HetInf: Social Influence Prediction With Heterogeneous Graph Neural Network

*Liqun Gao[1], Haiyang Wang[1], Zhouran Zhang[2], Hongwu Zhuang[1] and Bin Zhou[1]**

[1]Key Lab of Software Engineering for Complex Systems, School of Computer, National University of Defense Technology, ChangSha, China, [2]Department of Materials Science and Engineering, National University of Defense Technology, ChangSha, China

With the continuous enrichment of social network applications, such as TikTok, Weibo, Twitter, and others, social media have become an indispensable part of our lives. Web users can participate in their favorite events or pay attention to people they like. The "heterogeneous" influence between events and users can be effectively modeled, and users' potential future behaviors can be predicted, so as to facilitate applications such as recommendations and online advertising. For example, a user's favorite live streaming host (user) recommends certain products (event), can we predict whether the user will buy these products in the future? The majority of studies are based on a homogeneous graph neural network to model the influence between users. However, these studies ignore the impact of events on users in reality. For instance, when users purchase commodities through live streaming channels, in addition to the factors of the host, the commodity is also a key factor that influences the behavior of users. This study designs an influence prediction model based on a heterogeneous neural network HetInf. Specifically, we first constructed the heterogeneous social influence network according to the relationship between event nodes and user nodes, then sampled the user heterogeneous subgraph for each user, extracted the relevant node features, and finally predicted the probability of user behavior through the heterogeneous neural network model. We conducted comprehensive experiments on two large social network datasets. Furthermore, the experimental results show that HetInf is significantly superior to the previous homogeneous neural network methods.

Keywords: social network analysis, social influence analysis, heterogeneous neural network, user behavior prediction, deep learning

OPEN ACCESS

Edited by:
Chengyi Xia,
Tianjin University of Technology, China

Reviewed by:
Li Pan,
Shanghai Jiao Tong University, China
Di Jin,
Tianjin University, China

****Correspondence:***
Bin Zhou
binzhou@nudt.edu.cn

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

Received: *30 September 2021*
Accepted: *17 November 2021*
Published: *05 January 2022*

Citation:
Gao L, Wang H, Zhang Z, Zhuang H and Zhou B (2022) HetInf: Social Influence Prediction With Heterogeneous Graph Neural Network. Front. Phys. 9:787185. doi: 10.3389/fphy.2021.787185

1 INTRODUCTION

Nowadays, social networks are everywhere around us in our daily lives. Social influence occurs when we get information from social networks, which means that network events (such as network news, trending topics, publishing papers, or other events) or network users we are interested in constantly influence us through social media, and both of them can induce us to engage in social action (including retweet, comment, like, publish, and purchase). For example, live commerce is very popular nowadays, and we will choose our favorite live streaming host to buy necessary commodities. From another perspective, both the live streaming host (user) and the commodities (event) have a substantial impact on the target user's behavior. Similar to the definition of "event" in the study

