point coordinate (x, y), width w, height h, classification c, and detected confidence p.

3.2 Transfer Learning

Transfer learning aims to extract the knowledge from one or more source tasks and applies them to a target task [36]. According to the representation of transferred knowledge, transfer learning can be classified into four categories:

1) Instance-based transfer learning: Certain parts of data in the source task can be reused and combined with data in the target task to train a target model.
2) Parameter-based transfer learning: Some parameters, prior distributions, or hyperparameters of the models are shared between the source and the target task to improve the effectiveness of learning.
3) Feature-representation transfer learning: A good feature representation that reduces the difference between the source and target tasks is found and the knowledge used to transfer into the learned feature representation is encoded.
4) Relational-knowledge transfer learning: A mapping of relational knowledge between the source and target task is built, and the knowledge to be transferred is transformed into the relationship among the data.

3.3 Transfer Learning Based on YOLO

Traffic videos in this study are captured from fixed cameras above a straight road, including different light conditions, shooting directions, traffic conditions, and resolutions, where cars, buses, and trucks appear and the number of these three vehicle types are counted separately. In deep learning vehicle detection model construction, vehicle annotated data is needed as training data. Some open datasets, such as MS COCO [37], ImageNet [38] and PASCAL VOC [39], contain vehicle annotated data. Although they are different from vehicles in traffic videos, they can be used to build a target model through transfer learning. However, open datasets also have some limitations in this task because most of them are taken in the horizontal direction and fewer are taken from the top tilt down direction. Moreover, there are inevitably some annotation errors in some images. In addition, because of the difference between open datasets and traffic videos, the constructed model with high accuracy in open datasets may not perform well in traffic videos. Thus, it is necessary to build a supplemental dataset by labelling some traffic scenario images, and then using it as training data and testing data to further improve and evaluate the performance of the constructed vehicle detection model.

Based on YOLO and vehicle annotated data in open datasets, transfer learning is used to construct a deep learning vehicle detection model. The process is as follows:

1) Source model construction: YOLO trained by vehicle data in the MS COCO dataset is used as a source model. First, some annotated images containing vehicles (car, bus, and truck) in the MS COCO dataset are extracted. Then, k-means clustering is used on the bounding boxes of the extracted dataset, and nine anchor boxes (3 for each feature map in YOLO) for the model are obtained. Finally, the extracted dataset is used to train YOLO, and a source model Model-1 is obtained.
2) Supplemental dataset building: Model-1 is used to detect frame images in traffic videos, and the results are further processed into a refined annotated dataset. First, some frame images in traffic videos are extracted every fixed interframe space as a dataset to be annotated. Then, Model-1 is used to detect the extracted frame images and utilise the detection results as annotated data. Finally, the annotated data is further

processed to make the annotation more reliable, the false bounding boxes are removed, the missing detected objects are labelled, and the bounding boxes with inaccurate localisation and size are adjusted.

3) Transfer learning: Parameter-based transfer learning and instance-based transfer learning are adopted to construct a deep learning vehicle detection model. First, based on parameter-based transfer learning, the task of Model-1 and the target task are both considered vehicle detection, which have a strong correlation so that the parameters of Model-1 can be used to initialise the network of the target model. Then, based on instance-based transfer learning, the vehicle data in the MS COCO dataset are combined with the supplemental dataset, and k-means clustering is used on the combination to obtain another nine anchor boxes for YOLO. Finally, the merged dataset is used to train the initialised model, and a target model, Model-2, is obtained.
4) Target model optimization: Using training data in the supplemental dataset as training data, we further fine-tune the parameters of Model-2 to obtain a better target model Model-3.

4 VEHICLE COUNTING

In the vehicle counting stage, on the basis of vehicle bounding boxes obtained from vehicle detection, a new method of vehicle counting based on fusing virtual detection area and vehicle tracking is proposed, which considers the possible situation of missing detection and false detection, while combining the ideas of the traditional vehicle counting method based on virtual detection area and vehicle tracking.

4.1 Virtual Detection Area and Vehicle Counting

Considering the real-time requirement of vehicle counting, vehicle counting based on virtual detection area is employed as the basic method, and corresponding improvements are made for the input as a bounding box. The principle of the improved method is shown in **Figure 2**, and the process is as follows:

1) Virtual detection areas are set for each lane of the road section in the video. To ensure that at most one vehicle passes through the detection area at a time, the size of the detection area needs to have some restrictions. The length is close to the length of a car, and the width is similar to the width of the lane. In this case, there is not more than one vehicle passing through the detection area at the same time, which can reduce the complexity of vehicle tracking and vehicle counting.
2) The relative location relationship between the detected vehicle bounding box and the virtual detection area is calculated frame-by-frame, and the status of the detection area is updated. As shown in **Figure 2A**, if there is no central point of the bounding box located in the detection area, the status of that area is $S = 0$, indicating that no vehicle passes through. If there is a central point of the vehicle bounding box detected in the detection area, the status of that area is updated to $S = 1$, indicating that there is one vehicle passing through.
3) According to the status changes of each detection area in the frame sequence, the flow curve of each area can be obtained, and then the vehicle number can be counted. As shown in **Figure 2B**, when the status of an area in a frame is $S = 0$ and becomes $S = 1$ in the next frame, the vehicle number of that area is added to 1.

4.2 Missing Detection and False Detection

Missing detection (**Figure 3**, the yellow bounding box is missing detection) and false detection (**Figure 4**, the red bounding box is false detection) may occur in vehicle detection in continuous frame sequences, which cause vehicle counting errors. As shown in **Figure 5**, two vehicles pass through a virtual detection area, however, the first vehicle is determined to be a new vehicle when it is detected again after missing detection in one or several frames. Thus, the vehicle counting error occurs, and two vehicles are counted as three vehicles.

4.3 The Missing Alarm Suppression Module Based on Vehicle Tracking

To detect missing vehicles, considering that vehicle tracking has the ability to lock on each vehicle, a missing alarm suppression module based on vehicle tracking is added to the vehicle counting model. The module tracks detected vehicles in each area frame-by-frame and determines whether the detected passing vehicle in an area is a new vehicle to avoid incorrect counting caused by missing detection. The process is as follows:

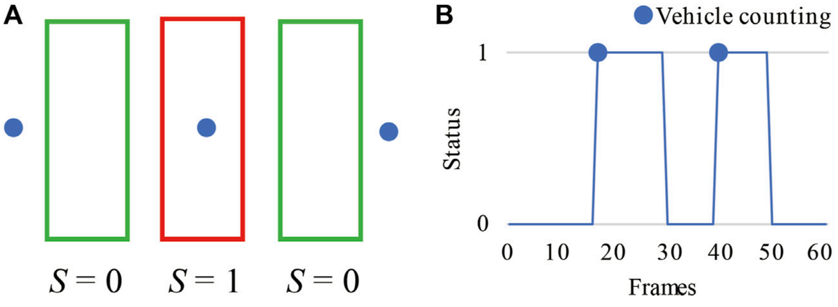


FIGURE 2 | The principle of vehicle counting based on virtual detection areas. **(A)** Status change of a virtual detection area; **(B)** Flow curve of a virtual detection area.

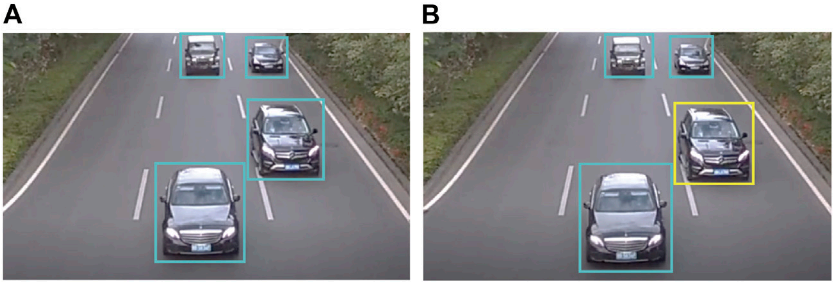


FIGURE 3 | Vehicle missing detection. **(A)** Results of vehicle detection in frame *i*; **(B)** Results of vehicle detection in frame *i* + 1.

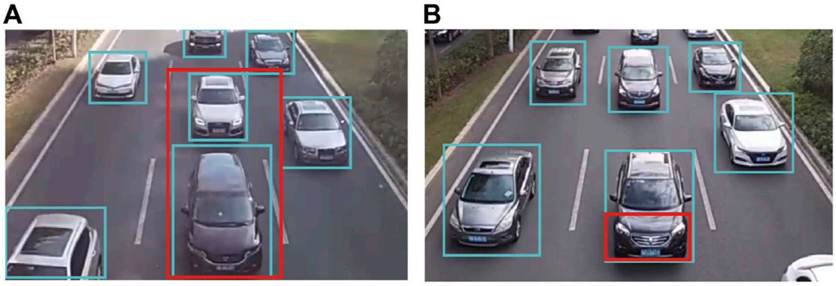


FIGURE 4 | Vehicle false detection. **(A)** Vehicle false detection situation a; **(B)** Vehicle false detection situation b.

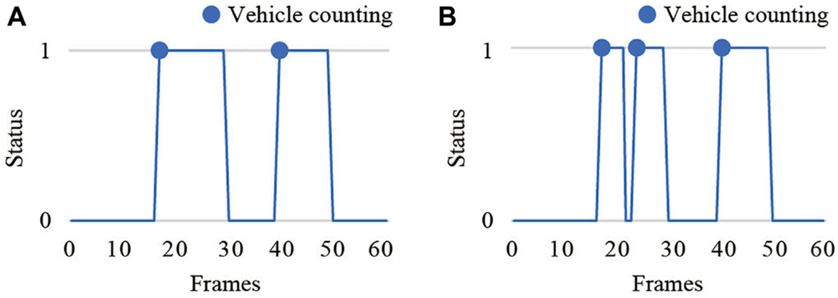


FIGURE 5 | Incorrect vehicle counting caused by vehicle missing detection. **(A)** Flow curve of correct counting; **(B)** Flow curve of incorrect counting.

1) Vehicles are detected in each frame. If the central point of a detected vehicle bounding box is located in an area, the bounding box of that vehicle is marked as the tracking vehicle of that area. bb_i^k is used to represent the detection result of area k in a frame, and tb_k is used to represent the tracking vehicle of area k.
2) Vehicles in the next frame are detected and match each detected vehicle with the tracking vehicle of the corresponding area according to the space-time distance of the vehicles. If there is a match, the tracking vehicle is updated to the detected vehicle. If no matching occurs, the detected vehicle is marked as a new tracking vehicle of the corresponding area.
3) Step 2 is carried out frame-by-frame, the vehicle number of each detection area is calculated based on the flow curve. In addition, to calculate the number of each vehicle type, such as car, bus, and truck, the flow curve of each vehicle type is generated according to the classification of the detected vehicles, and then each kind of number is calculated based on the corresponding kind of flow curve separately.

In vehicle tracking, to ensure the real-time performance of the module, an efficient vehicle tracking method based on the space-time distance of vehicles is used to match the detected vehicle with the tracking vehicle, including the space distance (*SD*) between central points of bounding boxes and the time distance (*TD*) of the bounding boxes detected frame number. The smaller the *SD* or *TD* is, the more likely two detected vehicles are the same vehicle. The calculation is as follows:

$$SD\left(bb_i^k,\ tb_k\right) = \frac{\sqrt{\left(x_{bb_i^k} - x_{tb_k}\right)^2 + \left(y_{bb_i^k} - y_{tb_k}\right)^2}}{\bar{h}_k^{car}} \tag{1}$$

$$TD\left(bb_i^k, tb_k\right) = f_{bb_i^k} - f_{tb_k} \tag{2}$$

where bb_i^k represents the bounding box detected in the current frame in area k, tb_k represents the bounding box of the tracking vehicle of area k, and $\bar{h}_k^{car}$ represents the average height of bounding boxes whose classification is the car in area k. Then, the new vehicle value (NV) is used to determine whether bb_i^k is a new vehicle or not. The calculation is as follows:

$$NV = \begin{cases} 1, & SD\left(bb_i^k,\ tb_k\right) > T_{SD}\ and\ TD\left(bb_i^k,\ tb_k\right) > T_{TD} \\ 0, & otherwise \end{cases} \tag{3}$$

where T_{SD} and T_{TD} represent the SD tracking threshold and the TD tracking threshold, respectively. Considering that a safe distance between vehicles will be kept when vehicles drive and that there is a relationship between this distance and the size of the vehicle, we set $T_{SD} = 1/3$ and $T_{TD} = 5$. If $SD(bb_i^k,\ tb_k)$ is greater than T_{SD} and $TD(bb_i^k,\ tb_k)$ is greater than T_{TD}, bb_i^k is determined to be a new vehicle.

Using this method to track vehicles has a low computing cost, especially compared with deep learning vehicle detection. The cost of this part is negligible, which does not affect the computational efficiency of vehicle counting.

4.4 The False Alarm Suppression Module Based on Bounding Box Size Statistics

For the problem of false vehicle detection, a false alarm suppression module based on the bounding box size statistics is added to the vehicle counting model. The false detection bounding box is usually larger or smaller than those of correct detections. Therefore, the false detection bounding box can be removed by comparing it with the size range of the correct detection. The process is as follows:

1) From the beginning of vehicle detection, the vehicle bounding box detected in each detection area is stored separately according to vehicle classification, and the height of every bounding box of each vehicle classification is averaged as $\bar{h}_k^c$, where c represents the vehicle classification and k represents the detection area.
2) Vehicles in each frame are detected, and the true detection value (TV) is used to determine whether a detected vehicle is a true detection. If it is a false detection, it is removed. The calculation is as follows:

$$TV = \begin{cases} 1, & \left|h_{bb_i^k} - \bar{h}_k^c\right| < T_{size} \times \bar{h}_k^c \\ 0, & otherwise \end{cases} \tag{4}$$

$$TV = \begin{cases} 1, & (1 + T_{size})\bar{h}_k^{car} < h_{bb_i^k} < (1 + T_{size})\bar{h}_k^{bus} \\ 0, & otherwise \end{cases} \tag{5}$$

where $h_{bb_i^k}$ represents the height of bounding box bb_i^k detected in area k and T_{size} is the threshold. Considering that vehicles of the same classification have different sizes, especially trucks, we use **Eq. 4** to determine a car and a bus, and use **Eq. 5** to determine a truck, setting the $T_{size} = 0.5$.

3) Step 1 and step 2 are executed cyclically, and $\bar{h}_k^c$ is updated until the end of the video.

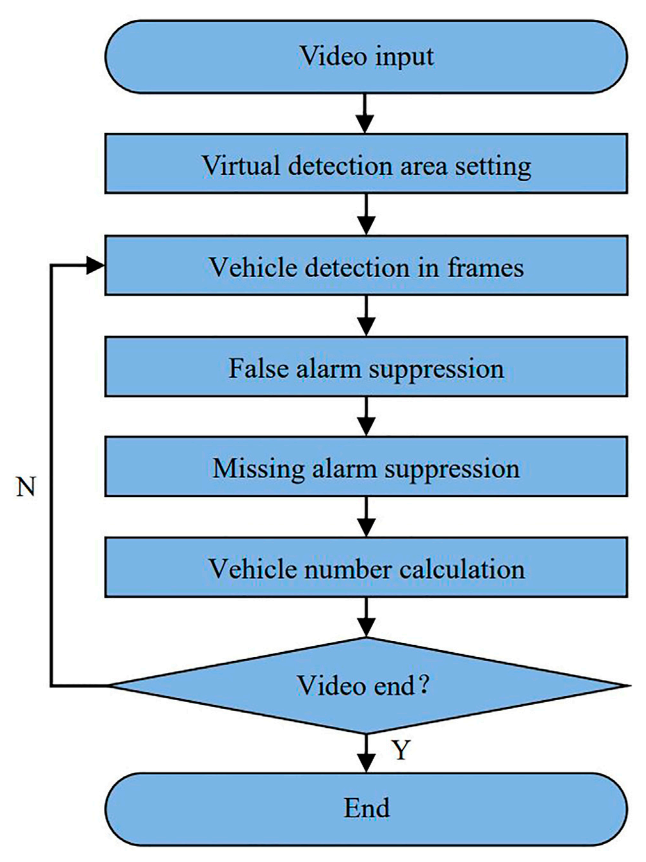

Video input
Virtual detection area setting
Vehicle detection in frames
False alarm suppression
Missing alarm suppression
Vehicle number calculation
Video end?
N
Y
End

FIGURE 6 | The workflow of vehicle counting.

The elimination of false detection may lead to missing detection, but it can be corrected by the abovementioned missing alarm suppression module based on vehicle tracking.

4.5 Vehicle Number Calculation

The process of vehicle counting based on fusing the virtual detection area and vehicle tracking is as follows (**Figure 6**):

1) Virtual detection area setting: according to the vehicle counting task for road sections or lanes, virtual detection areas are set up in each road section or lane.
2) Vehicle detection: Vehicles are detected frame-by-frame and the detected bounding boxes are taken as the output of the detection result, including classification, detected confidence, coordinate of the central point, width, and height.
3) False alarm suppression: In each detection area, according to the size of the detected bounding boxes, a detected bounding box is determined as false detection or not, and if so, false detection is eliminated.

TABLE 1 | Information on traffic videos.

Video	Light	Shooting direction	Traffic conditions	Resolution
1	Day	Front	Traffic lights	1,280 × 720
2	Day	Front	Light	1920 × 1,080
3	Day	Back	Heavy	1920 × 1,440
4	Night	Back	Heavy	1920 × 1,440
5	Day	Front	Traffic lights	1920 × 1,440
6	Night	Front	Traffic lights	1920 × 1,440
7	Night	Front	Traffic lights	1920 × 1,440
8	Night	Back	Light	1920 × 1,440
9	Day	Oblique front	Traffic lights	932 × 500
10	Day	Oblique back	Light	932 × 500

4) Missing alarm suppression: In each detection area, vehicle tracking based on the location of vehicle bounding boxes is carried out on the vehicles detected in the continuous frame sequence to determine whether the detected vehicle is a new vehicle.
5) Vehicle number calculation: Based on the virtual detection area and vehicle tracking, the flow curve of each vehicle type in every detection area is monitored to calculate the vehicle number.

5 EXPERIMENTS

To evaluate the performance of the proposed vehicle counting framework, four experiments are designed and carried out.

The experimental environment is a CPU: Intel Core i7-8700 3.20 GHz; Memory: 16 GB (2,666 MHz); GPU: NVIDIA GeForce GTX 1070, 8 GB.

The experimental data included ten traffic videos with different light conditions, shooting directions, traffic conditions, and resolutions (**Table 1**). All videos are captured for 5 min at 20 frames per second (fps). The MS COCO dataset is used as the basic training data because it has the characteristics of multiple small objects in a noncentral distribution in an image, which is more in line with the daily traffic scenario.

5.1 The Effectiveness of the Supplemental Dataset

Purpose: The purpose of the experiment is to evaluate the effectiveness of the supplemental dataset in the construction of the vehicle detection model.

The supplemental dataset is built as training data and testing data to further improve and evaluate the performance of the constructed vehicle detection model. According to the process introduced in **Section 3.3**, 1,660 images are refined annotated in total and then divided into training data and testing data; 1,000 for training and 660 for testing. In particular, only frame images in videos 1–8 are included, and videos 9 and 10 are used to evaluate the generalisability of the constructed vehicle counting model.

Based on vehicle data in the MS COCO dataset and the supplemental dataset, the following five vehicle detection models are trained, and their accuracies are evaluated. Model-coco is trained by 16,270 images with vehicles in the MS COCO dataset. Model-100, Model-250, Model-500 and Model-1000 are constructed as described in **Section 3.3**, and the number of supplemental training data points is 100, 250, 500 and 1,000, respectively. All models contain frame images of videos 1–8. The testing data for these five models are 660 images in the supplemental dataset, and the mean Average Precision (mAP) is used to represent the accuracy of these models. The mAP score is calculated by taking the mean AP over all classes and overall IoU thresholds, depending on different detection challenges that exist. In this study, AP for one object class is calculated for an Intersection over Union (IoU) threshold of 0.5. So the mAP@0.5 is averaged over all object classes. The accuracy of each model is shown in **Table 2**. As the amount of supplemental training data increases, the accuracy of the vehicle detection model is improved, especially from 48.78 to 73.07% with just 100 supplemental training data points. Thus, the supplemental dataset plays a role in vehicle detection model construction through transfer learning, which considerably improves vehicle detection accuracy by using only a small amount of supplemental data.

TABLE 2 | Vehicle detection accuracy of each model (mAP@0.5).

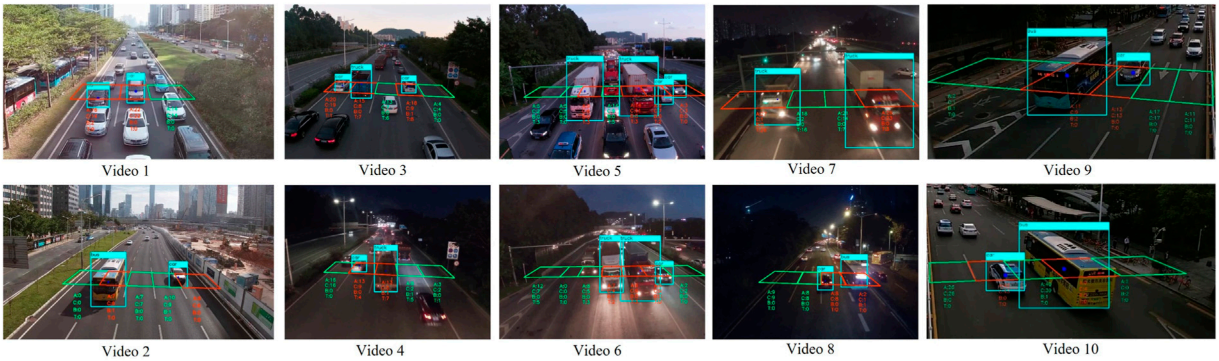
Model	Training data	All	Car	Bus	Truck
Model-coco	Coco	48.78	74.46	53.89	17.68
Model-100	coco +100	73.07	79.23	67.48	72.50
Model-250	coco +250	76.75	79.80	74.45	75.99
Model-500	coco +500	80.20	86.12	77.24	77.24
Model-1000	coco +1000	84.40	87.21	81.86	84.12

5.2 The Generalization of Vehicle Counting

Purpose: The purpose of the experiment is to evaluate the generalization of the vehicle counting model in videos with various light conditions, shooting directions, traffic conditions, and resolutions.

When employing Model-1000 constructed in Experiment 1 as the vehicle detection model, the vehicle numbers of the above 10 videos are counted, and the counting accuracy and computational efficiency are evaluated. **Figure 7** shows the screenshots of vehicle counting in traffic video. For direct-viewing expressions, only bounding boxes of vehicles passing through the detection area are displayed. First, in each video, virtual detection areas are set in each land. Then, vehicle counting is carried out in each detection area according to vehicle type, and the number of each type in each detection area is calculated as the counting results of the corresponding lane. Finally, the counting result of the road section is obtained by adding the counting results of all lanes. The counting results are shown in **Table 3**, where NR and ND refer to the real and detected number of vehicles, respectively, and CA (%) refers to the counting accuracy. The following conclusions can be drawn:

1) The model has strong performance in videos with various light conditions, shooting directions, traffic conditions, and resolutions, and the counting accuracy of the total number of vehicles and the number of three vehicle types all reach up to 99%. This shows that vehicle counting based on fusing virtual



Video 1 Video 3 Video 5 Video 7 Video 9

Video 2 Video 4 Video 6 Video 8 Video 10

FIGURE 7 | Screenshots of vehicle counting in traffic videos.

TABLE 3 | Results of vehicle counting.

Video	Total			Car			Bus			Truck			Efficiency (fps)
	NR	ND	CA (%)	NR	ND	CA (%)	NR	ND	CA (%)	NR	ND	CA (%)	
1	145	145	100.00	141	141	100.00	4	4	100.00	0	0	100.00	22.2
2	226	226	100.00	216	216	100.00	10	10	100.00	0	0	100.00	21.3
3	99	100	98.99	67	68	98.51	2	2	100.00	30	30	100.00	20.2
4	136	136	100.00	83	82	98.80	3	3	100.00	50	51	98.00	20.2
5	127	127	100.00	100	100	100.00	1	1	100.00	26	26	100.00	20.2
6	138	139	99.28	109	109	100.00	2	2	100.00	27	28	96.30	20.2
7	119	117	98.32	55	54	98.18	0	0	100.00	64	63	98.44	20.2
8	254	254	100.00	247	247	100.00	7	7	100.00	0	0	100.00	20.2
9	151	151	100.00	144	144	100.00	7	7	100.00	0	0	100.00	22.5
10	184	184	100.00	171	171	100.00	13	13	100.00	0	0	100.00	22.5
Average	—	—	99.66	—	—	99.55	—	—	100.00	—	—	99.27	21.0

detection area and vehicle tracking can avoid the errors caused by missing detection and false detection, which further improves the accuracy of vehicle counting, although the accuracy of vehicle detection is not very high.

2) The counting accuracy of video 9 and video 10 also reaches 99%, which shows that the model has good generalisability.
3) The computational efficiency of the model is faster than 20 fps, which meets the requirement of real-time vehicle counting.

5.3 The Effectiveness of Transfer Learning

Purpose: The purpose of the experiment is to evaluate the effectiveness of transfer learning in the construction of a deep learning vehicle counting model.

The following three vehicle detection models are constructed during transfer learning.

1) Model-1: This model is a source model trained by vehicle data in the MS COCO dataset.
2) Model-2: This model is a target model initialised by parameters of Model-1 and trained by the merged datasets of vehicle data in the MS COCO dataset and the supplemental dataset.
3) Model-3: This model is a target model fine-tuned from Model-2 through further training on the supplemental dataset.

To verify the effectiveness of instance-based and parameter-based transfer learning, another two vehicle detection models are constructed:

4) Model-4: This model is a target model initialised by parameters of Model-1 and trained by just the supplemental dataset to validate the effectiveness of instance-based transfer learning.
5) Model-5: This model is a target model trained by the merged datasets of vehicle data in the MS COCO dataset and supplemental dataset without initialisation by the parameters of Model-1. The model is then fine-tuned by the supplemental dataset to validate the effectiveness of parameter-based transfer learning.

The above five models are used to count the vehicle number of 10 videos, and the counting accuracies of each model are combined and compressed. Because the final task of the vehicle detection model is to count vehicles and the vehicle counting method proposed in this paper can further improve the accuracy of vehicle counting when the accuracy of vehicle detection is not high enough, the accuracy of vehicle counting is used as the evaluation index of the accuracy of the model. The average counting accuracy of each model in 10 videos is shown in **Table 4**. From the experimental results, the following conclusions can be drawn:

TABLE 4 | Vehicle counting accuracy of each model (%).

Model	Total	Car	Bus	Truck	Average
Model-1	95.88	89.32	41.53	52.25	69.75
Model-2	99.32	99.42	85.67	86.60	92.75
Model-3	99.66	99.55	100.00	99.27	99.62
Model-4	99.39	99.16	97.57	98.29	98.60
Model-5	99.55	98.94	87.57	97.63	95.92

TABLE 5 | Vehicle detection accuracy of Model-1000-Tiny (mAP@0.5).

Model	Training data	All	Car	Bus	Truck
Model-1000-Tiny	coco +1000	71.21	72.70	75.20	65.73

1) Compared with other models, Model-3 has the best detection accuracy, and the counting accuracy of the total number of vehicles, as well as the number of three vehicle types all reach 99%, which shows that the vehicle counting model based on transfer learning performs well.
2) Model-1, Model-2, and Model-3 improve the counting accuracy step-by-step, which shows that transfer learning plays a role. Annotated data of the target task can be obtained from the source model, and further processing into refined annotated data improves the performance of the target model through transfer learning.
3) Compared with Model-4 and Model-5, Model-3 has better detection accuracy. It shows that instance-based and parameter-based transfer learning are both working. A more reliable deep learning model can be constructed without annotated data through transfer learning.

5.4 The Robustness of Vehicle Counting

Purpose: The purpose of the experiment is to evaluate the robustness of vehicle counting under the condition of a vehicle detection model with higher efficiency but lower accuracy.

In the proposed framework, the vehicle detection process takes most of the computing cost and depends on the efficiency of the basic model used in the vehicle detection model construction stage. The vehicle counting process avoids the errors caused by missing detection and false detection and further improves the accuracy of vehicle number calculation. Thus, the framework is flexible, and the trade-off between accuracy and efficiency can be made according to the requirement by choosing a deep learning object detection model with suitable performance in accuracy and efficiency as the basic model.

Tiny-YOLO is a simplified version of YOLO that has higher efficiency but lower accuracy. Tiny-YOLO is used as the basic model, and a vehicle detection model (Model-1000-Tiny) is constructed in the same way as Model-1000. Then, it is used to perform vehicle counting in 10 videos and evaluate the counting accuracy and computational efficiency. The vehicle detection accuracy of Model-1000-Tiny is shown in **Table 5**, which is more than 10% lower than Model-1000. The vehicle counting result is shown in **Table 6**; it can see that the counting accuracy of the total number of vehicles and the number of three vehicle types are relatively lower than that of Model-1000, but the gap is not as large as vehicle detection accuracy. It is higher than 98% in the total vehicle number and car number, 90.37% in bus number and 97.61% in truck number. Although the average counting accuracy of the bus number is not very high, it performs well in eight of the 10 videos. It only makes large mistakes in video 6 and video 8, while two buses are counted into one bus in video 6, and 7 buses are counted into five buses in video 8. However, this is likely caused by the small real number of buses; once there is an error, the accuracy is seriously reduced, and thus, overall, the model has good performance. However, in computational efficiency, using Tiny-YOLO as the basic model is more efficient than using YOLO, which can reach 53.4 fps, more than twice that of Model-1000. Thus, the proposed framework can maintain high accuracy of vehicle counting, although the accuracy of the vehicle detection model is not very high, and the trade-off between accuracy and efficiency can be made according to the requirements.

6 CONCLUSION AND FUTURE WORK

In this paper, a deep learning framework for video-based vehicle counting is proposed. The framework has two main

TABLE 6 | Results of vehicle counting (Tiny-YOLO).

Video	Total			Car			Bus			Truck			Efficiency (fps)
	NR	ND	CA (%)	NR	ND	CA (%)	NR	ND	CA (%)	NR	ND	CA (%)	
1	145	143	98.62	141	139	98.58	4	4	100.00	0	0	100.00	86.5
2	226	227	99.56	216	216	100.00	10	11	90.00	0	0	100.00	42.5
3	99	101	97.98	67	68	98.51	2	2	100.00	30	31	96.67	32.4
4	136	141	96.32	83	83	100.00	3	3	100.00	50	55	90.00	32.4
5	127	129	98.43	100	102	98.00	1	1	100.00	26	26	100.00	32.4
6	138	140	98.55	109	110	99.08	2	1	50.00	27	29	92.59	32.4
7	119	116	97.48	55	54	98.18	0	0	100.00	64	62	96.88	32.4
8	254	255	99.61	247	250	98.79	7	5	71.43	0	0	100.00	32.4
9	151	152	99.34	144	145	99.31	7	7	100.00	0	0	100.00	105.3
10	184	188	97.83	171	174	98.25	13	14	92.31	0	0	100.00	105.3
Average	—	—	98.37	—	—	98.87	—	—	90.37	—	—	97.61	53.4

tasks: deep learning vehicle detection model construction and vehicle counting. In deep learning vehicle detection model construction, to solve the problem of lacking annotated data, based on an open dataset, instance-based transfer learning and parameter-based transfer learning are adopted to construct a vehicle detection model with good performance. In vehicle counting, for the possible situation of vehicle missing detection and false detection, vehicle counting based on fusing virtual detection area and vehicle tracking is proposed. Missing alarm suppression module based on vehicle tracking and false alarm suppression module based on bounding box size statistics are designed to avoid vehicle counting errors caused by missing detection or false detection, which further improves the accuracy of vehicle counting. In this framework, the trade-off between accuracy and efficiency can be made according to the requirement by choosing a deep learning object detection model with a suitable performance in accuracy and efficiency as the basic model. Moreover, the proposed framework can improve the accuracy of vehicle counting although the accuracy of vehicle detection is not very high.

All the traffic videos used in this study are shot on straight roads. However, there are other scenarios in traffic surveillance, such as intersections and T-junctions. Although the model in this study has strong performance in straight road scenarios, making the model work well in different scenarios is an important problem to solve. Future work will consider scene adaptation to build a vehicle counting framework for different scenarios.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

HL, ZY, BH, XK and XL: conceived the study, designed the vehicle detection and counting algorithms, conducted experiments, interpreted results, wrote and revised the article; RG: interpreted results, and revised the article.

FUNDING

We acknowledge the financial support by the National Natural Science Foundation of China (Project Nos. 71901147, 41901325, 41901329, 41971354, and 41971341), the Guangdong Science and Technology Strategic Innovation Fund (the Guangdong–Hong Kong-Macau Joint Laboratory Program, Project No. 2020B1212030009), the Research Program of Shenzhen S&T Innovation Committee (Project Nos. JCYJ20210324093600002 and JCYJ20210324093012033), the Science Foundation of Guangdong Province (Project Nos. 2121A1515012574 and 2019A1515010748), the Open Fund of Key Laboratory of Urban Land Resources Monitoring and Simulation, MNR, China (No. KF-2019-04-014), and Jilin Science and technology development plan project (No.20190303124SF).

REFERENCES

1. Zheng X, Cai Z. Privacy-preserved Data Sharing towards Multiple Parties in Industrial IoTs. *IEEE J Select Areas Commun* (2020) 38:968–79. doi:10.1109/jsac.2020.2980802
2. Cai Z, Zheng XS. Engineering, A Private and Efficient Mechanism for Data Uploading in Smart. *cyber-physical Syst* (2018) 7:766–75. doi:10.1109/TNSE.2018.2830307
3. Gao C, Fan Y, Jiang S, Deng Y, Liu J, Li X. Dynamic Robustness Analysis of a Two-Layer Rail Transit Network Model. *IEEE Transactions on Intelligent Transportation Systems* (2021). doi:10.1109/TITS.2021.3058185
4. Huang Q, Yang Y, Xu Y, Yang F, Yuan Z, Sun Y. Citywide Road-Network Traffic Monitoring Using Large-Scale mobile Signaling Data. *Neurocomputing* (2021) 444:136–46. doi:10.1016/j.neucom.2020.07.150
5. Huang Q, Yang Y, Xu Y, Wang E, Zhu KJWC, Computing M. Human Origin-Destination Flow Prediction Based on Large Scale Mobile Signal Data. *Wireless Communications and Mobile Computing* (2021). p. 2021. doi:10.1155/2021/1604268
6. Li Y, Li B, Tian B, Yao Q. Vehicle Detection Based on the and- or Graph for Congested Traffic Conditions. *IEEE Trans Intell Transport Syst* (2013) 14: 984–93. doi:10.1109/tits.2013.2250501
7. Barcellos P, Bouvié C, Escouto FL, Scharcanski J. A Novel Video Based System for Detecting and Counting Vehicles at User-Defined Virtual Loops. *Expert Syst Appl* (2015) 42:1845–56. doi:10.1016/j.eswa.2014.09.045
8. Kamkar S, Safabakhsh R. Vehicle Detection, Counting and Classification in Various Conditions. *IET Intell Transport Syst* (2016) 10:406–13. doi:10.1049/iet-its.2015.0157
9. Mandellos NA, Keramitsoglou I, Kiranoudis CT. A Background Subtraction Algorithm for Detecting and Tracking Vehicles. *Expert Syst Appl* (2011) 38: 1619–31. doi:10.1016/j.eswa.2010.07.083
10. Hofmann M, Tiefenbacher P, Rigoll G. *Background Segmentation with Feedback: The Pixel-Based Adaptive Segmenter*. Providence, RI: 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (2012). p. 38–43. doi:10.1109/CVPRW.2012.6238925
11. Rosin PL, Ellis TJ. *Image Difference Threshold Strategies and Shadow Detection*. Citeseer: BMVC (1995). p. 347–56.
12. Kasturi R, Goldgof D, Soundararajan P, Manohar V, Garofolo J, Framework for Performance Evaluation of Face, Text, and Vehicle Detection and Tracking in Video: Data, Metrics, and Protocol. *IEEE Trans Pattern Anal Mach Intell* (2009) 31:319–36. doi:10.1109/tpami.2008.57
13. Horn BK, Schunck BG. Determining Optical Flow. *Artif intelligence* (1981) 17: 185–203. doi:10.1016/0004-3702(81)90024-2
14. Chen Z, Cao J, Tang Y, Tang L. Tracking of Moving Object Based on Optical Flow Detection. In: Proceedings of 2011 International Conference on Computer Science and Network Technology, Harbin, China, December 24–26, 2011. IEEE (2011). p. 1096–9. doi:10.1109/iccsnt.2011.6182151
15. Lange S, Ulbrich F, Goehring D. Online Vehicle Detection Using Deep Neural Networks and Lidar Based Preselected Image Patches. In: 2016 IEEE Intelligent Vehicles Symposium (IV), Gothenburg, Sweden, June 19–22, 2016. IEEE (2016). p. 954–9. doi:10.1109/ivs.2016.7535503
16. Mundhenk TN, Konjevod G, Sakla WA, Boakye K. A Large Contextual Dataset for Classification, Detection and Counting of Cars with Deep Learning. In: European Conference on Computer Vision, Amsterdam, Netherlands, October 11–14, 2016. Springer (2016). p. 785–800. doi:10.1007/978-3-319-46487-9_48
17. Wang J, Zheng H, Huang Y, Ding X. Vehicle Type Recognition in Surveillance Images from Labeled Web-Nature Data Using Deep Transfer Learning. *IEEE Trans Intell Transportation Syst* (2017) 19:2913–22. doi:10.1109/TITS.2017.2765676

18. Suhao L, Jinzhao L, Guoquan L, Tong B, Huiqian W, Yu P. Vehicle Type Detection Based on Deep Learning in Traffic Scene. *Proced Comput Sci* (2018) 131:564–72. doi:10.1016/j.procs.2018.04.281
19. Krizhevsky A, Sutskever I, Hinton GE. Imagenet Classification with Deep Convolutional Neural Networks. *Adv Neural Inf Process Syst* (2012) 25–105. Available at https://proceedings.neurips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html.
20. Cao L, Wang C, Li J. Vehicle Detection from Highway Satellite Images via Transfer Learning. *Inf Sci* (2016) 366:177–87. doi:10.1016/j.ins.2016.01.004
21. Li X, Ye M, Liu Y, Zhu C. Adaptive Deep Convolutional Neural Networks for Scene-specific Object Detection. *IEEE Trans Circuits Syst Video Technol* (2017) 29 (9): 2538–2551. doi:10.1109/TCSVT.2017.2749620
22. Wang Y, Deng W, Liu Z, Wang J. Deep Learning-Based Vehicle Detection with Synthetic Image Data. *IET Intell Transport Syst* (2019) 13:1097–105. doi:10.1049/iet-its.2018.5365
23. Michalopoulos PG. Vehicle Detection Video through Image Processing: the Autoscope System. *IEEE Trans.veh.technol* (1991) 40:21–9. doi:10.1109/25.69968
24. Engel JI, Martin J, Barco R. A Low-Complexity Vision-Based System for Real-Time Traffic Monitoring. *IEEE Trans Intell Transportation Syst* (2017) 18: 1279–88. doi:10.1109/tits.2016.2603069
25. Rosas-Arias L, Portillo-Portillo J, Hernandez-Suarez A, Olivares-Mercado J, Sanchez-Perez G, Toscano-Medina K, et al. Vehicle Counting in Video Sequences: An Incremental Subspace Learning Approach. *Sensors* (2019) 19:2848. doi:10.3390/s19132848
26. Badenas Carpio J, Sanchiz Martí JM, Pla F. Motion-based Segmentation and Region Tracking in Image Sequences. *Pattern Recognition* (2001) 34:661–70. doi:10.1016/s0031-3203(00)00014-5
27. Unzueta L, Nieto M, Cortes A, Barandiaran J, Sanchez P. Adaptive Multi-Cue Background Subtraction for Robust Vehicle Counting and Classification. *IEEE Trans Intell Transportation Syst* (2012) 13:527–40. doi:10.1109/tits.2011.2174358
28. Dai Z, Song H, Wang X, Fang Y, Yun X, Zhang Z, et al. Video-based Vehicle Counting Framework. *IEEE Access* (2019) 7:64460–70. doi:10.1109/access.2019.2914254
29. Girshick R, Donahue J, Darrell T, Malik J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In: IEEE Conference on Computer Vision & Pattern Recognition, Columbus, OH, June 23–28, 2014 (2014). p. 580–7.
30. Girshick R. Fast R-CNN. In: IEEE International Conference on Computer Vision, Santiago, Chile, December 13–16, 2015 (2015). p. 1440–8.
31. Ren S, He K, Girshick R, Sun J. Faster R-Cnn: Towards Real-Time Object Detection with Region Proposal Networks. *Adv Neural Inf Process Syst* (2015) 28. Available at https://proceedings.neurips.cc/paper/2015/hash/14bfa6bb14875e45bba028a21ed38046-Abstract.html.
32. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C-Y, et al. SSD: Single Shot Multibox Detector. In: European conference on computer vision, Amsterdam, Netherlands, October 11–14, 2016. Springer (2016). p. 21–37. doi:10.1007/978-3-319-46448-0_2
33. Redmon J, Divvala S, Girshick R, Farhadi A. You Only Look once: Unified, Real-Time Object Detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, June 27–30, 2016 (2016). p. 779–88. doi:10.1109/cvpr.2016.91
34. Redmon J, Farhadi A. YOLO9000: Better, Faster, Stronger. In: Proceedings of the IEEE conference on computer vision and pattern recognition, Honolulu, HI, July 22–25, 2017 (2017). p. 7263–71. doi:10.1109/cvpr.2017.690
35. Redmon J, Farhadi A. *Yolov3: An Incremental Improvement*. arXiv preprint arXiv:1804.02767 (2018).
36. Pan SJ, Yang Q. A Survey on Transfer Learning. *IEEE Trans knowledge Data Eng* (2009) 22:1345–59. doi:10.1109/TKDE.2009.191
37. Lin T-Y, Maire M, Belongie S, Hays J, Perona P, Ramanan D, et al. Microsoft Coco: Common Objects in Context. In: European conference on computer vision, Zurich, Switzerland, September 6–12, 2014. Springer (2014). p. 740–55. doi:10.1007/978-3-319-10602-1_48
38. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, et al. ImageNet Large Scale Visual Recognition Challenge. *Int J Comp Vis* (2015) 115:211–52. doi:10.1007/s11263-015-0816-y
39. Everingham M, Eslami SMA, Gool LV, Williams CKI, Winn J, Zisserman A. The Pascal Visual Object Classes Challenge: A Retrospective. *Int J Comp Vis* (2015) 111:98–136. doi:10.1007/s11263-014-0733-5

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers in Physics

ORIGINAL RESEARCH
published: 22 February 2022
doi: 10.3389/fphy.2022.835412

A New Cooperative Recourse Strategy for Emergency Material Allocation in Uncertain Environments

Yuxin Liu[1], *Songxin Wang*[2] *and Xianghua Li*[3]*

[1]College of Information Engineering, Shanghai Maritime University, Shanghai, China, [2]School of Information Management and Engineering, Shanghai University of Finance and Economics, Shanghai, China, [3]School of Artificial Intelligence, Optics and Electronics (iOPEN), Northwestern Polytechnical University, Xi'an, China

Emergency material allocation is an important issue in the urgent handling of public health emergencies. This article models the relief allocation and transportation route planning as an uncertain capacitated arc routing problem, which is a classic combinatorial optimization problem that considers stochastic factors such as uncertain demand and travel time in the service. The stochastic of demand leads to the *route failure* that the vehicle cannot serve the tasks successfully unexpected. Most existing research uses the independent recourse strategy. That is, each vehicle takes a back-and-forth trip separately when its remaining capacity cannot meet the actual demand of the task. This leads to a considerable recourse cost. However, a few studies have considered vehicular cooperation to deal with route failure, which is beneficial for pooling the capacity of multiple vehicles. In this paper, we propose a new recourse strategy called OneFAll that lets one vehicle take charge of all the failed tasks. In this case, other vehicles can finish the service once they are full. We develop the genetic programming hyper-heuristic with the OneFAll recourse strategy for solving the uncertain capacitated arc routing problem. The experimental studies show that our proposed method outperforms the existing genetic programming hyper-heuristic with the independent recourse strategy to the uncertain capacitated arc routing problem for the *ugdb* and *uval* benchmark instances. Moreover, our strategy outperforms the recourse strategy that failed tasks are returned to the unassigned task set for any vehicle to complete. This reflects that there exists resource waste if all vehicles are involved to repair the failed routes.

Keywords: recourse strategy, uncertain capacitated arc routing problem, genetic programming hyper-heuristic, intelligent transportation, emergency material allocation

Edited by:
Zhanwei Du,
University of Texas at Austin,
United States

Reviewed by:
Wenbin Pei,
Victoria University of Wellington,
New Zealand
Yu-Jie Xiong,
Shanghai University of Engineering
Sciences, China

Correspondence:
Xianghua Li
li_xianghua@nwpu.edu.cn

Specialty section:
This article was submitted to
Social Physics,
a section of the journal
Frontiers in Physics

Received: *14 December 2021*
Accepted: *10 January 2022*
Published: *22 February 2022*

Citation:
Liu Y, Wang S and Li X (2022) A New Cooperative Recourse Strategy for Emergency Material Allocation in Uncertain Environments.
Front. Phys. 10:835412.
doi: 10.3389/fphy.2022.835412

1 INTRODUCTION

In the urgent handling of public health emergencies, the medical resources, including protective equipments, disinfection materials, drugs, and medical supplies, are the material basis [1]. The allocation efficiency has a direct impact on the timely control and elimination of public health emergencies, and safeguarding the physical health and life security of the general public. However, if the allocation is not improper, the emergencies cannot be contained timely, which has a great effect on the recovery of social functions [2]. It is bound to cause enormous losses for our society.

From the perspective of management decision, emergency material allocation is a dynamic decision-making problem in the complex road networks that allocate relief materials from suppliers to disaster areas as soon as possible. The relief allocation and transportation route planning can be categorized under the domain of capacitated arc routing problem (CARP) [3], which is a classic optimization problem that has been thoroughly studied in the operations research and has a wide range of applications for many real-world situations [4, 5]. The pharmacies, quarantine offices, and community offices can be seen as demand points along the streets in the road network, which correspond to *tasks* in CARP. A fleet of equipped vehicles is appointed to meet the demands of these points, and both the vehicles that can be dispatched and their capacities are limited, which can be modeled as constraints in CARP. The goal is to design the most economical routes, which corresponds to optimizing the objective functions (e.g., minimizing costs) in CARP.