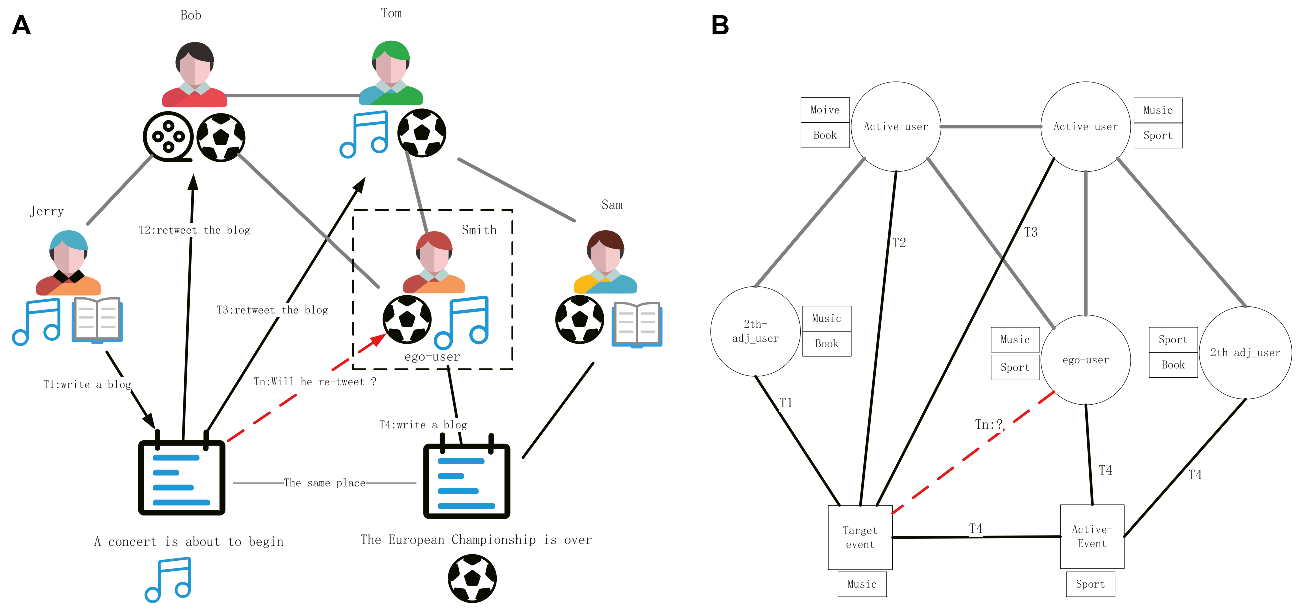

FIGURE 1 | Problem illustration of mining user–event influence in heterogeneous networks and predicting user behaviors. **Figure 1A** shows an instantiated prediction case, and the goal is to predict whether Smith (ego-user) will forward the concert blog (whether the red line will occur in the future). **Figure 1B** is obtained by abstracting **Figure 1A**, given 1) the relationship between the user and the event within the observable time (the connection in b includes three relationships: u-u, u-e, and e-e); 2) the embedding representation of the attributes of different nodes in the observable time (the rectangle next to the node in b); 3) the activation state of neighbor nodes (active or inactive); and 4) the embedding representation of each node in the network, then we predicted whether ego-user will participate in the target event.

mentioned in reference 1, social events can be regarded as a complete semantic unit in which network users participate and understand. One of the key computational problems is to predict the user's social behavior in social influence analysis. How to model the influence relationship to predict the behavior of network users on events is one of the key computational problems in user-level social influence prediction. This problem is applied to many fields, including but not limited to election [2], network marketing [3,4], recommendation [5], rumor detection [6], and information dissemination[7,8].

There are a large number of research studies on the role of the heterogeneity of nodes in social networks in social influence [9–12]. This kind of study mainly focuses on user nodes' interest in event nodes and predicts user behavior by capturing the influence of event nodes' topic level. The study in reference 10 improves the traditional cascade propagation mode and applies the topic distribution methods to an independent cascade model and linear threshold model. The study in reference 9 uses a graph generation method to predict user behavior through the relationship between event topics and network users. These methods use traditional machine learning models to predict users' social behavior through the manual feature representation of learning nodes. However, they do not consider the association between different types of network nodes in heterogeneous social networks, such as the dual impact of users and events on target users, which leads to the limited ability to capture the incentives that really affect user behavior.

Due to the progress of the graph neural network [13], the nodes of network have stronger representation ability. Many studies use graph neural network to model the problem of social influence prediction and make plenty of progress. The study in reference 14 uses the user's local network as the input of the graph neural network to learn the user's potential social representation and uses both network structures and user-specific features in convolutional neural and attention networks. Based on DeepInf, the study mentioned in reference 15 applies the multi-view impact prediction network to solve the social influence prediction problem. However, these methods are based on assumptions that users are only affected by other users, in order to model the relationship between users (homogeneous network), which lacks the analysis of the influence between heterogeneous nodes. Real social networks (such as Twitter, Digg, and Citation network) are heterogeneous and contain different types of entities [16], for example, user nodes and event nodes (stories, tweets, papers, and other objects), which inevitably interact with each other. For example, in **Figure 1A**, Bob may forward the concert event because he is affected by the user Jerry (because he is not interested in music), and Tom may forward the concert event because he is affected by the event (he likes music).

To tackle these challenges, we focused on user-level behavior prediction in a heterogeneous network. This network contains two types of nodes: user and event. It aims to construct a heterogeneous influence network of event nodes and user nodes based on attributes, and we use graph neural networks to model the influence between these nodes so as to better mine the inducement of user behavior. Inspired by the latest research on heterogeneous neural network [13], the local modeling of heterogeneous networks can capture both structure and content heterogeneity and provide more reliable heterogeneous node representation ability for downstream tasks. Therefore, we combine the benefits of heterogeneous neural networks and semantic representation methods to model the influence network of local neighbor nodes based on heterogeneous network graph. For example, in **Figure 1A**, with the aim of learning the influence of different types of nodes on him through historical semantic information and influence relationship, we input the heterogeneous neighbor local graph network with Smith as the ego-user, so as to predict his future behavior (whether to participate in the discussion of concert events).

Specifically, we proposed HetInf, a heterogeneous network influence prediction model based on two types of nodes. First, based on the influence relationship of network nodes, we constructed a heterogeneous relationship graph composed of them and hoped to build a more accurate influence model. Second, we sampled ego-user neighbor subgraphs. Specifically, an innovative heterogeneous network sampling strategy, based on restart random walk (RWR) [17], is used to sample the topology features and the semantic features (including event topics and user interests) of the heterogeneous nodes in ego-user neighbor subgraphs. Subsequently, an end-to-end heterogeneous neural network influence model is built, the historical topic features of events and the historical interest features of users based on semantics is embedded using Word2Vec [18], the node representation through the node semantics is aggregated, and the heterogeneous graph neural network model is used to learn the node relationship of event–user heterogeneous network. Finally, we learned the influence of different neighbor nodes on ego-user node through the graph attention networks [19], so as to predict whether users will have the social behavior of participating in events in the future.

Summarizing, our contributions are given as follows:

(1) We applied the heterogeneous graph neural network method to predict the influence of users at the micro-level in social networks. Specifically, we extend the deep learning method of homogeneity social influence networks and analyze the dynamic propagation mode of heterogeneous networks to infer more accurate influence network.
(2) As respect to heterogeneous networks, we design a local sampling method in line with time sequence process, established the influence relationship between events and users, and applied an innovative end-to-end heterogeneous graph neural network model to more accurately predict users' social behavior.
(3) Therefore, we tested two real large-scale social network data: Digg and Weibo. The experimental results show that HetInf exhibit significant improved accuracy when constructing a heterogeneous network compared with several state-of-the-art baselines.

The rest of this article is organized as follows: **Section 2** formulates social influence locality problem. **Section 3** introduces the proposed framework in detail. **Section 4** describes extensive experiments with two datasets, **Section 5** summarizes related study, and **Section 6** concludes this study.

2 PROBLEM FORMULATION

In this section, we introduced several related definition and then formally formulated the problem of heterogeneous social network influence locality.

2.1 Definition 1: R-Neighbors and r-Heterogeneous Neighbor Subgraph

A heterogeneous network $\mathcal{G} = (V, E; O_V, R_E, A_V)$ with two types of nodes V, three types of links E, and node attributes A_V is defined. In **Figure 1B**, O_V includes user node U (round), event nodes E (square), and $V = U \cup E$; R_E includes user–event relation R_{ue}, event–event relation R_{ee}, user–user relation R_{ue}, and $E = R_{ue} \cup R_{ee} \cup R_{uu}$. A_V is the attribute feature of the node, including the semantic attribute A_V^S(rectangle) and topology attribute A_V^T. For user u, its r-neighbors are defined to be $\Gamma_u^r = \{v\text{: } d(v, u) < = r\}$, where $d(v, v_u)$ is the distance (number of hops) from v to u, and v is different types of nodes in heterogeneous subgraph G. The r-ego heterogeneous subgraph of user u is the local heterogeneous subgraph induced by Γ_u^r and denoted by G_u^r.

2.2 Definition 2: Social Action

Social action refers to the behavior of users in the social network events, such as social network users retweet tweets (events) or publish papers (events). Formally, a social action can be regarded as the action of user u on event e at time t in the heterogeneous graph G_u. We define social action as a binary problem. Action status S_{ue}^t belongs to (0,1). When $S_{ue}^t = 1$, it means that user u has social action for an event e after time t; when $S_{ue}^t = 0$, it means that no social action has occurred.

2.3 Problem 1: Heterogeneous Social Network Influence Locality

Social influence locality models the probability of social action when ego-user u_i is influenced by neighbor nodes on his r-ego network $G_{u_i}^r$; formally, given a 6-tuple {u, e, t, G, A, S}, social influence locality aims to quantify the activation probability of user u_i's social action in response to event e after a given time t in G with attribute feature A and action status S as follows:

$$prob\left(S_{u_i}^{t+\Delta t} | e_j^t, G_{u^i}^r, A_G^t, S_G^t\right), \qquad (1)$$

where u_i represents the ego-user, e_j^t represents the event e_j at time t, $G_{u_i}^r$ represents r-heterogeneous neighbor subgraph, A_G^t represents the node attributes by time t, including semantic attribute A^S, topology attribute A^T, and S_G^t represents the action state of the subgraph node before timestamp t.