The emergency material allocation in the real world is much complex than the traditional CARP, which assumes that all the information in the environment such as the demands and traversal costs is static and can be exactly known in advance. However, this assumption does not always hold in the real world, especially in the environment of emergent disaster. In fact, the demands of tasks are uncertain, which are affected by many factors, such as the number of residents along the streets and the severity of the disaster. Hence, the exact value of demands cannot be exactly known beforehand. This may lead to that a vehicle reaches a task without enough capacity to meet the demand. Moreover, the roads may be interrupted or blocked, which leads to that the preplanned routes cannot be traversed. Hence, uncertain CARP (UCARP) has been a hot and active research topic in recent years [6]. Two of the above uncertainties are considered in UCARP, and they lead to two uncontrollable failures, i.e., the *route failure* and the *edge failure*, respectively.

For solving the UCARP, the existing approaches can be divided into three main categories [7, 8]: robust pro-active, completely reactive, and predictive–reactive. Among them, the completely reactive approaches aim to evolve policies, which can generate routes based on practical situations in real time. They have the advantages of flexibility and are very efficient in handling dynamic environments [9]. Among the completely reactive approaches, genetic programming (GP) has been proven to be an effective hyper-heuristic method (shorted as GPHH), which can automatically evolve routing policies for UCARP that are much better than the manually designed ones [10, 11]. For using GPHH to solve UCARP, an important issue is how to deal with failures, which could influence the efficiency of routing policies evolved by GPHH. For the edge failure, the most commonly used strategy is to find a detour to the destination. For the route failure, the situation is much more complex and has attracted more attention [12]. One of the naive recourse strategies to deal with route failure is that as soon as the capacity of the vehicle expires, the vehicle goes back to the depot to refill and then comes back to the interrupted place to continue the service [13]. This can be seen as an independent recourse strategy, and there is no collaboration between vehicles, which may lead to a considerable recourse cost. In recent years, collaborative transportation has been an emerging new mode, as it can bring together all the vehicles to improve the overall performance. Especially in the urgent disaster environment, the relief materials are very scarce and precious. There are not enough protective suits for workers or volunteers to participate in the rescue job. Hence, it is highly necessary to make full use of the cooperative abilities of vehicles in order to serve more tasks in a shorter time. That is to say, the independent recourse strategy is not suitable for urgent-disaster environments. We have to design more reasonable cooperative recourse strategies to deal with route failures, which have a great impact on the improvement of the whole efficiency.

Hence, we aim to propose a new recourse strategy for solving UCARP under the application of emergency material allocation in this paper. We develop a GPHH with the new recourse strategy to design routing policies for multi-vehicle UCARP. To be more specific, we have the following research objectives:

- To develop a new recourse strategy, named OneFAll, which considers cooperation between multiple vehicles.
- To develop a GPHH with the OneFAll recourse strategy to evolve routing policies for solving UCARP.
- To investigate the effectiveness of the OneFAll recourse strategy by comparing with existing state-of-the-art recourse strategies on benchmark UCARP datasets.
- To analyze the structure of the solutions obtained by different recourse strategies.

The rest of this paper is structured as follows. **Section 2** presents the background including UCARP definition and related work. **Section 3** describes the proposed OneFAll recourse strategy and the new GPHH algorithm for solving UCARP. **Section 4** shows the experimental studies and analysis. **Section 5** gives the conclusion and future work.

2 BACKGROUND

2.1 Problem definition

A UCARP instance [13, 14] can be represented by a connected graph $G = (V, E)$, where V is the set of vertices and E is the set of edges. Each edge $e \in E$ is associated with three features: a demand $d(e) > =0$, a serving cost $sc(e) > =0$, and a traversal cost (time to travel along the edge without serving it) $dc(e) > =0$. Edges with positive demands are called *tasks*. The set of all tasks is denoted as $T \in E$. A fleet of vehicles with a limited capacity Q is located at a special vertex called depot $v_0 \in V$ at the beginning. In the real scenario, it is assumed that the number of vehicles is restricted. The goal of the problem is to find out a least-cost routing plan for the vehicles to serve all the tasks subject to the following constraints:

1. Each vehicle starts from the depot and comes back to the depot after serving all the tasks allocated to it. Vehicles can replenish its capacity each time when they pass by the depot.
2. Each task is served exactly once in either direction.
3. The total demand served by each vehicle in a single trip cannot exceed its capacity.

A *sample* of a UCARP instance is obtained by sampling a value for each random variable of the corresponding UCARP instance. For example, a sample I_ξ of the UCARP instance I is obtained by sampling each random demand $d_\xi(e)$ and each random traversal cost $dc_\xi(e)$ under the environment (e.g., random seed) ξ. For solving a UCARP instance sample, both the task demand and edge traversal cost are unknown until the task is served or the edge is traversed. These lead to two unavoidable failures:

- Edge failure: the edge ahead of the route is inaccessible.
- Route failure: the actual demand of the task to be served exceeds the remaining capacity of the vehicle.

In the case of edge failure, one can find a detour to the destination. If the edge is a task assigned to the vehicle, the vehicle will abandon this task and go to serve the next task according to the routing plan. The route failure is not such kind of easy to cope with. A typical recourse operator uses an independent recourse strategy [11, 13]. When a route failure occurs, the vehicle returns to the depot to refill its capacity and then comes back to finish the remaining service of the failed task. However, this may introduce a large amount of extra cost. We will give a detailed review about the current recourse strategies for route failure in **Section 2.3**. To summarize, avoiding a route failure is more challenging and has a greater impact on solution quality [12]. A good recourse strategy is expected to minimize the extra refill cost. Therefore, in this paper, we aim to propose an efficient recourse strategy to tackle the uncertain demands.

A solution to a UCARP instance sample can be represented as $S = (X, Y)$. $X = \{X^{(1)}, X^{(2)}, \ldots, X^{(m)}\}$ is a set of routes, where each route $X^{(k)} = (x_1^{(k)}, \ldots, x_{L_k}^{(k)})$ is a sequence of vertices starting and ending at the depot vertex (i.e., $x_1^{(k)} = x_{L_k}^{(k)} = v_0$) and L_k is the number of vertices in the *kth* route. $Y = \{Y^{(1)}, Y^{(2)}, \ldots, Y^{(m)}\}$ is a set of real-valued vectors indicating the fraction of service of each edge along the routes. Specifically, $Y^{(k)} = (y_1^{(k)}, \ldots, y_{L_k-1}^{(k)})$ corresponds to $X^{(k)}$, where $0 \leq y_i^{(k)} \leq 1$. $y_i^{(k)} = 1$ means that the edge $(x_i^{(k)}, x_{i+1}^{(k)})$ is a task and is fully served, and $y_i^{(k)} = 0$ means that the edge $(x_i^{(k)}, x_{i+1}^{(k)})$ is traveled through by the vehicle without being served. For other values of $y_i^{(k)}$, it means that the edge $(x_i^{(k)}, x_{i+1}^{(k)})$ is partially served at the current route.

The total cost of a solution (X, Y) is calculated as **Eq. 1**,

$$C(S_\xi) = \sum_{k=1}^{m} \sum_{i=1}^{L_k-1} \Big(sc\big(S_\xi[x_i^{(k)}], S_\xi[x_{i+1}^{(k)}]\big) \times S_\xi[y_i^{(k)}] + dc_\xi\big(S_\xi[x_i^{(k)}], S_\xi[x_{i+1}^{(k)}]\big) \times \big(1 - S_\xi[y_i^{(k)}]\big) \Big) \tag{1}$$

where $S_\xi[x_i^{(k)}]$ and $S_\xi[y_i^{(k)}]$ stand for the $x_i^{(k)}$ and $y_i^{(k)}$ elements in the solution S_ξ on the environment ξ.

Note that S_ξ varies from one sample to another. For any sample ξ, a feasible solution S_ξ can be generated by a pre-optimized (robust) solution or a routing policy that generates the solution in an online fashion. In this paper, we focus on the latter.

2.2 Approaches to uncertain CARP

Based on when the decisions are made, the approaches to solve uncertain routing problems are categorized into three categories [7, 8]: robust pro-active, completely reactive, and predictive–reactive.

The robust proactive typically can be divided into two stages. It first constructs predictive solutions to satisfy performance requirements based on the prediction of the environment. Then, the solutions are executed, and the recourse strategies are taken to deal with failures, as in two-stage stochastic programming with recourse. The optimization algorithms used in the first stage in existing studies are the branch-and-price algorithm [15], the memetic algorithm [16, 17] and the estimation of the distribution algorithm [18]. The recourse actions in the second stage are summarized in the following subsection.

The advantage of the proactive approaches is that they can provide a robust and predictable solution when applied to new environments. However, they are non-flexible and cannot cope with real-time adjustment.

The completely reactive approaches treat the problem as an online decision-making process and construct the final solution step by step using the decision-making rule (called routing policy in UCARP) [8]. Some common heuristics such as path scanning [19] can be seen as a completely reactive approach. The keys to the success of these approaches are to obtain a good decision-making policy and the decision-making process of the policy on instances (i.e., meta-algorithm). The two main approaches based on the completely reactive approach are the GP algorithm and the rollout algorithm.

GP [20] belongs to the evolutionary computation field, which aims to evolve computer programs. In GP, populations of computer programs are genetically bred using the Darwinian principle of survival of the fittest and using a genetic crossover operator appropriate for genetically mating computer programs [21]. As a hyper-heuristic method, GPHH has been applied to scheduling tasks. A GPHH program can be seen as a routing policy for routing problems [22], or a dispatching rule for job shop scheduling problems [23] for different decision environments. Weise et al. [22] first proposed to apply GPHH for the automated design of routing policy for solving static CARP and tested the performance of the evolved rules for dealing with random disappearance of tasks. Liu et al. [11] extended the GPHH for solving UCARP with a new meta-algorithm. Later on, many researchers have proposed the improved GPHH for solving UCARP from the aspect of developing more effective meta-algorithms [24, 25], evolving more interpretable routing policies [26, 27], and discovering the reusability of routing policies [28, 29].

The design of the rollout algorithm is motivated by the idea of policy iteration in dynamic programming. It is a decision-making process algorithm based on Monte Carlo simulation. Dror et al. [30] first modeled the vehicle routing problem with stochastic demands as a Markov decision process in theory, but they did not provide any computational results. Secomandi [31] first proposed the rollout algorithm to solve the vehicle routing problem defined in [30]. Later on, the rollout algorithm was improved by many researchers for solving uncertain vehicle routing problems [32–34].

The advantage of completely reactive approaches is that the solutions are generated online, so the solutions are flexible, which is especially effective in uncertain environments [35]. However, their disadvantage is that no baseline solution (i.e., a set of routes) is generated, which reduces the stability of routes and causes difficulty for planning and measuring in advance [7].

The predictive–reactive can be seen as a hybridization of the pro-active and the completely reactive. They include a baseline solution obtained by the pro-active part, and a reoptimization strategy in charge of real-time reaction. Liu et al. [14] designed the first predictive–reactive approach for solving UCARP. They proposed a new solution representation, which is composed of two components: a baseline task sequence and a recourse policy. Meanwhile, a cooperative coevolution framework is designed to optimize these two components simultaneously.

The advantage of predictive–reactive approaches is that they generally consider both the quality of the predictive baseline solution (*efficiency*) and the degree of change to be made on the baseline solution to adapt to the new environment (*stability*) [12, 14].

2.3 Recourse strategies for route failure

In [36], Gendreau et al. pointed out that the development of new recourse strategies is one of the most critical issues and challenges that need to be addressed to advance research in this area. By now, there are three main kinds of recourse strategies: the independent, the pairing, and the global.

In the independent recourse strategy, upon failure, the vehicle returns to the depot, replenishes its capacity, and resumes its planned route at the point of failure [11, 13]. While somewhat simplistic, this recourse strategy has some advantages. First, it yields relatively tractable models that enable the development of exact algorithms. Second, from the practical view, the independent recourse strategies warrant stable tactical routes, which are operationally desirable, as they require little deviations of drivers' familiar driving environment and ensure that customers are consistently visited by the same drivers. Hence, it is the most widely used one for the routing problem with stochastic demands. However, the recourse actions are based on the realized demand of a route, independent of the demand realizations of the other routes. No cooperation between vehicles may cause a great degree of resource waste.

The pairing recourse strategy [37] considers a certain degree of collaboration between two vehicles. According to this strategy, the vehicles are paired to work, where one is identified as Type I and the other is Type II. When the Type I vehicle shows a failure, it returns to the depot. Moreover, the unserved tasks are appended to the end of the route of the Type II vehicle. The classical recourse policy is used to handle failure in the Type II route. Lei et al. [38] proposed another form of the paired strategy, where they allowed demands to be split between the paired routes while applying the classical recourse strategy upon failure. Erera et al. [39] proposed that customers were assigned to two planned routes, a primary and a backup. The recourse decisions allow reallocating customers to backup routes in the implementing of the planned routes. Their experiments verified that the paired recourse strategy can save the expected travel cost in a large degree.

The global recourse strategy aims to construct more collaborative forms of recourse involving multiple vehicles, which is likely to reduce the expected costs substantially. Unfortunately, only a few studies have considered the global recourse strategy to data. MacLachlan et al. [12] proposed that the failed tasks were returned back to the candidate task set. Any vehicle can potentially complete the remaining service at any time based on the routing policy. This can be seen as a kind of global recourse strategy. However, it may have the drawback that all vehicles could not cease their services until all the tasks are fulfilled. It may appear that many vehicles had to begin a second tour and only serve a few tasks (e.g., one or two), which may increase the total cost in a large degree. We compare our proposed strategy with this one in experiments and use a case study to show their differences.

Besides the above three categories, the preventive restocking strategy is also considered in many existing studies [40]. It assumes that whenever the residual capacity of a vehicle becomes low, the vehicle may execute a restocking trip to the depot actively. The preventive restocking can reduce the probability of route failures [36]. In our work, such strategy is used in the meta-algorithm to filter candidate tasks for the selection of vehicles [11]. Details are shown in **Section 3.1**.

As the information and communications technologies enable communications between vehicles [41], it is valuable to develop recourse strategies based on a high degree of vehicle collaboration. Moreover, taking the urgent disaster into consideration, the resources that can be used are limited and the basic requirement is to serve the area as larger as possible in

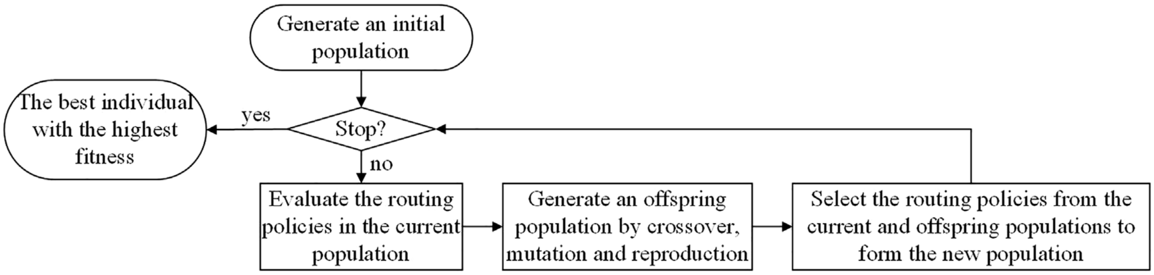

Generate an initial population
Stop?
yes
no
The best individual with the highest fitness
Evaluate the routing policies in the current population
Generate an offspring population by crossover, mutation and reproduction
Select the routing policies from the current and offspring populations to form the new population

FIGURE 1 | The flowchart of GPHH for UCARP.

the first time. Hence, we move a step forward to propose a new recourse strategy for route failure in the next section.

3 THE PROPOSED APPROACH

First, the general framework of GPHH for evolving routing policies to solve UCARP is described in **Figure 1**.

In the GPHH, a routing policy is represented as a Lisp tree, which is used as a priority function to select the task from the candidate task sets for a vehicle to serve next. To evaluate the fitness of each routing policy, a *meta-algorithm* is designed to generate a feasible solution given a sampled UCARP instance I_ξ and a routing policy $h(\cdot)$. At the beginning of the research, the meta-algorithm is designed as processes of building the routes one by one, which simulates a single-vehicle situation, where all the routes are executed by a single vehicle sequentially [11]. It cannot handle with the multi-vehicle UCARP, in which there are multiple vehicles on the road simultaneously. To fill this gap, Mei et al. [25] proposed the meta-algorithms that model a multi-vehicle decision-making process, where there can be any number of vehicles in service simultaneously. However, the strategies to deal with the route failure in multi-vehicle cases have been overlooked so far, which would influence the efficiency of the meta-algorithms. Considering the application of emergency material allocation, we proposed a new meta-algorithm with a new recourse strategy in the following.

3.1 The meta-algorithm with the new recourse strategy

For the emergency material allocation, a fleet of vehicles is appointed to allocate relief materials to communities distributed along streets in a certain area. These vehicles should complete the current tasks as soon as possible in order to traverse to the next area. Not all vehicles are needed to stay at the current area until all tasks are finished, as we want to help residents in wider areas. That is to say, when route failure occurs, only a few vehicles (e.g., one or two vehicles) need to stay at the current area to finish the remaining task; other vehicles can go to a new area straightforwardly.

Above all, an efficient cooperative recourse strategy in the emergency material allocation can be that vehicles are divided into two parts: flowing vehicles and stationary vehicles. Moreover, when route failure occurs, the failed tasks are returned back to the unassigned task set for any vehicles in the stationary category to potentially complete. For the vehicles in the flowing category, they just do one route, i.e., either when there is no candidate task for them to serve or when they encounter route failure, and they will finish the service. For the vehicles in the stationary category, they do not stop the service until all tasks are finished in the service. This proposal can also meet some other scenario in the real world. For example, some drivers want to do more job to earn more money.

Algorithm 1 describes the proposed meta-algorithm with the cooperative recourse strategy that executes a routing policy $h(\cdot)$ on a sampled UCARP instance I_ξ to construct a feasible solution.

TABLE 1 | The parameter settings.

Parameter	Value
Population size	1,024
Generations	51
Tournament section size	7
Crossover rate	0.8
Mutation rate	0.15
Reproduction rate	0.05
Maximal tree depth	8

Algorithm 1. The proposed meta-algorithm of the GPHH for UCARP

```
Input: A UCARP instance $I_\xi$, the number of vehicles $m$, a routing policy $h(\cdot)$
Output: A solution $S_\xi = (X_\xi, Y_\xi)$
 1: for $k = 1 \rightarrow m$ do
 2:     $X_\xi^{(k)} = (v_0), Y_\xi^{(k)} = (), q^{(k)}(\cdot) = Q$;
 3: end for
 4: $U \leftarrow T$                                        ▷ $U$ is the pool of unserved tasks
 5: Initiliase an empty time list $\Gamma$;
 6: for $k = 1 \rightarrow m$ do
 7:     add into $\Gamma$ an idle status for vehicle $k$;
 8: end for
 9: while not all tasks served do
10:     Select the earliest idle vehicle $\hat{k}$ from $\Gamma$, and remove it from $\Gamma$;
11:     $U' \leftarrow$ Filter(U)
12:     if $U' = \emptyset$ then
13:         GoTo($X_\xi^{(\hat{k})}, Y_\xi^{(\hat{k})}, v_0$);
14:         if the vehicle $\hat{k} \in$ stationary category then
15:             $q^{(\hat{k})}(\cdot) = Q$
16:             add the vehicle $\hat{k}$ into $\Gamma$ according to the idle time sequence;
17:         end if
18:     else
19:         Select the next task $u^\star$=argmin $\{h(u)|u \in U'\}$
20:         GoTo($X_\xi^{(\hat{k})}, Y_\xi^{(\hat{k})}, head(u^\star)$);
21:         if $q^{(k)}(\cdot) \geq dc_\xi(u^\star)$ then
22:             $X_\xi^{(\hat{k})} \leftarrow (X_\xi^{(\hat{k})}, tail(u^\star)), Y_\xi^{(\hat{k})} \leftarrow (X_\xi^{(\hat{k})}, 1)$;
23:             $U \leftarrow U \backslash u^\star, U \leftarrow U \backslash \hat{u}^\star$;
24:         else                                           ▷ route failure
25:             $\theta \leftarrow q^{(k)}(\cdot)/dc_\xi(u^\star)$;
26:             $X_\xi^{(\hat{k})} \leftarrow (X_\xi^{(\hat{k})}, tail(u^\star)), Y_\xi^{(\hat{k})} \leftarrow (X_\xi^{(\hat{k})}, \theta)$;
27:             GoTo($X_\xi^{(\hat{k})}, Y_\xi^{(\hat{k})}, v_0$);
28:             if the vehicle $\hat{k} \in$ stationary category then
29:                 $q^{(\hat{k})}(\cdot) = Q$
30:                 add the vehicle $\hat{k}$ into $\Gamma$ according to the idle time sequence;
31:             end if
32:         end if
33:     end if
34: end while
35: while $\Gamma \neq \emptyset$ do
36:     GoTo($X_\xi^{(k')}, Y_\xi^{(k')}, v_0$);                 ▷ $k'$ is the idle vehicle in $\Gamma$
37: end while
38: return $S_\xi = (X_\xi, Y_\xi)$;
```

Initially, the m vehicles with full capacity are located at the depot and ready to serve (lines 1–3). The algorithm uses a time list Γ to record the next idle time of each vehicle. Each recording in Γ has two parameters: 1) the vehicle ID and 2) the idle time. Recordings are stored according to the time sequence. Then, for each time slot when a vehicle becomes ready, the routing policy is used to decide the next task that the vehicle should go. For deciding the next destination of the vehicle, a subset of candidate tasks is firstly selected from the pool by the function Filter() (line 11). This is to eliminate the infeasible tasks whose demands are greater than the remaining capacity of the vehicle. Since the actual demands of tasks are unknown before the service, their expected values are used. If no candidate task is selected, then the vehicle goes back to the depot (line 13) to update the capacity. The function GoTo$(X_\xi^{(\hat{k})}, Y_\xi^{(\hat{k})}, v)$ updates the route $(X_\xi^{(\hat{k})}, Y_\xi^{(\hat{k})})$ of the $\hat{k}$th vehicle by traversing through the current location to the vertex v *via* the shortest path taking the possible edge failure into account. Details of the GoTo$(\cdot)$ function can be found in [11]. If the vehicle is a stationary vehicle,

TABLE 2 | The terminal set.

Notation	Description
CFH	Cost From Here (the current node) to the head node of the candidate task
CR	Cost to Refill (from the current node to the depot)
CTD	Cost from the tail node of the candidate task To the Depot
CTT1	Cost from the tail of the candidate task To its closest remaining unserved Task (the head)
DEM	DEMand of the candidate task
DEM1	DEMand of the closet unserved task to the candidata task
FRT	Fraction of the Remaining (unserved) Tasks
FULL	FULLness of the vehicle (current load over capacity)
RQ	Remaining Capacity of the vehicle
SC	Serving Cost of the candidata task
ERC	a random constant number between 0 and 1

TABLE 3 | Results on the *ugdb* dataset in terms of the average on the test set.

Name	(|*V*|, |*E*|)	Vehicle no	GPHH	GPHH-Re	GPHH-OneFAll
*ugdb*1	(12,22)	5	351.34(7.03) (–)	350.09 (5.55) (–)	**345.06 (6.21)**
*ugdb*2	(12,26)	6	368.99 (4.46) (–)	366.23 (5.50) (–)	**360.52 (3.53)**
*ugdb*3	(12,22)	5	307.56(1.53) (–)	305.39 (3.59)	**302.46 (7.60)**
*ugdb*4	(11,19)	4	321.26 (1.98) (–)	**317.19 (1.67)**	317.50 (2.59)
*ugdb*5	(13,26)	6	423.67 (6.14) (–)	422.05 (5.76) (–)	**418.09 (5.15)**
*ugdb*6	(12,22)	5	347.50 (11.22)	**345.66 (8.58)**	345.83 (1.98)
*ugdb*7	(12,22)	5	351.41 (4.94) (–)	347.22 (4.69)	**346.24 (4.26)**
*ugdb*8	(27,46)	10	445.17 (8.04) (–)	430.91 (7.90) (–)	**424.84 (6.04)**
*ugdb*9	(27,51)	10	382.95 (8.83) (-)	374.90 (8.63)	**369.73 (7.17)**
*ugdb*10	(12,25)	4	296.47 (3.39) (-)	**291.97 (3.29)**	293.01 (2.58)
*ugdb*11	(22,45)	5	431.88 (5.93) (–)	431.07 (5.27) (–)	**425.00 (5.85)**
*ugdb*12	(13,23)	7	612.69 (14.59) (–)	**592.23 (7.96)**	595.61 (7.18)
*ugdb*13	(10,28)	6	576.08 (4.05) (–)	572.21 (4.17) (–)	**568.04 (2.69)**
*ugdb*14	(7,21)	5	107.77 (1.28)	107.88 (1.37)	**107.34 (1.03)**
*ugdb*15	(7,21)	4	58.10 (0.03)	58.09 (0.03)	**58.08 (0.03)**
*ugdb*16	(8,28)	5	136.23 (1.35) (–)	136.05 (0.36) (–)	**133.17 (0.47)**
*ugdb*17	(8,28)	5	91.07 (0.04)	91.07 (0.03)	**91.06 (0.04)**
*ugdb*18	(9,36)	5	167.86 (2.47)	167.17 (1.61)	**166.50 (1.27)**
*ugdb*19	(8,11)	3	**61.58 (1.26)**	61.61 (1.37)	61.69 (1.24)
*ugdb*20	(11,22)	4	127.31 (1.89)	127.43 (1.17)	**127.24 (1.12)**
*ugdb*21	(11,33)	6	164.15 (2.10)	**163.99 (2.21)**	164.10 (1.87)
*ugdb*22	(11,44)	8	209.60 (0.94) (–)	208.52 (0.75)	**207.92 (0.78)**
*ugdb*23	(11,55)	10	251.19 (1.86) (–)	249.60 (1.47) (–)	**247.33 (1.32)**
Mean			286.60	283.41	**281.58**

TABLE 4 | Results on the *uval* dataset in terms of the average on the test set.

Name	(|*V*|, |*E*|)	Vehicle no.	GPHH	GPHH-Re	GPHH-OneFAll
*uval*1A	(24,39)	2	176.68 (2.44)	**175.70 (2.18)**	175.76 (2.81)
*uval*1B	(24,39)	3	184.88 (1.95) (–)	184.54 (2.17) (–)	**182.87 (1.72)**
*uval*1C	(24,39)	8	314.63 (8.06) (–)	303.24 (5.62)	**302.85 (6.68)**
*uval*2A	(24,34)	2	230.52 (3.47)	232.40 (3.99)	**230.05 (3.39)**
*uval*2B	(24,34)	3	277.15 (3.72)	276.54 (3.54)	**276.53 (3.19)**
*uval*2C	(24,34)	8	590.76 (15.67) (–)	558.72 (17.22)	**558.70 (13.45)**
*uval*3A	(24,35)	2	**81.75 (0.61)**	81.91 (0.78)	81.86 (0.61)
*uval*3B	(24,35)	3	97.51 (2.23) (–)	95.92 (1.35)	**95.67 (1.58)**
*uval*3C	(24,35)	7	174.73 (5.04) (–)	174.71 (7.24) (–)	**164.66 (7.61)**
*uval*4A	(41,69)	3	418.31 (5.79)	422.05 (9.41)	**416.80 (5.70)**
*uval*4B	(41,69)	4	440.96 (4.41)	439.39 (5.68)	**438.44 (3.10)**
*uval*4C	(41,69)	5	491.76 (9.16) (–)	487.25 (5.85) (–)	**478.08 (5.87)**
*uval*4D	(41,69)	9	712.77 (10.05) (–)	688.26 (23.45) (–)	**660.70 (19.11)**
Mean			322.49	316.97	**312.54**

TABLE 5 | The detailed information of one instance of *ugdb*13.

Task	Demand	Actual	Serving cost	Actual traversal
	(d)	demand (d_ξ)	sc	cost (dc_ξ)
(v0, v4)	12	11.93	7	4.82
(v0, v5)	2	1.41	18	15.76
(v0, v6)	13	11.76	4	5.13
(v0, v7)	12	14.57	24	20.25
(v0, v8)	16	11.75	11	8.18
(v1, v0)	13	15.24	15	15.38
(v1, v2)	11	10.26	5	5.7
(v1, v4)	7	9.24	12	9.61
(v1, v7)	13	15.57	13	21.62
(v2, v0)	16	18.38	8	8.2
(v2, v5)	7	7.21	1	0.92
(v2, v6)	5	4.84	10	9.18
(v2, v7)	3	3.17	24	24.67
(v3, v0)	13	17.45	6	5.66
(v3, v1)	4	4.55	3	2.91
(v3, v2)	7	8.87	28	34.13
(v3, v4)	15	16.46	2	1.45
(v4, v5)	9	9.65	20	23.01
(v4, v6)	5	6.37	42	39.36
(v4, v8)	5	5.38	12	10.5
(v5, v6)	8	9.19	9	8.64
(v5, v8)	3	2.1	13	15.23
(v6, v7)	9	8.61	16	10.67
(v6, v8)	3	3.23	60	55.69
(v6, v9)	14	10.73	5	5.68
(v7, v8)	7	6.95	22	18.51
(v7, v9)	8	5.61	99	152.56
(v8, v9)	5	5.52	20	20.38

it will continue the service. Hence, its capacity is updated and its status is added to the time list (lines 14–17). Otherwise, the vehicle finishes one route in the current network, and it continues to finish the service.

If there are candidate tasks selected by the vehicle $\hat{k}$, then the task $u^\star$ with the minimal heuristic value is selected to be served next, and the vehicle goes to its head node (lines 19–20). While serving the task, if the remaining capacity is larger than the actual demand, the task $u^\star$ is served successfully and both $u^\star$ and its opposite task $\hat{u}^\star$ are removed from the unserved task set (lines 21–23). Otherwise, a route failure occurs. The vehicle partially serves the task $u^\star$ before returning to the depot (lines 25–27). The collaborative effect shows that if the vehicle belongs to the stationary category, it will refill to wait for assigning a new task (i.e., it is added to the time list again, lines 28–31). Finally, all tasks are served and vehicles go to the depot (lines 35–37).

Note that the proposed recourse strategy can be applied to any approaches that confused by the route failure, including both proactive and reactive approaches.

3.2 GPHH with the new meta-algorithm

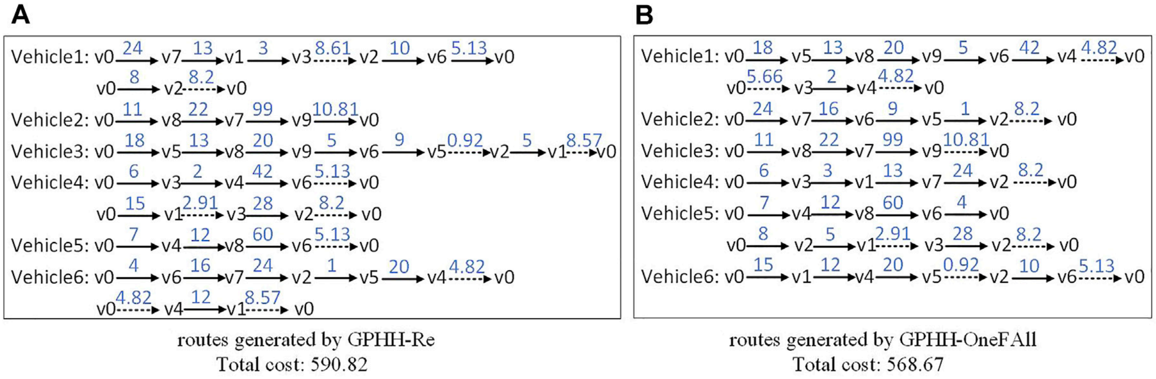
The training process of GPHH with the new meta-algorithm is described in Algorithm 2. It follows the standard GP process. During fitness evaluation (line 9), given a training set $\{I_\xi | \xi \in \Xi_{train}\}$, the fitness function of evaluating each routing policy $h(\cdot)$ is calculated as **Eq. 2**, i.e., the average total cost of the solutions obtained by applying this policy to the training samples.

$$fit(h(\cdot)) = \frac{1}{|\Xi_{train}|} \sum_{I_\xi \in \Xi_{train}} C(S_\xi, h(\cdot)) \tag{2}$$

where $C(S_\xi, h(\cdot))$ is the total cost of the solution S_ξ under the routing policy $h(\cdot)$.

Algorithm 2. The training process of GPHH

```
Input: A UCARP instance $I$, number of generations $G$
Output: A routing policy $h^*(\cdot)$
1: randomly initialise a GP population with $n$ routing policies;
2: $g$=0;
3: while $g < G$ do
4:     randomly generate a training subset $\Xi_{train}$ of $I$;
5:     for each policy $h(i)$ (i $\in$ [1, $n$]) do
6:         for each instance $I_\xi \in \Xi_{train}$ do
7:             generate a feasible solution (a set of routes) based on the meta-algorithm by Algorithm 1,
                and calculate its total cost based on Eq. (1);
8:         end for
9:         Calculate the average of costs as the fitness of the policy $h(i)$ based on Eq. (2)
10:    end for
11:    generate a new population of routing policies using GP search operators;
12: end while
13: return the best routing policy $h^*(\cdot)$ in the final population;
```


A
Vehicle1: v0 24 v7 13 v1 3 v3 8.61 v2 10 v6 5.13 v0
v0 8 v2 8.2 v0
Vehicle2: v0 11 v8 22 v7 99 v9 10.81 v0
Vehicle3: v0 18 v5 13 v8 20 v9 5 v6 9 v5 0.92 v2 5 v1 8.57 v0
Vehicle4: v0 6 v3 2 v4 42 v6 5.13 v0
v0 15 v1 2.91 v3 28 v2 8.2 v0
Vehicle5: v0 7 v4 12 v8 60 v6 5.13 v0
Vehicle6: v0 4 v6 16 v7 24 v2 1 v5 20 v4 4.82 v0
v0 4.82 v4 12 v1 8.57 v0
B
Vehicle1: v0 18 v5 13 v8 20 v9 5 v6 42 v4 4.82 v0
v0 5.66 v3 2 v4 4.82 v0
Vehicle2: v0 24 v7 16 v6 9 v5 1 v2 8.2 v0
Vehicle3: v0 11 v8 22 v7 99 v9 10.81 v0
Vehicle4: v0 6 v3 3 v1 13 v7 24 v2 8.2 v0
Vehicle5: v0 7 v4 12 v8 60 v6 4 v0
v0 8 v2 5 v1 2.91 v3 28 v2 8.2 v0
Vehicle6: v0 15 v1 12 v4 20 v5 0.92 v2 10 v6 5.13 v0

routes generated by GPHH-Re
Total cost: 590.82

routes generated by GPHH-OneFAll
Total cost: 568.67

FIGURE 2 | Examples of the routes generated by **(A)** GPHH-Re and **(B)** GPHH-OneFAll on the sampled *ugdb*13 instance in **Table 5**. The solid arrow connecting two vertices (*vi*, *vj*) means that (*vi*, *vj*) is a task and the vehicle travels from *vi* to *vj* to serve it, while the dotted arrow connecting two vertices (*vi*, *vj*) means that the vehicle travels from *vi* to *vj* following the shortest path between these two vertices without serving any tasks. The number above the solid arrow represents the serving cost, and the number above the dotted arrow represents the actual traversal cost of the shortest path.

4 EXPERIMENTAL STUDY

In order to examine the effectiveness of the proposed recourse strategy, we compare with the basic GPHH with the independent recourse strategy (named GPHH) and GPHH with the reassigned strategy proposed in [12] (named GPHH-Re, which is described as a kind of global recourse strategy in **Section 2.3**).

4.1 Experiment settings

All the compared GPHH algorithms share the same parameter settings, as shown in **Table 1**. The terminal set is given in **Table 2**. The function set is {+, −, ×,/, max, min}. The function / is protected, which returns to 1 if divided by zero. The *gdb* and *val* datasets are two commonly used benchmark datasets in the area of UCARP. In the *gdb* dataset, the number of vertices varies from 7 to 27, and the number of the arcs varies from 11 to 55. The *val* dataset is much bigger than the *gdb* dataset, and its vertices and arcs numbers vary from 24 to 41, from 34 to 69, respectively. For the number of vehicles, we suppose that it equals to the total demands of all tasks dividing the capacity of the vehicle in each scenario in the experiments. Hence, combined with the proposed new recourse strategy in **Section 3.1**, we let one vehicle as the stationary and called the new algorithm as GPHH-OneFAll. The UCARP instance generator in [13] is used to generate the training and test instances based on the static *gdb* and *val* datasets. The stochastic traversal costs and demands follow the normal distribution $\mathcal{N}(\mu, \mu \times \lambda)$, where μ is the deterministic value given in the static instance and λ is the uncertainty level. λ is set to 0.2 in our experiments. Especially, if the stochastic traversal cost $dc(e) < 0$, then it is set to ∞. That means that the arc becomes inaccessible. If the stochastic demand $d(e) < 0$, then it is set to 0. That means that the arc has no demand and is not a task in the current environment. In the experiments, each algorithm is trained on 5 randomly sampled instances in each generation, and the best routing policy $h^*(\cdot)$ is tested on 500 unseen instances. The test performance of GPHH is defined as the average total cost over the 500 samples.

4.2 RESULTS

Table 3 and **Table 4** show the test performances of the compared algorithms on the *ugdb* and *uval* UCARP instances, respectively. All the algorithms are run 20 times independently. The Wilcoxon rank-sum test with the significance level of 0.05 is conducted to compare GPHH-OneFAll with GPHH and GPHH-Re. The "(−)" means that the compared algorithms (i.e., GPHH or GPHH-Re) perform significantly worse than GPHH-OneFAll; otherwise, there is no significant difference between the two algorithms. The results with the minimum average are highlighted in bold.

As shown in **Table 3** and **Table 4**, it can be seen that the proposed GPHH-OneFAll outperformed GPHH on 22 out of the 36 instances and outperformed GPHH-Re on 12 out of the 36 instances. GPHH-OneFAll performed no worse than GPHH and GPHH-Re on any instances. Based on the results marked in bold, we can see that the GPHH algorithms with vehicle cooperations in recourse (i.e., GPHH-Re and GPHH-OneFAll) have better performance than the basic GPHH with the independent recourse strategy on almost all the instances. Only on two instances *ugdb*19 and *uval*3*A* whose vehicle number is small did the baseline GPHH obtain slightly better results. The experimental results show that the designed cooperative recourse strategy is useful for saving the total costs.

4.3 Further analysis

As we described in **Section 2.3**, GPHH-Re has the drawback that many vehicles may take a second small route to serve a few tasks because of the uncertainty. This usually increases the total cost. Furthermore, our proposed GPHH-OneFAll algorithm can overcome this drawback by appointing one vehicle to serve the remaining tasks if all other vehicles are full. To illustrate this idea more clearly, we randomly select a routing policy evolved by GPHH-Re and GPHH-OneFAll obtained from the *ugdb*13 instance, respectively. The two policies are applied on the same sampled scenario, whose detailed information (i.e., the tasks, the expected and the actual demands, the serving costs, and the actual traversal costs) is shown in **Table 5**.

The routes generated by the two routing policies are shown in **Figure 2**. It can be seen that the total cost of the routes generated by GPHH-OneFAll (i.e., 568.67) is much shorter than that of GPHH-Re (i.e., 590.82). When looking into the details of the route, we discover that half of the vehicles (i.e., Vehicle 1, Vehicle 4, and Vehicle 6) in routes generated by GPHH-Re (**Figure 2A**) take second routes with serving only one or two tasks. However, in routes generated by GPHH-OneFAll, Vehicle2, Vehicle3, Vehicle4, and Vehicle6 directly return to the depot when their remaining capacities cannot meet the requirements of the unserved candidate tasks. Vehicle1 is appointed as the *stationary vehicle*, taking a second route to serve the remaining unserved tasks. The reason why Vehicle5 takes a second tour is that when serving the task (*v*6, *v*0), its capacity is refilled because *v*0 is the depot.

5 CONCLUSION

In this paper, the emergency material allocation in the real world was formulated as UCARP, which was a classic combinatorial optimization problem under the uncertain environment. Addressing on the route failure caused by the uncertain demands of tasks, this paper proposed a new recourse strategy that divided the vehicles into two categories: flowing and stationary. The main idea of the new recourse strategy came from the real scenario that not all vehicles were needed to stay at the current area until all tasks were finished, as the goal was to help residents in wider areas. Moreover, it was easy to be applied to the real world. A GPHH algorithm based on the carefully designed meta-algorithm with the new recourse strategy for UCARP was proposed. In experiments, we let one vehicle to handle all the failed tasks or unserved tasks if other vehicles were full and called the new GPHH as GPHH-OneFAll. The experimental results showed that the proposed GPHH-OneFAll significantly outperformed the GPHH with existing recourse strategies.

For the future work, it is valuable to investigate the number of vehicles in each category, which may have a relationship with the total demands of tasks and the capacities of vehicles. As there is little knowledge on the manner of logistic companies that handle uncertain events, the recourse strategies remain mainly theoretical. Further work is to define more active recourse actions from the real data. Moreover, addressing on the global recourse strategies based on a high degree of vehicle coordination is also valuable to investigate.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Material; further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

YL and XL contributed to the conception and design of the study. SW organized the database and performed the statistical analysis. YL wrote the manuscript, and SW and XL contributed to the manuscript revision and read and approved the submitted version.

FUNDING

The work is sponsored by the National Natural Science Foundation of China (Nos. 61976181, 11931015) and the Shanghai Sailing Program (No. 21YF1416700).