After determining the problem, we sample N 6-tuple samples through preprocessing data. Similar to the definition of DeepInf [14], we regard social influence locality as a binary classification problem and calculate the model parameters by minimizing the negative log likelihood

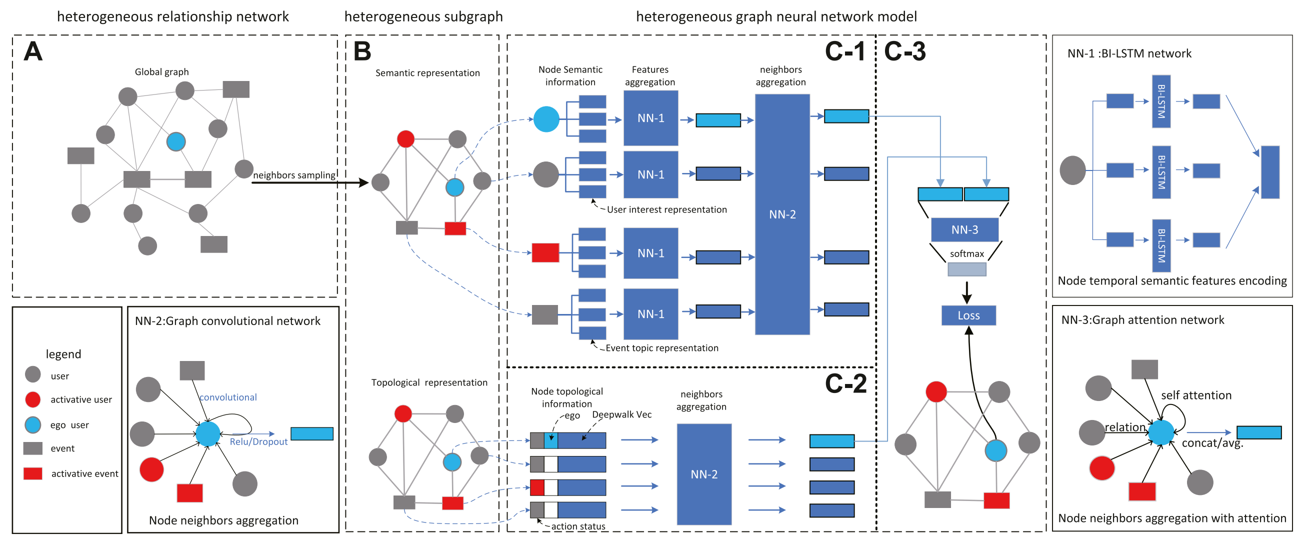
heterogeneous relationship network
heterogeneous subgraph
heterogeneous graph neural network model
A
Global graph
neighbors sampling
B
Semantic representation
Topological representation
C-1
C-3
Node Semantic information
Features aggregation
neighbors aggregation
NN-1
NN-2
User interest representation
Event topic representation
NN-3
softmax
Loss
NN-1 :BI-LSTM network
BI-LSTM
Node temporal semantic features encoding
legend
user
activative user
ego user
event
activative event
NN-2:Graph convolutional network
convolutional
Relu/Dropout
Node neighbors aggregation
C-2
Node topological information
ego
Deepwalk Vec
action status
neighbors aggregation
NN-3:Graph attention network
self attention
relation
concat/avg.
Node neighbors aggregation with attention

FIGURE 2 | Model framework of HetInf: It first **(A)** constructs heterogeneous relational networks with attributes), next **(B)** samples fix sized heterogeneous neighbors for each node, then aggregates the attribute features and topological features of the sampled heterogeneous neighbors *via* (C-1) and (C-2), and finally optimizes the model *via* (C-3). NN-1: Bi-LSTM, used to aggregate semantic features based on time series; NN-2: GCN network, used to aggregate heterogeneous node attributes; and NN-3: GAT network, used to learn the weights of interaction between nodes.

objective method. We use the following objective with parameters θ:

$$L(\theta) = -\sum_{j=1}^{M}\sum_{i=1}^{N}\left(S_{u_i}^{t+\Delta t}|e_j^t, G_{u^i}^r, A_G^t, S_G^t\right). \tag{2}$$

Similarly, we assume that time Δt is infinite, that is, we only predict user action outside the observation time window.