REFERENCES

1. Wang C, Deng Y, Yuan Z, Zhang C, Zhang F, Cai Q, et al. How to Optimize the Supply and Allocation of Medical Emergency Resources during Public Health Emergencies. *Front Phys* (2020) 8:383. doi:10.3389/fphy.2020.00383
2. Cui M, Han D, Wang J. An Efficient and Safe Road Condition Monitoring Authentication Scheme Based on Fog Computing. *IEEE Internet Things J* (2019) 6:9076–9084. doi:10.1109/JIOT.2019.2927497
3. Zhang S, Zhang J, Zhao J, Xin C. Robust Optimization of Municipal Solid Waste Collection and Transportation with Uncertain Waste Output: A Case Study. *J Syst Sci Syst Eng* (2021) 30:1–22. doi:10.1007/s11518-021-5510-8
4. Corberán Á, Eglese R, Hasle G, Plana I, Sanchis JM. Arc Routing Problems: A Review of the Past, Present, and Future. *Networks.* (2021) 77:88–115. doi:10.1002/net.21965
5. Thibbotuwawa A, Bocewicz G, Nielsen P, Banaszak Z. Unmanned Aerial Vehicle Routing Problems: A Literature Review. *Appl Sci* (2020) 10:4504. doi:10.3390/app10134504
6. Liu J, Tang K, Yao X. Robust Optimization in Uncertain Capacitated Arc Routing Problems: Progresses and Perspectives [Review Article]. *IEEE Comput Intell Mag* (2021) 16:63–82. doi:10.1109/mci.2020.3039069
7. Nguyen S, Mei Y, Ma H, Chen A, Zhang M. Evolutionary Scheduling and Combinatorial Optimisation: Applications, Challenges, and Future Directions. In: IEEE Congress on Evolutionary Computation; 2016 July 19–24. Vancouver, BC: IEEE (2016). p. 3053–60. doi:10.1109/cec.2016.7744175
8. Ouelhadj D, Petrovic S. A Survey of Dynamic Scheduling in Manufacturing Systems. *J Sched* (2009) 12:417–31. doi:10.1007/s10951-008-0090-8
9. Wang S, Mei Y, Zhang M, Yao X. Genetic Programming with Niching for Uncertain Capacitated Arc Routing Problem. *IEEE Transactions on Evolutionary Computation* (2022) 26 (1):73–87. doi:10.1109/TEVC.2021.3095261
10. Ardeh MA, Mei Y, Zhang M. Genetic Programming with Knowledge Transfer and Guided Search for Uncertain Capacitated Arc Routing Problem. *IEEE Trans Evol Computat* (2021). doi:10.1109/tevc.2021.3129278
11. Liu Y, Mei Y, Zhang M, Zhang Z. Automated Heuristic Design Using Genetic Programming Hyper-Heuristic for Uncertain Capacitated Arc Routing Problem. In: Proceedings of GECCO; 2017 July 15–19; Berlin, Germany. New York, NY: ACM (2017). p. 290–7. doi:10.1145/3071178.3071185
12. MacLachlan J, Mei Y, Branke J, Zhang M. Genetic Programming Hyper-Heuristics with Vehicle Collaboration for Uncertain Capacitated Arc Routing Problems. *Evol Comput* (2020) 28:563–93. doi:10.1162/evco_a_00267
13. Mei Y, Tang K, Yao X. Capacitated Arc Routing Problem in Uncertain Environments. In: IEEE Congress on Evolutionary Computation; 2010 July 18–23. Barcelona, Spain: IEEE (2010). p. 1–8. doi:10.1109/cec.2010.5586031
14. Liu Y, Mei Y, Zhang M, Zhang Z. A Predictive-Reactive Approach with Genetic Programming and Cooperative Coevolution for the Uncertain Capacitated Arc Routing Problem. *Evol Comput* (2020) 28:289–316. doi:10.1162/evco_a_00256
15. Christiansen CH, Lysgaard J, Wøhlk S. A branch-and-price Algorithm for the Capacitated Arc Routing Problem with Stochastic Demands. *Operations Res Lett* (2009) 37:392–8. doi:10.1016/j.orl.2009.05.008
16. Fleury G, Lacomme P, Prins C. *Evolutionary Algorithms for Stochastic Arc Routing Problems*. Berlin Heidelberg: Springer (2004). p. 501–12. doi:10.1007/978-3-540-24653-4_51
17. Wang J, Tang K, Yao X. A Memetic Algorithm for Uncertain Capacitated Arc Routing Problems. In: 2013 IEEE Workshop on Memetic Computing; 2013 April 16–19; Singapore (2013). p. 72–9. doi:10.1109/mc.2013.6608210
18. Wang J, Tang K, Lozano JA, Yao X. Estimation of the Distribution Algorithm with a Stochastic Local Search for Uncertain Capacitated Arc Routing Problems. *IEEE Trans Evol Computat* (2016) 20:96–109. doi:10.1109/tevc.2015.2428616
19. Lacomme P, Prins C, Ramdane-Cherif W. Competitive Memetic Algorithms for Arc Routing Problems. *Ann Operations Res* (2004) 131:159–85. doi:10.1023/b:anor.0000039517.35989.6d
20. Koza JR. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. Cambridge, MA, USA: MIT Press (1992).
21. Koza JR. Genetic Programming as a Means of Programming Computers by Natural Selection. *Stat Comput* (1994) 4:87–112. doi:10.1007/bf00175355
22. Weise T, Devert A, Tang K. A Developmental Solution to (Dynamic) Capacitated Arc Routing Problems Using Genetic Programming. In: Proceedings of GECCO; 2012 July 7–11; Philadelphia, Pennsylvania. New York, NY: ACM (2012). p. 831–8. doi:10.1145/2330163.2330278
23. Xu B, Mei Y, Wang Y, Ji Z, Zhang M. Genetic Programming with Delayed Routing for Multiobjective Dynamic Flexible Job Shop Scheduling. *Evolutionary Computation* (2021) 29 (1):75–105. doi:10.1162/evco_a_00273
24. MacLachlan J, Mei Y, Branke J, Zhang M. An Improved Genetic Programming Hyper-Heuristic for the Uncertain Capacitated Arc Routing Problem. In: Australasian Joint Conference on Artificial Intelligence; 2018 December 11–14; Wellington, New Zealand. Berlin, German: Springer (2018). p. 432–44. doi:10.1007/978-3-030-03991-2_40
25. Mei Y, Zhang M. Genetic Programming Hyper-Heuristics for Multi-Vehicle Uncertain Capacitated Arc Routing Problem. In: Proceedings of GECCO; 2018 July 15–19; Kyoto, Japan. New York, NY: ACM (2018). p. 141–2.
26. Wang S, Mei Y, Zhang M. Novel Ensemble Genetic Programming Hyper-Heuristics for Uncertain Capacitated Arc Routing Problem. In: Proceedings of GECCO; 2019 July 13–17; Prague, Czech Republic. New York, NY: ACM (2019). p. 1093–101. doi:10.1145/3321707.3321797
27. Wang S, Mei Y, Zhang M. A Multi-Objective Genetic Programming Hyper-Heuristic Approach to Uncertain Capacitated Arc Routing Problems. In: 2020 IEEE Congress on Evolutionary Computation (CEC); 2020 July 19–24; Glasgow, UK. Piscataway, NJ: IEEE (2020). p. 1–8. doi:10.1109/cec48606.2020.9185890
28. Ardeh M, Mei Y, Zhang M. Genetic Programming Hyper-Heuristic with Knowledge Transfer for Uncertain Capacitated Arc Routing Problem. In: Proceedings of GECCO; 2019 July 13–17; Prague, Czech Republic. New York, NY: ACM (2019). p. 334–5. doi:10.1145/3319619.3321988

29. Ardeh MA, Mei Y, Zhang M. A Parametric Framework for Genetic Programming with Transfer Learning for Uncertain Capacitated Arc Routing Problem. In: Australasian Joint Conference on Artificial Intelligence; 2020 November 29–30; Canberra, ACT, Australia. Berlin, German: Springer (2020). p. 150–62. doi:10.1007/978-3-030-64984-5_12
30. Dror M, Laporte G, Trudeau P. Vehicle Routing with Stochastic Demands: Properties and Solution Frameworks. *Transportation Sci* (1989) 23:166–76. doi:10.1287/trsc.23.3.166
31. Secomandi N. A Rollout Policy for the Vehicle Routing Problem with Stochastic Demands. *Operations Res* (2001) 49:796–802. doi:10.1287/opre.49.5.796.10608
32. Bertazzi L, Secomandi N. Faster Rollout Search for the Vehicle Routing Problem with Stochastic Demands and Restocking. *Eur J Oper Res* (2018) 270:487–97. doi:10.1016/j.ejor.2018.03.034
33. Goodson JC, Thomas BW, Ohlmann JW. Restocking-based Rollout Policies for the Vehicle Routing Problem with Stochastic Demand and Duration Limits. *Transportation Sci* (2016) 50:591–607. doi:10.1287/trsc.2015.0591
34. Novoa C, Storer R. An Approximate Dynamic Programming Approach for the Vehicle Routing Problem with Stochastic Demands. *Eur J Oper Res* (2009) 196: 509–15. doi:10.1016/j.ejor.2008.03.023
35. Nguyen S, Zhang M, Johnston M, Tan KC. A Computational Study of Representations in Genetic Programming to Evolve Dispatching Rules for the Job Shop Scheduling Problem. *IEEE Trans Evol Computat* (2013) 17: 621–39. doi:10.1109/TEVC.2012.2227326
36. Gendreau M, Jabali O, Rei W. 50th Anniversary Invited Article-Future Research Directions in Stochastic Vehicle Routing. *Transportation Sci* (2016) 50:1163–73. doi:10.1287/trsc.2016.0709
37. Ak A, Erera AL. A Paired-Vehicle Recourse Strategy for the Vehicle-Routing Problem with Stochastic Demands. *Transportation Sci* (2007) 41:222–37. doi:10.1287/trsc.1060.0180
38. Lei H, Laporte G, Guo B. The Vehicle Routing Problem with Stochastic Demands and Split Deliveries. *INFOR: Inf Syst Oper Res* (2012) 50:59–71. doi:10.3138/infor.50.2.059
39. Erera AL, Savelsbergh M, Uyar E. Fixed Routes with Backup Vehicles for Stochastic Vehicle Routing Problems with Time Constraints. *Networks* (2009) 54:270–83. doi:10.1002/net.20338
40. Salavati-Khoshghalb M, Gendreau M, Jabali O, Rei W. A Rule-Based Recourse for the Vehicle Routing Problem with Stochastic Demands. *Transportation Sci* (2019) 53:1334–53. doi:10.1287/trsc.2018.0876
41. Han D, Zhu Y, Li D, Liang W, Souri A, Li K-C. A Blockchain-Based Auditable Access Control System for Private Data in Service-Centric IoT Environment. *IEEE Trans Industr Inform* (2022) 18:3530–3540. doi:10.1109/TII.2021.3114621

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers in Physics

PERSPECTIVE
published: 22 February 2022
doi: 10.3389/fphy.2021.827468

Dynamic Influence Maximization *via* Network Representation Learning

Wei Sheng, Wenbo Song, Dong Li, Fei Yang and Yatao Zhang*

School of Mechanical, Electrical and Information Engineering, Shandong University, Weihai, China

Influence maximization is a hot research topic in the social computing field and has gained tremendous studies motivated by its wild application scenarios. As the structures of social networks change over time, how to seek seed node sets from dynamic networks has attracted some attention. However, all of the existing studies were based on network topology structure data which have the limitations of high dimensionality and low efficiency. Aiming at this drawback, we first convert each node in the network to a low-dimensional vector representation by network representation learning and then solve the problem of dynamic influence maximization in the low-dimensional latent space. Comprehensive experiments on NetHEPT, Twitter, UCI, and Wikipedia datasets show that our method can achieve influence diffusion performance similar to state-of-the-art approaches in much less time.

Keywords: dynamic social networks, influence maximization, network representation learning, incremental updating, efficiency

OPEN ACCESS

Edited by:
Chao Gao,
Southwest University, China

Reviewed by:
Hongbin Pei,
Xi'an Jiaotong University, China
Qi Tan,
The University of Hong Kong, Hong Kong SAR, China

****Correspondence:***
Dong Li
dongli@sdu.edu.cn

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *02 December 2021*
Accepted: *22 December 2021*
Published: *22 February 2022*

Citation:
Sheng W, Song W, Li D, Yang F and Zhang Y (2022) Dynamic Influence Maximization via *Network Representation Learning. Front. Phys. 9:827468. doi: 10.3389/fphy.2021.827468*

1 INTRODUCTION

With the development of online social websites, information diffusion over social networks has become a new and important channel for network users to receive information. How to optimize and control the spread of information is an important problem in the field of social computing, and influence maximization is used to solve the above problem. As all kinds of information (e.g., advertisement, rumor, and political opinion) propagate in the network, therefore, the studies of influence maximization have been widely applied in viral marketing, rumor control, political campaign, and so on.

The influence maximization problem originated from "viral marketing" and "word-of-mouth effect." In 2001, Domingos and Richardson [1, 2] first introduced the initial concept and evaluation index of the influence maximization problem. Later, Kempe et al. [3] first proposed the discrete optimization method to solve the influence maximization problem. However, the greedy algorithm with approximate accuracy guarantee proposed by them takes too much time. Leskovec et al. [4] reduced time consumption by optimizing the sub-model of the function, and the CELF they proposed was nearly 700 times faster than the traditional greedy algorithm. After that, researchers proposed other greedy algorithms [5–8]. In addition, subsequent researchers also proposed heuristic algorithms to improve operating efficiency. Chen et al. [9] developed a degree discount heuristic algorithm (DegreeDiscountIC), which has the same performance as the greedy algorithm but greatly reduces the computation time. Chen proposed the LDAG heuristic algorithm [10] based on the directed acyclic graph and MIA based on the tree structure [11]. Goyal et al. [12] proposed SIMPATH, which obtains the path of the node by backtracking and then uses the shortest path of the neighbor node to propagate the influence. Tang et al. [13] proposed TIM, which is an approximate linear solution based on reverse random sampling. Wang et al. [14] proposed CNCG considering an overlapping community structure [15, 16] and node coverage gain mechanism.

The social networks supporting information diffusion are not static but are temporal dynamic. Considering spread dynamics and structure dynamics together, exploring influence maximization in dynamic networks is an interesting and valuable problem. Zhuang et al. [17] updated the observed network by periodically detecting nodes in the real network and then selected seed nodes in the observed network as the approximate solution of the real network. Tong et al. [18] extended the static independent cascade model to the dynamic independent cascade (DIC) model and proposed two algorithms: A-greedy based on the greedy strategy and H-greedy based on the heuristic. Bao et al. [19] proposed the RSB algorithm based on multi-arm tiger machine optimization, which is suitable for dynamic non-stationary social networks. Wang [20] proposed an incremental algorithm based on the linear threshold model (DIM), which identifies top-K users in dynamic social networks based on information from the previous network. Liu et al. [21] proposed IncInf based on the independent cascade model and updated the seed set according to the topological changes of network evolution, which significantly reduces the running time of the algorithm. Chen et al. [22] extended the problem of dynamic network influence maximization and proposed the upper bound interchange (UBI) greedy algorithm to solve the problem of influence node tracking.

Although influence maximization in dynamic social networks has attracted some attention, all of the existing studies were constructed on network topology structure data which suffer from high dimensionality and low efficiency. Network representation learning aims to convert each node in the network to a low-dimensional latent representation, which has been widely applied in the tasks of visualization, clustering, classification, and link prediction. The network low-dimensional vector representation not only preserves structural feature relationships between nodes but also effectively alleviates the problem of network data sparsity. Based on the above discussions, leveraging network representation learning methods to help solve influence maximization in dynamic networks is a meaningful attempt.

In this paper, we develop dynamic influence maximization based on network representation learning, referred to as *DIMNRL*. First, we leverage network representation learning to obtain the low-dimensional vector representation of each node under different time steps and then construct the influence calculation method of node sets in the low-dimensional latent space. Next, aiming at the dynamic property of social networks, we propose an incremental node seed selection method to obtain the node set with maximum influence at different times. Comprehensive experimental results on NetHEPT, Twitter, UCI, and Wikipedia datasets demonstrate that compared with the state-of-the-art approaches, our method can yield similar performance in terms of influence spread but run much faster.

The rest of this paper is organized as follows: In **Section 2**, we introduce the definition of the problem of maximizing the influence of dynamic social networks and the design of the *DIMNRL* method in detail. **Section 3** shows the results and analysis of our experiments. Finally, in **Section 4**, we put forward conclusions and discussions.

2 MATERIALS AND METHODS

2.1 Problem Definition

We first define a dynamic network as $\mathcal{G} = (G^0, G^1, \ldots, G^t)$, where $G^t = (V^t, E^t, W^t)$ is the network snapshot of the dynamic social network at time t. We assume that the network snapshot G^0 at $t = 0$ is the initial network. $\triangle G^t = (\triangle V^t, \triangle E^t, \triangle W^t)$ is the change of network topology structure of G^t at time t, so the network topology at time $t + 1$ can be obtained by $G^{t+1} = G^t \cup \triangle G^t$. Based on the above definitions, the details of the problem we try to solve are as follows.

Influence maximization in the dynamic network. Given the topology structure G^t of a network and the network topology change $\triangle G^t$ at time t, the aim is to seek a seed set S^{t+1} with k nodes in G^{t+1} at time $t + 1$ such that the expected diffusion influence $\sigma(S^{t+1})$ reaches the maximum value. For ease of description, this problem is also referred to as dynamic influence maximization.

2.2 Framework of DIMNRL

In this section, we develop dynamic influence maximization based on network representation learning, referred to as *DIMNRL*. The main idea of *DIMNRL* is to update the seed nodes by combining the information provided by the structure change of the dynamic network with seed nodes selected from the network at the previous time, so as to greatly reduce the time to obtain the seed node set with maximum influence at the current time. The *DIMNRL* is designed in the low-dimensional latent space gained by network representation learning, which is helpful to reduce the computational complexity and improve the computational efficiency.

Figure 1 presents the framework of our proposed *DIMNRL*, which is divided into three stages: dynamic network representation learning, initial seed set calculation, and seed set incremental update. The first stage is to get the low-dimensional vector representation of the dynamic network. The second stage is to obtain the initial seed set with maximum influence from the initial network. The last stage aims to incrementally update seed nodes to gain the seed sets of the networks at all times.

2.2.1 Dynamic Network Representation Learning

The *DIMNRL* is to seek seed node sets from dynamic networks in the low-dimensional space. Therefore, we first need a network representation learning method to obtain low-dimensional vector representations of network snapshots at different time steps. Network representation learning is able to map large-scale and high-latitude networks to the low-dimensional space according to the relevant optimization objectives and use the vector representation of low-dimensional space to represent nodes, so as to preserve the topological structure and attribute characteristics of the original network as much as possible.

In this paper, we adopt the *DynamicTriad* [23] method which is suitable for dynamic social networks. *DynamicTriad* learns the embedding vector of each node at different time steps by quantifying the probability of an open triad evolving into a closed triad and proposes a semi-supervised learning

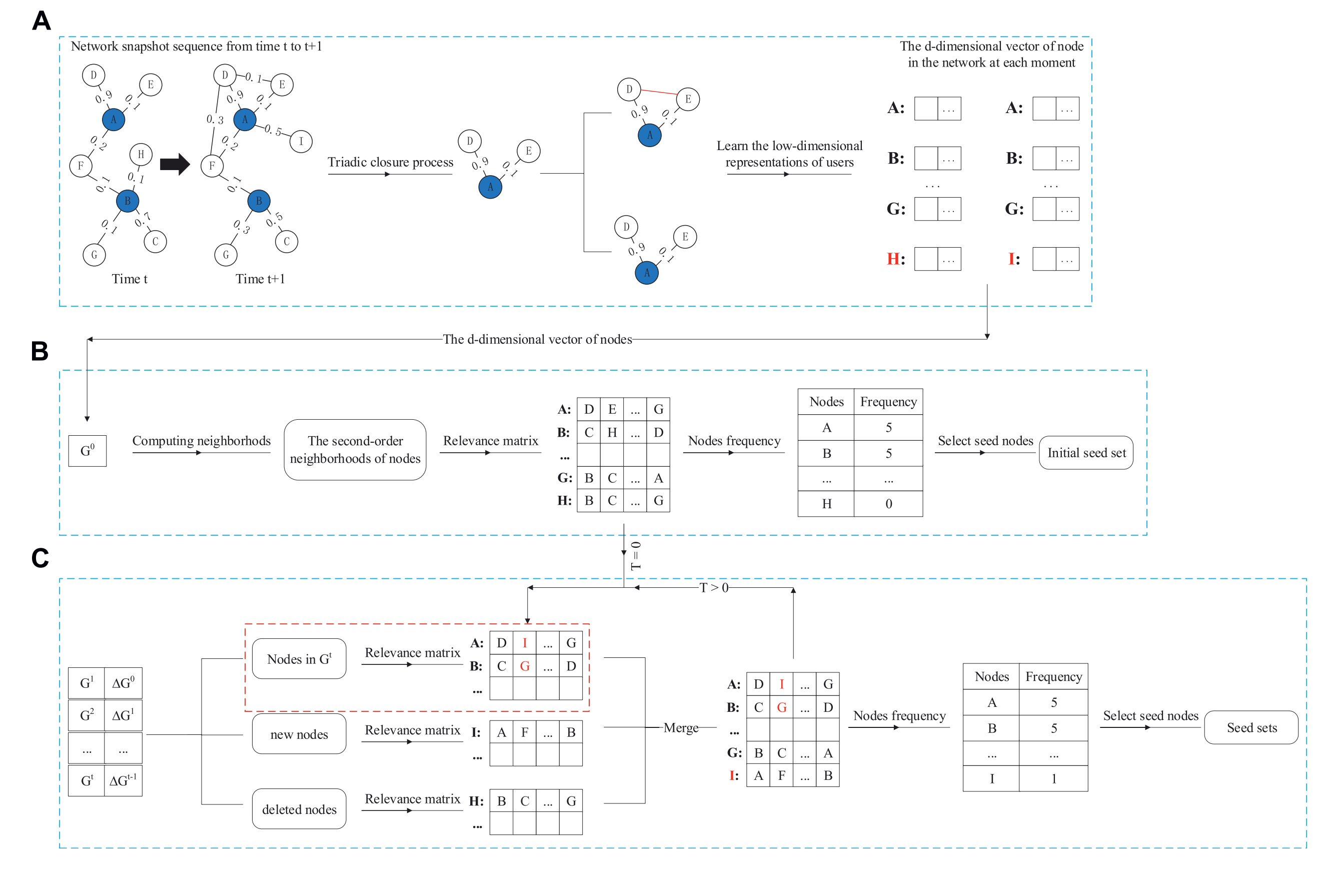

A
Network snapshot sequence from time t to t+1
Time t
Time t+1
Triadic closure process
Learn the low-dimensional representations of users
The d-dimensional vector of node in the network at each moment
A:
B:
G:
H:
I:
B
The d-dimensional vector of nodes
G⁰
Computing neighborhods
The second-order neighborhoods of nodes
Relevance matrix
Nodes frequency
Nodes
Frequency
Select seed nodes
Initial seed set
C
T = 0
T > 0
Nodes in Gᵗ
new nodes
deleted nodes
Merge
Seed sets

FIGURE 1 | Overall framework of DIMNRL. **(A)** Getting the low-dimensional vector representation of the dynamic network, **(B)** obtaining the initial seed set with maximum influence from the initial network, and **(C)** incrementally updating seed nodes to gain the seed sets of the networks at all times.

algorithm for effective parameter estimation to optimize the model parameters. This algorithm can embed the time-varying characteristics of the network into the vector representation of nodes while maintaining the network structure properties.

2.2.2 Initial Seed Set Calculation

Keikha et al. [24] proposed the *DeepIM* method to solve the influence maximization problem in the static network using network representation learning. The overall idea of *DeepIM* is using the network representation learning algorithm to generate the vectors of nodes and then calculating the similarity of chords between nodes to select *r* nodes with the highest similarity as the correlation vector of each node in the network. Next, the nodes are sorted according to the number of occurrences of nodes in the correlation vectors of all nodes in the network. Finally, the top *k* nodes are selected as the seed sets for the network. Here, we attempt to improve the *DeepIM* algorithm from two aspects so that it can be applied to dynamic networks.

First, researchers have found that the influence of nodes decreases with the increase of paths in the propagation process, and the influence propagation range of nodes in the network can usually reach the range of third-order neighbors [25] or second-order neighbors [26]. However, *DeepIM* calculates the similarity between one node and all other nodes, and this calculation method is time-consuming and unnecessary. Aiming at this drawback, we choose a pruning strategy to limit the computation of correlation vectors for each node from the whole network to the second-order neighborhood. This pruning strategy can improve the computational efficiency of the overall solution and also ensure that *r* nodes most similar to the target node are achievable.

Second, *DeepIM* selects seed nodes from social networks according to the number of nodes appearing in the correlation vectors of all nodes, and this may cause overlapping influence in the propagation process due to the situation that the k seed nodes have many common neighbors. Therefore, it is necessary to introduce a covering mechanism to disperse seed nodes. Here, we propose a threshold rule to overcome the above limitation. We adopt the FIDD algorithm [27] to calculate the degree of common neighbors of two nodes in the network, and its formula is as follows:

$$CN(i,j) = \frac{|N(i) \cap N(j)|}{|N(i) \cup N(j)|}, \qquad (1)$$

where *CN*(*i*, *j*) denotes the common neighbor degree of nodes *i* and *j* in the network and *N*(*i*) and *N*(*j*) denote the node sets in the secondary neighbors of nodes *i* and *j*, respectively. To reduce the influence overlap of seed set nodes during the propagation process, we set a threshold α to disperse the seed set nodes. If the *CN* values between the new selected node and the existing seed nodes are greater than the threshold α, this node will be ignored. Otherwise, if all *CN* values between the new selected node and the seed nodes are smaller than the threshold, this node will be added to the seed node set.

The specific algorithm is shown in **Algorithm 1**. In line 1, we obtain the second-order neighborhoods of each node. Lines 2–3 initialize the node's correlation vector and seed node set. In lines 4–8, we obtain the correlation vector of the node. Lines 5–7 calculate the cosine similarity between the node and all the nodes in its secondary neighborhood by **Eq 2**. u_u^t and u_v^t represent the vector representations of nodes u and v in the network at time t. The correlation vector of each node and the minimum cosine similarity will be stored:

$$\text{cosine}\left(u_u^t, u_v^t\right) = \frac{u_u^t \cdot u_v^t}{|u_u^t \| u_v^t|} = \frac{\sum_{i=1}^{d} u_u^t u_v^t}{\sqrt{\sum_{i=1}^{d} \left(u_u^t\right)^2} \sqrt{\sum_{i=1}^{d} \left(u_v^t\right)^2}}. \tag{2}$$

In lines 9–13, after obtaining the correlation vectors of all nodes, we count the number of times each node appears in the correlation vector. In lines 14–20, we select our initial seed node set from the initial network according to our threshold mechanism.

Algorithm 1. Initial seed set selection.

```
Input: initial network snapshot $G^0$, vector representation $embedding\_vector$ of nodes in network snapshot
    $G^0$, relevant vector size $r$, threshold of coverage mechanism $\alpha$, seed size $k$
Output: init seed node set $S^0$, $r$ correlation vector $r\_cosine$, minimum similarity $min\_cosine$ in $r$
    correlation vectors for each node in the network
 1: Initialize second-order neighborhood of each node in the network snapshot $neighborhoods$
 2: Initialize $relevant\_vector$, $occurance\_vector$
 3: Initialize $min\_cosine$, $r\_cosine$, $S^0 = \oslash$
 4: for each $v$ from $G^0$ do
 5:     $relevant\_vector[v] = neighborhoods[v] \cdot most\_similar(v, r)$
 6:     $r\_cosine[v] = relevant\_vector[v]$
 7:     $min\_cosine[v] = relevant\_vector[v][-1]$
 8: end for
 9: for each $i$ from $G^0$ do
10:     for $j = 1, \ldots, r$ do
11:         $occurance\_vector[relevant\_vector[i][j]] + +$
12:     end for
13: end for
14: while $|S^0| < K$ do
15:     select $v_{max} = max(occurance\_vector)$
16:     calculate $CN(v_{max}, S^0)$
17:     if all $CN > \alpha$ then
18:         $S^0 \cup v_{max}$
19:     end if
20: end while
```

2.2.3 Seed Set Incremental Update

When the network topology structure evolves slightly, the seed node set with maximum influence in this network also will not change violently. Therefore, based on network snapshots G^t at current time, seed set S^t of G^t, and network topology change $\triangle G^t$ at time *t*, designing an incremental algorithm to obtain S^{t+1} for network snapshots G^{t+1} quickly and accurately should be possible.

The evolution behaviors of social networks can be classified into six categories: inserting or deleting nodes, creating or deleting edges, and increasing or reducing edges' weights. Here, we denote insertNodes, deleteNodes, addEdges, removeEdges, increaseWeight, and reduceWeight as the above six evolution behaviors. When the topology of the network changes by adding edges or nodes, some nodes may propagate influence using added new edges or nodes. When the network topology changes by reducing edges or deleting nodes, the influence propagation of some nodes may be interrupted. In the network low-dimensional representation, the information of edge change and edge weight change is retained in the vector representations of nodes. Therefore, our incremental algorithm mainly considers the impact of node change behaviors (i.e., insertNodes and deleteNodes) on seed node selection.

Algorithm 2 presents the details of updating seed nodes. Lines 1–2 initialize variables such as the correlation vector of the node model. In line 3, the change of the second-order neighborhood of each node in the network G^{t+1} at time $t + 1$ is obtained and divided into three categories: nodes existing in G^t, new nodes, and deleted nodes. In lines 4–12, the correlation vectors of the network nodes at the previous moment are updated. For each added node, we calculate the cosine similarities between this node and its second-order neighborhoods and compare them with *min_cosine*. When the cosine similarity is greater than *min_cosine*, the node is added to the correlation vector. Otherwise, it is deleted. Lines 13–15 are to obtain the correlation vector of the new added node. This calculation process is the same as that of obtaining the initial seed set. Lines 16–18 are for the deleted nodes, and we directly delete the relevant vector of these nodes. Line 20 calculates the number of nodes in the correlation vector. In lines 21–27, we finally get the seed node set at time $t + 1$ according to the threshold mechanism.

Algorithm 2. Seed set incremental update.

```
Input: network snapshot $G^t$ and $G^{t+1}$, vector representation $embedding\_vector$ of nodes in network
    snapshot $G^{t+1}$ , relevant vector size $r$, threshold of coverage mechanism $\alpha$, seed size $k$, $r$ correlation
    vector $r\_cosine$ at time $t$ , minimum similarity $min\_cosine$ in $r$ correlation vectors for each node in
    the network at $t$ time
Output: seed set $S^{t+1}$, $r$ correlation vector $r\_cosine$ at time $t + 1$, minimum similarity $min\_cosine$ in $r$
    correlation vectors for each node in the network at $t + 1$ time
 1: Initialize $relevant\_vector$, $occurance\_vector$
 2: Initialize $min\_cosine$, $r\_cosine$, $S^{t+1} = \oslash$
 3: Obtain the change of second-order neighbor nodes $\triangle neighborhoods$ according to $\triangle G^t$
 4: for each $v$ from $G^t$ do
 5:     $relevant\_vector[v] = relevant\_vector[v] \cup r\_cosine[v]$
 6:     for each $u$ from $\triangle neighborhoods[v]$ do
 7:         if $cosine\_similarity(v, u) >= min\_cosine[v]$ then
 8:             $relevant\_vector[v] = relevant\_vector[v] \cup cosine\_similarity(v, u)$
 9:         end if
10:     end for
11:     $relevant\_vector[v] = relevant\_vector[v][: r]$
12: end for
13: for each $v$ from addNodes do
14:     $relevant\_vector[v] = \triangle neighborhoods[v] \cdot most\_similar(v, r)$
15: end for
16: for each $v$ from removesNodes do
17:     $relevant\_vector- = relevant\_vector[v]$
18: end for
19: update $r\_cosine$ and $min\_cosine$
20: Calculate the number of node occurrences and get $occurance\_vector$
21: while $|S^{t+1}| < K$ do
22:     select $v_{max} = max(occurance\_vector)$
23:     calculate $CN(v_{max}, S^{t+1})$
24:     if all $CN > \alpha$ then
25:         $S^{t+1} \cup v_{max}$
26:     end if
27: end while
```

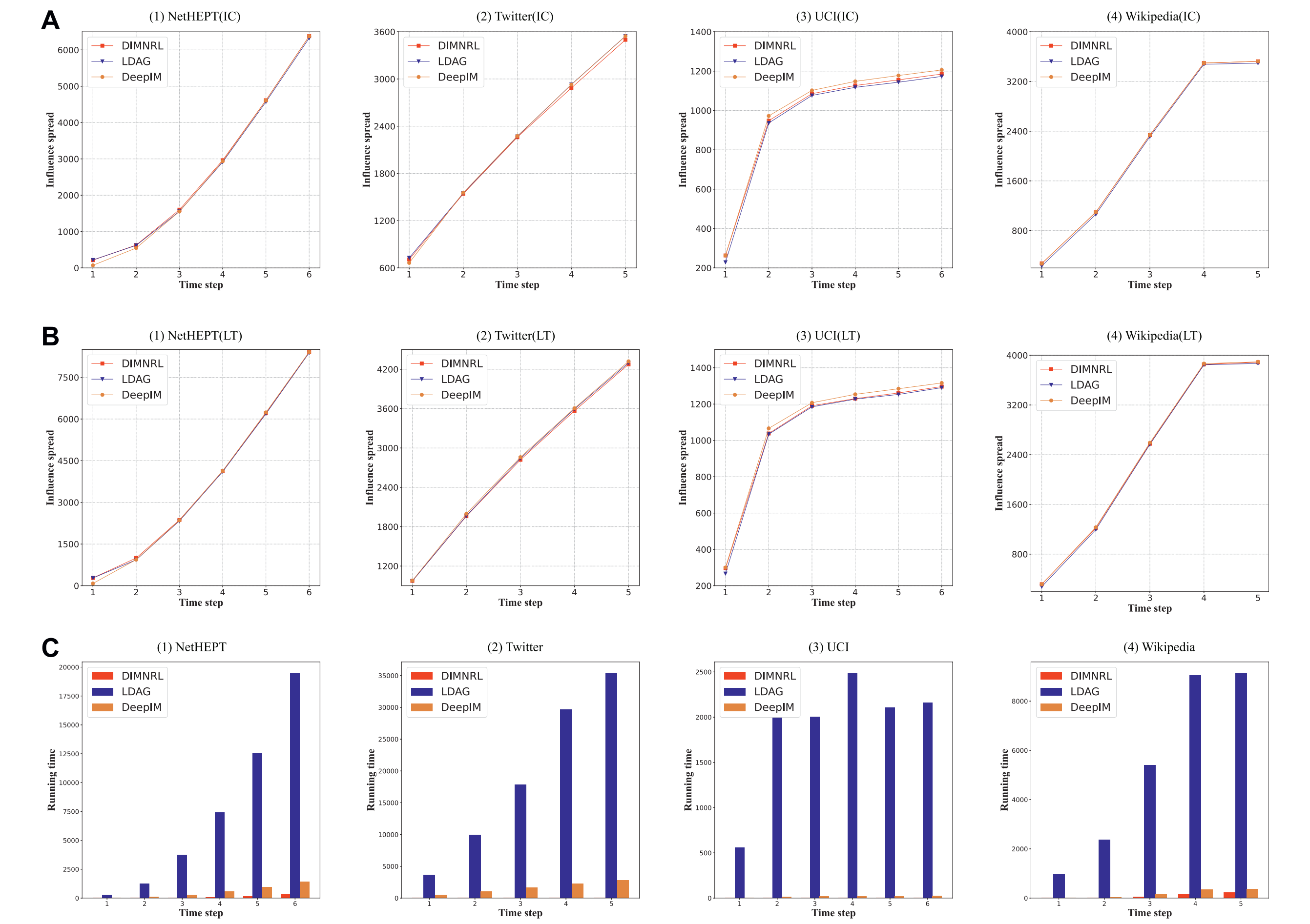


FIGURE 2 | Results of our experiments. **(A)** Influence coverage of different algorithms on four datasets under the IC model, **(B)** influence coverage of different algorithms on four datasets under the LT model, and **(C)** running time of three algorithms on four social network datasets.

3 RESULTS

3.1 Datasets and Baselines

We select four real social network datasets (NetHEPT, Twitter, UCI, and Wikipedia) to validate our method. **Supplementary Table S1** summarizes the details of the four datasets. As these four datasets do not provide the information of edge weights, here we adopt the uniformly model [28] to generate edge weights. Specifically, the weight of each edge is assigned to be 0.1 in the experiments.

- NetHEPT: It is a citation network of “High Energy Physics-Theory” from arXiv [29]. The dataset contains all the papers from January 1992 to April 2003. The edge between two nodes represents one paper citing another paper. We take the citation networks of 1992 and 1993 as the initial networks, and the network snapshot interval is 1 year. We use all the data from 1992 to 1998.
- Twitter: These data are extracted from Twitter, which records forwarded tweets between users from September 2010 to November 2010 [30]. Each edge indicates that one user has forwarded tweets from another. We set 10 days as the network snapshot interval.
- UCI: These data come from a Facebook-like online community at the University of California, Irvine, which records the data from April to October 2004 [31]. Each edge indicates that two users have communicated at least one piece of information. The network snapshot is set to be 1 month.
- Wikipedia: These data are from Wikipedia, which records historical data on all Wikipedia administrator elections and votes from 2004 to 2008 [32]. We record network snapshots every 1 year.

We adopt *LDAG* [10] and *DeepIM* [24] as the baseline methods and compare them with our *DIMNRL* solution in terms of influence diffusion range and running time. *LDAG* is a static network influence maximization algorithm based on the directed cyclic graph. The threshold parameter θ of *LDAG* in the experiment is set to 1/320, which is consistent with that in the original paper. *DeepIM* is an influence maximization algorithm based on deep learning. The correlation vector size *r* of *DeepIM* is set to 50 in the experiment.

To compare different methods under unified standards, we run the simulations using the independent cascade (IC) model and linear threshold (LT) model to obtain the influence of these seed node sets selected by *LDAG*, *DeepIM*, and *DIMNRL*. The propagation probability of the IC model and LT model is set to 0.1. All our experiments were carried out on the laptop of Inter(R) Core(TM) i7-10750H CPU @ 2.60Ghz and 16 GB RAM.

3.2 Effectiveness Evaluation

Figure 2A shows the influence spread ranges achieved by three methods on four datasets at different time steps, under the IC model. Each subfigure is corresponding to the result of a dataset, where red, blue, and brown curves represent *DIMNRL*, *DeepIM*, and *LDAG* methods, respectively. The *x*-axis represents the time step, and the *y*-axis represents the influence spread ranges of seed nodes selected by different methods. Similarly, **Figure 2B** shows the influence spread ranges achieved by three methods on four datasets at different time steps, under the LT model.

It can be seen from **Figures 2A,B** that, in UCI and Wikipedia datasets, *LDAG* has the worst performance compared with the other two methods (*DIMNRL*, *DeepIM*). On NetHEPT and Twitter datasets, *LDAG* has a bit of advantage over other methods in starting a few time steps and then achieves similar performance. *DIMNRL* and *DeepIM* achieve similar performance on different datasets at different time steps; *DIMNRL* improves *DeepIM* from the aspect of similar node selection, and the above experimental results validate the rationality of our improvement that only focuses on the second-order neighborhood.

3.3 Efficiency Evaluation

Figure 2C presents the running time of different methods for selecting 50 seed nodes from four datasets. Red, blue, and brown columns represent running time consumed by *DIMNRL*, *DeepIM*, and *LDAG*, respectively. It can be seen from **Figure 2C** that the running time of our *DIMNRL* method in each network snapshot is much shorter than that of the other two baseline methods. The *LDAG* method takes the most time, and the running time is especially longer in the Twitter dataset. The *DeepIM* algorithm finds seed nodes by traversing all nodes of the entire network, and the running time is still very terrible in a large-scale network with a large number of nodes. On NetHEPT and Twitter datasets, the running time of the *DeepIM* method is 2–3 orders of magnitude higher than our solution on each network snapshot. The above results fully demonstrate the high efficiency of our *DIMNRL* method.

In **Figure 2C**, we observe that the time consumption of the *DIMNRL* method at different times is not monotonous on the UCI dataset. The time consumption of *DIMNRL* in time step 2 is 3.64 s, while the time consumption in time steps 3 and 4 is 3.47 and 3.5 s, respectively. This is because our incremental seed selection method is closely related to the severity of network evolution. When the topology of the network snapshot changes greatly, the update algorithm takes a long time. When the topology of the network snapshot changes slightly, the running time of the update algorithm is short.

Based on the results shown in **Figure 2**, our *DIMNRL* method can achieve a similar or better influence performance than baseline methods, but the running time is much less. This means that the *DIMNRL* has the potential to effectively solve the influence maximization problem in large-scale dynamic social networks.

4 CONCLUSION

How to find seed node sets from temporal dynamic networks is an important extension direction in the research of influence maximization. In this paper, we combine network representation learning and influence maximization together and try to solve the influence maximization problem in dynamic networks via network low-dimensional vector representations. Extensive experiments on NetHEPT, Twitter, UCI, and Wikipedia datasets show that our method is able to achieve influence spread performance similar to existing methods but run much faster. These results fully illustrate the necessity and effectiveness of using network representation learning to maximize influence propagation over dynamic networks.

Network representation learning represents nodes as low-dimensional dense vectors and retains all information in the network as much as possible. It is possible to use network representation learning to obtain special information in social networks. Recently, network representation learning has made progress in signed networks [33], location-based networks [34], and hypernetworks [35]. Our solution in the low-dimensional latent space is not limited to dynamic influence maximization but applicable to polarity influence maximization, location-related influence maximization, and influence maximization in hypernetworks.

In addition to structural dynamics considered in this paper, relationship polarity, user preference, and geographic location also affect the effect of information/influence diffusion. How to integrate these factors into influence maximization research at the same time as much as possible is our next research direction. In today's society, traditional media still play an important role in information dissemination. Information diffusion driven by traditional media and information spread over social networks are not isolated. How to leverage both traditional and online media together to maximize information propagation will be an interesting problem.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/**Supplementary Material**, and further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

DL conceptualized the research idea. WS analyzed the data and ran the software. WS and DL performed the methodology and wrote the original draft. WS, FY, DL, and YZ reviewed the paper, and WS, FY, and YZ edited the paper.

FUNDING

This work was funded by the National Natural Science Foundation of China (Nos. 62076149, 61702138, 61602128, 61672322, and 61672185), the Shandong Province Natural Science Foundation of China (Nos. ZR2019MF011 and ZR2016FQ13), the China Postdoctoral Science Foundation (Nos. 2020T130368, 2019M662360, 2017M621275, and 2018T110301), the Young Scholars Program of Shandong University, China, Weihai (No. 1050501318006), and the Science and Technology Development Plan of Weihai City, China (No. 1050413421912).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fphy.2021.827468/full#supplementary-material

REFERENCES

1. Richardson M, Domingos P. Mining Knowledge-Sharing Sites for Viral Marketing. In: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, Edmonton, AB, July 23–26, 2002 (2002). p. 61–70. doi:10.1145/775047.775057
2. Domingos P, Richardson M. Mining the Network Value of Customers. In: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, California, San Francisco, August 26–29, 2001 (2001). p. 57–66. doi:10.1145/502512.502525
3. Kempe D, Kleinberg J, Tardos É. Maximizing the Spread of Influence through a Social Network. In: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, Washington, DC, August 24–27, 2003 (2003). p. 137–46. doi:10.1145/956750.956769
4. Leskovec J, Krause A, Guestrin C, Faloutsos C, VanBriesen J, Glance N. Cost-effective Outbreak Detection in Networks. In: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, San Jose, CA, August 12–15, 2007 (2007). p. 420–9.
5. Goyal A, Lu W, Lakshmanan LV. Celf++ Optimizing the Greedy Algorithm for Influence Maximization in Social Networks. In: Proceedings of the 20th international conference companion on World wide web, Hyderabad, India, March 28–April 1, 2011 (2011). p. 47–8.
6. Cheng S, Shen H, Huang J, Zhang G, Cheng X. Staticgreedy: Solving the Scalability-Accuracy Dilemma in Influence Maximization. In: Proceedings of the 22nd ACM international conference on Information & Knowledge Management, Atlanta, GA, October 17–21, 2012 (2013). p. 509–18.
7. Heidari M, Asadpour M, Faili H. Smg: Fast Scalable Greedy Algorithm for Influence Maximization in Social Networks. *Physica A: Stat Mech its Appl* (2015) 420:124–33. doi:10.1016/j.physa.2014.10.088
8. Kundu S, Pal SK. Deprecation Based Greedy Strategy for Target Set Selection in Large Scale Social Networks. *Inf Sci* (2015) 316:107–22. doi:10.1016/j.ins.2015.04.024
9. Chen W, Wang Y, Yang S. Efficient Influence Maximization in Social Networks. In: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, Paris, France, June 28–July 1, 2009 (2009). p. 199–208. doi:10.1145/1557019.1557047
10. Chen W, Wang C, Wang Y. Scalable Influence Maximization for Prevalent Viral Marketing in Large-Scale Social Networks. In: Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining; 25-28, 2010; Washington, DC, USA (2010). p. 1029–38. doi:10.1145/1835804.1835934
11. Chen W, Yuan Y, Zhang L. Scalable Influence Maximization in Social Networks under the Linear Threshold Model. In: Proceeding of the 2010 IEEE international conference on data mining; 13-17 Dec. 2010; Sydney, NSW, Australia. IEEE (2010). p. 88–97. doi:10.1109/ICDM.2010.118
12. Goyal A, Lu W, Lakshmanan LVS. SIMPATH: An Efficient Algorithm for Influence Maximization under the Linear Threshold Model. In: Proceeding of the 2011 IEEE 11th international conference on data mining; 11-14 Dec. 2011; Vancouver, BC, Canada. IEEE (2011). p. 211–20. doi:10.1109/ICDM.2011.132
13. Tang Y, Xiao X, Shi Y. Influence Maximization: Near-Optimal Time Complexity Meets Practical Efficiency. In: Proceedings of the 2014 ACM SIGMOD international conference on Management of data, Snowbird, UT, June 22–27, 2014 (2014). p. 75–86.
14. Wang Z, Sun C, Xi J, Li X. Influence Maximization in Social Graphs Based on Community Structure and Node Coverage Gain. *Future Generation Comput Syst* (2021) 118:327–38. doi:10.1016/j.future.2021.01.025
15. Zhang F, Liu H, Leung Y-W, Chu X, Jin B. Cbs: Community-Based Bus System as Routing Backbone for Vehicular Ad Hoc Networks. *IEEE Trans Mobile Comput* (2016) 16(8):2132–46. doi:10.1109/TMC.2016.2613869
16. Zhang F, Zhang D, Xiong J, Wang H, Niu K, Jin B, et al. From Fresnel Diffraction Model to fine-grained Human Respiration Sensing with Commodity Wi-Fi Devices. *Proc ACM Interact Mob Wearable Ubiquitous Technol* (2018) 2(1):1–23. doi:10.1145/3191785
17. Zhuang H, Sun Y, Tang J, Zhang J, Sun X. Influence Maximization in Dynamic Social Networks. In: Proceeding of the 2013 IEEE 13th International Conference on Data Mining; 7-10 Dec. 2013; Dallas, TX, USA. IEEE (2013). p. 1313–8. doi:10.1109/ICDM.2013.145
18. Tong G, Wu W, Tang S, Du DZ. Adaptive Influence Maximization in Dynamic Social Networks. *IEEE/ACM Trans Networking* (2016) 25(1):112–25. doi:10.1109/TNET.2016.2563397
19. Bao Y, Wang X, Wang Z, Wu C, Lau FC. Online Influence Maximization in Non-stationary Social Networks. In: 2016 IEEE/ACM 24th International Symposium on Quality of Service (IWQoS), Changsha, China, June 20–21, 2016. IEEE (2016). p. 1–6. doi:10.1109/iwqos.2016.7590438
20. Wang Y, Zhu J, Ming Q. Incremental Influence Maximization for Dynamic Social Networks. In: International Conference of Pioneering Computer Scientists, Engineers and Educators, Harbin, China, September 22–24, 2017. Springer (2017). p. 13–27. doi:10.1007/978-981-10-6388-6_2
21. Liu X, Liao X, Li S, Zheng S, Lin B, Zhang J, et al. On the Shoulders of Giants: Incremental Influence Maximization in Evolving Social Networks. *Complexity* (2017) 2017 p. 1–14. doi:10.1155/2017/5049836
22. Chen X, Song G, He X, Xie K. On Influential Nodes Tracking in Dynamic Social Networks. In: Proceedings of the 2015 SIAM International Conference on Data Mining, Vancouver, BC, April 30–May 2, 2015. SIAM (2015). p. 613–21. doi:10.1137/1.9781611974010.69
23. Zhou L, Yang Y, Ren X, Wu F, Zhuang Y. Dynamic Network Embedding by Modeling Triadic Closure Process. *Proc AAAI Conf Artif Intelligence* (2018) 32. p. 571–8.
24. Keikha MM, Rahgozar M, Asadpour M, Abdollahi MF. Influence Maximization across Heterogeneous Interconnected Networks Based on Deep Learning. *Expert Syst Appl* (2020) 140:112905. doi:10.1016/j.eswa.2019.112905
25. Christakis NA, Fowler JH. *Connected: The Surprising Power of Our Social Networks and How They Shape Our Lives*. New York, NY: Little, Brown Spark (2009).
26. Pei S, Muchnik L, Andrade JS, Jr, Zheng Z, Makse HA. Searching for Superspreaders of Information in Real-World Social media. *Sci Rep* (2014) 4(1):5547–12. doi:10.1038/srep05547
27. Sheikhahmadi A, Nematbakhsh MA, Shokrollahi A. Improving Detection of Influential Nodes in Complex Networks. *Physica A: Stat Mech its Appl* (2015) 436:833–45. doi:10.1016/j.physa.2015.04.035

28. Li D, Xu Z-M, Chakraborty N, Gupta A, Sycara K, Li S. Polarity Related Influence Maximization in Signed Social Networks. *PloS one* (2014) 9(7): e102199. doi:10.1371/journal.pone.0102199
29. Cornell. arxiv nethept dataset (2003). Available from: http://www.cs.cornell.edu/projects/kddcup/datasets.html (Accessed on: September 19, 2021).
30. Conover MD, Ratkiewicz J, Francisco M, Gonçalves B, Menczer F, Flammini A. Political Polarization on Twitter. In: Proceeding of the Fifth international AAAI conference on weblogs and social media; July 17-21, 2011; Barcelona, Catalonia, Spain (2011).
31. Opsahl T, Panzarasa P. Clustering in Weighted Networks. *Social networks* (2009) 31(2):155–63. doi:10.1016/j.socnet.2009.02.002
32. Leskovec J, Huttenlocher D, Kleinberg J. Signed Networks in Social media. In: Proceedings of the SIGCHI conference on human factors in computing systems, Atlanta, GA, April 10–15, 2010 (2010). p. 1361–70. doi:10.1145/1753326.1753532
33. Shen X, Chung FL. Deep Network Embedding for Graph Representation Learning in Signed Networks. *IEEE Trans Cybern* (2020) 50(4):1556–68. doi:10.1109/TCYB.2018.2871503
34. Qiao Y, Luo X, Li C, Tian H, Ma J. Heterogeneous Graph-Based Joint Representation Learning for Users and Pois in Location-Based Social Network. *Inf Process Manage* (2020) 57(2):102151. doi:10.1016/j.ipm.2019.102151
35. Baytas IM, Xiao C, Wang F, Jain AK, Zhou J. Heterogeneous Hyper-Network Embedding. In: Proceeding of the 2018 IEEE International Conference on Data Mining (ICDM); 17-20 Nov. 2018; Singapore. IEEE (2018). p. 875–80. doi:10.1109/ICDM.2018.00104

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers in Physics

PERSPECTIVE
published: 03 March 2022
doi: 10.3389/fphy.2022.824369

Local Surveillance of the COVID-19 Outbreak

Caifen Liu[1,2†], *Lingfeng Xu*[1,2,3†], *Yuan Bai*[1,2], *Xiaoke Xu*[3], *Eric H. Y. Lau*[1,2], *Benjamin J. Cowling*[1,2] *and Zhanwei Du*[1,2*]

[1]WHO Collaborating Centre for Infectious Disease Epidemiology and Control, School of Public Health, Li Ka Shing Faculty of Medicine, The University of Hong Kong, Hong Kong, Hong Kong SAR, China, [2]Laboratory of Data Discovery for Health, Hong Kong Science and Technology Park, Hong Kong, Hong Kong SAR, China, [3]College of Information and Communication Engineering, Dalian Minzu University, Dalian, China

Given the worldwide pandemic of the novel coronavirus disease 2019 (COVID-19) and its continuing threat brought by the emergence of virus variants, there are great demands for accurate surveillance and monitoring of outbreaks. A valuable metric for assessing the current risk posed by an outbreak is the time-varying reproduction number (R_t). Several methods have been proposed to estimate R_t using different types of data. We developed a new tool that integrated two commonly used approaches into a unified and user-friendly platform for the estimation of time-varying reproduction numbers. This tool allows users to perform simulations and yield real-time tracking of local epidemic of COVID-19 with an R package.

Keywords: epidemics (covid 19), surveillance, infectious disease, package, modeling

OPEN ACCESS

Edited by:
Huijia Li,
Central University of Finance and Economics, China

Reviewed by:
Yongxing Li,
Beijing University of Technology, China
Xueyan Liu,
Jilin University, China

****Correspondence:***
Zhanwei Du
duzhanwei0@gmail.com

†These authors have contributed equally to this work and share first authorship

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *29 November 2021*
Accepted: *13 January 2022*
Published: *03 March 2022*

Citation:
Liu C, Xu L, Bai Y, Xu X, Lau EHY, Cowling BJ and Du Z (2022) Local Surveillance of the COVID-19 Outbreak.
Front. Phys. 10:824369.
doi: 10.3389/fphy.2022.824369

INTRODUCTION

The novel coronavirus disease 2019 (COVID-19) pandemic caused by severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) has led to 257 million confirmed cases and 5.15 million deaths worldwide by November 22, 2021 [1]. The COVID-19 pandemic continues to pose substantial risks to public health, and the situation is worsened by the emergence of SARS-CoV-2 variants with potentially higher transmissibility [2].

Quantification of the transmissibility during epidemics is fundamental for designing and adjusting public health responses. The time-varying reproduction number R_t, defined as the expected number of secondary cases of disease caused by a single infected individual at time t, is a key epidemiological measure of transmissibility, with $R_t < 1$ indicating that incidence is in decline because of either successful control measures or population immunity having reached a sufficiently high level to limit further transmission. The real-time monitoring of R_t provides feedback on the effectiveness of interventions and on the need to intensify control efforts [3, 4].

A large number of methods have been proposed to estimate R_t from surveillance data [5–12]. There are generally two categories. One is based on fitting mechanistic transmission models to incidence data, and the other is a statistical approach requiring case incidence data and the distribution of the serial interval (the time between symptom onsets in a primary case and secondary case) [13]. The mechanistic models are often complicated to deal with because of the potential for biases in the reported incidence data and the context-specific assumptions made. The statistical method proposed by Wallinga and Teunis [13] is relatively simpler but still has drawbacks. Estimates of R_t can vary considerably over a short period when the data aggregation time step is small. To overcome these limits, Cori et al. [14] developed a generic tool for estimating R_t with a ready-to-use R software package *EpiEstim*, which has been frequently used to analyze the recent

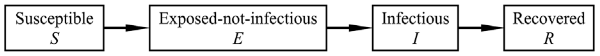


FIGURE 1 | The SEIR structure model used to describe the transmission of infections.

outbreaks of COVID-19. After searching CRAN package data which retrieve package download information from the RStudio mirror, we found that the most popular packages providing estimation of time-varying reproduction number include *EpiEstim*, *EpiNow2*, *R0*, *epidemia*, and *nbTransmission*. All of these tools use statistical methods to estimate R_t from surveillance data and are widely adopted to study COVID-19.

Recently, a new method was proposed by Hay et al. [15] using information inherent in cycle threshold (Ct) values from reverse transcription quantitative polymerase chain reaction (RT-qPCR) tests to estimate the time-varying reproduction number from positive samples. Ct values are semiquantitative results provided by RT-qPCR tests. It is common when testing for infectious diseases to use this quantification of sample viral load. Lower Ct values indicate higher viral loads, and a Ct value below 40 gives a positive result. Based on cross-sectional virologic surveys (observed viral loads), this method overcomes the biases in traditional approaches resulting from testing constraint, unrepresentative sampling, and reporting delays. They also developed the R package *virosolver* to infer epidemic dynamics including estimation of R_t.

In this study, we chose *EpiEstim* and *virosolver* as the representatives of traditional and new methods, respectively. Although the accuracies of the two approaches have been separately demonstrated, there is still a lack of comparison between the two methods to the best of our knowledge. Therefore, we quantify the accuracies of *EpiEstim* and *virosolver* in different transmission scenarios by individual-based simulations and develop a ready-to-use R package for researchers to compare different R_t methods with the synthetic truth.

METHODS

SEIR-based simulation

To assess the performance of the methods, we simulate outbreaks in three scenarios with different basic reproduction numbers (R_0 = 2, 5, 10, respectively) using SEIR-based simulations as the baselines. The three scenarios could represent the situations of wild type, Delta variants, and potential variants of SARS-CoV-2 with higher transmissibility, according to R_0 estimates given by previous studies [16, 17]. The model parameters were determined on the basis of existing literature and epidemiological characteristics of COVID-19 in Hong Kong in early 2020 [14, 15, 18]. In particular, we adopted the prior distributions for the parameters of the SEIR model given by Hay et al. [15]. The SEIR model is a compartmental model which assumes that the growth rate of new infections depends on the current prevalence of infectious and susceptible individuals by modeling the proportion of the population who are susceptible (S), exposed not infectious (E), infectious (I), and recovered (R) with respect to disease over time, as illustrated in **Figure 1**. A stochastic SEIR model is implemented, and the R package *odin* is used to solve the model and obtain true infections over time. The true value of R_t is estimated as $R_t = S_t \times \frac{\beta_t}{\gamma}$, where S_t is the proportion of susceptible population, β_t is the transmission rate at time t derived from the compartmental transition equations, and $1/\gamma$ is the average infectious period.

EpiEstim and *virosolver* methods were run separately on the same simulations for comparison. For *EpiEstim*, it relies on two inputs: incidence time series and the serial interval distribution. Incidence data by days since the start of outbreak were generated from the simulated SEIR epidemic. We used an empirical serial interval distribution informed by a previous outbreak of COVID-19 in Hong Kong in early 2020 [18], and we also used the simulated serial interval distribution for comparison, denoted by *EpiEstim* (empirical SI) and *EpiEstim* (simulated SI) in **Figure 2**. We assumed that the simulated serial interval distribution has the same standard variation as that of the empirical serial interval distribution and inferred the mean of the simulated serial interval distribution by conducting numerical experiments on a range of means from 1 to 10 with a step of 0.1 and chose the one yielding the least root mean square error (RMSE). For *virosolver*, the input data include population-level Ct values over days since the start of outbreak, and individual-level viral kinetics model over days since the infection. The Ct values were generated for all exposed, infectious, and recovered individuals when they were samples based on the Ct value model proposed by Hay et al. [15], and the viral kinetics parameters were also given in their study. We assumed that the Ct values were observed from randomized samples of the population at selected testing days, and **Figure 2** shows the simulated Ct values of the sampled people every 14 days. Each panel presents the distribution of observed Ct values among sampled infected individuals on that testing day. Day 14 and Day 28 had no data because there was no infection among the samples at the early stage of the epidemic.

EpiEstim

The framework of *EpiEstim* is based on statistical assumptions and Bayesian estimation. Transmission is modeled by a Poisson process so that the rate, at which individuals infected between infection and symptom onset generate new infections, is equal to $R_t w_s$, where s is the time postinfection; t is the time post symptom onset; R_t is the time-varying reproduction number at time t; and w_s is a probability distribution describing the average infectiousness profile after infection. The incidence at time t is assumed to be Poisson distributed with mean $R_t \sum_{s=1}^{t} I_{t-s} w_s$, and the likelihood of the incidence I_t given the reproduction number R_t is:

$$P(I_t | I_0, \ldots, I_{t-1}, w, R_t) = \frac{(R_t \Lambda_t)^{I_t} e^{-R_t \Lambda_t}}{I_t!}$$

where $\Lambda_t = \sum_{s=1}^{t} I_{t-s} w_s$. R_t is estimated in a time window τ, under the assumption that the time-varying reproduction number is

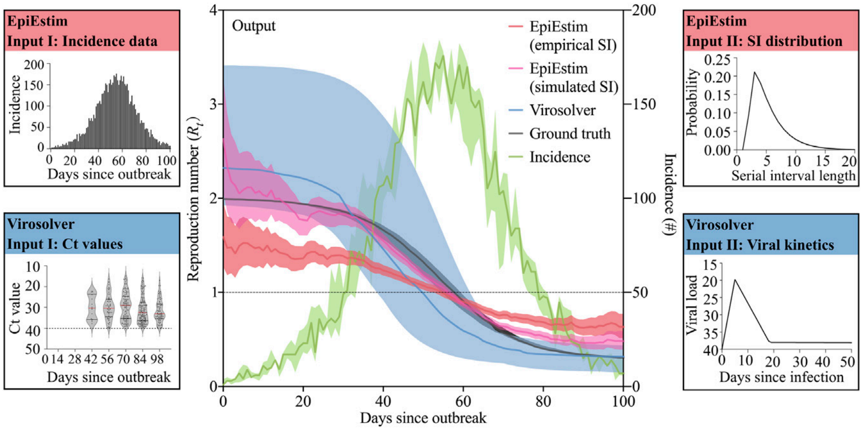


FIGURE 2 | A schematic illustrating how our simulation platform generates a comparison of the estimated R_t from *EpiEstim* and *virosolver*. Incidence data and ground truth were generated from 100 simulations based on the SEIR model (green/gray line and shaded ribbon show mean and the range). Estimates of R_t were obtained using *EpiEstim* (red line and shaded ribbon show posterior median and 95% CrI using mean incidence data) and *virosolver* (blue line and shaded ribbon show posterior mean and 95% CrI using Ct value model), respectively. *EpiEstim* using the empirical value of serial interval distribution [18] and the simulated serial interval distribution are denoted by *EpiEstim* (empirical SI) and *EpiEstim* (simulated SI), respectively.

constant within that time window. Therefore, over time period $[t-\tau+1, t]$, the likelihood of the incidence during this time period $I_{t-\tau+1}, \ldots, I_t$ given the reproduction number $R_{t,\tau}$, conditional on the previous incidences, is as follows:

$$P\left(I_{t-\tau+1}, \ldots, I_t \middle| I_0, \ldots, I_{t-\tau}, w, R_{t,\tau}\right) = \prod_{s=t-\tau+1}^{t} \frac{\left(R_{t,\tau}\Lambda_s\right)^{I_s} e^{-R_{t,\tau}\Lambda_s}}{I_s!}$$

Using a Bayesian framework with a Gamma distributed prior with parameters of shape $\boldsymbol{a}$ and scale $\boldsymbol{b}$ for $R_{t,\tau}$, the posterior distribution of $R_{t,\tau}$ is assumed to be a Gamma distribution with parameters $(a + \sum_{s=t-\tau+1}^{t} I_s, \frac{1}{\frac{1}{b}+\sum_{s=t-\tau+1}^{t}\Lambda_s})$. Hence, inference of $R_{t,\tau}$ is straightforward from the posterior distribution. Note that the choice of the time window size τ has an impact on the estimates of R_t: small values of τ lead to a more rapid detection of changes in transmission but also more statistical noise; large values lead to more smoothing and reductions in statistical noise. By conducting simulation experiments on $\tau = 7$, 14, 21, respectively, we found that $\tau = 14$ exhibited the best compromise between high accuracy and easy interpretation, so the window size was set to be 14 in this study. Readers can refer to Gostic et al. [19] for a detailed discussion on the sliding window of the *EpiEstim* method.

Virosovler

The R package *virosolver* was developed by Hay et al. [15] using virological data and Ct values, to infer epidemic dynamics. Ct values are inversely correlated with $\log_{10}$ viral loads, which depend on the time since infection. The distribution of Ct values across positive specimens at a single time point reflects the epidemic trajectory: a growing epidemic will have a high proportion of recently infected individuals with high viral loads, whereas a declining epidemic will have more individuals with older infections and thus lower viral loads. Using a mathematical model for population-level viral load distributions calibrated to known features of the SARS-CoV-2 viral load kinetics, we can use Ct values from a single random cross section of virologic testing to estimate the time-varying reproduction number in a population. For individual i sampled on day u, the Ct value X_i is assumed to follow the Gumbel distribution as

$$X_i \sim Gumbel\left[C_{mode}\left(u - t_{inf}\right), \sigma\left(u - t_{inf}\right)\right],$$

where t_{inf} is the time of infection, and $C_{mode}\left(u - t_{inf}\right)$ and $\sigma\left(u - t_{inf}\right)$ are the location and scale parameters, respectively. The details of the parameterization are found in [15]. In practice, *virosolver* takes an input data frame of Ct values with associated sample collection dates from RT-qPCR testing and reconstructs the incidence curve that gave rise to those measurements. By capturing this logic in a mathematical model, we can obtain a probabilistic estimate of the underlying incidence curve, thus time-varying reproduction number having observed a set of Ct values at some point in time. Noting that the sampling scheme has an impact on the estimate of incidence, we set the population number to be 8,000 and sampled 1,000 (1/8) of the population to fit the local prevalence data of COVID-19 in Hong Kong in early 2020 as a case study [20].

RESULTS

We assessed the performance of *EpiEstim* and *virosovler* in three scenarios where $R_0 = 2, 5, 10$, respectively, in which $R_0 = 2$ can serve as a demonstration of the outbreak of COVID-19 in Hong Kong in early 2020. For each scenario, we generated the incidence data over 100 days based on the SEIR model from 100 stochastic simulations and estimated the mean incidence. **Figure 3** gives the estimated R_t with the uncertainties (95% credible intervals) across 100 simulations using *EpiEstim* and *virosolver*,

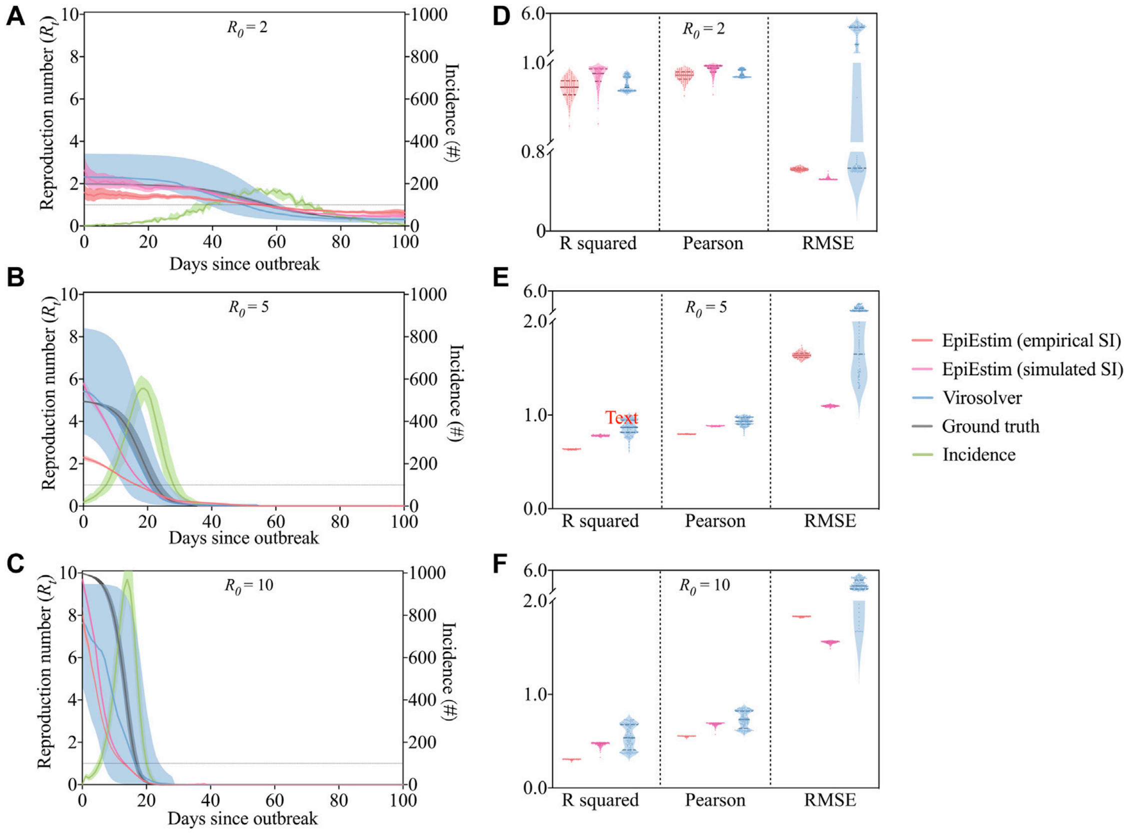


FIGURE 3 | The output of R_t estimates in three designed scenarios and the corresponding outcomes of accuracy assessment. **(A–C)** The graphical interface by setting R_0 = 2, 5, 10, respectively. We parameterized the serial interval distribution used by *EpiEstim* with the empirical study [18] and the simulated serial interval distribution, which are denoted by *EpiEstim* (empirical SI) and *EpiEstim* (simulated SI) in figure legends. **(D–F)** Results of R squared, Pearson correlation coefficient, and RMSE for both methods in scenarios with R_0 = 2, 5, 10, respectively.

respectively, and the ground truths for the R_t values are presented for comparison. *EpiEstim* with the empirical serial interval distribution [18] would underestimate R_t. In contrast with *EpiEstim*, *virosovler* provided less biased estimates but exhibited wider intervals of uncertainty. However, both approaches performed well in detecting the timing point when $R_t < 1$.

To quantify and compare the accuracies of the methods, we used multiple metrics including coefficient of determination (R^2), Pearson correlation coefficient, and root mean square error (RMSE). As **Figures 3D–F** show, in all scenarios, *virosolver* almost had the highest R^2 and Pearson correlation coefficient with the ground truth of R_t, suggesting that *virosolver* had the highest accuracy and strongest correlation with the synthetic epidemic growth. In terms of RMSE, the performance of *EpiEstim* with the simulated serial interval distribution was the best (lowest RMSE), and *virosolver* had the largest RMSE due to its large estimation uncertainties. We noted that *EpiEstim* with simulated SI always performed better than *EpiEstim* with empirical SI. In conclusion, *virosovler* provided more accurate estimates of R_t, and *EpiEstim* relied on the adjustment of serial interval distribution for better performance.

DISCUSSION

Quantifying disease transmissibility during outbreaks is crucial for designing effective control measures and assessing their effectiveness once implemented. In the situation where the incidence is still increasing while the time-varying reproduction number is actually dropping, there might be a very different outlook compared to if the incidence and the reproduction number are both increasing. The platform for estimating R_t provided here can therefore help epidemiologists and policymakers to monitor temporal changes in the transmissibility of COVID-19. The key contributions of our platform are as follows: 1) our software package integrates the most popular method (*EpiEstim*) and the newest approach (*virosovler*) into a unified framework, allowing users to infer real-time viral transmissibility from different perspectives; 2) by setting the value of R_0, users can conduct simulation experiments on our platform to study the epidemic development and compare the performances of two approaches accordingly; 3) this platform is easy enough for nonspecialists to apply by simply inputting the required data and is also flexible for specialists to use by changing the parameters setting if needed.

The estimation tools we used here have several limitations and thus may result in potential bias. For *EpiEstim*, a preexisting estimate of serial interval distribution is required as the input data, which may account for the underestimation of reproduction number in our simulation study (**Figure 3**). If data on pairs of infector-infected individuals are available, the serial interval distribution can be estimated jointly, which leads to more precise estimates of transmissibility [21]. In addition, the inevitable delay between infection and case reporting (the incubation period) could also result in biased estimation of R_t. If data on the incubation period are available, a possible strategy would be to use the incubation period distribution to back-calculate the incidence of infections from the incidence of symptoms and then apply *EpiEstim* to estimate the reproduction number from those inferred data.

The virologic data-based method, *virosolver*, as mentioned above, exhibits greater uncertainty of estimated than *EpiEstim*. This is probably caused by insufficient information on Ct value distribution and viral kinetics model. The viral load kinetics model used in *virosolver* was generated on the basis of observed properties of measured viral loads in the literature, and these results were applied to inform priors on key parameters when estimating reproduction numbers. The estimates can therefore be improved by choosing more precise, accurate priors relevant to the observations used during model fitting. For example, the model should be adjusted by specifying different distributions if results come from multiple testing platforms. Results may also be improved if individual-level features such as symptom status, age, antiviral treatment, and vaccination record are available and incorporated into the Ct value model.

Apart from the two methods presented in our study, many other approaches are still available, which we will include in this platform in the future to track disease transmissibility by using other data sources (e.g., hospitalization and death). Additionally, genomic data are also of great importance in the inference of transmissibility of COVID-19 considering recent emergence of virus variants [22]. We only provide incidence and R_t estimates as the outputs; other epidemiological metrics such as prevalence, hospitalization, admission to ICU, death, and the economic analysis, such as net monetary benefit (NMB), are not included in our platform. Besides, we used the SEIR model for simulation in our package, because Hay et al. [15] had studied four other epidemic models for fitting cross-sectional viral load data, namely, the SEIR model, exponential growth model, SEEIRR model, and Gaussian process model, and they also made a comparison of these models. They found that the SEIR model was the most appropriate as it consistently provided unbiased, constrained estimates of transmissibility during the epidemic growth. We may explore these models and other individual-based models (branching process, for example) in future studies.

Our tool can also be applied to the new variants of SARS-CoV-2 as long as incidence data and Ct values of the infected people are available. Users can obtain a more accurate estimation by adopting the updated parameters of serial interval distribution and viral kinetics model for objective variants informed by recent studies [23–26]. In conclusion, we have established a platform for simulation and inference of time-varying reproduction numbers by incorporating two commonly used approaches. We would ensure our tool to epidemiologists and public health organizations in a wide range of future outbreak response scenarios.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

CL, LX, ZD, and BC: conceived the study, designed statistical and modeling methods, conducted analyses, interpreted results, and wrote and revised the manuscript; YB, XX, and EL: interpreted the results and revised the manuscript.

ACKNOWLEDGMENTS

We acknowledge the financial support from the Collaborative Research Fund (Project No. C7123-20G) of the Research Grants Council of the Hong Kong SAR Government.

REFERENCES

1. Home - Johns Hopkins Coronavirus Resource center. Available from: https://coronavirus.jhu.edu/. Accessed 22 November 2021.
2. Sars- CDC. CoV-2 Variant Classifications and Definitions (2021). Available from: https://www.cdc.gov/coronavirus/2019-ncov/variants/variant-info.html?CDC_AA_refVal=https%3A%2F%2Fwww.cdc.gov%2Fcoronavirus%2F2019-ncov%2Fcases-updates%2Fvariant-surveillance%2Fvariant-info.html (Accessed November 3, 2021).
3. Anderson RM, May RM. *Infectious Diseases of Humans: Dynamics and Control*. Oxford: OUP Oxford (1992).
4. Ferguson NM, Cummings DAT, Fraser C, Cajka JC, Cooley PC, Burke DS. Strategies for Mitigating an Influenza Pandemic. *Nature* (2006) 442: 448–52. doi:10.1038/nature04795
5. Riley S, Fraser C, Donnelly CA, Ghani AC, Abu-Raddad LJ, Hedley AJ, et al. Transmission Dynamics of the Etiological Agent of SARS in Hong Kong: Impact of Public Health Interventions. *Science* (2003) 300:1961–6. doi:10.1126/science.1086478
6. Fraser C, Donnelly CA, Cauchemez S, Hanage WP, Van Kerkhove MD, Hollingsworth TD, et al. Pandemic Potential of a Strain of Influenza A (H1N1): Early Findings. *Science* (2009) 324:1557–61. doi:10.1126/science.1176062
7. Ferguson NM, Donnelly CA, Anderson RM. Transmission Intensity and Impact of Control Policies on the Foot and Mouth Epidemic in Great Britain. *Nature* (2001) 413:542–8. doi:10.1038/35097116
8. Amundsen EJ, Stigum H, Røttingen J-A, Aalen OO. Definition and Estimation of an Actual Reproduction Number Describing Past Infectious Disease Transmission: Application to HIV Epidemics Among Homosexual Men in Denmark, Norway and Sweden. *Epidemiol Infect* (2004) 132:1139–49. doi:10.1017/s0950268804002997

9. Bettencourt LMA, Ribeiro RM. Real Time Bayesian Estimation of the Epidemic Potential of Emerging Infectious Diseases. *PLoS One* (2008) 3: e2185. doi:10.1371/journal.pone.0002185
10. Cintrón-Arias A, Castillo-Chávez C, Bettencourt LM, Lloyd AL, Banks HT. The Estimation of the Effective Reproductive Number from Disease Outbreak Data. *Math Biosci Eng* (2009) 6:261–82. doi:10.3934/mbe.2009.6.261
11. Howard SC, Donnelly CA. Estimation of a Time-Varying Force of Infection and Basic Reproduction Number with Application to an Outbreak of Classical Swine Fever. *J Epidemiol Biostat* (2000) 5:161–8.
12. Kelly HA, Mercer GN, Fielding JE, Dowse GK, Glass K, Carcione D, et al. Pandemic (H1N1) 2009 Influenza Community Transmission Was Established in One Australian State when the Virus Was First Identified in North America. *PLoS One* (2010) 5:e11341. doi:10.1371/journal.pone.0011341
13. Wallinga J, Teunis P. Different Epidemic Curves for Severe Acute Respiratory Syndrome Reveal Similar Impacts of Control Measures. *Am J Epidemiol* (2004) 160:509–16. doi:10.1093/aje/kwh255
14. Cori A, Ferguson NM, Fraser C, Cauchemez S. A New Framework and Software to Estimate Time-Varying Reproduction Numbers during Epidemics. *Am J Epidemiol* (2013) 178:1505–12. doi:10.1093/aje/kwt133
15. Hay JA, Kennedy-Shaffer L, Kanjilal S, Lennon NJ, Gabriel SB, Lipsitch M, et al. Estimating Epidemiologic Dynamics from Cross-Sectional Viral Load Distributions. *Science* (2021) 373:373. doi:10.1126/science.abh0635
16. Alimohamadi Y, Taghdir M, Sepandi M. Estimate of the Basic Reproduction Number for COVID-19: A Systematic Review and Meta-Analysis. *J Prev Med Public Health* (2020) 53:151–7. doi:10.3961/jpmph.20.076
17. Liu Y, Rocklöv J. The Reproductive Number of the Delta Variant of SARS-CoV-2 Is Far Higher Compared to the Ancestral SARS-CoV-2 Virus. *J Trav Med* (2021) 28. doi:10.1093/jtm/taab124
18. Zhao S, Gao D, Zhuang Z, Chong MKC, Cai Y, Ran J, et al. Estimating the Serial Interval of the Novel Coronavirus Disease (COVID-19): A Statistical Analysis Using the Public Data in Hong Kong from January 16 to February 15, 2020. *Front Phys* (2020) 8:347. doi:10.3389/fphy.2020.00347
19. Gostic KM, McGough L, Baskerville EB, Abbott S, Joshi K, Tedijanto C, et al. Practical Considerations for Measuring the Effective Reproductive Number, Rt. *Plos Comput Biol* (2020) 16:e1008409. doi:10.1371/journal.pcbi.1008409
20. Real-time Dashboard. Available from: https://covid19.sph.hku.hk/. Accessed 10 November 2021.
21. Thompson RN, Stockwin JE, van Gaalen RD, Polonsky JA, Kamvar ZN, Demarsh PA, et al. Improved Inference of Time-Varying Reproduction Numbers during Infectious Disease Outbreaks. *Epidemics* (2019) 29:100356. doi:10.1016/j.epidem.2019.100356
22. Leung K, Shum MH, Leung GM, Lam TT, Wu JT. Early Transmissibility Assessment of the N501Y Mutant Strains of SARS-CoV-2 in the United Kingdom, October to November 2020. *Eurosurveillance* (2021) 26: 2002106. doi:10.2807/1560-7917.es.2020.26.1.2002106
23. Ryu S, Kim D, Lim J-S, Ali ST, Cowling BJ. Serial Interval and Transmission Dynamics during SARS-CoV-2 Delta Variant Predominance, South Korea. *Emerg Infect Dis* (2022) 28:407–10. doi:10.3201/eid2802.211774
24. Pung R, Mak TM, Kucharski AJ, Lee VJCMMID COVID-19 working group. Serial Intervals in SARS-CoV-2 B.1.617.2 Variant Cases. *The Lancet* (2021) 398:837–8. doi:10.1016/s0140-6736(21)01697-4
25. Singanayagam A, Hakki S, Dunning J, Community Transmission and Viral Load Kinetics of the SARS-CoV-2 delta (B. 1.617. 2) Variant in Vaccinated and Unvaccinated Individuals in the UK: a Prospective, Longitudinal, Cohort Study. *Lancet Infect Dis* (2021) 22(2):183–195. doi:10.1016/s1473-3099(21)00648-4
26. Chia PY, Ong SWX, Chiew CJ, Virological and Serological Kinetics of SARS-CoV-2 Delta Variant Vaccine Breakthrough Infections: a Multicentre Cohort Study. *Clin Microbiol Infect* (2021). doi:10.1016/j.cmi.2021.11.010

Conflict of Interest: BC reports honoraria from AstraZeneca, GSK, Moderna, Pfizer, Roche, and Sanofi Pasteur.

CL, LX, YB, EL, BC, and ZD were employed by the company Laboratory of Data Discovery for Health, Hong Kong Science and Technology Park.

The remaining author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers
in Physics

ORIGINAL RESEARCH
published: 07 March 2022
doi: 10.3389/fphy.2022.807099

Construction of the Social Network Information Dissemination Index System Based on CNNs

Weihong Han[1†], *Linhe Xiao*[1†], *Xiaobo Wu*[2], *Daihai He*[3], *Zhen Wang*[4] *and Shudong Li*[1*]

[1]Cyber Space Institute of Advanced Technology, Guangzhou University, Guangzhou, China, [2]School of Computer Science and Cyber Engineering, Guangzhou University, Guangzhou, China, [3]Department of Applied Mathematics, Hong Kong Polytechnic University, Hong Kong, China, [4]School of Artificial Intelligence, OPtics and Electronics (iOPEN), Northwestern Polytechnical University, Xi'an, China

The information dissemination index system is an effective way to measure the dissemination of public opinion events in social networks. Due to the complexity, variability, and asymmetry of information, the construction of traditional information dissemination index systems demands excessive reliance on manual intervention, has large deviations, and is applied in a limited range. Such shortcomings cannot meet the requirements of constructing an objective, comprehensive, and highly credible index system. Therefore, we propose a method of constructing a multilevel and multigranular information dissemination index system with complex perspectives. In addition, we use the deep learning method of the convolutional neural network to extract the rich convolution features of public opinion events in the information dissemination process. Then, we train the weight, and it forms the corresponding weight of the information dissemination index systems. The experimental results prove that the method we use is superior to other methods and has better performance on the data set of a specific field.

Keywords: convolutional neural network, deep learning, index system, social network, information dissemination

OPEN ACCESS

Edited by:
Zhanwei Du,
University of Texas at Austin,
United States

Reviewed by:
Lixiang Li,
Beijing University of Posts and
Telecommunications (BUPT), China
Yan Li,
BP America, United States

****Correspondence:***
Shudong Li
lishudong@gzhu.edu.cn

[†]These authors have contributed equally to this work and share first authorship

Specialty section:
This article was submitted to
Social Physics,
a section of the journal
Frontiers in Physics

Received: *01 November 2021*
Accepted: *28 January 2022*
Published: *07 March 2022*

Citation:
Han W, Xiao L, Wu X, He D, Wang Z and Li S (2022) Construction of the Social Network Information Dissemination Index System Based on CNNs.
Front. Phys. 10:807099.
doi: 10.3389/fphy.2022.807099

1 INTRODUCTION

With the increasing convenience of Internet social networks, the cost and time of information flow are rapidly decreasing. How to accurately and quantitatively evaluate the dissemination of online public opinion information from the complex and massive amount of online public opinion information is a problem that needs to be solved urgently [1]. But nowadays, the understanding of the evolution mechanism of online public opinion is not deep enough. Many communication laws still need to be explored and summarized. The communication of online information involves a wide range of factors. The role of the constituent elements is interactive promotion and restriction. The main body and role of each stage factors are changing, making it difficult for managers to systematically assess the situation of information dissemination in an all-round way [2]. Therefore, establishing a reasonable and easy-to-use information dissemination evaluation index system, quantifying the information dissemination per unit time, and evaluating and calculating the situation to help managers make accurate decisions have become a current key research topic.

2 RELATED WORK

The research on information dissemination situation assessment mainly includes two aspects: qualitative public opinion situation assessment and quantitative public opinion situation assessment.

2.1 Qualitative Evaluation

In terms of qualitative evaluation, the current research mainly focuses on the evaluation of online public opinion, scope of the event, clustering effect, and tendency of netizens to the event.

Warshaw developed a multilevel regression and post-hierarchical model combining survey and population data so as to be able to detect specific online public opinion on lower-level geographic aggregation and evaluate the spread of public opinion [3]. Gil-Garcia developed two specific dynamic clustering algorithms from the framework proposed by previous scholars for clustering public opinion texts, which can automatically correct the clustering without re-clustering when the massive document data change [4]. Matsumura proposed an impact diffusion model to discover the network opinion leaders on the designated topics on the forum and the spread of network events [5]. Some studies proposed network topology and infectious disease strategies to explore the characteristics of information dissemination in social networks [6,7]. Cha made a detailed comparison of the three indicators for measuring user influence based on Twitter's massive data. Then, based on these three indicators, he also explored the dynamic performance of influence in terms of time and themes and gave a report based on the evaluation method of spreading public opinion of users [8]. Bermingham grabbed large-scale data on possible radical topics from YouTube and used social network analysis and sentiment analysis tools to dig out topics and their emotional tendencies and portray the characteristics of radicals. An analysis method of user emotional orientation in public opinion events [9].

In recent years, topic discovery technology and communication range evaluation technology for new Internet media, especially Twitter short texts, have also received extensive attention from scholars [10]. Li proposed a detection algorithm based on sparse self-encoding to study the mutual influence between nodes in social networks and analyze the information dissemination situation from the perspective of network topology [11]. Han proposed a topic representation model based on user behavior analysis and evaluated the importance of vocabulary in the process of information dissemination [12]. Zhang proposed a method based on vocabulary decomposition to rate the sentiment of Twitter short texts and used machine learning methods to automatically classify the sentiment of new texts to support the qualitative evaluation of user sentiments and responses of specific online public opinion events [13].

2.2 Quantitative Evaluation

In terms of quantitative evaluation of online public opinion situation, the research on online public opinion decision-making can be summarized from three aspects. The first is to design an index system for online public opinion situation decision-making. Based on the analysis of the spatial structure and evolution of online public opinion, corresponding indicators can be designed to represent the situation of online public opinion from different aspects. Designing a systematic, comprehensive, and scientific indicator system is the primary task of making decisions on the online public opinion situation. Identifying key and representative indicators can not only reduce the complexity of the indicator system but also improve the accuracy of decision-making [14].

Zeng took public opinion risk as the core research object, starting from the source of risk and the internal and external performance of risk, constructed an early warning indicator system for network public opinion emergencies, and introduced the expert evaluation and AHP methods to determine the index weight [15]. Dai designed a three-tier security state decision-making index system from four dimensions: event sensitivity, communication, netizen attitude, and attention to public opinion and explained the statistical methods of each quantitative indicator [16]. Zhu proposed a mechanism to study the role of personal reputation and strategy in social network interactions [17] and also studied the information dissemination phenomenon of social networks in complex networks [18]. Shang regards the public opinion index system as an investment method, and under the guidance of portfolio theory integrates three sets of network public opinion index systems with their own characteristics into a dynamic integrated index system and uses radar chart analysis tools to evaluate the network public opinion. Comprehensive scores can also be used to analyze deep-seated reasons through online public opinion evaluation [19].

Zhang abstracted the key indicator system and logical relationship of the situation evaluation of network public opinion into a Bayesian network, then added some subjective evaluations, and gave a public opinion situation evaluation method based on the Bayesian network model. On the basis of constructing a public opinion index system with three dimensions, [20] Rong proposed a gray prediction method for network public opinion based on the variable-length mechanism of a data sequence. The prediction effect can be iteratively tested to find the optimal data sequence length to achieve the maximum good forecasting effect [21]. He chose the causal loop diagram to represent the mutual influence between the network public opinion indicators and constructed a network public opinion system dynamics model with 38 variables and four core subsystems targeting public opinion popularity [22]. Zhang used the principal component analysis method to refine the primary index system into a few comprehensive indicators without correlation and established an online public opinion early warning model based on the SVM machine learning method [23]. Nie measured and judged the influence of different users on the information dissemination situation in social networks from the perspective of user identity [24]. Huang used the PLSA model to extract the feature word space with no sentimentality, constructed the sentiment word space through word segmentation and TF-IDF function, and then integrated the inclination of all sentiment words based on the HowNet similarity algorithm to support the judgment of the sentiment trend of the network public opinion [25].

Most of the abovementioned scholars use subjective methods to construct indicator system models and use analytic hierarchy process, subjective assignment way, and other weight distribution methods to totally evaluate the situation of social network information dissemination, which have been verified by certain practical applications, but their focus is more limited. For more

TABLE 1 | Dataset.

Dataset	#Hours	#Blogs	#Events
Dataset	6,354	1785034	25

complex social network communication status, as well as the situational assessment of different types of events, the portability is inadequate. In the abovementioned former information dissemination index system construction and evaluation calculation method, the weight distribution of each fundamental index depends on manual operation, and the value is relatively vague, and sometimes the deviation is large. Considering the real-time nature of social network information dissemination data, it is required to construct a real-time, self-adaptive nonlinear system. The system should include a dynamic index weight learning mechanism and iterate with the constantly updated social network information dissemination situation data to form an absolute index system that can effectually evaluate the information dissemination situation.

TABLE 2 | Information dissemination index system.

Primary index	Secondary index	Tertiary index
Public opinion events	Characteristics of the evolution stage	Propagation time Evolution stage Propagation rate
	Characteristics of post content	Forwarding ratio Include picture ratio Include audio ratio Include video ratio Include topic ratio @ other number
	Characteristics of information dimension	Blog title length Blog character length Blog vectorization feature Blog average TF-IDF
	Characteristics of network structure	Network density Convergence factor Connection strength
Public opinion audience	Characteristics of emotion	Positive affective word frequency Neutral affective word frequency Reverse affective word frequency Emotional intensity rating Positive emotion ratio Neutral emotion ratio Reverse emotion ratio
	Characteristics of user identity	Sex ratio Identity authentication ratio Account registration duration Age distribution of users Education level of users Political distribution User name ratio User head portrait ratio
Media aspect	Characteristics of media	Number of media Total news Proportion of media news reports News likes News forwarding number Media concerning number
	Characteristics of propagation	Total blogs Forwarding number Number of comments Number of likes Number of participating platforms Number of users covered Proportion of original posts
	Characteristics of distribution	Regional coverage Regional aggregation Domain name distribution Proportion of foreign blogs Proportion of domestic blogs

3 CONVOLUTIONAL NEURAL NETWORK

The convolutional neural network (CNN) is currently one of the key research directions in the field of computer vision, based on deep learning. It performs well in applications such as image classification and segmentation, and its powerful feature learning and feature expression capabilities are increasingly being valued by researchers. The convolution operation is a multilayer feedforward neural network model. The network structure is shown in **Figure 1**. Each layer uses a set of convolution kernels separately, which helps extract useful features from locally related data points. In the training process, the CNN learns through the backpropagation algorithm. The objective function optimized by this backpropagation algorithm uses a response-based human-like brain learning mechanism. The CNN imitates the biological neural network, adopting the core weight sharing network structure so that it can adjust the network model magnitude by adjusting the depth and width of the neural network.

3.1 Convolutional Layer

The convolutional neural network model has powerful assumptions about physical images, that is, statistical smoothness and local connection. It can validly reduce the learning complexity of the deep neural network model, making the network connection and weight parameters less, which makes it more than the same scale. The fully connected network is easier to train. It uses the convolution kernel to slide on the image and finally completes the process of calculating the gray value of all image pixels after a series of matrix operations.

3.2 Pooling Layer

The pooling layers can lower the dimensionality of the data by imitating the human visual system and using higher-level features to represent the image. The pooling layer can very effectively reduce the size of the matrix, that is, it can perform set statistical operations on the features of different positions in the local area of the image, thereby alleviating excessive sensitivity of the convolutional layer to image position and reducing the final fully connected layer. Parameters to speed up the calculation speed. The most commonly used pooling methods in practice are Max pooling and Average pooling. In addition to reducing model calculations and reducing information redundancy, they also improve scale invariance and rotation invariance of the model to varying degrees, effectively preventing overfitting. The improvement of various pooling methods also helps better realize feature compression and feature extraction, which greatly reduces the time required for model training.

3.3 Fully Connected Layer

A fully connected layer is consisted of several hidden layers in the CNN and usually appears in the last few layers. Each layer contains multiple neurons, and each neuron is fully connected to the neuron of the next layer, which is used to compare the characteristic structure and the structure designed in the previous section. Through the calculation of the layer and the layer, the feature obtained by the feature map is used as the input of the fully connected layer. The essence of the fully connected layer space is to linearly transform from one feature to another feature space. In addition, at the end of the CNN, we use different classification functions to calculate the results.

4 MODEL DESIGN AND EXPERIMENT

4.1 Dataset and Environment

We grab a total of 15 from different social platforms including: Weibo, Twitter, and WeChat official accounts; some news websites: Yahoo, Sina, Tencent, and World Wide Web; and various online information dissemination platforms such as forums and blogs. The relevant data of the event and each event is divided on an hourly basis according to the time window, forming a total of 6,354h of data as shown in **Table 1**. We randomly use 10 percent of them as the test set and the remaining as the training set. Each set of data is scored by 10 experts and 100 ordinary users of social networks in accordance with the evaluation criteria, which are used as tags for the data set.

We performed this research in the following environment: CentOs 7.5, Intel(R) Xeon(R) Silver 4210, and Intel(R) Core(TM) i7-8750H CPU.

4.2 Construction of the Index System

The work of this module focuses on the diversified characteristics of the factors involved in the information dissemination situation, and each factor has a different impact on the information dissemination situation at different levels and different granularities. In-depth analysis and mining of influencing information from various perspectives, such as public opinion events, communication media, and public opinion audiences. The characteristics of the attributes of various factors in the communication situation and the law of their influence on the information dissemination situation have constructed a multilevel, multigranular, and multidimensional information dissemination situation indicator system. The three-tier indicators of the information dissemination trend index system are determined through research and use of the Delphi method, and the principal component analysis method is used to determine the main factors affecting the information dissemination trend by analyzing the correlation between different indicators, and finally, the information dissemination trend evaluation index is established, as shown in **Table 2**.

Among them, the first-level index public opinion event is analyzed from the perspective of the public opinion event, including the characteristics of evolution, post content, information dimension, and network structure. It is a class of indicators that describe the state of the public opinion event in the process of dissemination. It is mainly used to judge the communication stage of public opinion so as to analyze the communication trend. In the second-level indicators of the characteristics of the evolution stage of propaganda events, there are three three-level indicators of propagation time span, evolution stage, and propagation rate. In the second-level indicators of post content characteristics, there are three-level indicators such as the proportion of forwarding; the proportion of pictures, audio, video, and topics; and the number of others mentioned. The secondary indicators of the information dimension feature include the length of the post title, the length of the blog post, and the TF-IDF of the blog post, and the

TABLE 3 | Setting the initial model parameters.

Paramenters	Setting
Convolution kernel size	2*2
Number of convolution kernels	64*1
Convolution stride	1
Pooling method	Max
Pooling window size	2*2
Pooling stride	2
Padding	SAME
Activation function	Relu
Dropout	0.25
Loss function	Cross-entropy loss
Fully connected layer neurons	128

dissemination situation of the post is measured from the character level. Audience tendency analysis is an indispensable part of public opinion analysis, and tendency analysis also reflects the size, structure, and psychological condition of the audience from another perspective, and is an important component of public opinion dissemination. For the secondary indicators of audience sentiment tendency, it includes the word frequency of positive, neutral, and negative sentiment words as well as the average sentiment intensity of each event and the proportion of positive, neutral, and negative sentiment posts. The spread of posts is measured from the sentiment analysis level. The secondary indicators of user identity characteristics include user information such as the user's gender ratio, age distribution, education level distribution, and political affiliation distribution, as well as account registration time, whether it is identity authentication, and whether the account has a user name and avatar, and other account information.