3 MODEL FRAMEWORK

Our goal is to design a heterogeneous graph network model based on the interaction between events and users, named HetInf, which aims to learn the dynamic preferences of individuals and the influence of heterogeneous neighbors in detail. Building the HetInf model needs three steps: 1) constructing heterogeneous relational networks with attributes; 2) sampling r-heterogeneous neighbor subgraph; and 3) building heterogeneous graph neural network model. **Figure 2** shows the framework of HetInf.

3.1 Construct Heterogeneous Graph

3.1.1 Constructing Heterogeneous Relational Networks

Considering the heterogeneous network based on influence prediction user action, an intuitive way is to construct heterogeneous influence graph [20] to obtain the influence between nodes from the heterogeneous graph. We first obtain two types of nodes, including users and events. The event node can be regarded as a specific network event, such as a hashtag in social network dataset or a "story" in Digg dataset.

Then we establish the relationship between nodes according to the data characteristics, including the relationship between user–user, user–event, and event–event. Specifically, we use follow relationship and interact relationship as the user–user relationship; the user–event relationship between users and events can be determined by the user's historical behavior, for example, the user has participated in the event; and the event–event relationship can use co-occurrence association [21] or semantic association [22]. In this way, we construct the global heterogeneous relationship network $\mathcal{G}$ (Definition 2.1), as shown in **Figure 2A**.

3.1.2 Extract Node Attributes

For different types of nodes, we select two features as the initial node representation of the heterogeneous relationship network:

Semantic features: Since the user's behavior is more influenced by the semantic information of the social event [9], the semantic feature A_v^S of each node was used to indicate the "bias" in the semantic level. For the event nodes, TF-IDF [23] was used to sample K keywords Ew_i, $i \in K$ in each e and to distinguish between different events. For the user nodes, the stopwords (non-meaningful) was first removed, then the most frequent I keywords Uw_i $(i \in I)$ of a user was sampled to represent the user's interest, and finally, these keywords with timestamp was sorted to represent the semantic evolution of nodes over time. Formalization is given as follows:

$$A_v^S = \begin{cases} Sort\_by\_time(Ew_i) & \text{if } v \in E \\ Sort\_by\_time(Uw_i) & \text{if } v \in U. \end{cases} \tag{3}$$

Topology features: In addition, inspired by the study in reference 14, the DeepWalk [24] algorithm was used to obtain the topology feature $A_v^T, A_v^T \in R^d$, where d represents the embedding dimension, $v \in V$.

3.2 Sample r-Heterogeneous Neighbor Subgraph

Generally, the graph neural network uses the feature information of the node's n-order neighbor (e.g., first older) for the aggregation process of node features, such as GAT [19]. Therefore, the breadth first search (BFS) strategy [25] used in the graph localization process will make the weight between users and events too large in hot events. For example, in popular events, due to a large number of related events, most of the neighbor nodes sampled by a user are event type nodes, which will lead to the event type nodes becoming the dominant influence, and ignore the influence between users. In order to solve this problem, an improved random walk strategy was used to comply with the law of information dissemination. The sampling strategy includes two steps:

1. Sample a fixed-length random walk sequence. We took each user $u \in U$ as the starting point, utilizing the RWR [17] method to sample a fixed number of N_r neighbor nodes.
2. Use the meta-path method to perform sub-sampling in the sequence of step 1. We used random walk probability and u-e-u (user publishes an event, which is then forwarded by other users) and U-U (users directly forward through other users) meta-paths, sampled neighbor nodes with a fixed length of $N < N_R$, and then used these neighbor nodes to construct a subgraph G_u.

This strategy satisfies the law of information dissemination and helps avoid the problem of too little sampling of some types of nodes. Similar to the definition in DeepInf [14], a positive instance of a local heterogeneous subgraph was generated if a user has social action with an event after the timestamp t, and a negative instance was generated if the user is not observed in the watch window to be associated with the event.

3.3 Build Heterogeneous Graph Neural Network Model

In this way, 6-tuple (2.3) was used as a set of examples and a deep learning model was designed to predict the action state $S_u^{t+\Delta t}$ of the ego-user. Our neural network model consists of three parts: a semantic feature aggregation module (shown in **Figure 2c-1**), topological feature aggregation module (shown in **Figure 2c-2**), and heterogeneous multi-attribute hidden layer aggregation module (shown in **Figure 2c-3**). In this section, the different modules are introduced to express the process of model building.