Public opinion audience is analyzed from the perspective of users participating in a certain public opinion event, including characteristics such as emotional tendency and identity. Audience tendency analysis is an indispensable part of public opinion analysis. Tendency analysis also reflects the size, structure, and psychological status of the audience from another angle and is an important component of public opinion dissemination. Secondary indicators of media participation include the proportion of news reports in posts involving publicity events, number of news media reported, total number of news reports, total number of likes on the news, total number of retweets, and total number of followers of the news media. Leaders play an important intermediary or filtering role in the formation of mass communication effects, and they spread information to audiences to form two-level dissemination of information. An important role in two-level communication is played by the person in the crowd who is first or more exposed to mass media information and disseminates the information that has been reprocessed by himself to others. With the ability to influence the attitudes of others, they intervene in mass communication, speeding it up and expanding its influence. The secondary indicators of dissemination heat include information such as the number of posts, retweets, comments, likes, participating platforms, and users of the propaganda event. These indicators directly reflect the dissemination situation of the propaganda event in social networks.

The media aspect is analyzed from the perspective of the media of public opinion events, including indicators such as media participation, communication popularity, and regional distribution. It is an important standard for measuring the spread of public opinion events. Secondary indicators of media participation include the proportion of news reports in posts involving publicity events, number of news media reports, total number of news reports, total number of likes, total number of news retweets, and total number of news media followers. The secondary indicators of dissemination heat include information such as the number of posts, retweets, comments, likes, participating platforms. and users of the propaganda event. These indicators directly reflect the dissemination situation of the propaganda event in social networks.

4.3 Construction of the Model Structure

4.3.1 Input

We calculated 25 three-level indicators separately and use the following method to standardize the original statistical data:

$$Z_{ij} = \left(X_{ij} - \overline{X_j}\right) \Big/ \sigma_j, \sigma_j = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\left(X_{ij} - \overline{X_j}\right)}. \quad (1)$$

This method first calculates the average value and standard deviation of the original data and then standardizes the data. The data processed in this way conform to the standard normal distribution and conduct the neural network input data. We combine the indicators of each event into a 5*5 two-dimensional matrix as the input of the CNN. The neural network training process is as follows.

4.3.2 Convolution

We used a two-dimensional convolution operation to filter and extract features. The input vector is combined into a 5*5 two-dimensional single-channel grayscale image, and the calculation of the feature on the convolutional layer is as follows:

$$c_\sigma^a(x) = f\left(\sum W^{(a,\sigma)} p_i^{(a-1)} + b^{(a,\sigma)}\right), \sigma = 1, 2, \ldots, n, \quad (2)$$

$c_\sigma^a(x)$ represents the output of the latter layer of features,W represents the weight of the kernel, p represents the feature map of the layer, and f is activation function Relu.

4.3.3 Pooling

In addition to reducing model calculations and reducing information redundancy, pooling operation improves the scale invariance and rotation invariance of the model to varying degrees, effectively preventing overfitting. The improvement of various pooling methods also better realizes feature compression and feature extraction, which greatly reduces the time required for model training. The calculation of features is as follows:

$$s_\sigma^a(x) = maxpooling\left(c_\sigma^{a-1}\right) + b^{(m,\sigma)}. \quad (3)$$

4.3.4 Output

The convolutional neural network filters the matrix through the aforementioned convolutional layer and pooling layer for feature screening, and then the obtained vector constructs a fully connected layer. After two fully connected layers, finally, the output is analyzed through the Softmax function to obtain the

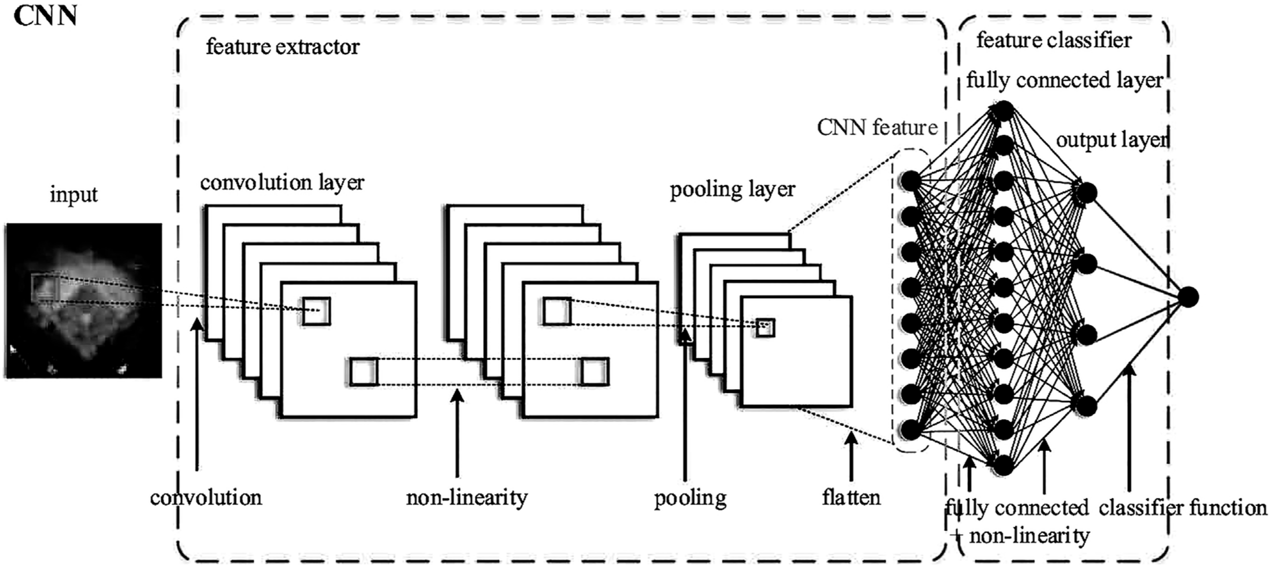


FIGURE 1 | Structure of the convolutional neural network.

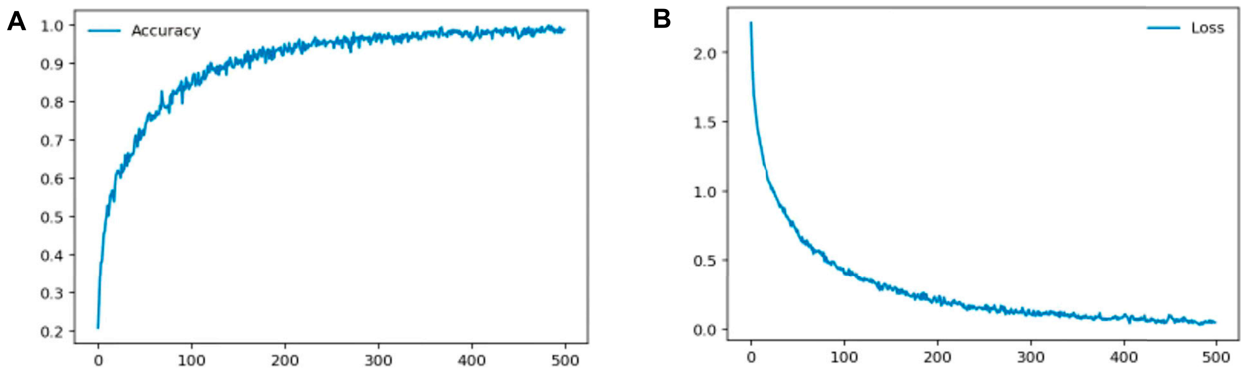


FIGURE 2 | Result of the convolutional neural network. **(A)** is accuracy, and **(B)** is loss.

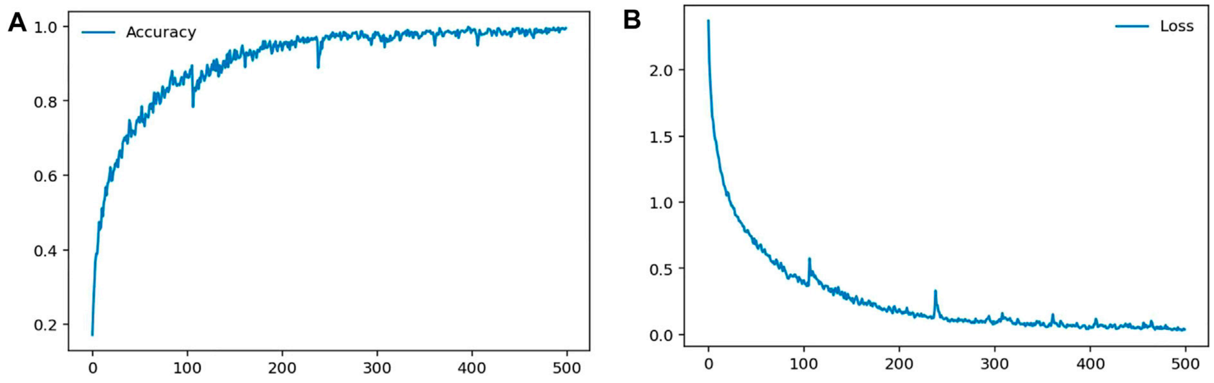


FIGURE 3 | Result of the BP neural network. **(A)** is accuracy, and **(B)** is loss.

TABLE 4 | Comparison of regression evaluation indexes.

Model	Accuracy	Recall	Precision	F1-score
LR	0.924	0.916	0.937	0.924
BPNN	0.966	0.950	**0.968**	0.966
SVM	0.938	0.925	0.958	0.938
CNN	**0.986**	**0.973**	0.956	**0.986**

prediction result, and the offset value and weight value are adjusted at the same time.

4.3.5 Calculation

In the model, we use the Relu function as the activation function of each layer and cross-entropy as the loss function.

$$L = \frac{1}{N}\sum_{i} = -\frac{1}{N}\sum_{i}\sum_{c=1}^{M} y_{ic}\, log\left(p_{ic}\right). \tag{4}$$

In addition, the Adam algorithm is used for backpropagation optimization. Finally, the model is trained and the Captum interpretability tool is used to obtain the three-level indicator weights. The lower-level index weight is calculated by weighted summation to obtain the upper-level index weight:

$$f\left(x\right) = \sum_{i=1}^{n} w_i x_i. \tag{5}$$

In this way, we calculated the first-level, second-level, and third-level index weights of the index system. When we use the index system, we input the calculated value of the third-level index of an event to be detected. Therefore, we can obtain the corresponding second-level and first-level index values using **Eq. 5** and obtain the propagation situation value of the event using model prediction.

4.4 Setting of Model Parameters

Convolutional neural networks can extract features at multiple levels and granularities through convolution and pooling operations and finally learn the propaganda posture features that can really distinguish the propaganda posture. We evaluate the needs and test the propaganda posture at different levels and granularities. The effect of different depth convolutional neural networks and different combinations of data input methods on the performance of the algorithm is to test the effect of convolutional neural networks on the quantitative evaluation of publicity situations. Based on the abovementioned experiments and tests, we have obtained a network structure with excellent experimental results. Two layers of convolutional layer and pooling layer are used, and two layers of fully connected layers, and finally, use the Softmax function is used for output, as shown in **Table 3**.

4.5 Experiment Results

We preprocessed the data and input it into the constructed convolutional neural network. After training, the results of fitting expert calibration are found to have very good performance and good convergence effect, as shown in **Figure 2** and we used the BP neural network which also has a good performance as in **Figure 3**.

We also used linear regression, SVM, and other methods to calculate and compared the results with the effect of the convolutional neural network. As shown in **Table 4**, we used the following standards: accuracy, recall, precision, and F1-score to measure the performance of the models. The bold values represent the best effects. We found that the convolutional neural network has more excellent results.

5 CONCLUSION

In summary, the existing methods have several shortcomings, which are mainly concentrated in the following aspects: 1) the calculation of the information dissemination index system relies too much on manual evaluation; 2) the determined index system is relatively limited and one-sided; and 3) the weight value of the index system fluctuates greatly, and sometimes, it cannot accurately reflect the dissemination trend of information dissemination events. In response to these shortcomings, this study put forward a way for constructing an information dissemination index system based on the CNN. We use the convolutional layer for multidimensional and multigranular feature extraction and apply the pooling layer to quickly reduce the size of the information dissemination network and highlight the main features. Through the deep-network structure with several hidden layers of the CNN, we have realized the evolution of simulating expert experience to assess all-round indicators. In addition, it has adaptive features such as self-learning. Through experimental comparison, the calculated results perform better than the other mentioned models. However, this CNN model lacks the best parameter proof, and for specific information dissemination events on different topics, it lacks a more targeted index system construction. This is the direction of our future improvement and research.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusion of this article will be made available by the authors, without undue reservation.

AUTHOR CONTRIBUTIONS

WH and LX contributed equally to this study and share first authorship. SL and LX contributed to the conception and design of the study and performed the experiments. XW and ZW grabbed and analyzed the dataset. SL and LX wrote the first draft of the manuscript.

FUNDING

This research was funded by NSFC (Nos U1803263, 62072131, and 61972106), Grant of Guangdong Recruitment Program of Foreign Experts (2020A1414010081), Grant of (18-163-15-ZD-002-003-10), Guangdong Higher Education Innovation Group, China (No. 2020KCXTD007), and Guangzhou Higher Education Innovation Group, China (No. 202032854).

REFERENCES

1. Gao C, Su Z, Liu J, Kurths J. Even central Users Do Not Always Drive Information Diffusion. *Commun ACM* (2019) 62:61–7. doi:10.1145/3224203
2. Gao C, Liu J. Network-based Modeling for Characterizing Human Collective Behaviors during Extreme Events. *IEEE Trans Syst Man, Cybernetics: Syst* (2016) 47:171–83. doi:10.1109/tsmc.2016.2608658
3. Warshaw C, Rodden J. How Should We Measure District-Level Public Opinion on Individual Issues. *J Polit* (2012) 74:203–19. doi:10.1017/s0022381611001204
4. Gil-García R, Pons-Porrata A. Dynamic Hierarchical Algorithms for Document Clustering. *Pattern Recognition Lett* (2010) 31:469–77. doi:10.1016/j.patrec.2009.11.011
5. Matsumura N, Ohsawa Y, Ishizuka M. Influence Diffusion Model in Text-Based Communication. *Trans Jpn Soc Artif Intelligence* (2002) 17:259–67. doi:10.1527/tjsai.17.259
6. Li S, Zhao D, Wu X, Tian Z, Li A, Wang Z. Functional Immunization of Networks Based on Message Passing. *Appl Maths Comput* (2020) 366:124728. doi:10.1016/j.amc.2019.124728
7. Su Z, Gao C, Liu J, Jia T, Wang Z, Kurths J. Emergence of Nonlinear Crossover under Epidemic Dynamics in Heterogeneous Networks. *Phys Rev E* (2020) 102:052311. doi:10.1103/PhysRevE.102.052311
8. Cha M, Haddadi H, Benevenuto F, Gummadi K. Measuring User Influence in Twitter: The Million Follower Fallacy. In: *Proceedings of the International AAAI Conference on Web and Social media*, 4 (2010).
9. Bermingham A, Conway M, McInerney L, O'Hare N, Smeaton AF. Combining Social Network Analysis and Sentiment Analysis to Explore the Potential for Online Radicalisation. In: 2009 International Conference on Advances in Social Network Analysis and Mining. IEEE (2009). p. 231–6. doi:10.1109/asonam.2009.31
10. Li X, Wang Z, Gao C, Shi L. Reasoning Human Emotional Responses from Large-Scale Social and Public media. *Appl Maths Comput* (2017) 310:182–93. doi:10.1016/j.amc.2017.03.031
11. Li S, Jiang L, Wu X, Han W, Zhao D, Wang Z. A Weighted Network Community Detection Algorithm Based on Deep Learning. *Appl Maths Comput* (2021) 401:126012. doi:10.1016/j.amc.2021.126012
12. Han W, Tian Z, Huang Z, Li S, Jia Y. Topic Representation Model Based on Microblogging Behavior Analysis. *World Wide Web* (2020) 23:3083–97. doi:10.1007/s11280-020-00822-x
13. Zhang L, Ghosh R, Dekhil M, Hsu M, Liu B. *Combining Lexicon-Based and Learning-Based Methods for Twitter Sentiment Analysis*. Palo Alto, CA: Hewlett-Packard Laboratories (2011). p. 89. Technical Report HPL-2011.
14. Quan Y, Jia Y, Zhou B, Han W, Li S. Repost Prediction Incorporating Time-Sensitive Mutual Influence in Social Networks. *J Comput Sci* (2018) 28:217–27. doi:10.1016/j.jocs.2017.11.015
15. Runxi Z. Construction of Early Warning index System for Internet Public Opinion Emergencies. Beijing: Beijing University of Chemical Technology. (2013) (Masters Thesis).
16. Yuan D. *Research on My Country's Internet Public Opinion Security Evaluation Index System*. Beijing: Beijing University of Chemical Technology (2008). Ph.D. thesis.
17. Zhu P, Wang X, Jia D, Guo Y, Li S, Chu C. Investigating the Co-evolution of Node Reputation and Edge-Strategy in Prisoner's Dilemma Game. *Appl Maths Comput* (2020) 386:125474. doi:10.1016/j.amc.2020.125474
18. Zhu P, Wang X, Li S, Guo Y, Wang Z. Investigation of Epidemic Spreading Process on Multiplex Networks by Incorporating Fatal Properties. *Appl Math Comput* (2019) 359:512–24. doi:10.1016/j.amc.2019.02.049
19. Runan S, Xiaolong D. Research on the index System of Internet Public Opinion Cases Based on Improved Radar Chart. In: *Beijing; Journal of Beijing University of Posts and Telecommunications (Social Sciences Edition)* (2015). p. 15–20.
20. Yiwen Z, Jiayin Q, Binxing F, Yuxiao L. Research on Internet Public Opinion Early Warning of Unconventional Crisis Events Based on Bayesian Network Modeling. *Libr Inf Work* (2012) 56:76.
21. Zizhan R. *Research and Implementation of Online Public Opinion Warning and Pre-plan System*. Beijing: Beijing University of Posts and Telecommunications (2013). Ph.D. thesis.
22. Fei H. *Study on Early Warning of Enterprise Network Public Opinion Crisis Based on System Dynamics*. Beijing: Capital University of Economics and Business (2013). Master's thesis.
23. Yanyan Z. *Study on Early Warning of Internet Public Opinion Crisis Based on Support Vector Machine*. Nanchang: Jiangxi University of Finance and Economics (2013). Ph.D. thesis, Master's Thesis.
24. Nie Y, Jia Y, Li S, Zhu X, Li A, Zhou B. Identifying Users across Social Networks Based on Dynamic Core Interests. *Neurocomputing* (2016) 210:107–15. doi:10.1016/j.neucom.2015.10.147
25. Weidong H, Ping L, Yi D, Hongwei L. Emotional Evolution Analysis of Internet Public Opinion Participants Based on Topic Feature Words. *Intelligence Mag* (2015) 34:117–22.

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers in Physics

ORIGINAL RESEARCH
published: 11 March 2022
doi: 10.3389/fphy.2022.830805

Contrastive Graph Learning for Social Recommendation

Yongshuai Zhang[1,2]*, Jiajin Huang*[1,2,3,4]*, Mi Li*[1,2,3,4] *and Jian Yang*[1,2,3,4]*

[1]Faculty of Information Technology, Beijing University of Technology, Beijing, China, [2]Beijing International Collaboration Base on Brain Informatics and Wisdom Services, Beijing, China, [3]Engineering Research Center of Intelligent Perception and Autonomous Control, Ministry of Education, Beijing, China, [4]Engineering Research Center of Digital Community, Ministry of Education, Beijing, China

Owing to the strength in learning representation of the high-order connectivity of graph neural networks (GNN), GNN-based collaborative filtering has been widely adopted in recommender systems. Furthermore, to overcome the data sparsity problem, some recent GNN-based models attempt to incorporate social information and to design contrastive learning as an auxiliary task to assist the primary recommendation task. Existing GNN and contrastive-learning-based recommendation models learn user and item representations in a symmetrical way and utilize social information and contrastive learning in a complex manner. The above two strategies lead to these models being either ineffective for datasets with a serious imbalance between users and items or inefficient for datasets with too many users and items. In this work, we propose a contrastive graph learning (CGL) model, which combines social information and contrastive learning in a simple and powerful way. CGL consists of three modules: diffusion, readout, and prediction. The diffusion module recursively aggregates and integrates social information and interest information to learn representations of users and items. The readout module takes the average value of user embeddings from all diffusion layers and item embeddings at the last diffusion layer as readouts of users and items, respectively. The prediction module calculates prediction rating scores with an interest graph to emphasize interest information. Three different losses are designed to ensure the function of each module. Extensive experiments on three benchmark datasets are implemented to validate the effectiveness of our model.

Keywords: recommender system, contrastive learning, graph neural network, social graph, interest graph

Edited by:
Peican Zhu,
Northwestern Polytechnical University, China

Reviewed by:
Xiaodong Yue,
Shanghai University, China
Zihan Zhou,
Louisiana State University, United States

****Correspondence:***
Jian Yang
jianyang@bjut.edu.cn

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *07 December 2021*
Accepted: *12 January 2022*
Published: *11 March 2022*

Citation:
Zhang Y, Huang J, Li M and Yang J (2022) Contrastive Graph Learning for Social Recommendation. Front. Phys. 10:830805. doi: 10.3389/fphy.2022.830805

1 INTRODUCTION

With the rapid development of networks, it is becoming harder and harder for a user to extract useful information from a mass of redundant information. Recommender systems play an important role in solving this problem and have become a promising solution for enhancing economic benefits in many domains like e-commerce, social media, and advertising [1, 2]. The main task of recommender systems is to provide interesting items for each user, which saves a lot of time for users and increases turnover for companies. One of the most popular recommendation techniques is collaborative filtering (CF), which infers each user's interests to items based on collaborative behaviors of all users without requiring the creation of explicit user and item profiles [3]. Matrix factorization (MF) is a key component in most learnable CF models [4, 5], which decomposes the user–item interaction matrix into two low-dimensional latent matrices for user and item representations [6]. However, MF-based models do not encode user and item embeddings well as a result of insufficient collaborative signals.

To yield satisfactory embeddings for CF, a graph neural network (GNN) [7] has been successfully applied to recommender systems [8–10]. Due to the high-order connectivity of GNN [11], GNN-based models can mine collaborative signals including high-order neighbors from abundant historical interactions; thus, it can generate more powerful node representations. But in real-world scenarios, it is always very hard to collect enough historical interaction information. To overcome the problem of data sparsity, many prior CF models [12–14] combine social information with historical interaction information and thus upgrade the recommendation performance. Recently, to further solve the data sparsity, many studies introduced self-supervised learning into recommender systems. The most popular self-supervised learning technology is contrastive learning, which has been successfully employed in many fields [15–18]. Contrastive learning utilizes self-supervision signals in a contrastive way, which pulls positive signals and target embeddings together and pushes negative signals and target embeddings away. To achieve better recommendation performance, many existing recommendation models [19–21] encode node embeddings by a GNN framework and simultaneously resort to contrastive learning in the learning process. Though these models take "social recommendation" as one of their aims, they still have the following inherent limitations that need to be addressed [13].

Firstly, existing GNN-based models [11, 22] learn representations of users and items in the same way and do not consider the different sparsities of users and items. In real-world scenarios, users and items are usually different in number or sparsity. For instance, in the Flickr dataset [23, 24], the number of items is almost 10 times that of users. Due to the huge difference in sparsity, high-quality representations of users may be obtained earlier than items in the learning stage. Thus, representations of users and items learned in the same way might decrease the embedding quality and then degrade the recommendation performance of models.

Secondly, although some social recommendation studies have made efforts to combine contrastive learning and social behaviors, it is still difficult to apply contrastive learning to social recommendation in an appropriate way. On the one hand, existing social recommendation models [19, 25] utilize contrastive learning in a quite complex manner such as encoding hypergraph and data augmentation. This complex manner may destroy the original social graph structure and waste social information. On the other hand, existing social recommendation models [19, 25] do not directly utilize social information in the prediction process, which might reduce the influence of social information. In a word, existing contrastive learning manners increase the complexity of social recommendation models but do not make full use of useful information contained in social behaviors.

This paper explores how to overcome the above limitations in existing recommendation models based on GNN and contrastive learning and proposes a contrastive graph learning (CGL) model for social recommendation. In social recommender systems, there are two kinds of neighbors for each user: historically interacted items and interacted users. Generally speaking, users with similar preferences are more likely to be friends with each other in daily life; likewise, intimate friends often have similar preferences. Hence, it is reasonable to characterize each user's preference by item aggregation and friend aggregation separately and to require user representations learned from the two views (user–item graph or user–user graph) to have consistent agreement [26]. This argument motivates us to simplify the contrastive learning task between social user embeddings and interest user embeddings. Moreover, to ensure that social user embeddings take part in the prediction process as well, we learn user embeddings in a diffused way inspired by some diffusion models [9, 24]. At each layer in the diffusion process, integrated user embeddings are obtained by taking the average value of social user and interest user embeddings. Besides, to avoid the negative effect of different sparsities in users and items [27], we design an asymmetrical readout strategy for users and items, which takes the average value of user embeddings from all diffusion layers and item embeddings from the last diffusion layer as their readouts, respectively. Since the task of our model is to find out the right items for users instead of mining potential social friends, we set up an extra interest graph for the final aggregation at the end of our model. Therefore, it is very essential to assure that historically interacted behaviors have a more powerful effect on recommendation results than social behaviors.

To summarize, this work makes the following main contributions:

- We successfully combine social information and interaction information by contrastive learning in recommender systems, which significantly improves the quality of recommendation results.
- We design a new readout strategy to alleviate the imbalance problem in the sparsity of users and items, which takes the average value of user embeddings from all layers and item embeddings from the last layer in the diffusion process as their readouts, respectively.
- We construct a pointwise loss between users and items in a contrastive way, which provides positive and negative signals for items. We place this loss and a pairwise loss in different modules to further promote the recommendation performance.
- We compare the proposed model with six state-of-the-art baselines on three real-world datasets, and experimental results demonstrate the effectiveness of our model.

The rest of our paper is organized as follows. **Section 2** summarizes some related works in recommender systems. **Section 3** introduces necessary notations and describes our model. In **Section 4**, we give some experimental results to validate the effectiveness of the proposed model and analyze the effect of hyper-parameters. **Section 5** concludes our work and discusses some possible issues in future work.

2 RELATED WORK

We simply review MF-based recommendation models and GNN-based recommendation models.

2.1 MF-Based Recommendation

In recommender systems, many classic collaborative filtering algorithms fall into the class of matrix factorization (MF) [28]. MF-based models project users and items into a low-dimensional latent space and represent a user or an item by a vector [29]. The inner product of a user vector and an item vector represents the user's satisfaction degree to the item. MF-based models have been widely used as baselines in recommender systems. SocialMF [30] employs the MF technique in social networks and assumes each user's features are dependent on his/her direct neighbors. So the feature vector of each user is supposed to keep consistent with social neighbors. TrustMF [31] considers a twofold influence of trust propagation, which analyzes different implications between truster to trustee and trustee to truster. To take advantage of these two different effects on performance, TrustMF also proposes a synthetic strategy to combine the truster model with the trustee model [32]. NeuMF [33] is the first model that combines the linearity of MF with the nonlinearity of the neural network. It indicates the importance of pre-training in the combination of two different models and achieves better performance than MF-based models and deep neural network (DNN)-based models. DASO [34] dynamically learns representations of users and items by using the generative adversarial network (GAN) [35].

2.2 GNN-Based Recommendation

In recent years, GNNs have shown their effectiveness in the recommendation field. GNNs aim to learn an embedding for each node which contains the information of neighbors [36, 37]. As the simplest GNN, LightGCN [22] does not have any complex operations other than neighbor aggregation, but it still achieves state-of-the-art performance for recommendation. In social recommender systems, GNN is first used in GraphRec [38]. In this model, the attention mechanism is extensively applied to aggregate neighbor information. After the aggregation process, rating scores can be obtained by putting user and item embeddings into a DNN. ConsisRec [39] takes social inconsistency into consideration and categorizes it into the context level and relation level. To solve the inconsistency at the context level, it obtains consistency scores by calculating the distance between neighbor embeddings and query embeddings and then samples consistent neighbors by relating sampling probability with consistency scores. After that, the attention mechanism is adopted to solve the inconsistency at the relation level. DiffNet++ [24] builds a unified framework to diffuse the social influence of social networks and interest influence of interest networks. Because information from social networks and interest networks can be spread into each other, it can receive different information in a recursive way, thereby learning more powerful representations of graph nodes. SGL [26] first introduces contrastive learning to recommendation and improves the accuracy and robustness of GNNs for recommendation. SGL generates graphs with different views by changing the graph structure in different manners and then utilizes supervised signals generated from these views to set an auxiliary self-supervised learning task. SEPT [19] adopts contrastive learning to social recommendation. It builds three different views by data augmentation, and each view provides supervision signals to other views. It employs contrastive learning for social recommendation for the first time, which takes recommendation and contrastive learning as the primary task and auxiliary task, respectively.

3 CGL MODEL

In this section, we present our CGL model. An overview of CGL is illustrated in **Figure 1**, which takes a user and an item as an example. CGL consists of three modules with different functions. The first one is a diffusion module, which builds connections between interest interactions and social links and guides the learning of representations in a recursive way. The second one is a readout module, which constructs user embeddings and item embeddings in an asymmetrical way to avoid the imbalance problem of users and items. The third one is a prediction module, which generates recommendations for users.

Some necessary notations are defined in **Section 3.1**. **Section 3.2**, **Section 3.3**, and **Section 3.4** introduce the diffusion module, readout module, and prediction module, respectively. The training of the model is given in **Section 3.5**. Finally, the complexity of CGL is analyzed in **Section 3.6**.

3.1 Notations

To facilitate the reading, matrices appear in bold capital letters and vectors appear in bold lowercase letters. Let $\mathcal{U}$ and $\mathcal{V}$ be the set of m users and n items, respectively. Denote by $\mathcal{G}_r = (\mathcal{N}, \mathcal{E}_r)$ the user–item interest graph, where $\mathcal{N} = \mathcal{U} \cup \mathcal{V}$ and $\mathcal{E}_r$ is the edge set indicating interactions between users and items. Let $\mathcal{G}_s = (\mathcal{T}, \mathcal{E}_s)$ be the user–user social graph, where $\mathcal{T} \subseteq \mathcal{U}$ and $\mathcal{E}_s$ is the edge set indicating social links among users. In this paper, we keep all users in a social network, so $\mathcal{T} = \mathcal{U}$. For $\mathcal{G}_r$, the binary matrix $\mathbf{R} = [r_{ij}]_{m\times n}$ represents its user–item interactions, where $r_{ij} = 1$ if user i has an interaction with item j; otherwise, $r_{ij} = 0$. For $\mathcal{G}_s$, the binary matrix $\mathbf{S} = [s_{ij}]_{m\times m}$ represents its user–user social links, where $s_{it} = 1$ if user i has a link with another user t; otherwise, $s_{it} = 0$.

For users and items, we encode two basic embedding matrices, $\mathbf{U}^{(0)} = [\mathbf{u}_1^{(0)}, \mathbf{u}_2^{(0)}, \ldots, \mathbf{u}_{|\mathcal{U}|}^{(0)}]$ and $\mathbf{V}^{(0)} = [\mathbf{v}_1^{(0)}, \mathbf{v}_2^{(0)}, \ldots, \mathbf{v}_{|\mathcal{V}|}^{(0)}]$, where $\mathbf{u}_i^{(0)}$ is a d-dimensional embedding of user i and $\mathbf{v}_j^{(0)}$ is a d-dimensional embedding of item j. Starting from $\mathbf{U}^{(0)}$ and $\mathbf{V}^{(0)}$, a graph convolution operation is implemented on $\mathcal{G}_r$ and $\mathcal{G}_s$ to produce user and item representations, respectively.

3.2 Diffusion Module

The diffusion module has L layers, and each layer consists of aggregation on the interest graph, aggregation on the social graph, and their integration. The input of the first layer is the initialized user latent embedding $\mathbf{u}_i^{(0)}$ and item latent embedding $\mathbf{v}_j^{(0)}$, and the input of other layers is the output of their respective previous layers. These layers recursively model the user's latent preference and the item's latent preference propagation in two graphs with layer-wise convolutions. LightGCN [22] is a brief graph convolution network (GCN)-based general recommendation model, which discards two standard operations in GCNs:

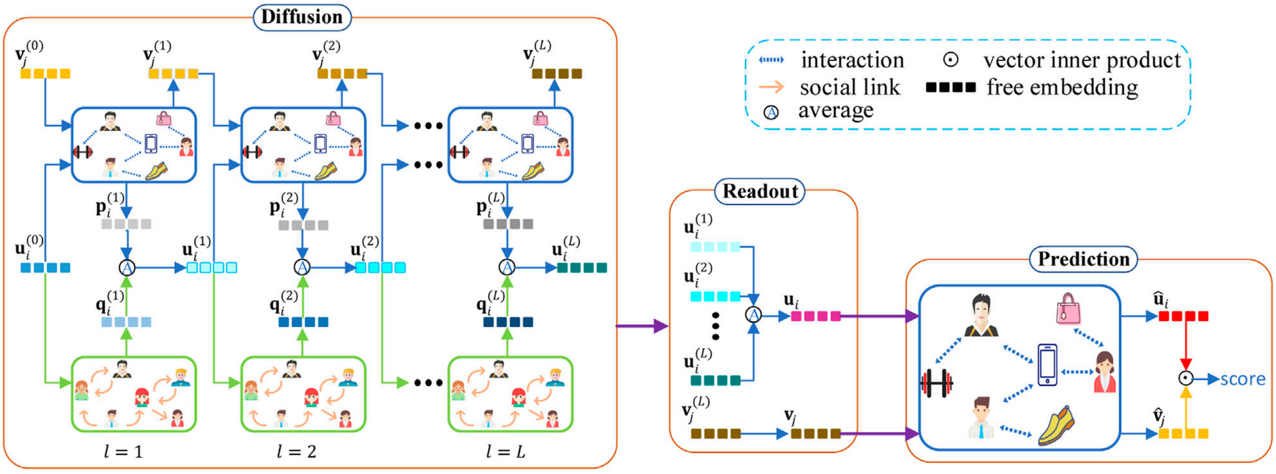


FIGURE 1 | The overall architecture of the proposed model.

feature transformation and nonlinear activation. We utilize LightGCN to realize aggregation operations in CGL.

To aggregate interaction information in the interest graph $\mathcal{G}_r$, we collect the neighbor information of each node by the simple way of LightGCN. Specifically, for a given user i and item j, let $\mathbf{u}_i^{(l-1)}$ and $\mathbf{v}_j^{(l-1)}$ represent user embedding and item embedding from the $(l-1)$-th layer, respectively. Then the l-th layer interest aggregation process is given by

$$\mathbf{p}_i^{(l)} = Agg_{items}\left(\mathbf{v}_j^{(l-1)}, \forall j \in \mathcal{N}_i\right) = \sum_{j \in \mathcal{N}_i} \frac{\mathbf{v}_j^{(l-1)}}{\sqrt{|\mathcal{N}_i||\mathcal{N}_j|}}, \tag{1}$$

$$\mathbf{v}_j^{(l)} = Agg_{users}\left(\mathbf{u}_i^{(l-1)}, \forall i \in \mathcal{N}_j\right) = \sum_{i \in \mathcal{N}_j} \frac{\mathbf{u}_i^{(l-1)}}{\sqrt{|\mathcal{N}_j||\mathcal{N}_i|}}, \tag{2}$$

where $\mathcal{N}_i$ is the set of items that are interacted by user i and $\mathcal{N}_j$ is the set of users that interact with item j. LightGCN keeps the same normalization operation as standard GCNs [40]; i.e., the neighbor number of the current node and aggregated node is utilized for normalization during aggregating information. This strategy is rather essential to avoid the unreasonable increase of embedding in graph convolution operations.

To aggregate social information of the social graph $\mathcal{G}_s$, we perform node aggregation based on LightGCN as well. Note that there are only user nodes in social graphs. Let $\mathbf{u}_i^{(l-1)}$ and $\mathcal{T}_i$ be user embeddings from the $(l-1)$-th layer and the set of social neighbors linked with user i, respectively. The l-th layer social aggregation process is defined by

$$\mathbf{q}_i^{(l)} = Agg_{neighbors}\left(\mathbf{u}_t^{(l-1)}, \forall t \in \mathcal{T}_i\right) = \sum_{t \in \mathcal{T}_i} \frac{\mathbf{u}_t^{(l-1)}}{\sqrt{|\mathcal{T}_i||\mathcal{T}_t|}}. \tag{3}$$

As the above equation shows, LightGCN designs normalization by taking the neighbor number of current trustee node and aggregated truster node.

Aggregation on the interest graph generates user embedding $\mathbf{p}_i^{(l)}$ and aggregation on the social graph produces user embedding $\mathbf{q}_i^{(l)}$. These two user embeddings contain different information about user i and will be further integrated to generate user embedding $\mathbf{u}_i^{(l)}$ at the l-th layer. We simply integrate $\mathbf{p}_i^{(l)}$ and $\mathbf{q}_i^{(l)}$ by taking their average value. That is to say, we get integrated user embedding at the l-th layer by

$$\mathbf{u}_i^{(l)} = \frac{\mathbf{p}_i^{(l)} + \mathbf{q}_i^{(l)}}{2}. \tag{4}$$

Then integrated user embedding $\mathbf{u}_i^{(l)}$ and aggregated item embedding $\mathbf{v}_j^{(l)}$ are input into the $(l+1)$-th layer so as to diffuse interest and social information into two graphs. By integrating social user embeddings and interest user embeddings at each layer, we can update integrated user embeddings consecutively. Consequently, the fusion between social information and interest information becomes closer and closer in the convolution process. Moreover, although item embeddings do not take part in the diffusion process, due to the high-order connectivity in GNN, social user embeddings and item embeddings can still exert a powerful influence on each other, especially when the diffusion process becomes deeper.

The diffusion process of CGL is inspired by the validity of the diffusion model DiffNet++, but it is much more concise than that of DiffNet++. Unlike the Diffnet++, CGL does not absorb any information from other attribute features of users or items and introduce any attention mechanism. The former will take us a lot of time to process attribute features, and the latter will use DNNs. As such, the time complexity of our diffusion process is lower compared with that of DiffNet++.

3.3 Readout Module

After the above L layer diffusion process, we construct readouts for users and items, separately, and then these readouts are sent to the last interest graph in the subsequent prediction module. Our strategy for constructing readouts is different from existing GNN-based recommendation models. In existing GNN-based models, there are two strategies to prepare embeddings for the prediction phase or other subsequent phases. One is taking the average value of all layers' embeddings for users and items, and the other is taking the embedding value of the last stacked layer. Both these strategies deal with users and items in a symmetrical manner, which results in the model being unable to learn good representations of users and items while the numbers of users and items in the training data are very different. To alleviate the problem, we integrate these two strategies to constitute readouts of users and items and adopt the former for users and the latter for items. Because of the massive use of social information, user embeddings in each layer of the diffusion process contain some collaborative signals, which can improve the quality of user embeddings. Thus high-quality representations of users may be generated in early stages of the diffusion process. That encourages us to use all user embeddings in the diffusion process to build readouts of users. Specifically, the readout $\mathbf{u}_i$ of user i is defined as the average of embeddings from all L layers in the diffusion process:

$$\mathbf{u}_i = Rdout_{users}\left(\left\{\mathbf{u}_i^{(l)} | l = [1,\ldots,L]\right\}\right) = \frac{1}{L}\sum_{l=1}^{L}\mathbf{u}_i^{(l)}. \tag{5}$$

For an item, its embedding in the early stage of the diffusion process may be poor due to the large item number, sparsity, and lack of auxiliary information. So the readout $\mathbf{v}_j$ of item j is defined as its embedding at the last layer in the diffusion process:

$$\mathbf{v}_i = Rdout_{items}\left(\left\{\mathbf{v}_i^{(l)} | l = [1,\ldots,L]\right\}\right) = \mathbf{v}_j^{(L)}. \tag{6}$$

From **Eq. 5** and **Eq. 6**, it can be seen that our readout strategy is asymmetrical for users and items.

3.4 Prediction Module

Considering that the task of recommendation is to predict interactions between users and items, we add a separate interest graph to emphasize the influence of interaction information between their readouts. The interest graph generates the final user embedding $\hat{\mathbf{u}}_i$ and item embedding $\hat{\mathbf{v}}_j$ by aggregating $\mathbf{v}_j$s and $\mathbf{u}_i$s, respectively. That is,

$$\hat{\mathbf{u}}_i = Agg_{items}\left(\mathbf{v}_j, \forall j \in \mathcal{N}_i\right), \tag{7}$$

$$\hat{\mathbf{v}}_j = Agg_{users}\left(\mathbf{u}_i, \forall i \in \mathcal{N}_j\right). \tag{8}$$

Then the prediction module defines the inner product between the final user and item embeddings

$$\hat{r}_{ij} = <\hat{\mathbf{u}}_i, \hat{\mathbf{v}}_j>. \tag{9}$$

The inner product is taken as the ranking score to generate recommendations.

3.5 Model Training

To train the model, we design a self-supervised loss for the diffusion module and a supervised loss for the readout module and prediction module. At each layer of the diffusion module, the interest graph and social graph generate two user embeddings for each user separately. In order to make the integration on them more reasonable, we design a self-supervised loss to close two embeddings of the same user. Users and items are two aspects of recommender systems, and we can recommend items for a given user or select users for a given item. According to these two tasks, we design a supervised loss in the readout module and prediction module, respectively. By jointly optimizing the above three losses, we learn parameters in the model.

3.5.1 Self-Supervised Loss

We integrate social user embeddings and interest user embeddings by **Eq. 4** at each layer of the diffusion process. Such a direct integration strategy can make the two groups of embeddings complement each other but cannot guarantee that they know each other in the training process. So, we introduce a social contrastive learning loss in the diffusion module. The main idea comes from the assumption that social behaviors can usually reflect the preference of a user. As a result, the user's embedding from the social graph is supposed to have a close connection with that from the interest graph.

For convenience, let $\mathbf{P}^{(l)} = [\mathbf{p}_1^{(l)}, \mathbf{p}_2^{(l)}, \ldots, \mathbf{p}_{|\mathcal{U}|}^{(l)}] \in \mathbb{R}^{m\times k}$ be the l-th interest aggregation matrix on the interest graph, where $\mathbf{p}_i^{(l)}$ is calculated by **Eq. 1**. Similarly, $\mathbf{Q}^{(l)} = [\mathbf{q}_1^{(l)}, \mathbf{q}_2^{(l)}, \ldots, \mathbf{q}_{|\mathcal{V}|}^{(l)}] \in \mathbb{R}^{m\times k}$ denote the l-th social aggregation matrix, where $\mathbf{q}_i^{(l)}$ is calculated by **Eq. 3**. We first produce two user embedding matrices $\mathbf{P}$ and $\mathbf{Q}$ by

$$\mathbf{P} = \frac{\sum_{l=1}^{L}\mathbf{P}^{(l)}}{L}, \tag{10}$$

$$\mathbf{Q} = \frac{\sum_{l=1}^{L}\mathbf{Q}^{(l)}}{L}. \tag{11}$$

We randomly ruffle matrix $\mathbf{Q}$ row-wise and column-wise in turn and denote the ruffled one as $\tilde{\mathbf{Q}}$

$$\tilde{\mathbf{Q}} = r(\mathbf{Q}), \tag{12}$$

where $r(\cdot)$ is the ruffle operation. Then the i-th row of the user embedding matrix can be a latent embedding of user i, and the j-th row of the item embedding matrix can be a latent embedding of item j. Thus, for user i and item j, we can get the interest user embedding $\mathbf{p}_i$ from $\mathbf{P}$, real social user embedding $\mathbf{q}_i$ from $\mathbf{Q}$, and fake social user embedding $\tilde{\mathbf{q}}_i$ from $\tilde{\mathbf{Q}}$. We assume that fake social user embeddings are quite different from their corresponding interest user embeddings. We minimize the agreement between each fake social user embedding and its corresponding interest user embedding, and the social contrastive learning loss in the diffusion module can be formulated as

$$\mathcal{L}_{c-uu} = \sum_{i\in\mathcal{U}} -\left(log\left(\sigma\left(<\mathbf{p}_i, \mathbf{q}_i>\right)\right) + log\left(1 - \sigma\left(<\mathbf{p}_i, \tilde{\mathbf{q}}_i>\right)\right)\right), \tag{13}$$

where $\sigma(\cdot)$ is the sigmoid function.

3.5.2 Supervised Loss

In the prediction module, we can get rating scores by **Eq. 9**. To assure observed interactions can achieve higher scores than unobserved interactions, we employ the pairwise Bayesian personalized ranking (BPR) loss as our primary loss to induce model learning, which is proposed to make the observed interactions be ranked in front of unobserved interactions. The BPR loss in CGL is formulated as

$$\mathcal{L}_{bpr} = \sum_{(i,j^{+},j^{-})\in\mathcal{R}} -log\left(\sigma\left(\hat{r}_{ij^{+}} - \hat{r}_{ij^{-}}\right)\right), \tag{14}$$

where $\mathcal{R} = \{(i, j^{+}, j^{-}) | (i, j^{+}) \in R^{+}, (i, j^{-}) \in R^{-}\}$ denotes training data, R^{+} indicates observed interactions, and R^{-} indicates unobserved interactions. By minimizing the BPR loss, the predictive score of an observed interaction could be enforced to be higher than that of its unobserved counterparts. However, as a pairwise loss, it ignores the entrywise consistency between predictive scores and real scores. Besides, the BPR loss only provides positive and negative signals for a given user, which is unfair for items and makes it hard to learn good item representations.

To overcome this shortcoming, we employ a pointwise loss as a complement to the BPR loss and formulate it as

$$\mathcal{L}_{c-uv} = \sum_{j\in\mathcal{V}} \sum_{i\in\mathcal{N}_j} -log \frac{\exp\left(<\mathbf{u}_i, \mathbf{v}_j>\right)}{\sum_{t\in\mathcal{U}} \exp\left(<\mathbf{u}_t, \mathbf{v}_j>\right)}, \tag{15}$$

where each $\mathbf{u}_i$ and $\mathbf{v}_j$ are given by **Eqs 5** and **6**, respectively. Obviously, this loss provides positive and negative supervised signals for a given item, instead of a given user. By minimizing **Eq. 15**, predictive scores could be consistent with real scores. It is worth mentioning that we define the loss in the readout module instead of in the prediction module. This is different from most existing heterogeneous losses that typically combine the pointwise loss and pairwise loss in the prediction module. The main reason is that we separate the predictive task and the ranking task by using different user and item representations. The pointwise loss focuses on users for a given item and enhances the user representation directly derived from the aggregation operation on both interest and social graphs. And these enhanced user and item representations are used to aggregate information on an interest graph for the ranking task.

3.5.3 Final Loss

With the above three losses in three modules of the CGL model, we set its final loss as

$$\mathcal{L}_{CGL} = \mathcal{L}_{bpr} + \alpha\mathcal{L}_{c-uu} + \beta\mathcal{L}_{c-uv} + \lambda\|\Theta\|_2^2, \tag{16}$$

where α and β are two additional regularizers to control the strength of $\mathcal{L}_{c-uu}$ and $\mathcal{L}_{c-uv}$, respectively; Θ is the set of all learnable parameters; and λ controls the strength of L_2 regularization. The overall training process of CGL is given in Algorithm 1.

Algorithm 1. The training process of CGL.

```
Input: interest graph $\mathcal{G}_r$, social graph $\mathcal{G}_s$, trainable parameters $\Theta$.
for each iteration do
    for each diffusion layer do
        Obtain user embeddings from $\mathcal{G}_r$ and $\mathcal{G}_s$ using Eq. (1) and (3);
        Minimize the contrastive loss function $\mathcal{L}_{c-uu}$ using Eq. (13);
        Obtain user and item embeddings of this diffusion layer using Eq. (2) and (4);
    end
    Select diffused embeddings using Eq. (5) - (6);
    Minimize the contrastive loss function $\mathcal{L}_{c-uv}$ using Eq. (15);
    Obtain final embeddings by Eq. (10) - (11);
    Minimize the primary loss $\mathcal{L}_{bpr}$ using Equation (14);
    Minimize the final loss $\mathcal{L}_{CGL}$ in Eq. (16);
end
Output: optimized model parameter set $\Theta$.
```

3.6 Complexity Analysis

The overall time complexity of CGL mainly comes from two parts: aggregation on graphs and calculation on three losses. At each iteration, training data aggregate neighbor information on interest graphs $L + 1$ times and on social graphs L times. Thus, the complexity of aggregation on interest graphs is $\mathcal{O}(|\mathbf{R}|d(L+1))$, which is only dependent on the latent dimension and the size of rating data. Similarly, the complexity of aggregation on social graphs is $\mathcal{O}(|\mathbf{S}|dL)$. Compared with interest aggregation, besides the difference in aggregation number, social aggregation has fewer nodes in the graph structure. In short, the time complexity of the aggregation operation on graphs increases linearly with the size of training data, the dimension of latent embedding, and the number of aggregation. For the complexity of calculation on three losses, we only take the inner product operation into consideration, since it produces the major complexity in our model. Within a batch, the complexity of the contrastive loss $\mathcal{L}_{c-uu}$ is $\mathcal{O}(2Bd)$, where B is the batch size. For another contrastive learning loss $\mathcal{L}_{c-uv}$, we get its numerator by calculating the inner product between each positive pair in a batch; hence, the complexity of the numerator is $\mathcal{O}(Bd)$. Likewise, the denominator is obtained by the product between the user embedding matrix and item embedding matrix, and its complexity is $\mathcal{O}(B^2d)$. For the BPR loss, we calculate the inner product of all positive pairs and negative pairs in each batch, and its complexity is $\mathcal{O}(2Bd)$. Therefore, the total time complexity of training CGL in one batch is $\mathcal{O}(|\mathbf{R}|d(L+1) + |\mathbf{S}|dL + 5Bd + B^2d)$.

4 EXPERIMENTS

We conduct multiple experiments to verify the effectiveness of CGL in this section. Experimental setup is introduced in **Section 4.1**. The performance of CGL is compared with six baselines in **Section 4.2**. **Section 4.3** analyzes the effects of different strategies for the readout strategy and pointwise contrastive loss. **Section 4.4** shows the performance of CGL under different hyper-parameters. Note that we omit the percent sign of model performance in all tables.

4.1 Experimental Setup

Experimental setup contains datasets, evaluation metrics, baselines, and parameter settings.

4.1.1 Datasets

The task of our experiments is a top-K recommendation, and we conduct experiments on three representative real-world datasets: Yelp [24], Flickr [23], and Ciao [41].

- **Yelp:** This dataset is crawled from an online location-based review site, Yelp. Users on the site are encouraged to interact with others and express their opinions through the form of reviews and ratings. The ratings data are converted into implicit feedback as the dataset. The itemset of this dataset includes a variety of locations visited or reviewed by users.

TABLE 1 | The statics of datasets.

Dataset	Users	Items	Ratings	Links	Rating density (%)	Link density (%)
Yelp	17,237	38,342	204,448	143,765	0.03	0.05
Flickr	8,358	82,120	327,815	187,273	0.05	0.27
Ciao	7,317	104,975	283,319	111,781	0.04	0.21

And the relationship among users can be found out directly in terms of the friend list of users.

- **Flickr**: This dataset is crawled from one of the largest social image-sharing platforms, Flickr. In this platform, users can share their preferences in images with their social followers and follow the people they are interested in. So, social images make up the itemset, and the social relationship can be confirmed through followers of users.
- **Ciao**: This dataset is crawled from an online shopping site, Ciao. On the site, people not only write critical reviews for various products but also read and rate the reviews written by others. Furthermore, people can add members to their trust networks or “Circle of Trust”, if they find their reviews consistently interesting and helpful [41]. The itemset of this dataset includes a variety of goods.

The above three datasets can be publicly downloaded online (Yelp and Flickr,[1] and Ciao[2]) provided by [19, 23], and their statistics are summarized in **Table 1**. Following many previous works [9, 24], we convert all explicit ratings into implicit ratings and remove repeated ratings in each dataset. Similar to [30], we only focus on the social information in these datasets and do not consider the attribute information of users and items. Finally, we randomly select 10% of rating data as testing set and the rest as training set.

4.1.2 Evaluation Metrics

For all models, we perform item ranking on all candidate items and evaluate the performance of each model with Precision@K, Recall@K, and NDCG@K metrics, which are three widely used evaluation protocols. The NDCG metric is sensitive to rank, and the other two metrics can measure the relevancy of the recommendation list.

4.1.3 Baselines

- **MF-BPR** [42]: It exploits how to represent users and items in a low-dimensional latent space, which is optimized by the BPR loss.
- **SocialMF** [30]: It combines social information and purchase information through the form of MF. For a specific user, this model focuses on not only items purchased by the user but also social neighbors around the user.
- **LightGCN** [22]: It is a light version of the GNN-based model and only performs aggregation operations. As a state-of-the-art model, it has been widely used as a baseline in many recommender system studies.
- **SEPT** [19]: It builds complementary views of the raw data so as to provide self-supervision signals (pseudo-labels) to participate in the contrastive learning. To ensure the effectiveness of contrastive learning, it simultaneously employs the tri-training scheme to coordinate the labeling process.
- **SGL** [26]: It constructs different views by perturbing the raw data graph with uniform node/edge dropout and then conducts self-discrimination-based contrastive learning over these views to learn node representations.
- **DiffNet++** [24]: It recursively diffuses different information into social networks and interest networks, which can be used without attribute information fusion and attention mechanism.

4.1.4 Settings

For a fair comparison, all models are optimized by the Adam optimizer [43] and initialized in the same way. The size of each batch is set to 2,048, and the dimension of the latent vector is tuned in {8, 16, 32, 64, 128}. It is worth noting that MF-based models are often different from GNN-based models in L_2 regularization. In order to ensure that all models can achieve good results, the L_2 regularization coefficient λ is chosen from $\{1.0 \times 10^{-1}, 1.0 \times 10^{-2}\}$ for MF-based models and from $\{1.0 \times 10^{-4}, 5.0 \times 10^{-5}, 1.0 \times 10^{-5}\}$ for GNN-based models. The learning rate is tuned in $\{1.0 \times 10^{-2}, 5.0 \times 10^{-3}, 1.0 \times 10^{-3}\}$, and the maximum train epoch is 150. For α and β, we tune them in different ranges on three datasets. On Yelp, α and β are searched in $\{4.0 \times 10^{-4}, 4.0 \times 10^{-5}, 4.0 \times 10^{-6}, 4.0 \times 10^{-7}, 4.0 \times 10^{-8}\}$ and $\{6.5 \times 10^{-1}, 6.5 \times 10^{-2}, 6.5 \times 10^{-3}, 6.5 \times 10^{-4}, 6.5 \times 10^{-5}\}$, respectively. On Flickr, α and β are tuned in $\{1.6 \times 10^{-4}, 1.6 \times 10^{-5}, 1.6 \times 10^{-6}, 1.6 \times 10^{-7}, 1.6 \times 10^{-8}\}$ and $\{8.5 \times 10^{-5}, 8.5 \times 10^{-6}, 8.5 \times 10^{-7}, 8.5 \times 10^{-8}, 8.5 \times 10^{-9}\}$, respectively. On Ciao, α and β are tuned in $\{2.0 \times 10^{-5}, 2.0 \times 10^{-6}, 2.0 \times 10^{-7}, 2.0 \times 10^{-8}, 2.0 \times 10^{-9}\}$ and $\{1.0 \times 10^{-0}, 1.0 \times 10^{-1}, 1.0 \times 10^{-2}, 1.0 \times 10^{-3}, 1.0 \times 10^{-4}\}$, respectively. Our experiments are implemented in PyTorch.

4.2 Overall Performance Comparison

We evaluate our proposed CGL model by comparing it with six baselines. For all models, we exhibit the performance of the top 10, top 15, and top 20 in **Table 2**, **Table 3**, and **Table 4**, respectively. From these tables, we have the following observations:

- CGL outperforms all baselines on Yelp and Flickr by a large margin. On Yelp, the average top-K (10, 15, 20) improvement of CGL on Precision, Recall, and NDCG is 6.60%, 5.77%, and 6.63%, respectively. On Flickr, the average top-K (10, 15, 20) improvement of CGL on Precision, Recall, and NDCG is 9.76%, 10.80%, and

[1] https://github.com/PeiJieSun/diffnet

[2] https://https://github.com/Coder-Yu/QRec

TABLE 2 | Overall comparison (*K* = 10). The best results are in bold.

Models	Yelp			Flickr			Ciao		
	Prec	Rec	NDCG	Prec	Rec	NDCG	Prec	Rec	NDCG
MF-BPR	0.3962	2.222	1.238	0.2534	0.5698	0.4669	1.065	3.127	2.199
SocialMF	0.4030	2.251	1.247	0.2500	0.5707	0.4634	1.163	3.255	2.406
LightGCN	0.4797	2.650	1.563	0.3168	0.7847	0.6366	1.699	4.240	3.531
SEPT	0.4418	2.479	1.366	0.2620	0.6090	0.5016	1.647	4.082	3.260
SGL	0.4826	2.662	1.535	0.3356	0.8635	0.6424	**1.775**	4.344	**3.753**
DiffNet++	0.5156	2.819	1.633	0.3442	0.8843	0.7056	1.596	4.026	3.342
CGL	**0.5351**	**2.940**	**1.726**	**0.3750**	**0.9913**	**0.7664**	1.757	**4.522**	3.610

TABLE 3 | Overall comparison (*K* = 15). The best results are in bold.

Models	Yelp			Flickr			Ciao		
	Prec	Rec	NDCG	Prec	Rec	NDCG	Prec	Rec	NDCG
MF-BPR	0.3761	3.165	1.523	0.2249	0.7485	0.5210	0.9643	4.133	2.544
SocialMF	0.3671	3.072	1.497	0.2306	0.7983	0.5321	1.050	4.326	2.765
LightGCN	0.4512	3.738	1.881	0.2979	1.071	0.7287	1.444	5.405	3.868
SEPT	0.4137	3.459	1.664	0.2260	0.7321	0.5427	1.426	5.147	3.602
SGL	0.4577	3.814	1.884	0.2957	1.112	0.7181	1.494	5.520	**4.093**
DiffNet++	0.4590	3.761	1.951	0.3185	1.202	0.8120	1.371	5.096	3.683
CGL	**0.4998**	**4.101**	**2.077**	**0.3493**	**1.365**	**0.9006**	**1.540**	**5.788**	4.036

TABLE 4 | Overall comparison (*K* = 20). The best results are in bold.

Models	Yelp			Flickr			Ciao		
	Prec	Rec	NDCG	Prec	Rec	NDCG	Prec	Rec	NDCG
MF-BPR	0.3583	4.042	1.760	0.2089	0.9296	0.5750	0.8791	4.942	2.802
SocialMF	0.3651	4.095	1.776	0.2175	0.9913	0.5907	0.9794	5.385	3.094
LightGCN	0.4452	4.946	2.236	0.2756	1.270	0.7940	1.298	6.478	4.181
SEPT	0.3977	4.487	1.946	0.2295	0.9810	0.6273	1.237	5.916	3.832
SGL	0.4292	4.774	2.144	0.2714	1.321	0.7855	1.314	6.332	**4.344**
DiffNet++	0.4346	4.723	2.238	0.3005	1.516	0.9059	1.235	6.105	4.017
CGL	**0.4773**	**5.217**	**2.409**	**0.3348**	**1.618**	**0.9942**	**1.368**	**6.669**	4.287

10.01%, respectively. On Ciao, the average top-K (10, 15, 20) improvement of CGL on Precision, Recall, and NDCG is −2.17%, 3.97%, and 2.06%, respectively. Although CGL's performance on Ciao is not as good as that on Yelp and Flickr, it is still the best or second best among all models in terms of the three metrics. Since the overall performance of all models is similar when $K = 10$, $K = 15$, and $K = 20$, we only discuss the case of $K = 20$ in the subsequent experiments.

- Nearly all GNN-based models (e.g., LightGCN, DiffNet++, and SGL) perform much better than MF-based models (MF-BPR and SocialMF), which demonstrates the important role of GNNs for the recommendation task. We also observe that some GNN-based models (DiffNet++ and SEPT) incorporate the social information, but cannot beat other models without social information on some metrics. It illustrates that though social behaviors may reflect a person's interest information to an item indeed, it is hard to mathematically design a reasonable and effective manner to utilize social information. Fortunately, CGL performs better than almost all baselines, which proves that contrastive learning in CGL can ensure the effectiveness of the social information.
- Compared with its performance on Yelp and Ciao, CGL brings more improvement on Flickr. As Flickr has much denser social information, this may indicate that CGL can make full use of social information. Besides, by analyzing the performance of DiffNet++ and CGL on Ciao, we conclude that there may be much noise in the raw data of Ciao so that it is hard to improve the recommendation performance by directly diffusing social information and interest information. Naturally, SGL shows better performance on some metrics because self-discrimination-based contrastive learning can help to alleviate noise effect greatly.

Moreover, to concretely investigate the impact of contrastive learning and the diffusion process on the model, we further compare CGL with two other GNN-based models (LightGCN and DiffNet++) at each layer and show the relevant results in

TABLE 5 | Comparison of CGL and two GNN-based baselines at each layer (K = 20). The best results at each layer are in bold.

Models	# Layer	Yelp			Flickr			Ciao		
		Prec	Rec	NDCG	Prec	Rec	NDCG	Prec	Rec	NDCG
LightGCN	1	0.405 9	4.434	2.033	0.197 8	0.970	0.557 1	1.135	6.105	4.017
DiffNet++	1	0.400 6	4.403	2.019	0.230 3	1.213	0.675 2	1.181	5.952	3.735
CGL	1	**0.450 1**	**4.600**	**2.287**	**0.309 1**	**1.377**	**0.913 7**	**1.368**	**6.669**	**4.287**
LightGCN	2	0.434 5	4.891	2.177	0.275 6	1.270	0.794 0	**1.298**	6.478	**4.181**
DiffNet++	2	0.432 1	4.723	2.218	0.300 5	**1.517**	**0.905 9**	1.235	6.105	4.017
CGL	2	**0.475 3**	**5.324**	**2.405**	**0.303 1**	1.475	0.861 1	1.255	**6.538**	4.062
LightGCN	3	0.445 2	4.946	2.236	0.264 6	1.214	0.751 5	1.251	**6.578**	**4.018**
DiffNet++	3	0.434 6	4.723	2.238	0.297 1	1.477	0.862 5	1.226	6.031	3.885
CGL	3	**0.477 3**	**5.217**	**2.409**	**0.334 8**	**1.618**	**0.994 2**	**1.275**	6.501	3.947

TABLE 6 | Average epoch runtime of all models (seconds).

Datasets	Time/Epoch						
	MF-BPR	SocialMF	LightGCN	SEPT	SGL	DiffNet++	CGL
Yelp	5.32	6.44	10.71	45.69	22.22	12.71	15.20
Flickr	7.03	8.57	19.56	134.61	45.87	22.81	28.19
Ciao	8.30	9.23	20.54	93.40	46.04	21.88	26.80

Table 5 ($K = 20$). From the table, we have the following observations:

- CGL outperforms LightGCN and DiffNet++ at the first and the third layers on both Yelp and Flickr with respect to all metrics. At the second layer, CGL still achieves the best performance except Recall and NDCG on Flickr. On Ciao, our model reaches its best performance at the first layer, and the performance is better than that of DiffNet++ and LightGCN at any layer. These experiment results illustrate the superiority of our model and the necessity of constructing supervised loss for social user representations.
- The overall performance of LightGCN at each layer is better than that of Diffnet++ on Yelp. However, DiffNet++ performs better than LightGCN on Flickr. A possible reason is that directly diffusing social information without any supervised measure cannot guarantee the validity of social information, especially while social information is inadequate. However, the performance of DiffNet++ is inferior to that of LightGCN on Ciao; we infer there may be much noise in this dataset and it makes a negative effect on the diffusion process. We will further validate our argument in the parameter analysis experiments.
- LightGCN reaches its best performance at the third layer on Yelp and at the second layer on Flickr and Ciao. DiffNet++ has a similar trend with LightGCN at each layer, even though it incorporates social information. CGL achieves its best results at the third layer on Yelp and Flickr and at the first layer on Ciao. The difference between the trend of CGL and the other two models may be attributed to two aspects. Firstly, when there exists few noise in the raw data of rating data and social links, a deeper diffusion layer can help the model extract useful information from them. Secondly, if there exists a lot of noise in the raw data, a deeper diffusion layer will have a negative effect on the model since noise will be diffused at each layer.

Lastly, we show the runtime of each model on three datasets in **Table 6** so as to evaluate the model performance thoroughly. The layer number is set as three for all GNN-based models to assure the fairness of comparison. From **Table 6**, we can see that the average time of our model for each epoch is about 25 s, which is much less than SEPT and SGL. Considering the excellent performance of CGL, we conclude it is efficient and effective.

4.3 Component Study

We analyze the impact of the readout strategy and the pointwise loss in the readout module of CGL in this section. For the readout strategy, CGL employs the average values of user embeddings from all layers in the diffusion process and item embeddings from the last layer, which is an asymmetrical manner for users and items and originates from the huge difference in number and sparsity of users and items. To validate the effectiveness of this asymmetrical readout strategy in CGL, its two variants are designed to compare their performance, which uses two common symmetrical readout strategies. One variant is denoted by CGL-M, which takes the average value of user and item embeddings from all layers in the diffusion process as readouts of users and items, respectively. The other is CGL-L, which treats embeddings from the last layer in the diffusion process as readouts of users and items. Experimental results of CGL and its variants on three datasets are given in **Table 7** ($K = 20$).

From **Table 7**, we can see that CGL outperforms CGL-M and CGL-L on all three datasets, which indicates the superiority of our asymmetrical readout strategy. Note that the structures of CGL

TABLE 7 | Comparison of CGL and its variants with different readout strategies (*K* = 20). The best results are in bold.

Models	Yelp			Flickr			Ciao		
	Prec	Rec	NDCG	Prec	Rec	NDCG	Prec	Rec	NDCG
CGL-M	0.442 3	4.943	2.240	0.244 9	1.152	0.723 1	1.186	5.827	3.730
CGL-L	0.456 0	5.025	2.294	0.297 1	1.438	0.858 4	**1.368**	**6.669**	**4.287**
CGL	**0.477 3**	**5.217**	**2.409**	**0.334 8**	**1.618**	**0.994 2**	**1.368**	**6.669**	**4.287**

TABLE 8 | Comparison of CGL and its variant with different pointwise loss positions (*K* = 20). The best results are in bold.

Models	Yelp			Flickr			Ciao		
	Prec	Rec	NDCG	Prec	Rec	NDCG	Prec	Rec	NDCG
CGL-P	0.471 0	5.206	2.386	0.333 9	1.616	0.987 9	1.333	6.577	4.260
CGL	**0.477 3**	**5.217**	**2.409**	**0.334 8**	**1.618**	**0.994 2**	**1.368**	**6.669**	**4.287**

TABLE 9 | Performance of CGL with respect to different values of α (*K* = 20). The best results are in bold.

	Yelp				Flickr				Ciao		
α	Pre	Rec	NDCG	α	Prec	Rec	NDCG	α	Prec	Rec	NDCG
10^{-4}	0.434 1	4.735	2.201	10^{-4}	0.297 1	1.578	9.201	10^{-5}	1.323	6.543	4.141
10^{-5}	0.465 1	**5.241**	2.358	10^{-5}	0.322 8	**1.651**	9.556	10^{-6}	1.339	6.579	4.291
10^{-6}	**0.477 3**	5.217	**2.409**	10^{-6}	**0.334 8**	1.618	**9.942**	10^{-7}	**1.368**	6.669	4.287
10^{-7}	0.470 5	5.232	2.376	10^{-7}	0.326 2	1.607	9.613	10^{-8}	1.363	**6.670**	4.304
10^{-8}	0.469 0	5.187	2.352	10^{-8}	0.326 2	1.598	9.522	10^{-9}	1.363	6.669	**4.307**

and CGL-L are identical when we fix the layer number at one, which explains why they have the same performance on Ciao (they achieve their best results at the first layer). Moreover, CGL exceeds CGL-M and CGL-L by a bigger margin on Flickr and Ciao than on Yelp. It is worth noting that the imbalance between the number of users and items in Flickr (8,358 *vs.* 82,120) and Ciao (7,317 *vs.* 104,975) is much more serious than that in Yelp (17,237 *vs.* 38,342). So, the bigger margin further demonstrates that the asymmetrical readout strategy in CGL can effectively alleviate the bad effect of the imbalance between users and items.

CGL has a pointwise loss $\mathcal{L}_{c-uv}$ in the readout module and a pairwise BPR loss in the prediction module, but existing models usually combine these two losses in the prediction module. To investigate the benefit of placing two losses separately in CGL, we transfer $\mathcal{L}_{c-uv}$ together with the BPR loss in the prediction module, take corresponding mode as a CGL variant, and denote it by CGL-P. Experimental results of CGL-P and CGL on all datasets are given in **Table 8** ($K = 20$).

From **Table 8**, we can see that CGL achieves better performance on all datasets with respect to all three metrics compared with CGL-P. This indicates that the performance of CGL is affected by the position of $\mathcal{L}_{c-uv}$ and that it is better to put it in the readout module. Actually, the pointwise $\mathcal{L}_{c-uv}$ loss and the pairwise BPR loss emphasize different aspects of user and item embeddings, and they should be placed in different positions according to different goals. The pointwise loss focuses on the entrywise consistency between user embeddings and item embeddings, which is used to provide accurate profiles of users and items and should be put before the aggregation in the prediction module. However, the pairwise loss focuses on the consistency between observed and unobserved interactions, which is used to improve ranking performance and should be set after the aggregation in the prediction module.

4.4 Parameter Analysis

Impacts of α, β, and embedding size d on CGL are analyzed, where α and β are two unique parameters of our model to control the strength of contrastive losses $\mathcal{L}_{c-uu}$ and $\mathcal{L}_{c-uv}$. Experimental results of CGL with different α, β, and d values are given in **Table 9**, **Table 10**, and **Table 11** ($K = 20$), respectively. For the simplicity of expression, we omit the coefficient of α (4.0 on Yelp, 2.0 on Ciao, and 6.5 on Flickr) in **Table 9** and the coefficient of β (1.6 on Yelp, 1.0 on Ciao, and 8.5 on Flickr) in **Table 10**.

From **Table 9**, we can see that on Yelp and Flickr, all three metrics achieve their peaks while tuning α in its parameter range. This illustrates that $\mathcal{L}_{c-uu}$ can have a stable influence on the model even though it faces different sparsities of social information in different datasets. When α is bigger than a certain number (10^{-5} for Yelp and Flickr and 10^{-7} for Ciao), the performance of CGL degrades. This phenomenon indicates that placing too much emphasis on the consistency between social behaviors and historical interactions will destroy the learning process of model parameters. When α is smaller than 10^{-6}, CGL generates worse performance on Yelp and Flickr as well. This is because social behaviors and historical interactions can provide supervised signals for each other, and it is

TABLE 10 | Performance of CGL with respect to different values of β (K = 20). The best results are in bold.

	Yelp				Flickr				Ciao		
β	Prec	Rec	NDCG	β	Prec	Rec	NDCG	β	Prec	Rec	NDCG
10^{-1}	0.4015	4.315	1.939	10^{-5}	**0.3348**	1.600	9.779	10^{-0}	1.108	5.063	3.365
10^{-2}	0.4651	5.155	2.313	10^{-6}	**0.3348**	**1.618**	**9.942**	10^{-1}	1.310	6.151	3.868
10^{-3}	**0.4773**	5.217	**2.409**	10^{-7}	0.3341	1.596	9.825	10^{-2}	**1.368**	**6.669**	**4.287**
10^{-4}	0.4734	5.230	2.373	10^{-8}	0.3305	1.595	9.822	10^{-3}	1.330	6.566	4.284
10^{-5}	0.4748	**5.279**	2.361	10^{-9}	0.3305	1.595	9.825	10^{-4}	1.312	6.501	4.219

TABLE 11 | Performance of CGL with respect to different embedding sizes d (K = 20). The best results are in bold.

	Yelp			Flickr			Ciao		
β	Prec	Rec	NDCG	Prec	Rec	NDCG	Prec	Rec	NDCG
8	0.4278	4.811	2.142	0.2671	1.297	7.584	1.064	5.532	3.001
16	0.4467	4.932	2.285	0.2937	1.420	8.664	1.196	5.898	3.338
32	0.4671	5.154	2.352	0.3185	1.549	9.211	1.274	6.386	3.898
64	**0.4773**	5.217	2.409	**0.3348**	**1.618**	**9.942**	1.368	6.669	4.287
128	**0.4773**	**5.240**	**2.421**	0.3245	1.589	9.555	**1.384**	**6.747**	**4.486**

unreasonable to set α to be too small. So it is important to find a suitable α value to balance the effect between two behaviors. However, on Ciao, CGL performs well when α is set to be small. This shows evidence that there exists too much noise in the dataset and explains the reason why DiffNet++ performs badly. In general, we suggest to carefully tune α in the range of [10^{-5}, 10^{-7}].

From **Table 10**, we observe that on Yelp all three metrics have relatively large fluctuation ranges while increasing β from 10^{-1} to 10^{-5} and Recall achieves its best result at a β value (10^{-5}) quite different from that of Precision and NDCG (10^{-3}). However, on Flickr and Ciao, all three metrics keep relatively stable, and they achieve their best results at the same β value. Moreover, β is tuned in quite different ranges on three datasets. CGL achieves the best performance at a larger β value on Ciao and Flickr and a smaller value on Yelp. The reason might be that these datasets have different sparsities in rating information. Compared with Yelp and Ciao, Flickr provides denser social information and interest information to train the model, which makes the model insensitive to parameter β and easy to train. On Yelp and Ciao, there is not enough social information or there exists too much noise, so the model depends more on the supervised signals in $\mathcal{L}_{c-uv}$. In general, a larger value is suggested for β for sparse datasets and a smaller value for dense datasets. By considering the overall performance, we suggest to carefully choose β in [10^{-2}, 10^{-3}] on spare datasets and in [10^{-5}, 10^{-6}] on dense datasets.

To investigate the influence of the embedding size d, we adjust its value from 8 to 128 and give the corresponding results in **Table 11** ($K = 20$). On Yelp and Ciao, CGL improves its performance gradually while increasing the value of d. This is easy to understand, since a larger embedding size corresponds to more powerful user and item representations. On Flickr, the model performance increases while d increases from 8 to 64 and then decreases. One possible reason is that Flickr has denser interest information and social links than the other two datasets, and a too large embedding size might result in overfitting in learning user and item representations. In a nutshell, we may set the embedding size to 64 to compromise the complexity and effectiveness.

5 CONCLUSIONS AND FUTURE WORK

In this work, we present a CGL-based model for social recommendation, which explores how to effectively combine social information and interest information in a contrastive way. Aiming to overcome the problem of imbalance of users and items, we design an asymmetrical readout strategy to get user embeddings and item embeddings. Besides, to make full use of social information and alleviate the problem of data sparsity, we also introduce a self-supervised loss and a supervised pointwise loss, respectively. We conduct multiple experiments on three real-world datasets, and the experimental performance verifies that our model is simpler but more powerful than other social recommendation models [19, 25].

Although our model improves the recommendation performance significantly, there still exist some limitations. For example, we only fuse social user embeddings and interest user embeddings in the diffusion module, which may limit the influence of items. Our model can utilize social information efficiently, but the model's performance degrades when the social information is not enough. In addition, the proposed model may be extended by modeling multiple auxiliary information [33], such as user reviews [44], knowledge bases [45], and temporal signals [46]. And supervised signals are mined from the auxiliary information, and then different losses are designed to drive the model training. Inspired by the effectiveness of adversarial learning, an adversarial process between different views can also be considered in the model. Although some works [34, 47] have explored adversarial learning in recommendation, they are complex and hard to train. A simple way to utilize adversarial learning in social recommendation is worth being investigated.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

YZ, JH, and JY conceived and designed the study and established the proposed model. JH and YZ wrote the algorithms and executed the experiments. YZ, JH, and JY made great contributions to the writing of the study. And ML provided valuable comments on the manuscript writing. All authors approved the submitted version.

REFERENCES

1. Wei Y, Wang X, Nie L, He X, Hong R, Chua T-S. Mmgcn. In: Proceedings of the 27th ACM International Conference on Multimedia (MM 2019); October 21-25, 2019; Nice, France (2019). p. 1437–45. doi:10.1145/3343031.3351034
2. Huang T, Dong Y, Ding M, Yang Z, Feng W, Wang X, et al. MixGCF. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD 2021); August 14-18, 2021; Virtual Event, Singapore (2021). p. 665–74. doi:10.1145/3447548.3467408
3. Wu L, Sun P, Hong R, Ge Y, Wang M. Collaborative Neural Social Recommendation. *IEEE Trans Syst Man Cybern, Syst* (2021) 51:464–76. doi:10.1109/TSMC.2018.2872842
4. Deng Z-H, Huang L, Wang C-D, Lai J-H, Yu PS. DeepCF: A Unified Framework of Representation Learning and Matching Function Learning in Recommender System. In: Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI 2019); January 27 - February 1, 2019; Hawaii, USA, 33 (2019). p. 61–8. doi:10.1609/aaai.v33i01.330161*Aaai*
5. Xue H-J, Dai X, Zhang J, Huang S, Chen J. Deep Matrix Factorization Models for Recommender Systems. In: Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI 2017); August 19-25, 2017; Melbourne, Australia (2017). p. 3203–9. doi:10.24963/ijcai.2017/447
6. Mao K, Zhu J, Wang J, Dai Q, Dong Z, Xiao X, et al. SimpleX. In: Proceedings of the 30th ACM International Conference on Information and Knowledge Management (CIKM 2021); November 1 - 5, 2021; Virtual Event, Queensland, Australia (2021). p. 1243–52. doi:10.1145/3459637.3482297
7. Scarselli F, Gori M, Ah Chung Tsoi AC, Hagenbuchner M, Monfardini G. The Graph Neural Network Model. *IEEE Trans Neural Netw* (2009) 20:61–80. doi:10.1109/TNN.2008.2005605
8. Song W, Xiao Z, Wang Y, Charlin L, Zhang M, Tang J. Session-Based Social Recommendation via Dynamic Graph Attention Networks. In: Proceedings of the 20th ACM International Conference on Web Search and Data Mining (WSDM 2019); February 11-15, 2019; Melbourne, Australia (2019). p. 555–63. doi:10.1145/3289600.3290989
9. Wu L, Sun P, Fu Y, Hong R, Wang X, Wang M. A Neural Influence Diffusion Model for Social Recommendation. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2019); July 21-25, 2019; Paris, France (2019). p. 235–44. doi:10.1145/3331184.3331214
10. Yu J, Yin H, Li J, Gao M, Huang Z, Cui L. Enhance Social Recommendation with Adversarial Graph Convolutional Networks. *IEEE Trans Knowl Data Eng* (2020) 1. doi:10.1109/TKDE.2020.3033673
11. Wang X, He X, Wang M, Feng F, Chua T-S. Neural Graph Collaborative Filtering. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2019); July 21-25, 2019; Paris, France (2019). p. 165–74. doi:10.1145/3331184.3331267
12. Ma H, Yang H, Lyu MR, King I. SoRec. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM 2008); October 26-30, 2008; Napa Valley, USA (2008). p. 931–40. doi:10.1145/1458082.1458205
13. Ma H, Zhou D, Liu C, Lyu MR, King I. Recommender Systems with Social Regularization. In: Proceedings of the 4th International Conference on Web Search and Web Data Mining (WSDM 2011); February 9-12, 2011; Hong Kong, China (2011). p. 287–96. doi:10.1145/1935826.1935877
14. Ma H, King I, Lyu MR. Learning to Recommend with Social Trust Ensemble. In: Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2009); July 19-23, 2009; Boston, USA (2009). p. 203–10. doi:10.1145/1571941.1571978
15. Devlin J, Chang M, Lee K, Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2019); June 2-7, 2019; Minneapolis, USA (2019). p. 4171–86. doi:10.18653/v1/n19-1423
16. Lan Z, Chen M, Goodman S, Gimpel K, Sharma P, Soricut R. ALBERT: A Lite BERT for Self-Supervised Learning of Language Representations. In: Proceedings of the 8th International Conference on Learning Representations (ICLR 2020); April 26-30, 2020; Addis Ababa, Ethiopia (2020).
17. van den Oord A, Li Y, Vinyals O. *Representation Learning with Contrastive Predictive Coding*. CoRR abs/1807.03748 (2018).
18. Gidaris S, Singh P, Komodakis N. Unsupervised Representation Learning by Predicting Image Rotations. In: Proceedings of 6th International Conference on Learning Representations (ICLR 2018); April 30-May 3, 2018; Vancouver, Canada (2018).
19. Yu J, Yin H, Gao M, Xia X, Zhang X, Viet Hung NQ. Socially-Aware Self-Supervised Tri-training for Recommendation. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD 2021); August 14-18, 2021; Virtual Event, Singapore (2021). p. 2084–92. doi:10.1145/3447548.3467340
20. Zhu D, Sun Y, Du H, Tian Z. *Self-Supervised Recommendation with Cross-Channel Matching Representation and Hierarchical Contrastive Learning*. CoRR abs/2109.00676 (2021).
21. Xia X, Yin H, Yu J, Shao Y, Cui L. Self-Supervised Graph Co-training for Session-Based Recommendation. In: Proceedings of the 30th ACM International Conference on Information and Knowledge Management (CIKM 2021); November 1 - 5, 2021; Virtual Event, Queensland, Australia (2021). p. 2180–90. doi:10.1145/3459637.3482388
22. He X, Deng K, Wang X, Li Y, Zhang Y, Wang M. LightGCN. In: Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval (SIGIR 2020); July 25-30, 2020; Virtual Event, China (2020). p. 639–48. doi:10.1145/3397271.3401063
23. Wu L, Chen L, Hong R, Fu Y, Xie X, Wang M. A Hierarchical Attention Model for Social Contextual Image Recommendation. *IEEE Trans Knowl Data Eng* (2020) 32:1854–67. doi:10.1109/TKDE.2019.2913394
24. Wu L, Li J, Sun P, Hong R, Ge Y, Wang M. DiffNet++: A Neural Influence and Interest Diffusion Network for Social Recommendation. *IEEE Trans Knowl Data Eng* (2021) 1. doi:10.1109/TKDE.2020.3048414
25. Yu J, Yin H, Li J, Wang Q, Hung NQV, Zhang X. Self-Supervised Multi-Channel Hypergraph Convolutional Network for Social Recommendation. In: Proceeding of the Web Conference 2021 (WWW 2021); April 19-23, 2021; Virtual Event, Ljubljana, Slovenia (2021). p. 413–24. doi:10.1145/3442381.3449844
26. Wu J, Wang X, Feng F, He X, Chen L, Lian J, et al. Self-Supervised Graph Learning for Recommendation. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2021); July 11-15, 2021; Virtual Event, Canada (2021). p. 726–35. doi:10.1145/3404835.3462862
27. Wang W, Feng F, He X, Nie L, Chua T-S. Denoising Implicit Feedback for Recommendation. In: Proceedings of the 14th ACM International Conference

on Web Search and Data Mining (WSDM 2021); March 8-12, 2021; Virtual Event, Israel (2021). p. 373–81. doi:10.1145/3437963.3441800

28. van den Berg R, Kipf TN, Welling M. *Graph Convolutional Matrix Completion*. CoRR abs/1706.02263 (2017).
29. Wang Z, Wang C, Gao C, Li X, Li X. An Evolutionary Autoencoder for Dynamic Community Detection. *Sci China Inf Sci* (2020) 63:1–16. doi:10.1007/s11432-020-2827-9
30. Jamali M, Ester M. A Matrix Factorization Technique with Trust Propagation for Recommendation in Social Networks. In: Proceedings of the 2010 ACM Conference on Recommender Systems (RecSys 2010); September 26-30, 2010; Barcelona, Spain (2010). p. 135–42. doi:10.1145/1864708.1864736
31. Yang B, Lei Y, Liu D, Liu J. Social Collaborative Filtering by Trust. In: Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI 2013); August 3-9, 2013; Beijing, China (2013). p. 2747–53. doi:10.1109/tpami.2016.2605085
32. Guo G, Zhang J, Yorke-Smith N. TrustSVD: Collaborative Filtering with Both the Explicit and Implicit Influence of User Trust and of Item Ratings. In: Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI 2015); January 25-30, 2015; Austin, USA (2015). p. 123–9.
33. He X, Liao L, Zhang H, Nie L, Hu X, Chua T. Neural Collaborative Filtering. In: Proceedings of the 26th International Conference on World Wide Web (WWW 2017); April 3-7, 2017; Perth, Australia (2017). p. 173–82. doi:10.1145/3038912.3052569
34. Fan W, Derr T, Ma Y, Wang J, Tang J, Li Q. Deep Adversarial Social Recommendation. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI 2019); August 10-16, 2019; Macao, China (2019). p. 1351–7. doi:10.24963/ijcai.2019/187
35. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al. Generative Adversarial Networks. In: Proceedings of the 27th International Conference on Neural Information Processing Systems; December 8-13, 2014; Montreal, Canada, 63 (2020). p. 139–44. doi:10.1145/3422622
36. Liu Y, Liang C, He X, Peng J, Zheng Z, Tang J. Modelling High-Order Social Relations for Item Recommendation. *IEEE Trans Knowl Data Eng* (2020) 1. doi:10.1109/TKDE.2020.3039463
37. Zhu J, Li X, Gao C, Wang Z, Kurths J. Unsupervised Community Detection in Attributed Networks Based on Mutual Information Maximization. *New J Phys* (2021) 23:113016. doi:10.1088/1367-2630/ac2fbd
38. Fan W, Ma Y, Li Q, He Y, Zhao E, Tang J, et al. Graph Neural Networks for Social Recommendation. In: Proceedings of the World Wide Web Conference (WWW 2019); May 13-17, 2019; San Francisco, USA (2019). p. 417–26. doi:10.1145/3308558.3313488
39. Yang L, Liu Z, Dou Y, Ma J, Yu PS. ConsisRec: Enhancing GNN for Social Recommendation via Consistent Neighbor Aggregation. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2021); July 11-15, 2021; Virtual Event, Canada (2021). p. 2141–5. doi:10.1145/3404835.34630280
40. Kipf TN, Welling M. Semi-Supervised Classification with Graph Convolutional Networks. In: Proceedings of the 5th International Conference on Learning Representations (ICLR 2017); April 24-26, 2017; Toulon, France (2017).
41. Tang J, Gao H, Liu H. mTrust. In: Proceedings of the 5th International Conference on Web Search and Web Data Mining (WSDM 2012); February 8-12, 2012; Seattle, WA, USA. Seattle, WA: ACM (2012). p. 93–102. doi:10.1145/2124295.2124309
42. Rendle S, Freudenthaler C, Gantner Z, Schmidt-Thieme L. BPR: Bayesian Personalized Ranking from Implicit Feedback. In: Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence (UAI 2009); June 18-21, 2009; Montreal, Canada (2009). p. 452–61.
43. Kingma DP, Ba J. Adam: A Method for Stochastic Optimization. In: Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015); May 7-9, 2015; San Diego, USA (2015).
44. He X, Chen T, Kan M-Y, Chen X. TriRank. In: Proceedings of the 24th ACM International Conference on Information and Knowledge Management (CIKM 2015); October 19 - 23, 2015; Melbourne, Australia (2015). p. 1661–70. doi:10.1145/2806416.2806504
45. Zhang F, Yuan NJ, Lian D, Xie X, Ma W-Y. Collaborative Knowledge Base Embedding for Recommender Systems. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2016); August 13-17, 2016; San Francisco, USA (2016). p. 353–62. doi:10.1145/2939672.2939673
46. Bayer I, He X, Kanagal B, Rendle S. A Generic Coordinate Descent Framework for Learning from Implicit Feedback. In: Proceedings of the 26th International Conference on World Wide Web (WWW 2017); April 3-7, 2017; Perth, Australia (2017). p. 1341–50. doi:10.1145/3038912.3052694
47. He X, He Z, Du X, Chua T-S. Adversarial Personalized Ranking for Recommendation. In: Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2018); July 08-12, 2018; Ann Arbor, USA (2018). p. 355–64. doi:10.1145/3209978.3209981

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

ORIGINAL RESEARCH
published: 30 March 2022
doi: 10.3389/fphy.2022.858225

Detecting Local Opinion Leader in Semantic Social Networks: A Community-Based Approach

Hailu Yang[1]*, *Qian Liu*[1]*, *Xiaoyu Ding*[2], *Chen Chen*[1] *and Lili Wang*[1]

[1]School of Computer Science and Technology, Harbin University of Science and Technology, Harbin, China, [2]School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing, China

Online social networks have been incorporated into people's work and daily lives as social media and services continue to develop. Opinion leaders are social media activists who forward and filter messages in mass communication. Therefore, competent monitoring of opinion leaders may, to some extent, influence the spread and growth of public opinion. Most traditional opinion leader mining approaches focus solely on the user's network structure, neglecting the significance and role of semantic information in the generation of opinion leaders. Furthermore, these methods rank the influence of users globally and lack effectiveness in detecting local opinion leaders with low influence. This paper presents a community-based opinion leader mining approach in semantic social networks to address these issues. Firstly, we present a new node semantic feature representation method and community detection algorithm to generate the local public opinion circle. Then, a novel influence calculation method is proposed to find local opinion leaders by combining the global structure of the network and local structure of the public opinion circle. Finally, nodes with high comprehensive influence are identified as opinion leaders. Experiments on real social networks indicate that the proposed method can accurately measure global and local influence in social networks, as well as increase the accuracy of local opinion leader mining.

Keywords: social networks, local opinion leader, influence calculation, semantic representation, community detection

OPEN ACCESS

Edited by:
Peican Zhu,
Northwestern Polytechnical University, China

Reviewed by:
Erik Cambria,
Nanyang Technological University, Singapore
Jie Cao,
Nanjing University of Finance and Economics, China

****Correspondence:***
Hailu Yang
yanghailu@hrbust.edu.cn
Qian Liu
529572578@qq.com

Specialty section:
This article was submitted to Social Physics, a section of the journal Frontiers in Physics

Received: *19 January 2022*
Accepted: *02 March 2022*
Published: *30 March 2022*

Citation:
Yang H, Liu Q, Ding X, Chen C and Wang L (2022) Detecting Local Opinion Leader in Semantic Social Networks: A Community-Based Approach. Front. Phys. 10:858225. doi: 10.3389/fphy.2022.858225

1 INTRODUCTION

The social network is a complex structure made up of people or entities who are linked together by some kind of relationship or shared interest (friendship, professional relationship, kinship, etc.) [1]. As tens of thousands of people around the world utilize social networks to interact, the Internet can continue to share an enormous quantity of data, resulting in the exponential expansion of social media and online social networks (e.g., Facebook, Twitter, Weibo) in recent years. Simultaneously, online social networks have grown in popularity as a result of their convenience, openness, anonymity, and virtual character, and have steadily evolved into a major carrier of online opinion and information distribution [2–5]. Texts, emojis, hashtags, and gif videos all contribute to the propagation of public opinion on social media [6]. As a result, online public opinion has evolved into a distinct form of public opinion with increasing social clout.

Most studies on online public opinion, including online opinion mining, dissemination patterns, and data mining, currently focus on the spread of online public opinion on social media [7–9]. The status of users in social networks is unequal, those in the center play a leading and driving role in the development of online public opinion, while those in the periphery are easily influenced by other factors (Aleahmad et al., 2016). Internet opinion leaders are often at the center, and their communication can easily push certain events to the forefront of the public opinion wave (Walter and Brüggemann, 2020). Public opinion leadership has the potential to not only actively guide public life, but also to trigger a wide range of negative emotions [12]. As a result, mining public opinion leaders is an important factor of guiding correct public opinion and sustaining network order.

Community detection is the process of dividing social users into tightly connected and highly relevant groups so that each group can be well separated from the others (Chunaev, 2020). Community detection has important applications in the fields of social network analysis, data mining, spatial database technology, statistics, biology, and smart grids [14–18]. This paper uses community detection to build a public opinion circle of users with the same ideas and opinions in social networks. Different from directly detecting opinion communities by leveraging connections between nodes [19], we use the semantic cohesiveness [20] and structural compactness of the community to further enhance the impact and effect of opinion leaders in the local environment.

Thanks to the development of online web technology, we can easily extract semantic information released by any individual from popular semantic social networks such as Facebook, Twitter, and Sina Weibo. At present, online opinion leaders detection mostly uses user behavior analysis [21,22], text semantic or sentiment analysis [23,24], node centrality analysis [25,26]. These methods, however, do not adequately account for the complexity of individuals in the local environment, and there is a lack of effective identification of opinion leaders with strong local influence.

To improve the performance of online opinion leader mining, we propose using the individual's local structure to weight the influence score when mining opinion leaders. Firstly, an LDA topic model is introduced to obtain the topic distribution of semantic information of users and complete the construction of social networks; then a community detection algorithm based on σ-norm is proposed to obtain the community structure of social networks and form multiple opinion circles; Finally, using the graph structure of the community, we propose an influence calculation method based on the global and local structure of the graph to detect the opinion leaders of social networks. Experiments on real social networks show that the method proposed in this paper can extract online opinion leaders accurately and effectively.

In summary, the contributions of this paper are as follows:

(1) We propose a new opinion leader mining method that considers both semantic information and network structure of users.

(2) We construct a new node semantic feature representation method by computing the similarity between user documents and global topics to map user semantic to topology spaces.

(3) A community detection algorithm based on σ-norm is proposed, which can accurately output high-quality community partition results by exploiting the robustness of l21-norm and F-norm.

(4) We present a new influence calculation method that combines the global and local structure of the graph, successfully avoiding the impact of global high-influence nodes in local influence calculation.

2 SEMANTIC INFORMATION DISCOVERY OF SOCIAL NETWORK

In social networks, users express their views or opinions in response to various message. We define the social network as $G = (V, E, D)$, where V is the set of nodes, E is the set of edges, D represents the semantic information. The semantic information published by user node $v \in V$ is $d \in D$. Meanwhile, we abstract semantic information into topics and topic keywords and use them as feature attributes of nodes. Afterward, the connections $e \in E$ are established based on the similarity of the topics to which the nodes belong. We use the LDA topic model to process node semantic information.

2.1 LDA Representation of Semantic Information

LDA (Latent Dirichlet Allocation) is a three-level Bayesian model for document generation, which considers an article as having multiple topics, and each topic corresponds to a different word. **Figure 1** shows the semantic information published by users in the social network, which contains three documents with the words marked with different colors. For example, words related to the biological environment are "coronavirus" and "vaccines," which are marked with green; words related to political life are "government" and "official," which are marked with yellow; words related to economy are "economy" and "opening," which are marked with blue. If all the words in the document are marked, it can be found that each post mixes different topics in different proportions. For example, the first post mixes bioenvironmental and political themes, and the bioenvironmental theme has a higher proportion. With this idea, the topic distribution of semantic information in social networks can be extracted and the exploration of semantic information can be realized.

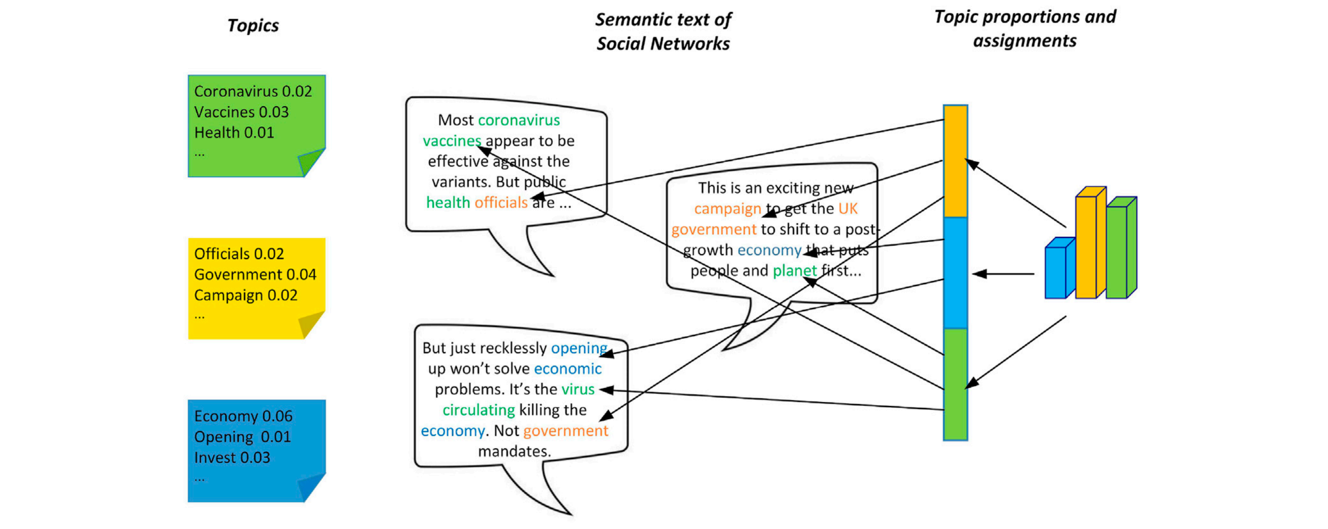
The mathematical notation involved in the LDA topic model is shown in **Table 1**, and it is generated for each node as follows:

(1) $\theta_d \sim Dirichlet(\alpha)$: The topic distribution θ_d of document d follows the Dirichlet distribution with hyperparameter α, where α determines the proportion of the distribution of topics in document d.

(2) $\beta_z \sim Dirichlet(\eta)$: The word distribution β_z of topic z follows the Dirichlet distribution with hyperparameter η, where η determines the proportion of words distributed in the topic.

(3) $z_i \mid \theta_d \sim Multinomial(\theta_d)$: The topic number z_i follows a polynomial distribution under the topic distribution θ_d.

(4) $w_i \mid z_i, \beta_{z_i} \sim Multinomial(\beta_{z_i})$: The probability of occurrence of keyword w_i in topic z_i follows a polynomial distribution under word distribution β_{z_i}.



Most coronavirus vaccines appear to be effective against the variants. But public health officials are ...

This is an exciting new campaign to get the UK government to shift to a post-growth economy that puts people and planet first...

But just recklessly opening up won't solve economic problems. It's the virus circulating killing the economy. Not government mandates.

FIGURE 1 | LDA model for document generation process of semantic information published by users in social networks.

TABLE 1 | Description of notations.

Notation	Description
θ_d	Topic distribution probability of document d
$\vec{\theta}_d$	The vector of topic distribution probability
β_{z_i}	Keywords distribution probability of topic z_i
$\vec{\beta}_z$	The vector of keywords distribution probability
w_i	The *i*th keyword in vector $\vec{w}$
$\vec{w}$	The vector of keywords
z_i	The *i*th topic in vector $\vec{z}$
$\vec{z}$	The vector of topics
$\|D\|$	Total number of documents
T	The number of topics in total documents
H	The number of keywords in a topic distribution probability
α	priori parameter over topic distribution probability specific to a document
$\vec{\alpha}$	a vector of priori parameter to each document
η	priori parameter over keyword distribution probability specific to a topic
$\vec{\eta}$	a vector of priori parameter to each topic

In summary, n documents will correspond to n independent Dirichlet-Multinomial conjugate structures, and K topics will correspond to K independent Dirichlet-Multinomial conjugate structures. Use α to generate topic distribution θ, and topic distribution θ determines the specific topic. Use η to generate word distribution β, which determines the specific keyword, i.e

$$\begin{gathered}\vec{\alpha} \rightarrow \vec{\theta}_d \rightarrow \vec{z} \\ \vec{\eta} \rightarrow \vec{\beta}^{z_i} \rightarrow \vec{w}\end{gathered} \tag{1}$$

2.2 Gibbs Sampling Process

Gibbs sampling is a Markov-Chain-Monte-Carlo (MCMC) method and is widely used in probability inference (Su et al., 2018). Gibbs sampling approximately samples a group of random variables from a complex joint distribution to obtain the conditional probability distribution of each characteristic dimension. Specifically for the LDA model, our goal is to obtain the overall probability distribution $\vec{z}$ and $\vec{w}$, corresponding to each z_i and w_i, i.e., topic distribution of documents and word distribution of topics.

Using the relationship existing in **Eq. 1**, the joint probability distribution $p(\vec{w}, \vec{z}|\vec{\alpha}, \vec{\eta})$ of topics and words can be obtained as follows:

$$\begin{aligned} p\left(\vec{w}, \vec{z}\right) &\propto p\left(\vec{w}, \vec{z}|\vec{\alpha}, \vec{\eta}\right) \\ &= p\left(\vec{z}|\vec{\alpha}\right)p\left(\vec{w}|\vec{z}, \vec{\eta}\right) = \prod_{d=1}^{|D|} \frac{\Delta\left(\vec{n}_d + \vec{\alpha}\right)}{\Delta\left(\vec{\alpha}\right)} \prod_{t=1}^{T} \frac{\Delta\left(\vec{n}_t + \vec{\eta}\right)}{\Delta\left(\vec{\eta}\right)} \end{aligned} \tag{2}$$

Where $\Delta(\vec{\alpha})$, $\Delta(\vec{\eta})$ are the normalization parameters, $\vec{n}_d = (n_d^{(1)}, n_d^{(2)}, \ldots, n_d^{(T)})$, $n_d^{(t)}$ is the number of occurrences of the word for the tth topic in the dth document; $\vec{n}_t = (n_t^{(1)}, n_t^{(2)}, \ldots, n_t^{(H)})$, $n_t^{(h)}$ is the number of occurrences of the h-th word in the t-th topic. Given the conditional distribution $p(\vec{z}|\vec{w})$ of the observable variable $\vec{w}$ under the hidden content $\vec{z}$, Bayesian analysis can be performed on it using the joint distribution (**Eq. 2**). The Bayesian relationship between $\vec{z}$ and $\vec{w}$ is expressed as:

$$\begin{aligned} & p\left(z_i = o|\vec{z}_{\neg i}, \vec{w}\right) \propto p\left(z_i = o, w_i = y|\vec{z}_{\neg i}, \vec{w}_{\neg i}\right) \\ & = \int p\left(z_i = o, \vec{\theta}_d \mid \vec{w}_{\neg i}, \vec{z}_{\neg i}\right) p\left(w_i = y, \vec{\beta}_{z_i} \mid \vec{w}_{\neg i}, \vec{z}_{\neg i}\right) d\vec{\theta}_d d\vec{\beta}_{z_i} \\ & = \int p\left(z_i = o \mid \vec{\theta}_d\right) p\left(\vec{\theta}_d \mid \vec{w}_{\neg i}, \vec{z}_{\neg i}\right) p\left(w_i = y \mid \vec{\beta}_{z_i}\right) p\left(\vec{\beta}_{z_i} \mid \vec{w}_{\neg i}, \vec{z}_{-i}\right) d\vec{\theta}_d d\vec{\beta}_{z_i} \end{aligned} \tag{3}$$

When $z_i = o$, $w_i = y$, the probability $p(z_i = o, w_i = y|\vec{z}_{\neg i}, \vec{w}_{\neg i})$ only involves the conjugate distribution of the dth document and the tth topic under the Dirichlet-Multinomial model. where y is one of the keywords in w; o corresponding to y, is one of the topics in z; $\vec{w}_{\neg i}, \vec{z}_{\neg i}$ represents the corresponding topic distribution and word distribution after removing topics and words with subscript *i*; the posterior distributions of $\vec{\theta}_d$ and $\vec{\beta}_{z_i}$ can be calculated by the following equation:

$$\begin{aligned} p\left(\vec{\theta}_d \mid \vec{w}_{\neg i}, \vec{z}_{\neg i}\right) = Dirichlet\left(\vec{\theta}_d \mid \vec{n}_{d,\neg i} + \vec{\alpha}\right) \\ p\left(\vec{\beta}_{z_i} \mid \vec{w}_{\neg i}, \vec{z}_{\neg i}\right) = Dirichlet\left(\vec{\beta}_{z_i} \mid \vec{n}_{t,\neg i} + \vec{\eta}\right) \end{aligned} \tag{4}$$

Thus, **Eq. 3** can be reduced to:

$$\int p(z_i = o|\vec{\theta}_d) Dirichlet\,(\vec{\theta}_d|\vec{n}_{d,\neg i} + \vec{\alpha})d\vec{\theta}_d \cdot \int p\left(w_i = y \mid \vec{\beta}_{z_i}\right)$$

$$Dirichlet\left(\vec{\beta}_{z_i}|\vec{n}_{t,\neg i} + \vec{\eta}\right)d\vec{\beta}_{z_i} = \frac{n^o_{d,\neg i} + \alpha_o}{\sum_{t=1}^{T} n^t_{d,\neg i} + \alpha_t} \cdot$$

$$\frac{n^y_{k,\neg i} + \eta_y}{\sum_{h=1}^{H} n^h_{k,\neg i} + \eta_h} \Rightarrow p(z_i = o|\vec{w}, \vec{z}_{\neg i}) \propto \frac{n^o_{d,\neg i} + \alpha_o}{\sum_{t=1}^{T} n^t_{d,\neg i} + \alpha_t} \cdot \frac{n^y_{k,\neg i} + \eta_y}{\sum_{h=1}^{H} n^h_{k,\neg i} + \eta_h} \tag{5}$$

Where $\alpha_o(\alpha_t)$ is the hyperparameter α of the topic distribution corresponding to the topic of $o(t)$. $\eta_y(\eta_h)$ is the hyperparameter η of the word distribution corresponding to the keyword of $y(h)$. $n^k_{d,-i}$ is the number of topics when $z_i = o$. $n^t_{k,\neg i}$ is the number of keywords when $w_i = y$.

Gibbs sampling is performed on the topics of all words by **Eq. 5**, and when the sampling converges, the topics corresponding to all words are obtained; then, according to the correspondence between the sampled words and topics, we can get the topic distribution θ_d of each document and the distribution β_k of keywords in each topic.

3 COMMUNITY DETECTION BASED ON TOPIC DISTRIBUTION

3.1 Node Representation

In this paper, the semantic information corresponding to the user nodes v_i in the social network is used as the document d_i to generate the topic distribution θ_{d_i}. Therefore, each node is represented as a *K*-dimensional vector and is equal to the topic probability distribution of the node corresponding to the document. The set of all node vectors is formed into a data matrix *X* of $K \times n$ to implement the node representation, where the matrix *X* is calculated as follows:

$$x_{i,j} = \begin{cases} 0, & z_j = 0 \\ \theta^{z_j}_{d_i}, & z_j \neq 0 \end{cases} \tag{6}$$

In **Eq. 6**, $x_{i,j}$ denotes the value of the *i*th row and *j*th column of the data matrix *X*, and $\theta^{z_j}_{d_i}$ denotes the probability that document d_i belongs to the *j*th topic. Therefore, when the probability of the *j*th topic in the topic distribution θ_{d_i} is zero, $X = 0$; when the probability of the *j*th topic in the topic distribution θ_{d_i} is non-zero, $x_{i,j} = \theta^{z_j}_{d_i}$, which constitutes the user node vector, and the data matrix *X* is obtained to complete the node representation.

3.2 Establishing Associations

Calculating the similarity between node vectors can establish association for nodes. Two users with high correlation in a social network will correspond to a large similarity value, low correlation users will correspond to a low similarity value, and uncorrelated users will have zero similarity value. Commonly used similarity calculation methods include cosine similarity, Pearson correlation coefficient, and Gaussian kernel similarity calculation methods. These methods are sensitive to noise and outliers and are easy to ignore the local structure of data and the size of the vector itself. Therefore, this paper chooses a data similarity matrix learning method based on sparse representation [28] that is robust to noise kernel outliers in the data [29] and fits the requirement of connecting social network users. We can obtain the similarity matrix between users in a social network by solving the following equation:

$$\begin{aligned} &\min_{a_{i,j}} a_{i,j} \left\|\vec{x}_i - \vec{x}_j\right\|_2^2 + \varepsilon \sum_i^n \sum_j^n a_{i,j} \\ &s.t. \mathbf{1}^T \vec{a}_i = 1, a_{i,i} = 0, a_{i,j} \geq 0 \end{aligned} \tag{7}$$

Where $a_{i,j}$ is the value of the *i*th row and *j*th column of the similarity matrix *A*, $A \in \boldsymbol{R}^{n \times n}$. *n* is the number of users in the social network. $\vec{a}_i$ is the vector of the *i*th row of *A*, which represents the similarity value between user *i* and other users. ε is the sparse adjustment factor. **1** is a vector with all values of 1, constraint $\mathbf{1}^T \vec{a}_i = 1$ makes the second term in **Eq. 7** to be constant. That is, the constraint $\mathbf{1}^T \vec{a}_i = 1$ is equivalent to a sparse constraint on *A*.

After calculation and derivation, the following results can be obtained:

$$\hat{a}_{i,j} = \begin{cases} \dfrac{c_{i,m+1} - c_{i,j}}{m c_{i,m+1} - \sum\limits_{h=1}^{m} c_{i,h}} & j \leq m \\ 0 & j > m \end{cases} \tag{8}$$

Where $c_{i,j} = \|x_i - x_j\|_2^2$, Sort them from small to large so that the learning of $c_{i,j}$ satisfies $\hat{c}_{i,m} > 0$, and $\hat{c}_{i,m+1} = 0$. *m* is the number of adaptive neighbors. The similarity matrix calculated using cosine similarity, Pearson correlation coefficient, and other methods is usually presented in the form of fully connected or *K*-nearest neighbors. The similarity matrix *A* calculated by **Eq. 8** can adapt to the number of neighbors *m* of users in the social network, compensating for the disadvantage that community detection requires high node similarity. This will improve the quality of the community structure and, as a result, accurately detect network opinion leaders.

3.3 Constructing Community Detection Algorithm

Loss function is usually constructed using the l1-norm and the l2-norm. The loss function constructed using the l1-norm has the disadvantage of being insensitive to larger outliers but sensitive to smaller ones; the l2-norm does the opposite. σ-norm [30] neutralizes the above two problems and is defined as follows:

$$\|\vec{x}\|_\sigma = \sum_i^n \frac{(1+\sigma)x_i^2}{x_i^2 + \sigma} \tag{9}$$

Where σ is the adaptive parameter. The generalization of the vector $\vec{x}$ into matrix X is equivalent to neutralizing the l21-norm and F-norm of the matrix. Thus, the σ-norm takes advantage of the robustness of the l21-norm and F-norm precisely for both large and small outliers, and $\|X\|_\sigma$ is nonnegative, convex, and quadratically differentiable.

$$\|X\|_\sigma = \sum_i^n \frac{(1+\sigma)\|\vec{x}_i\|_2^2}{\|\vec{x}_i\|_2 + \sigma} \tag{10}$$

After constructing the similarity matrix A of the social network by **Eq. 8**, we introduce the rank constraint and propose the following objective function:

$$\begin{aligned} &\min_U \|A - U\|_\sigma \\ &s.t.\mathbf{1}^T \vec{u}_i = 1, u_{i,j} \geq 0, \operatorname{rank}(L) = n - k \end{aligned} \tag{11}$$

In **Eq. 11**, U is the target matrix obtained from learning, due to the introduction of rank constraint, the target matrix U will have k connected components, so it can directly output the k community structures of the social network; L is the Laplace matrix of U; $L = R - S$, where R is diagonal matrix, $r_{ii} = \sum_{j=1}^n u_{i,j}$; and $S = (U^T + U)/2$; the constraint $\mathbf{1}^T \vec{u}_i = 1$ is set to avoid anomalous nodes (without any neighbors), so that the sum of each row of U is 1.

However, the dependence of L on the variable S and the nonlinearity of the rank constraint leads to the difficulty in solving **Eq. 11**. To solve this puzzle, we define $\lambda_i(L)$ to denote the ith smallest eigenvalue of L. Since the matrix L is a symmetric semi-positive definite matrix, the eigenvalues of L are real and non-negative [31], so there exists $\lambda_i(L) \geq 0$. Then, if the first k smallest eigenvalues of L satisfy $\sum_{i=1}^k \lambda_i(L) = 0$, the rank constraint rank(L) = $n - k$ is achieved, and **Eq. 11** can be expressed as:

$$\begin{aligned} &\min_S \|A - U\|_\sigma + \rho \sum_{i=1}^k \lambda_i(L) \\ &s.t.\mathbf{1}^T \vec{u}_i = 1, u_{i,j} \geq 0 \end{aligned} \tag{12}$$

Where ρ is a balancing factor that can increase or decrease its value to cope with the cases that the connected components of the target matrix U are greater or less than k until k connected components of U exist. According to Fan's study [32], there is the following theorem:

$$\begin{aligned} &\sum_{i=1}^k \lambda_i(L) = \min \operatorname{Tr}\left(F^T L F\right) \\ &s.t.F^T F = I \end{aligned} \tag{13}$$

Where $F = \{\vec{f}_1, \vec{f}_2, \ldots, \vec{f}_k\}$ is the clustering indicator matrix, which is used to output clustering results in spectral clustering; I is the identity matrix. Substituting **Eq. 13** into **Eq. 12** gives:

$$\begin{aligned} &\min_{U,F} \|A - U\|_\sigma + \rho \operatorname{Tr}\left(F^T L F\right) \\ &s.t.\mathbf{1}^T \vec{u}_i = 1, u_{i,j} \geq 0, F^T F = I \end{aligned} \tag{14}$$

Eq. 14 is the final objective function, where the objective matrix U has k connected components, so that the final community detection results can be obtained directly using this algorithm.

3.4 Algorithm Optimization

We introduce an iterative optimization algorithm for solving **Eq. 14** and the target variable U therein. Since the target variable U and other variables F are coupled in one equation, solving **Eq. 14** and deriving all variables at once is a challenging problem. In addition, the constraints in the objective function are not smooth. Assuming that A, F has been obtained, the target variable U can be computed using ALM (Augmented Lagrange Multiplier). ALM performs superiorly on matrix analysis problems [33]. Similarly, when the variable U is fixed, F can be computed. The detailed computational strategy is as follows:

(1) Keep F fixed, update U.

When F is fixed, using the Laplace matrix nature $\sum_{i,j} \frac{1}{2}\|\vec{f}_i - \vec{f}_j\|_2^2 s_{i,j} = \operatorname{Tr}(F^T L F)$. The **Eq. 14** is rewritten as:

$$\begin{aligned} &\min_{U,F} \|A - U\|_\sigma + \rho \sum_{i,j} \frac{1}{2}\left\|\vec{f}_i - \vec{f}_j\right\|_2^2 u_{i,j} \\ &s.t.\mathbf{1}^T \vec{u}_i = 1, u_{i,j} \geq 0 \end{aligned} \tag{15}$$

Define the matrix $Q \in \boldsymbol{R}^{n\times n}$, where $\vec{e}_i \in \boldsymbol{R}^{n\times 1}$ is the ith column of Q and its jth element is $q_{i,j} = \|\vec{f}_i - \vec{f}_j\|_2^2$. Since each row in U has independence and according to the work of Nie et al [34], **Eq. 15** can be written in vector form as:

$$\begin{aligned} &\min_{\vec{u}_i} s_i \|\vec{a}_i - \vec{u}_i\|_2^2 + \rho \vec{u}_i^{\mathrm{T}} \vec{q}_i \\ &s.t.\mathbf{1}^T \vec{u}_i = 1, \vec{u}_i \geq 0 \end{aligned} \tag{16}$$

Where $\vec{u}_i$ is the column vector consisting of the elements of the ith row of the target matrix U; $\vec{a}_i$ is the column vector consisting of the elements of the ith row of the similarity matrix A; s_i is taken as:

$$s_i = (1+\sigma)\frac{\|\vec{a}_i - \vec{u}_i\|_2 + 2\sigma}{2\left(\|\vec{a}_i - \vec{u}_i\|_2 + \sigma\right)^2} \tag{17}$$

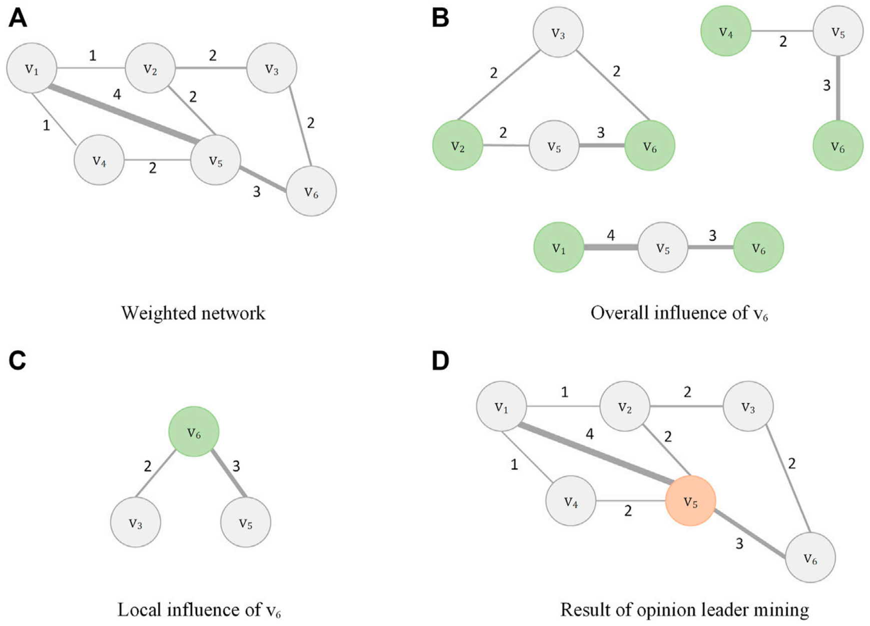
The simplification of **Eq. 16** yields:

$$\begin{aligned} &\min_{\vec{u}_i} \frac{1}{2} s_i \vec{u}_i^{\mathrm{T}} \vec{u}_i - \vec{u}_i^{\mathrm{T}}\left(s_i \vec{a}_i - \frac{\rho}{2}\vec{q}_i\right) \\ &s.t.\mathbf{1}^T \vec{u}_i = 1, \vec{u}_i \geq 0 \end{aligned} \tag{18}$$

Let $\vec{h}_i = s_i \vec{a}_i - \frac{\rho}{2}\vec{q}_i$, and using ALM we have $\ell(\vec{u}_i, \varphi, \xi) = \frac{1}{2} s_i \vec{u}_i^{\mathrm{T}} \vec{u}_i - \vec{u}_i^{\mathrm{T}} \vec{h}_i - \varphi(\mathbf{1}^T \vec{u}_i - 1) - \xi^T \vec{u}_i$, where ξ is a Lagrangian coefficient vector and ξ is a scalar.

Suppose the optimal solution to **Eq. 18** is $\hat{u}_i$, and the corresponding Lagrange multipliers are $\hat{\varphi}$ and $\hat{\xi}$ respectively. According to the Karush-Kuhn-Tucker conditions, we have:

$$\begin{cases} \forall j, s_i \hat{u}_{i,j} - h_{i,j} - \hat{\varphi} - \hat{\xi}_j = 0 \\ \forall j, \hat{\varphi} \geq 0 \\ \forall j, \hat{\xi}_j \geq 0 \\ \forall j, \hat{u}_{i,j} \hat{\xi}_j = 0 \end{cases} \tag{19a}$$



Weighted network

Overall influence of v₆

Local influence of v₆

Result of opinion leader mining

FIGURE 2 | An example of online opinion leader mining method based on global and local influence, **(A)** is weighted network. **(B)** is overall influence of v_6, **(C)** is local influence of v_6, **(D)** is result of opinion leader mining.

Equation 19 written in vector form has $s_i\hat{u}_i - h_i - \hat{\varphi}\mathbf{1} - \hat{\xi} = 0$. Due to the constraint $\mathbf{1}^T\vec{u}_i = 1$, we have $\hat{\varphi} = \frac{s_i - \mathbf{1}^T\vec{h}_i - \mathbf{1}^T\hat{\xi}}{n}$. Thus, the optimal solution $\hat{u}_i$ is formulated as:

$$\hat{u}_i = \frac{\vec{h}_i}{s_i} + \frac{1}{n} + \frac{\mathbf{1}^T\vec{h}_i\mathbf{1}}{ns_i} - \frac{\mathbf{1}^T\hat{\xi}\mathbf{1}}{ns_i} + \frac{\hat{\xi}}{s_i} \tag{20}$$

We further denote $\vec{g} = \frac{\vec{h}_i}{s_i} + \frac{1}{n} + \frac{\mathbf{1}^T\vec{h}_i\mathbf{1}}{ns_i}$ and $\hat{\xi}^* = \frac{\mathbf{1}^T\hat{\xi}}{ns_i}$. As a result, **Eq. 20** for $\forall j$ has:

$$\hat{u}_{i,j} = \vec{g}_j - \hat{\xi}^* + \frac{\hat{\xi}_j}{s_i} \tag{21}$$

According to **Eqs 19**, **21**, we know that the optimal solution $\hat{u}_{i,j} = \max(g_j - \hat{\xi}^*, 0)$. That is, the optimal solution $\hat{u}_{i,j}$ can be obtained if $\hat{\xi}^*$ is know. Furthermore, we can derive $\hat{\xi}_j = s_i(\hat{u}_{i,j} - \vec{g}_j + \hat{\xi}^*)$ from **Eq. 21**. Similarly, according to **Eq. 19**, we then have:

$$\hat{\xi}_j = s_i \max\left(\hat{\xi}^* - g_j, 0\right) \tag{22}$$

As denoted above $\hat{\xi}^* = \frac{\mathbf{1}^T\hat{\xi}}{ns_i}$, the optimal solution $\hat{\xi}^*$ is represented as $\hat{\xi}^* = \frac{1}{n}\sum_{j=1}^{n}\max(\hat{\xi}^* - g_j, 0)$. Now we define a function $f(\xi^*) = \frac{1}{n}\sum_{j=1}^{n}\max(\xi^* - g_j, 0) - \xi^*$ with respect to $\xi\star$. As can be seen, $\hat{\xi}^*$ is determined by solving the root finding problem when $f(\hat{\xi}^*) = 0$. Since $\xi\star \geq 0$, $f'(\xi^*) \leq 0$ and $f'(\xi^*) \leq 0$ are piece-wise linear and convex functions, the roots of $f'(\xi^*) = 0$ can be computed *via* the Newton method efficiently, shown below:

$$\xi^*_{t+1} = \xi^*_t - \frac{f(\xi^*_t)}{f'(\xi^*_t)} \tag{23}$$

(2) Keep U fixed, update F.

When U is fixed, it is equivalent to solving the following problem:

$$\begin{aligned}&\min_F \mathrm{Tr}\left(F^T LF\right)\\&s.t. F^T F = I\end{aligned} \tag{24}$$

The study in [31] indicates that the optimal solution to F is formed by the k eigenvectors of L corresponding to the k smallest eigenvalues.

The stopping condition for algorithm optimization is that the relative change in U is less than 10^{-3} or the number of iterations is greater than 150. Compared with other traditional community detection algorithms, our proposed community detection algorithm based on σ-norm requires the computation of **Eq. 14**. The time complexity of **Eq. 14** is $O(itn^2)$, where it is the number of iterations; $it \ll n$. Therefore, the time complexity of the proposed community detection algorithm is $O(n^2)$. The process of community detection for semantic social networks has been given above, and the whole framework is shown in Algorithm 1.

Algorithm 1. A community analysis approach to semantic social networks

```
Input: Social network $G$; Number of nearest neighbors $m$; Number of communities $k$; Number of topics
    $T$ ; Initialization parameters $\rho$, $\sigma$.
Output: The objective matrix $U$ with $k$ connected components.
Clean and filter the semantic information representing users in social network $G$;
Using LDA topic model to obtain the topic distribution $\theta_d$ of Social Network $G$;
Complete the node representation of the social network by Eq.6;
Form the node vector $x_i$ into a data matrix $X$;
According to Eq.8, the similarity matrix $A$ of social network $G$ is calculated;
Initialize the target matrix $U$;
Use Eq.24 to calculate the matrix $F$;
repeat
    Fix $F$, use Eq.21 to update objective matrix $U$;
    Fix $U$, the matrix consisting of the eigenvectors corresponding to the first $k$ smallest eigenvalues
    of the Laplacian matrix $L$ to update $F$;
until the relative change in $U$ is less than $10^{-3}$ or the number of iterations is greater than 150;
return The objective matrix$U$ containing the $k$ connected components.
```

TABLE 2 | Overall influence results for weighted network *G*.

Node	$\omega(v_i)$	v_i with $\lvert PN(v_i, v_j)\rvert$ for each node	Overall influence
v_1	6	0,1,1,1,2,1	33.66
v_2	5	1,0,0,2,1,2	23.10
v_3	4	1,0,0,0,2,0	9.24
v_4	3	1,2,0,0,1,1	13.53
v_5	11	2,1,2,1,0,0	50.82
v_6	5	1,2,0,1,0,0	17.05

TABLE 3 | Local influence results for weighted network *G*.

Node	$p'(v_i)$	$C_D^{\omega}(v_i)$	Local influence
v_1	0.32	4.24	5.19
v_2	0.24	3.87	6.31
v_3	0.40	2.83	1.98
v_4	0.18	2.45	5.18
v_5	0.58	6.63	3.75
v_6	0.33	3.16	4.97

TABLE 4 | The influence score for each node.

Node	Influence score
v_1	174.85
v_2	145.79
v_3	18.26
v_4	70.09
v_5	190.48
v_6	84.77

4 OPINION LEADERS MINING IN SOCIAL NETWORK

4.1 Definitions

Before explaining the opinion leader mining approach, we formalize some definitions that will be used subsequently. In **Section 2** we define the social network as $G = (V, E, D)$, where V is the set of nodes; E is the set of edges; D is the semantic information. The community structure A^k (k is the number of communities; A^k is the weighted networks) can be obtained by the community detection algorithm in **Section 3**. If v_i, $v_j \in V$, $\exists a_{i,j} \neq 0$, then v_i, v_j are adjacent, i.e. $\exists e_{v_i,v_j} \in E$. **Figure 2A** is an example of weighted network to explain the following definitions.

Definition 1 (Node neighborhood). *The neighborhood of node v_i is a node set composed of the neighbors of v_i. The neighborhood of node v_i denoted as $M(v_i)$ is defined as follows:*

$$M(v_i) = \left\{v_j \middle| v_j \in V, \exists e_{v_i,v_j} \in E\right\}, v_i \in V \tag{25}$$

In **Figure 2A**, nodes v_2, v_4 and v_5 are neighbors of node v_1. Thus, $M(v_1) = \{v_2, v_4, v_5\}$.

Definition 2 (public neighbor). *The nodes v_i, v_j represent two different nodes in the network G. The public neighbor nodes between these two nodes are represented by $PN(v_i, v_j)$, which is defined as follows:*

$$PN(v_i, v_j) = \left\{v_k \in V, v_k = M(v_i) \cap M(v_j)\right\}, v_i, v_j \in V \tag{26}$$

In **Figure 2A**, the neighbors of node v_1 and v_5 are $M(v_1) = \{v_2, v_4\}$ and $M(v_3) = \{v_2, v_6\}$, respectively. Thus, $PN(v_1, v_5) = \{v_2\}$.

Definition 3 (Sum of Weights). *The sum of weights is an extension of the degree and is usually used when analyzing weighted networks [35]. The sum of weights of v_1 denoted as $\omega(v_i)$ is defined as follows:*

$$\omega(v_i) = \sum_{v_j = M(v_i)} a_{i,j} \tag{27}$$

In **Figure 2A**, The sum of weights for the set of nodes $\{v_1, v_2, v_3, v_4, v_5, v_6\}$ are $\{6, 5, 4, 3, 11, 5\}$.

Definition 4 (Degree Centrality). *Degree centrality is the most direct metric for portraying node centrality in network analysis and the simplest measure of node influence, denoted by $C_D(v_i)$ and defined as follows:*

$$C_D(v_i) = \frac{d(v_i)}{n-1} \tag{28}$$

Where n is the total number of nodes and $d(v_i)$ is the degree of the node v_i.

In **Figure 2A**, the degree centrality of node v_2 is 0.6 and the degree centrality of node v_5 is 0.8. Therefore, the influence of node v_2 is higher than v_5 analyzed from the perspective of degree centrality.

Definition 5 (Comprehensive Node Centrality). *Comprehensive node centrality is an extension of degree centrality that considers not only the number of connections between nodes but also the degree of participation of nodes in the network, i.e., a node centrality measure that combines degrees and weights [36]. Denoted by $C_D^{\omega}(v_i)$, it is defined as follows:*

$$C_D^{\omega}(v_i) = d(v_i) \times \left(\frac{\omega(v_i)}{d(v_i)}\right)^{\tau} = d(v_i)^{(1-\tau)}\omega(v_i)^{\tau} \tag{29}$$

Where $d(v_i)$ is the degree of node v_i; $\omega(v_i)$ is the sum of the weights of node v_i; τ is the positive tuning parameter (default $\tau = 1.1$), which can be set on a situational basis. If τ is between 0 and 1, then it is favorable for nodes with high degree, while if τ is set above 1, then it is favorable for nodes with low degree.

In **Figure 2A**, the comprehensive node centrality of node v_3 is 2.83 and the degree centrality of node v_6 is 3.16. Therefore, the influence of node v_6 is higher than v_3 analyzed from the perspective of comprehensive node centrality.

Definition 6 (Average Degrees). *The average degree of node v_i is the sum of the degrees of all neighboring nodes of v_i over the degree of v_i, denoted by $\bar{d}(v_i)$, which is defined as follows:*

$$\bar{d}(v_i) = \frac{\sum_{v_j = M(v_i)} d(v_j)}{d(v_i)} \tag{30}$$

Where $d(v_i)$ and $d(v_j)$ are the degrees of nodes v_i and v_j; $M(v_i)$ is the set of neighboring nodes of v_i.

Definition 7 (Average Weights). *The Average Weight of node v_i is the sum of the weights of all neighboring nodes of v_i over the weight of v_i, denoted by $\bar{\omega}(v_i)$, which is defined as follows:*

$$\bar{\omega}(v_i) = \frac{\sum_{v_j = M(v_i)} \omega\left(v_j\right)}{\omega(v_i)} \tag{31}$$

Where $\omega(v_i)$ and $\omega(v_j)$ are the weights of nodes v_i and v_j; $M(v_i)$ is the set of neighboring nodes of v_i.

Definition 8 (Contribution Probability). *The influence of the node v_i itself is measured by its location in the network. In the unweighted network, we take the inverse of the average degree as the probability that neighbor nodes contribute to the influence of node v_i, which is defined as follows:*

$$p(v_i) = \frac{1}{\bar{d}(v_i)} \tag{32}$$

In the weighted network, we take the inverse of the average weight as the probability that neighbor nodes contribute to the influence of node v_i, which is defined as follows:

$$p'(v_i) = \frac{1}{\bar{\omega}(v_i)} \tag{33}$$

In **Eqs 32**, **33**, *$\bar{d}(v_i)$ is the average degree of all neighbor nodes; $\bar{\omega}(v_i)$ is the average weight of all neighbor nodes; $p(v_i)$ and $p'(v_i)$ is contribution probability of the node v_i in unweighted network and weighted network, respectively.*

4.2 Influence Calculation

After completing community discovery, it is necessary to perform opinion leader mining on different community structures. We propose a social network opinion leader mining method based on the overall and local structure of graphs by using the information interaction ability between nodes and the local characteristics of nodes.

(1) Users of overall influence.

Social network is a relatively stable social relationship system formed by the information interaction among users. Strong information interaction ability indicates that users are in the hub position in social networks and can promote network information sharing. Therefore, opinion leaders, as key nodes in social networks, will have high intensity information interaction ability.

The information interaction ability between nodes v_i and v_j in social network G can be measured by counting the number of common nodes between them. A higher number of common nodes for v_i and v_j indicates a higher closeness between them, which means a higher information interaction capability between v_i and v_j [37]. The metric of information interaction ability of node v is formulated as follows:

$$Total(v_i) = d(v_i) \sum_{v_j \in V} pow\left(B\left|PN\left(v_i, v_j\right)\right|\right) \tag{34}$$

Where $PN(v_i, v_j)$ is public neighbors of v_i and v_j; $|PN(v_i, v_j)|$ is the number of public neighbors for v_i and v_j; $pow(x, y)$ denotes the yth power of x; B is a constant, and is usually set $B = 1.1$ for convenience of calculation.

Since the social network constructed in **Section 3** is a weighted undirected graph, considering only the degrees of the nodes will bias the results. Therefore, we extend the sum of degrees to the sum of weights when analyzing the weighted network. The information interaction capacity of nodes in the weighted network is calculated as follows:

$$Total'(v_i) = \omega(v_i) \sum_{v_j \in V} pow\left(B, \left|PN\left(v_i, v_j\right)\right|\right) \tag{35}$$

Where $\omega(v_i)$ is the sum of the weights of node v_i. **Equations 34**, **35** utilize the information interaction ability of v_i with other nodes as a measure of the overall structural influence, i.e., the sum of the information interaction ability for users in the social network. This is a relationship between the user and other users in the social network, that is, from the overall structure of the graph.

(2) Users of local influence.

The local influence of a user is the influence of the user itself and the surrounding users on it. The local influence of node v_i is defined as follows:

$$Local(v_i) = \sum_{v_j = M(v_i)} C_D\left(v_j\right) p\left(v_j\right) \tag{36}$$

Where $C_D(v_j)$ is the degree centrality of the neighbor nodes v_j for v_i and $p(v_j)$ is the node contribution probability. For the weighted network, it is obviously not possible to consider only the node degree. For example, in **Figure 2A**, $d(v_4) = d(v_6) = 2$, but $\omega(v_6) > \omega(v_4)$, so v_6 has a higher degree of participation in the network. Therefore, the local influence calculation of users for the weighted network will consider both degree and weight, which are defined as follows:

$$Local'(v_i) = \sum_{v_j = M(v_i)} C_D^{\omega}\left(v_j\right) p'\left(v_j\right) \tag{37}$$

Where $C_D^{\omega}(v_j)$ is the comprehensive node centrality of v_j.

(3) Influence Ranking.

The user's influence will be evaluated by taking into account the user's ability to interact with information, as well as the influence of the user itself and the surrounding users on it, i.e., by integrating the overall and local structure of the graph. Its calculation formula is as follows:

$$Influence(v_i) = Total(v_i) \cdot Local(v_i) \tag{38}$$

$$Influence'(v_i) = Total'(v_i) \cdot Local'(v_i) \tag{39}$$

Equation 38 is applied to the unweighted network and **Eq. 39** is applied to the weighted network. These two equations enable the node influence assessment of all users in the social network. The higher influence of a node means that it is more important in the network, and the most important node is the opinion leader in the social network.

4.3 An Illustrative Example

Figure 2A is a weighted network, containing six nodes $\{v_1, v_2, \ldots, v_6\}$. **Figure 2B** shows the information interaction of v_6 with other nodes, and the number of common neighbor nodes

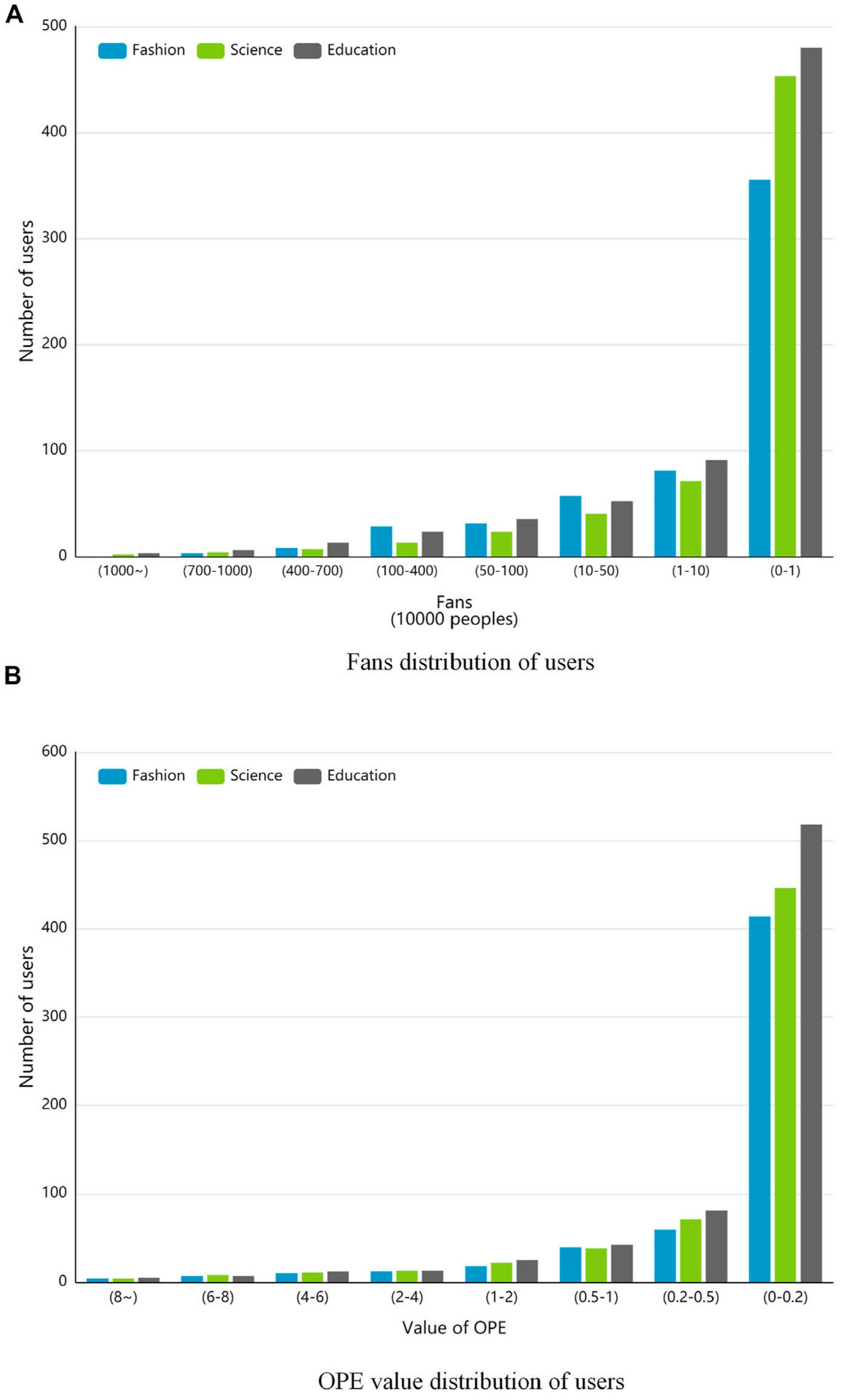


FIGURE 3 | Distribution of fans and *OPE* in micro-blog dataset, **(A)** is fans distribution of users. **(B)** is *OPE* value distribution of users.

between v_6 and other nodes is used to measure the information interaction ability between two nodes. The stronger information interaction ability of v_6 means the higher overall influence of v_6 in G. With **Eq. 35**, the overall influence of each node in G can be obtained, and the results are shown in **Table 2**.

Figure 2C shows all neighboring nodes of node v_6 in G. Each neighbor node has an inconsistent impact on v_6. The

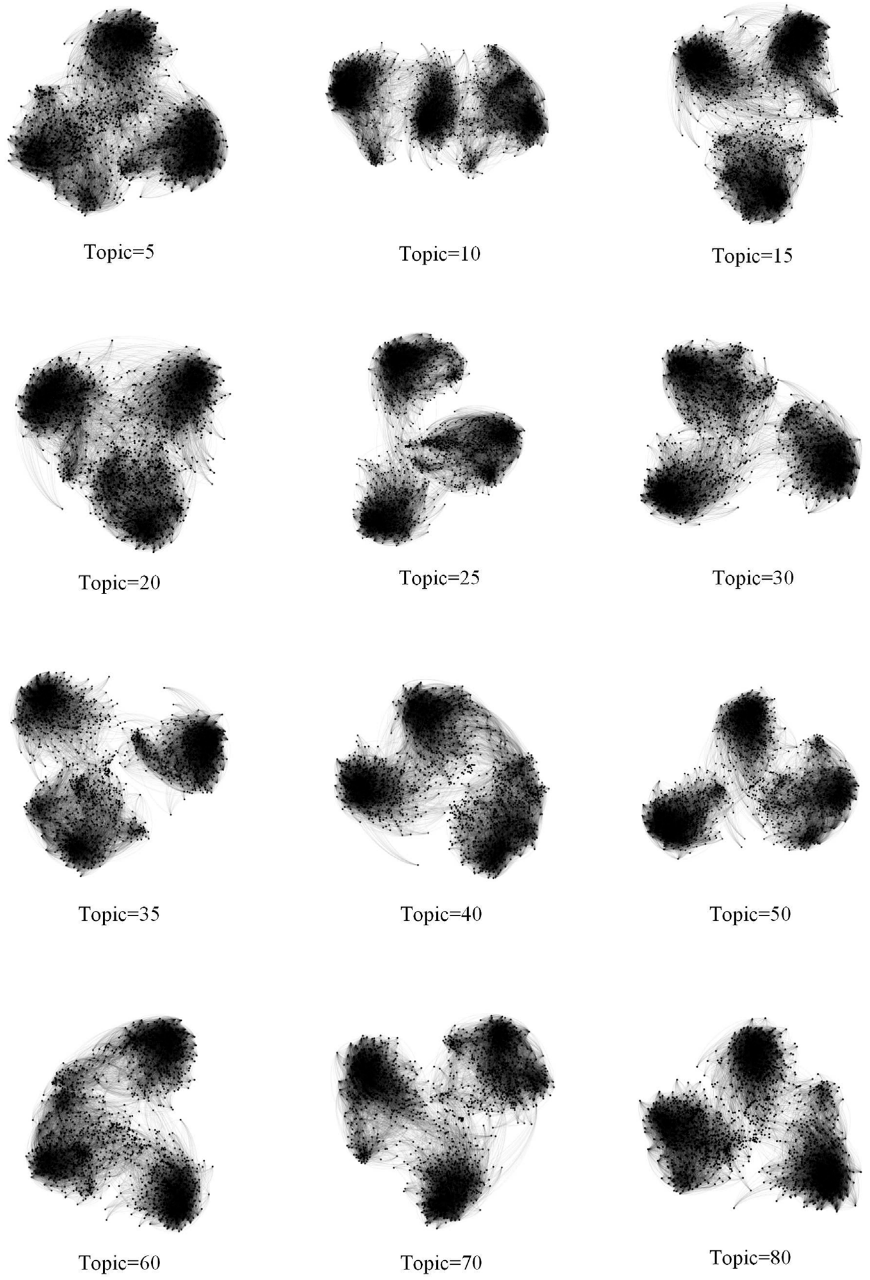


FIGURE 4 | Relationship network diagram of different topic numbers.

higher **Eq. 37** contribution probability of neighbor nodes, the higher influence on node v_6. The local influence of each node can be obtained using ($\tau = 1.1$), and the results are shown in **Table 3**.

The influence scores of all nodes can be obtained using **Eq. 39**, as shown in the **Table 4**. According to **Table 4**, it is known that v_5 has the highest influence. Therefore v_5 has the highest importance in the weighted network G, which is the opinion leader. **Figure 2D** shows the result of opinion leader mining, and the opinion leaders have been marked with different colors.

5 EXPERIENCE AND ANALYSIS

5.1 Dataset Analysis

Microblogging has now become an important social platform for most people to get information and communicate. Opinion leaders at the center of social networks are essential communication media for providing information to others. Analysis of online opinion leaders through microblog data can effectively identify the source of negative information and control it. Therefore, to validate the method proposed in this paper, we collected 37,590 posts by 1,879 users from

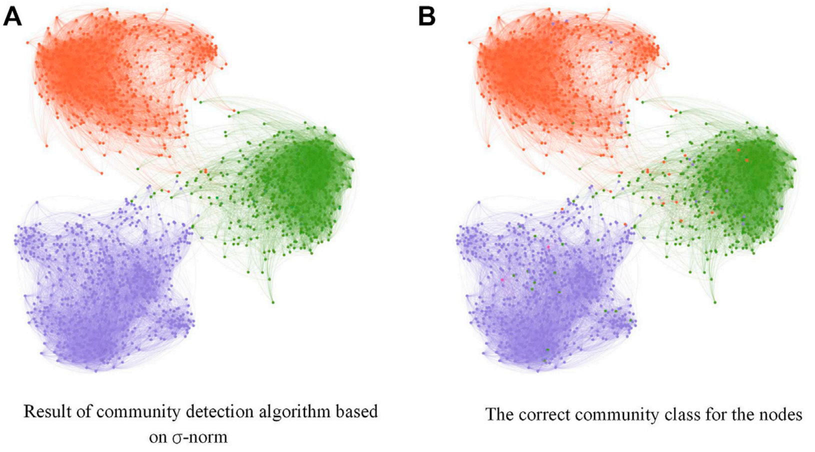

A
B
Result of community detection algorithm based on σ-norm
The correct community class for the nodes

FIGURE 5 | The results of community detection, **(A)** is the result of community detection algorithm based on σ-norm, **(B)** is the correct community class for all nodes.

three domains of Sina Weibo: fashion, technology and education as the experimental dataset, among which fashion, technology and education contain 11,260, 12,262 and 14,068 posts, respectively, and all posts made by a single user represent his semantic information.

According to some current studies, there is no precise evaluation system for opinion leaders. Therefore, we tag users with community labels by the domain they belong to and use the number of user followers and the activity indexes provided by Sina Weibo platform (number of users reading, number of interactions, number of super topics) as the basis for evaluating online opinion leaders. Opinion leaders were determined according to the ratio of 40 and 60%, expressed as OPE, which was calculated as follows:

$$OPE_i = 4\cdot\frac{Fans_i - \min(Fans)}{\max(Fans)-\min(Fans)} + 2\cdot\frac{\mathrm{Read}_i - \min(Read)}{\max(Read)-\min(Read)} + 2\cdot\frac{Inter_i - \min(Inter)}{\max(Inter)-\min(Inter)} + 2\cdot\frac{STop - \min(STop)}{\max(STop)-\min(STop)} \tag{40}$$

In **Eq. 40**, OPE_i represents the opinion leader evaluation indexes of user i, and a larger value indicates that the user i is more likely to become opinion leader; $Fans_i$, $Read_i$, $Inter_i$, and $STop_i$ denote the number of fans, readers, interactions, and super topics of user i, respectively; max($Fans$), min($Fans$) indicate the maximum and minimum values of the number of fans among all users, and other similar variables have similar meanings.

Figure 3 shows the distribution of the number of followers and *OPE* values in the Weibo dataset, where different colors represent different domains. It can be seen that the number of users with fans greater than 1000 W and *OPE* greater than eight is extremely small, and the influence of these users will also be at the top of the dataset, so we define the top 10% of users with *OPE* values in each domain as online opinion leaders for subsequent verification of the effectiveness and performance of the opinion leader mining method proposed in this paper.

5.2 Evaluation Metrics

To compare the performance of the community discovery method and online opinion leader mining method proposed in this paper with other methods, we use several widely used evaluation metrics.

Accuracy (AC) [38] is used to evaluate the correctness of the results for community detection algorithms and the correctness of the results for online opinion leader mining, which is defined as follows:

$$AC = \frac{\sum_{i=1}^{n}\delta(pc_i, cc_i)}{n} \tag{41}$$

Where n is the total number of nodes; pc_i denotes the predicted consequence; cc_i denotes the practical consequence; and $\delta(pc_i, cc_i)$ is the Kronecker function, indicating that it is equal to 1 if pc_i and cc_i are the same and 0 otherwise.

Normalized mutual information (NMI) [39] is used to compare the similarity between ground-truth and detected communities and to evaluate the quality of community segmentation in social networks. It is defined as follows:

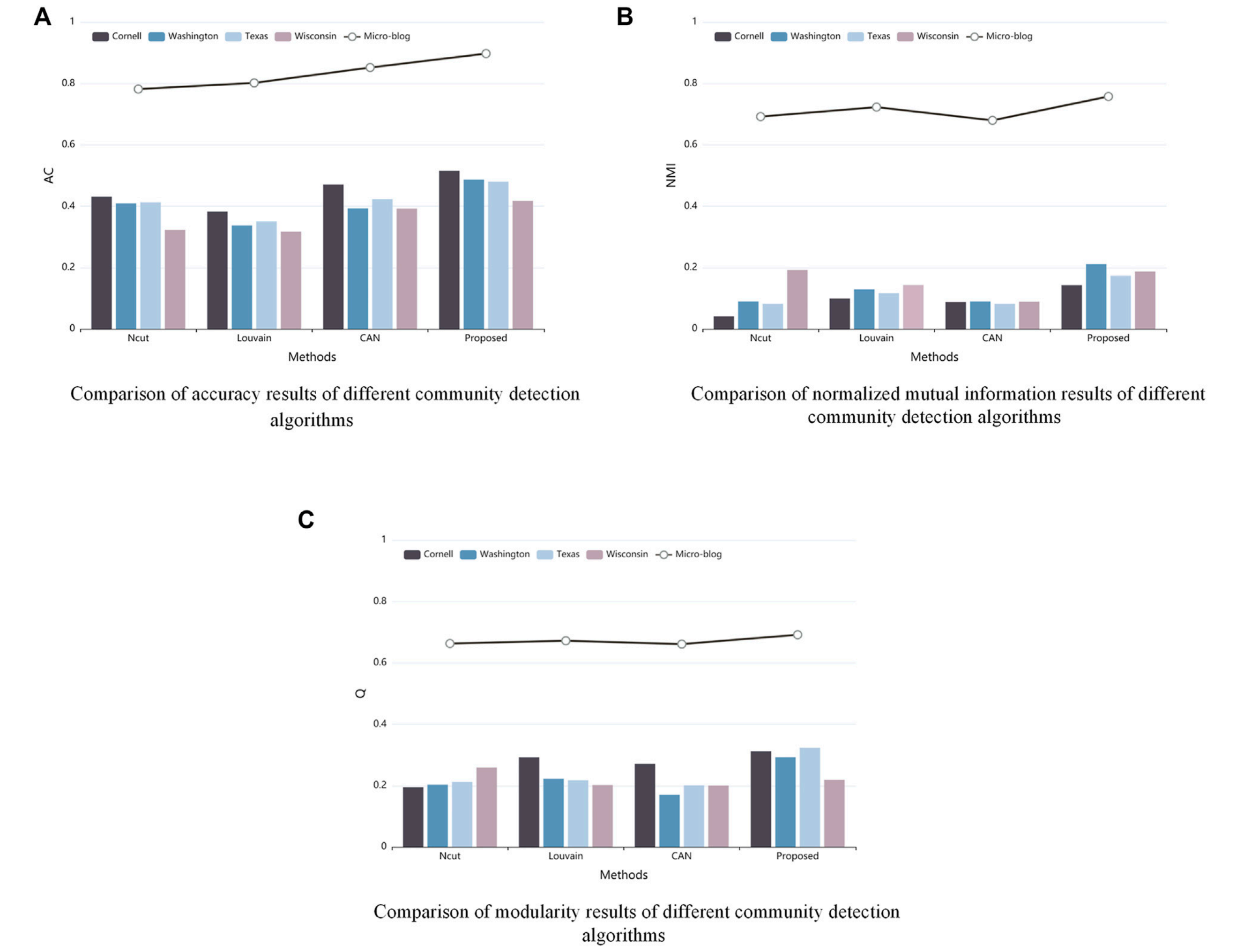
$$NMI = \frac{\frac{1}{2}(H(X)+H(Y)-H(X|Y)-H(Y|X))}{\max(H(X),H(Y))} \tag{42}$$

Where $H(X)$ and $H(Y)$ are the information entropy of the random variables X and Y; $H(X|Y)$ and $H(Y|X)$ are the conditional entropy of the random variables X and Y.

F1-score [40] is a composite metric that balances accuracy and recall which is defined as follows:

$$F1 - score = 2\times\frac{Recall\times Accuracy}{Recall + Accuracy} \tag{43}$$

Where $Accuracy = \frac{TP+TN}{TP+FN+FP+TN}$ and $Recall = \frac{TP}{TP+FN}$ denote accuracy and recall, respectively; True Positive (TP) includes the estimated observations identified true by both actual model and proposed model; True Negative (TN) includes the estimated observations identified false by both the actual model and proposed model; False Positive (FP) includes the estimated observations



Comparison of accuracy results of different community detection algorithms

Comparison of normalized mutual information results of different community detection algorithms

Comparison of modularity results of different community detection algorithms

FIGURE 6 | Comparative analysis of WebKB dataset and Micro-blog dataset based on AC, NMI and Q, **(A)** is comparison of accuracy results for different community detection algorithms, **(B)** is comparison of normalized mutual information results for different community detection algorithms, **(C)** is comparison of modularity results for different community detection algorithms.

identified false by the proposed model and true by actual model; False Negative (FN) includes the estimated observations identified true by the proposed model and false by actual model.

The modularity(Q) [41] is used to assess the quality of the community structure and is defined as follows:

$$Q = \frac{1}{|E|}\sum_{i,j}\left(Sim_{i,j} - \frac{d(v_i)d(v_j)}{|E|}\right)\delta(v_i, v_j) \tag{44}$$

Where $|E|$ denotes the sum of all edges in the network; $Sim_{i,j}$ is the value of the ith row and jth column of the similarity matrix; $d(v_i)$, $d(v_j)$ is the degree of node v_i and v_j; $\delta(v_i, v_j)$ is the Kronecker function, which is 1 if v_i and v_j are in the same community, and 0 otherwise.

5.3 Results of Community Detection

After cleaning the dataset (advertisement, duplicate, brief), all semantic information published by each user is used as one document, and all semantic information of all users is used as corpus. After that, the topic distribution of each document is obtained using the LDA topic model, and node representation and data matrix construction are performed. Then the similarity matrix is calculated using **Eq. 8** to achieve the construction of social networks, where the parameter m is set to 30 by default.

However, in the node representation process, the number of topics is an important parameter to determine the combined similarity of two users and to identify the community structure. In order to obtain the optimal value of the number of topics, the relationship between the number of different topics and the constructed similarity matrix is discussed. To obtain the optimal value of the number of topics, the relationship between the number of different topics and the constructed similarity matrix is discussed.

Figure 4 shows the relationship network diagrams constructed by the similarity matrix corresponding to different topic numbers. It can be clearly seen that regardless of the number of topics three main

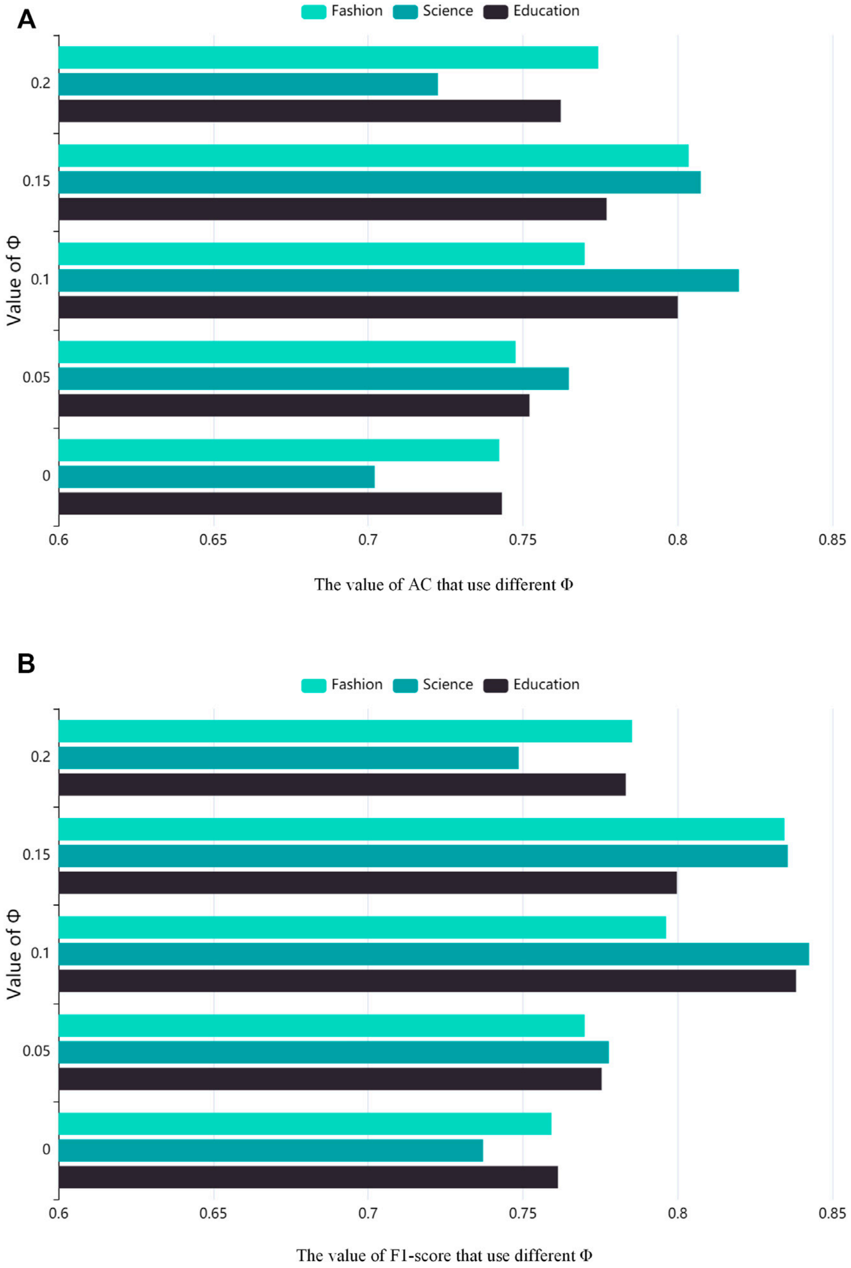


FIGURE 7 | Accuracy and F1-score analysis of opinion leaders mining with different similarity thresholds ϕ, **(A)** is the value of AC that use different ϕ, **(B)** is the value of F1-score that use different ϕ.

community structures are presented, corresponding exactly to users from three different domains, so it is reasonable to use the semantic information of individual users for the construction of the network. Regarding the choice of topic number, it is obvious from **Figure 4** that the community structure boundaries will not be obvious when the topic number is smaller and larger, and when the topic number is equal to 25 and 35, the community structure is of higher quality with clear contours, which is obviously better than the relationship network graph presented by other topic numbers. Therefore, we set the number of topics to 25 and conduct subsequent experiments.

After completing the construction of the social network, the results shown in **Figure 5A** are obtained using the σ-norm-based community detection algorithm proposed in this paper (where the initial value of the parameter ρ is set to 1, which is automatically adjusted according to the number of iterations, and $\rho = \rho^{*}2$ when the connected component of the target matrix U is smaller than the number of communities k, and $\rho = \rho/2$ when it is larger than the number of communities k. The adaptive loss parameter is set to 0.1 according to [34]), with each color representing a community. **Figure 5B** then represents the correct community to which the node belongs. By comparing **Figures 5A,B**, it can be observed that

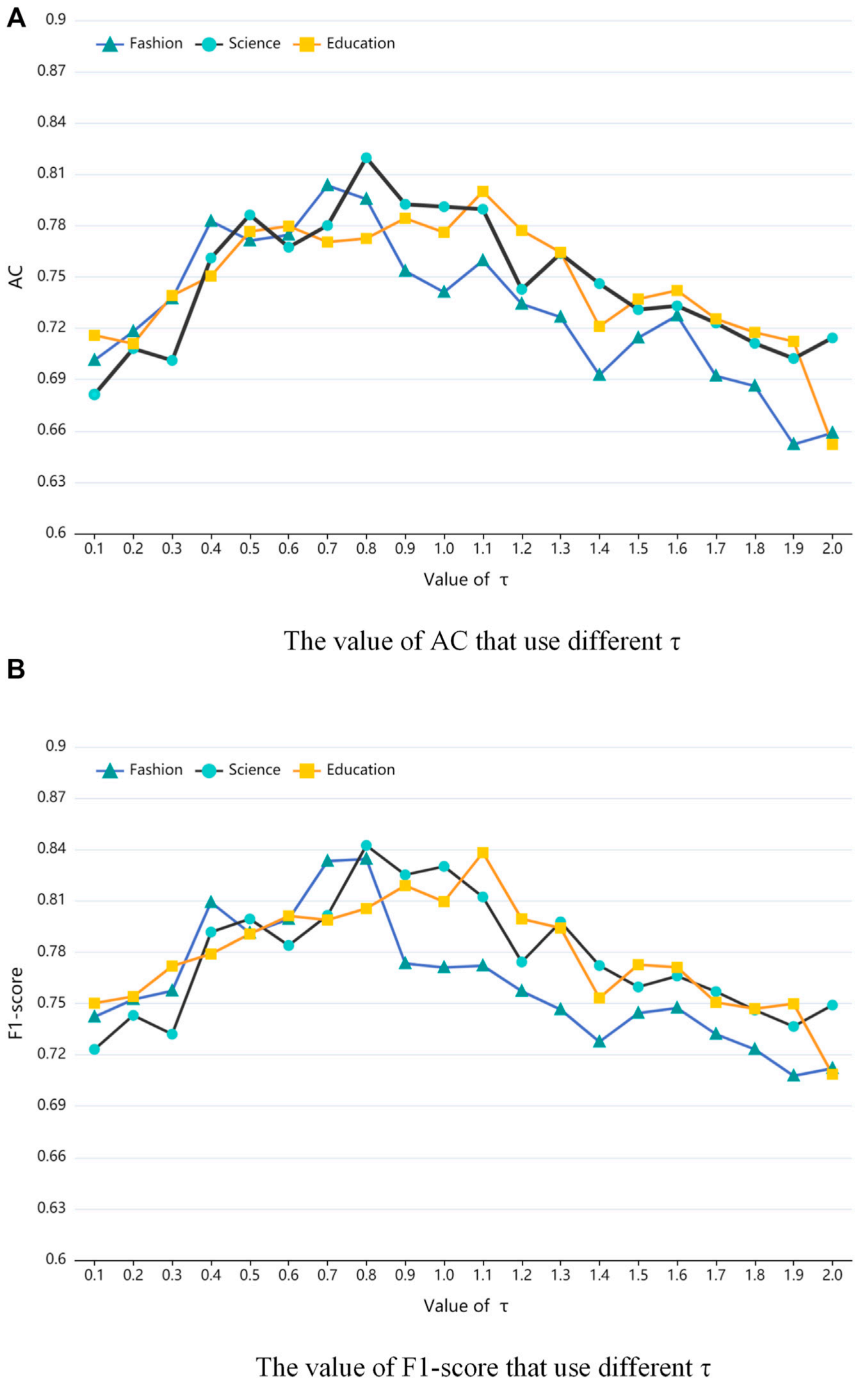


FIGURE 8 | Accuracy and F1-score analysis of opinion leaders mining with different positive tuning parameters τ, **(A)** is the value of AC that use different τ, **(B)** is the value of F1-score that use different τ.

the community detection algorithm proposed in this paper performs very well, and there are relatively few cases of misclassified communities, and only a small number of nodes are misclassified sporadically.

To better validate the performance of the algorithm proposed in this paper, we compare it with three community detection algorithms, Normalized cut (Ncut) [42], Fast unfolding algorithm (Louvain) [43] and Clustering with Adaptive Neighbors (CAN) [44], on the Weibo dataset and the WebKB dataset[1] [45]. Among them, Ncut is a classical graph-based approach; Louvain is a modularity-based community discovery algorithm; CAN is similar to the algorithm proposed in this

[1] http://www.cs.cmu.edu/~WebKB/

TABLE 5 | Time Complexity of different evaluation methods.

Method	Time complexity
BC	$O(\|V\|\cdot\|E\|)$
CC	$O(\|V\|^2 \cdot \log(\|V\|) + \|V\|\cdot\|E\|)$
EC	$O(\|V\|^2)$
PL	$O(\|V\|\cdot\langle G\rangle)$ ($\langle G\rangle$ is the average degree of the social network G)
PR	$O(it \cdot\|E\|)$ (it is the number of iterations)
Proposed	$O(\|V\|^2)$

paper and is an algorithm that learns both the data similarity matrix and the clustering structure. WebKB dataset is composed of approximately 6,000 web pages from computer science departments of four schools (Cornell, Texas, Washington, and Wisconsin), which are classified into seven categories.

Figure 6 shows the performance of each algorithm in terms of AC, NMI and Q on the WebKB and Weibo datasets. By observing **Figure 6**, we can find that the community detection method proposed in this paper only has slightly lower NMI and Q values than Ncut in the Wisconsin dataset, but is in the leading position in all other aspects, and is significantly more stable than the Ncut algorithm, which can be applied to multiple types of datasets well. In the Weibo dataset, the performance of AC, NMI and Q is better than other methods, which indicates that the community detection algorithm proposed in this paper can be perfectly applied to social networks composed of individual semantic information as features, and provides high-quality preconditions for the subsequent extraction of online opinion leaders.

5.4 Results of Online Opinion Leader Mining

After completing the community detection, each community structure can be considered as an opinion circle, from which the online opinion leaders are mined. Since the similarity matrix calculated by **Eq. 8** uses a sparsity constraint, the sum of the edge weights of the nodes is 1, which will lead to the existence of some edges with very small weights (very low similarity between nodes) within the community structure, as well as nodes whose weight sizes and degrees do not reach a balance. Therefore, to obtain the optimal experimental results, we need to determine a similarity threshold ϕ and keep the edges with weights greater than ϕ. **Figure 7** depicts the effects of using the opinion leader mining method proposed in this paper on the AC and F1-score metrics under different similarity thresholds. From **Figure 7**, we can find that the values of AC and F1-score increase and then decrease as ϕ increases, and when the ϕ reaches 0.1, the indicators in Science and Education communities reach the maximum value; when ϕ reaches 0.15, the indicators in Fashion community reach the maximum value. Therefore, the similarity threshold ϕ is set to 0.1 for Science and Education communities and 0.15 for Fashion communities. Also, to balance the size of the weight values of the nodes with the size of the degree, we found that multiplying the edge weights of each node by three performs best.

Finally, it is also necessary to determine the optimal value of the parameter τ (**Eq. 30**) used in this online opinion leader mining method. **Figure 8** depicts the effect of different τ on the AC and F1-score metrics, and it can be observed that the Fashion community reaches the maximum AC at τ equal to 0.7 and the maximum F1-score at τ equal to 0.8; the Science community reaches the maximum for each metric at τ equal to 0.8; the Education community reaches the maximum for each metric at τ equal to 1.1 maximum. Therefore, considering the magnitude of AC and F1-score indicators, the parameter τ is set to 0.7, 0.8, and 1.1 for Fashion, Science,and Education communities, respectively, for online opinion leader mining.

To verify the effectiveness and performance of the methods proposed in this paper, we compare the AC and F1-score metrics performance of the five methods on the Weibo dataset and further discuss the performance effectiveness of each method. Before giving the experimental results, a brief introduction of the five methods is given.

BC (Betweenness Centrality) [46]: The method uses betweenness centrality to mine opinion leaders. In most real networks, information flows randomly according to its intent rather than following the shortest path, so using betweenness centrality to measure node importance is not applicable in some networks.

CC (Closeness Centrality) [47]: This method is similar to betweenness centrality and combines the global and local effects of nodes in complex networks, effectively solving the complexity of node deletion methods and direct computation of betweenness centrality.

EC (Eigenvector Centrality) [48]: This method is based on the assumption that the importance of a node depends on the number of neighboring nodes and also on the influence of each neighboring node, so that the importance of the node is evaluated only from the other nodes connected to the node.

ProfitLeader (PL) [49]: This method ranks the key nodes in the network by measuring the profit that the nodes can provide.

PageRank (PR) [50]: This method ranks pages according to their link structure, i.e. the influence of a page depends on the number and quality of the other pages pointing to it. If a page has many high quality pages pointing to it, then it is also of high quality.

TABLE 6 | Comparison of AC and F1-score results with other evaluation methods.

Communities	Metrics (%)	BC	CC	EC	PL	PR	Proposed
Fashion	AC	67.60	71.65	74.03	74.82	75.52	80.35
	F1-score	74.91	73.04	77.60	81.77	78.65	83.44
Science	AC	69.33	72.43	76.95	75.76	77.12	81.97
	F1-score	73.02	74.97	79.22	78.50	80.82	84.34
Education	AC	68.22	71.06	73.21	74.33	74.20	80.00
	F1-score	72.30	73.41	75.45	79.09	78.42	83.82

TABLE 7 | The results of AC for different percentages of opinion leaders.

Communities	Top k percent of rank users				
	1	3	5	7	10
Fashion	100	82.36	82.14	79.49	80.35
Science	66.67	77.78	83.87	83.72	81.97
Education	85.71	80.95	82.85	79.59	80.00

TABLE 8 | The Top-15 users of influence evaluation results.

User id	Total influence	Local influence	Influence	Communities
196	162.29	12.27	1991.56	Education
1,545	144.50	11.23	1622.91	Science
357	157.25	8.41	1323.49	Education
242	140.28	8.28	1160.90	Education
432	101.65	11.27	1145.45	Education
1,483	123.19	8.81	1086.56	Science
1,209	117.80	8.79	1034.42	Fashion
1,226	100.04	8.73	874.0	Fashion
1,167	96.97	8.89	862.22	Fashion
530	96.58	8.31	803.46	Education
539	127.37	5.81	740.10	Education
1,349	111.19	6.53	726.59	Science
1,191	129.99	5.45	709.57	Fashion
829	125.16	5.32	666.02	Fashion
1,781	101.76	6.38	650.01	Science

Table 5 summarizes the time complexity of different influence calculation methods ($|V|$ is the total number of nodes of the network, $|E|$ is the total number of edges of the network). The time complexity of the method proposed in this paper can be divided into two parts: global and local. The time complexity of the global influence calculation is $O(|PN|\cdot|V|^2)$, where $|PN|$ is the public neighbors between nodes, $|PN|\ll|V|$; the time complexity of the local influence calculation is $O(|M|\cdot|V|)$, where $|M|$ is the number of node neighborhoods, $|M|\ll n$. Therefore, the time complexity of the influence calculation method proposed in this paper is $O(|V|^2)$.

Table 6 shows the performance results of the proposed method in this paper with the above five methods on the Weibo dataset. The AC and F1-score values are obtained by comparing the calculation results of each method with the actual network opinion leaders (the top 10% of important user nodes). We can find that the method in this paper has better results compared with other methods, and the AC and F1-score can reach more than 80% in all three community structures, which can prove the effectiveness and correctness of the method proposed in this paper. **Table 7** lists the mining accuracy of our proposed method for opinion leaders ranked in the top k% of influence, and it can be found that the results tend to be smooth and do not have excellent performance only for mining opinion leaders in specific positions, so the method can be applied to mining opinion leaders with different percentage requirements.

Table 8 presents the local influence, overall influence, and combined influence values of the top 15 users and the communities they belong to in the Weibo dataset using the results obtained from the proposed method. From **Table 8**, it can be found that as the ranking decreases, the values of both the local influence and the overall influence of the user show a relatively large decrease, which means that the user's information interaction ability with other users and the influence of neighboring nodes on it are decreasing. This also verifies the scarcity of users with followers greater than 1000 W and *OPE* values greater than eight in the Weibo dataset, further illustrating the effectiveness of the network opinion leader mining method proposed in this paper.

6 CONCLUSION

This paper studies the detection of local opinion leaders in semantic social networks. In the aspect of semantic information quantification, we introduce the LDA model to extract the global topics of network documents and construct the semantic feature representation of nodes by calculating the similarity between the global topics and the posts produced by users. To detect local opinion leaders, a community detection method based on σ-norm is presented to split the network and users with topic consistency create a public opinion circle. The proposed strategy efficiently prevents the exclusion of local opinion leaders with low global influence by taking into account local influence within the public opinion circle and global influence outside the public opinion circle. We conduct experiments on real social networks, and the results show that the proposed method is capable of a high-quality semantic social network partition and accurate mining of local opinion leaders. Future research will focus on the design of adaptive algorithms to achieve fast identification of opinion leaders in dynamic networks.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: http://www.cs.cmu.edu/~WebKB/.

AUTHOR CONTRIBUTIONS

HY proposed the core idea of the paper. QL and XD collected data and built the experimental platform. CC and LW wrote the main part of the paper and verified the performance of the algorithm. All authors listed approved the paper for publication.

FUNDING

This work is sponsored by National Natural Science Foundation of China (No. 61402126, No. 62101163); Nature Science Foundation of Heilongjiang Province of China (No. F2016024, No. LH2021F029), China Postdoctoral Science Foundation (No.

2021M701020); Heilongjiang Postdoctoral Science Foundation (No. LBH-Z15095, No. LBH-Z20020); University Nursing Program for Young Scholars with Creative Talents in Heilongjiang Province (No. UNPYSCT-2017094); Fundamental Research Foundation for Universities of Heilongjiang Province (No. 2020-KYYWF-0341).

REFERENCES

1. Camacho D, Panizo-LLedot Á, Bello-Orgaz G, Gonzalez-Pardo A, Cambria E. The Four Dimensions of Social Network Analysis: An Overview of Research Methods, Applications, and Software Tools. *Inf Fusion* (2020) 63:88–120. doi:10.1016/j.inffus.2020.05.009
2. Camacho D, Luzón MV, Cambria E. New Research Methods & Algorithms in Social Network Analysis. *Future Generation Comput Syst* (2021) 114:290–3. doi:10.1016/j.future.2020.08.006
3. Bello-Orgaz G, Jung JJ, Camacho D. Social Big Data: Recent Achievements and New Challenges. *Inf Fusion* (2016) 28:45–59. doi:10.1016/j.inffus.2015.08.005
4. Hussain A, Cambria E. Semi-supervised Learning for Big Social Data Analysis. *Neurocomputing* (2018) 275:1662–73. doi:10.1016/j.neucom.2017.10.010
5. Zhang M, Wang W. Study on Public Opinion Propagation in Self media Age Based on Time Delay Differential Model. *Proced Comput Sci* (2017) 122: 486–93. doi:10.1016/j.procs.2017.11.397
6. Chen X, Zhang W, Xu X, Cao W. A Public and Large-Scale Expert Information Fusion Method and its Application: Mining Public Opinion via Sentiment Analysis and Measuring Public Dynamic Reliability. *Inf Fusion* (2022) 78: 71–85. doi:10.1016/j.inffus.2021.09.015
7. He W, Tian X, Tao R, Zhang W, Yan G, Akula V. Application of Social media Analytics: A Case of Analyzing Online Hotel Reviews. *Online Inf Rev* (2017) 41:1. doi:10.1108/oir-07-2016-0201
8. Ramakrishnan J, Mavaluru D, Srinivasan K, Mubarakali A, Narmatha C, Malathi G. Opinion Mining Using Machine Learning Approaches: A Critical Study. In: 2020 International Conference on Computing and Information Technology (ICCIT-1441). Tabuk, Saudi Arabia: IEEE (2020). p. 1–4. doi:10.1109/iccit-144147971.2020.9213747
9. Chen T, Shi J, Yang J, Cong G, Li G. Modeling Public Opinion Polarization in Group Behavior by Integrating Sirs-Based Information Diffusion Process. *Complexity* (2020) 2020. doi:10.1155/2020/4791527
10. Aleahmad A, Karisani P, Rahgozar M, Oroumchian F. Olfinder: Finding Opinion Leaders in Online Social Networks. *J Inf Sci* (2016) 42:659–74. doi:10.1177/0165551515605217
11. Walter S, Brüggemann M. Opportunity Makes Opinion Leaders: Analyzing the Role of First-Hand Information in Opinion Leadership in Social media Networks. *Inf Commun Soc* (2020) 23:267–87. doi:10.1080/1369118x.2018.1500622
12. Jain L, Katarya R, Sachdeva S. Recognition of Opinion Leaders Coalitions in Online Social Network Using Game Theory. *Knowledge-Based Syst* (2020) 203: 106158. doi:10.1016/j.knosys.2020.106158
13. Chunaev P. Community Detection in Node-Attributed Social Networks: a Survey. *Comput Sci Rev* (2020) 37:100286. doi:10.1016/j.cosrev.2020.100286
14. Leskovec J, Lang KJ, Mahoney M. Empirical Comparison of Algorithms for Network Community Detection. In: WWW '10: Proceedings of the 19th International Conference on World Wide Web. New York, NY: Association for Computing Machinery (2010). p. 631–40. doi:10.1145/1772690.1772755
15. Papadopoulos S, Kompatsiaris Y, Vakali A, Spyridonos P. Community Detection in Social media. *Data Min Knowl Disc* (2012) 24:515–54. doi:10.1007/s10618-011-0224-z
16. Chien I, Lin C-Y, Wang I-H. Community Detection in Hypergraphs: Optimal Statistical Limit and Efficient Algorithms. In: International Conference on Artificial Intelligence and Statistics (PMLR) (2018). p. 871–9.
17. Garcia JO, Ashourvan A, Muldoon S, Vettel JM, Bassett DS. Applications of Community Detection Techniques to Brain Graphs: Algorithmic Considerations and Implications for Neural Function. *Proc IEEE* (2018) 106:846–67. doi:10.1109/jproc.2017.2786710
18. Cao J, Bu Z, Wang Y, Yang H, Jiang J, Li H-J. Detecting Prosumer-Community Groups in Smart Grids from the Multiagent Perspective. *IEEE Trans Syst Man Cybern, Syst* (2019) 49:1652–64. doi:10.1109/tsmc.2019.2899366
19. Bu Z, Li H-J, Zhang C, Cao J, Li A, Shi Y. Graph K-Means Based on Leader Identification, Dynamic Game, and Opinion Dynamics. *IEEE Trans Knowledge Data Eng* (2019) 32:1348–61.
20. Cao J, Wang Y, Bu Z, Wang Y, Tao H, Zhu G. Compactness Preserving Community Computation via a Network Generative Process. *IEEE Trans Emerging Top Comput Intelligence* (2021). doi:10.1109/tetci.2021.3110086
21. Zhao Y, Kou G, Peng Y, Chen Y. Understanding Influence Power of Opinion Leaders in E-Commerce Networks: An Opinion Dynamics Theory Perspective. *Inf Sci* (2018) 426:131–47. doi:10.1016/j.ins.2017.10.031
22. Liu X, Liu C. Information Diffusion and Opinion Leader Mathematical Modeling Based on Microblog. *IEEE Access* (2018) 6:34736–45. doi:10.1109/access.2018.2849722
23. Jain L, Katarya R. Identification of Opinion Leader in Online Social Network Using Fuzzy Trust System. In: 2018 IEEE 8th International Advance Computing Conference (IACC). Greater Noida, India: IEEE (2018). p. 233–9. doi:10.1109/iadcc.2018.8692095
24. Wang C, Du YJ, Tang MW. Opinion Leader Mining Algorithm in Microblog Platform Based on Topic Similarity. In: 2016 2nd IEEE International Conference on Computer and Communications (ICCC). Chengdu: IEEE (2016). p. 160–5. doi:10.1109/compcomm.2016.7924685
25. Dewi FK, Yudhoatmojo SB, Budi I. Identification of Opinion Leader on Rumor Spreading in Online Social Network Twitter Using Edge Weighting and Centrality Measure Weighting. In: 2017 Twelfth International Conference on Digital Information Management (ICDIM). Fukuoka, Japan: IEEE (2017). p. 313–8. doi:10.1109/icdim.2017.8244680
26. Yang L, Qiao Y, Liu Z, Ma J, Li X. Identifying Opinion Leader Nodes in Online Social Networks with a New Closeness Evaluation Algorithm. *Soft Comput* (2018) 22:453–64. doi:10.1007/s00500-016-2335-3
27. Su J, Xu J, Qiu X, Huang X. Incorporating Discriminator in Sentence Generation: a Gibbs Sampling Method. In: Proceedings of the AAAI Conference on Artificial Intelligence (2018). vol. 32.
28. Nie F, Wang X, Jordan M, Huang H. The Constrained Laplacian Rank Algorithm for Graph-Based Clustering. In: Proceedings of the AAAI conference on artificial intelligence (2016). vol. 30.
29. Wright J, Yang AY, Ganesh A, Sastry SS, Ma Y. Robust Face Recognition via Sparse Representation. *IEEE Trans Pattern Anal Mach Intell* (2008) 31:210–27. doi:10.1109/TPAMI.2008.79
30. Zhang R, Nie F, Guo M, Wei X, Li X. Joint Learning of Fuzzy K-Means and Nonnegative Spectral Clustering with Side Information. *IEEE Trans Image Process* (2018) 28:2152–62. doi:10.1109/TIP.2018.2882925
31. Oellermann OR, Schwenk AJ. *The Laplacian Spectrum of Graphs* (1991).
32. Fan K. On a Theorem of Weyl Concerning Eigenvalues of Linear Transformations: Ii. *Proc Natl Acad Sci* (1950) 36:31–5. doi:10.1073/pnas.36.1.31
33. Bertsekas DP. *Constrained Optimization and Lagrange Multiplier Methods*. Cambridge, MA: Academic Press (2014).
34. Nie F, Wang H, Huang H, Ding C. Adaptive Loss Minimization for Semi-supervised Elastic Embedding. In: Twenty-Third International Joint Conference on Artificial Intelligence (2013).
35. Li H-J, Bu Z, Wang Z, Cao J, Shi Y. Enhance the Performance of Network Computation by a Tunable Weighting Strategy. *IEEE Trans Emerg Top Comput Intell* (2018) 2:214–23. doi:10.1109/tetci.2018.2829906
36. Opsahl T, Agneessens F, Skvoretz J. Node Centrality in Weighted Networks: Generalizing Degree and Shortest Paths. *Social networks* (2010) 32:245–51. doi:10.1016/j.socnet.2010.03.006
37. Sheng J, Dai J, Wang B, Duan G, Long J, Zhang J, et al. Identifying Influential Nodes in Complex Networks Based on Global and Local Structure. *Physica A: Stat Mech its Appl* (2020) 541:123262. doi:10.1016/j.physa.2019.123262
38. Kynkäänniemi T, Karras T, Laine S, Lehtinen J, Aila T. Improved Precision and Recall Metric for Assessing Generative Models. *arXiv preprint arXiv: 1904.06991* (2019).

39. Zhang P. Evaluating Accuracy of Community Detection Using the Relative Normalized Mutual Information. *J Stat Mech* (2015) 2015:P11006. doi:10.1088/1742-5468/2015/11/p11006
40. Fawcett T. An Introduction to Roc Analysis. *Pattern recognition Lett* (2006) 27:861–74. doi:10.1016/j.patrec.2005.10.010
41. Newman ME, Girvan M. Finding and Evaluating Community Structure in Networks. *Phys Rev E Stat Nonlin Soft Matter Phys* (2004) 69:026113. doi:10.1103/PhysRevE.69.026113
42. Cour T, Benezit F, Shi J. Spectral Segmentation with Multiscale Graph Decomposition. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). San Diego, CA, USA: IEEE (2005). p. 1124–31. vol. 2.
43. Blondel VD, Guillaume J-L, Lambiotte R, Lefebvre E. Fast Unfolding of Communities in Large Networks. *J Stat Mech* (2008) 2008:P10008. doi:10.1088/1742-5468/2008/10/p10008
44. Nie F, Wang X, Huang H. Clustering and Projected Clustering with Adaptive Neighbors. In: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining (2014). p. 977–86. doi:10.1145/2623330.2623726
45. Getoor L. Link-based Classification. In: *Advanced Methods for Knowledge Discovery from Complex Data*. Berlin, Germany: Springer (2005). p. 189–207.
46. Freeman LC. Centrality in Social Networks Conceptual Clarification. *Soc networks* (1978) 1:215–39. doi:10.1016/0378-8733(78)90021-7
47. Okamoto K, Chen W, Li X-Y. Ranking of Closeness Centrality for Large-Scale Social Networks. In: *International Workshop on Frontiers in Algorithmics*. Berlin, Germany: Springer (2008). p. 186–95.
48. Solá L, Romance M, Criado R, Flores J, García del Amo A, Boccaletti S. Eigenvector Centrality of Nodes in Multiplex Networks. *Chaos* (2013) 23:033131. doi:10.1063/1.4818544
49. Yu Z, Shao J, Yang Q, Sun Z. Profitleader: Identifying Leaders in Networks with Profit Capacity. *World Wide Web* (2019) 22:533–53. doi:10.1007/s11280-018-0537-6
50. Page L, Brin S, Motwani R, Winograd T. The PageRank Citation Ranking: Bringing Order to the Web. In: *Tech. Rep.* Stanford, CA, USA: Stanford InfoLab (1999).

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Advantages of publishing in Frontiers

OPEN ACCESS
Articles are free to read
for greatest visibility
and readership

FAST PUBLICATION
Around 90 days
from submission
to decision

HIGH QUALITY PEER-REVIEW
Rigorous, collaborative,
and constructive
peer-review

TRANSPARENT PEER-REVIEW
Editors and reviewers
acknowledged by name
on published articles

Frontiers
Avenue du Tribunal-Fédéral 34
1005 Lausanne | Switzerland

Visit us: www.frontiersin.org
Contact us: frontiersin.org/about/contact

REPRODUCIBILITY OF RESEARCH
Support open data
and methods to enhance
research reproducibility

DIGITAL PUBLISHING
Articles designed
for optimal readership
across devices

FOLLOW US
@frontiersin

IMPACT METRICS
Advanced article metrics
track visibility across
digital media

EXTENSIVE PROMOTION
Marketing
and promotion
of impactful research

LOOP RESEARCH NETWORK
Our network
increases your
article's readership