3.3.1 Semantic Feature Aggregation Module

A neural network module was designed to learn the deep association between user and event semantics. The module uses the node semantic attribute A_v^S (obtained in 3.1.2) and the local heterogeneous subgraph G_u as input and realizes the aggregation function of semantic features through a neural network $H_S'(v)$. Specifically, we denoted the semantic feature as W_v^i, indicating the ith semantic feature of node v, and utilized Word2Vec [18] to pre-train W_v^i as x_i. Inspired by the study in reference 13, the neural network structure of Bi-LSTM was used to capture the association between semantic features of nodes at a deeper level, and the average value of all hidden states was used to represent the general aggregation embedding as follows:

$$H_S(v) = \frac{\sum_{i\in W_v^i}\left[\overrightarrow{LSTM}_\theta(x_i)\|\overleftarrow{LSTM}_\theta(x_i)\right]}{|W_v|}, \tag{4}$$

where $H_S(v) \in R^{d\times 1}$ (semantic feature embedding dimension), v represents a node in subgraph G_u, and x_i represents the ith feature word of node v (refer to [13]). $\overrightarrow{LSTM}$ represents the forward LSTM network, $\overleftarrow{LSTM}$ represents the backward LSTM network, θ represents the neural network parameter, and operator $\|$ represents concatenation. Bi-LSTM can learn the potential evolution process of node semantics, leading to a strong expression capability [26].

Then the GCN [27] framework was used to aggregate semantic nodes of $H_S(5)$ to learn the influence relationship between different nodes, which is formally expressed as follows:

$$\begin{aligned} H_S'(v) &= GCN(H_S(v)) = g(A(G_u)H_S(v)W^T + b) \\ A(G_u) &= D^{-1/2}AD^{-1/2} \end{aligned}, \tag{5}$$

where $W \in R^{d\times d}$, $b \in R^d$ are model parameters, g is a non-linear activation function, A is the adjacency matrix of $G(u)$, and D represents $diag(A)$. Since the number of subgraph nodes is fixed, A (G_u) can be calculated efficiently.

3.3.2 Topological Feature Aggregation Module

A topological feature can represent the importance of nodes in the network [28]. To aggregate topological feature embeddings of heterogeneous neighbors for each node, a layer of the GCN model was used for feature aggregation; in particular, the input vector consists of topological features A_v^T and node state S_v^t, which is inspired by the study in reference 14. Then the concatenated vector into the GCN layer was input to generate the node topological features hidden layer vector $H_T'(v)$. The formalization is given as follows:

$$\begin{aligned} H_T(v) &= A_v^T\|S_v^t \\ H_T'(v) &= GCN(H_T(v)) = g(A(G_u)H_T(v)W^T + b). \\ A(G_u) &= D^{-1/2}AD^{-1/2} \end{aligned} \tag{6}$$

Eq. 6 is the same as **Eq. 5**, except for the input and model parameters. We used the GCN to aggregate the topological embeddedness of all heterogeneous neighbors. Obviously, GCN has excellent performance in relation aggregation capability [27].

3.3.3 Heterogeneous Multi-Attribute Hidden Layer Aggregation Module

We can obtain the semantic feature embedding $H_T'(v)$ and structural feature embedding $H_S'(v)$ of each node in the heterogeneous subgraph G_u. To combine these features based on neural network module for each node v of

TABLE 1 | Statistics of the datasets.

Statistics	Weibo	Digg
Num of users	410764	279631
Num of events	7124	3155
User–user	92819	1731657
User–event	1196489	3018196
Event–event	22518	—
Num/dim of user interest topics	20/32	—
Num/dim of event historical topics	20/32	—
Dim of node topology embedding	128	128
Num of samples	52130	34113

subgraph G, the graph attention network [19] was employed. The advantage of this is that since different nodes have different influence contributions to the results, the multi-head GAT learns the influence between different attributes of heterogeneous nodes.

First, we concatenated the hidden layer results of the previous steps, then, following the study in reference 14, we used multi-head GAT and calculated the normalized attention coefficients $H'{}_f(v)$ as follows:

$$
\begin{aligned}
H_f(v) &= H'_T(v) \| H'_S(v) \\
H'_f(v) &= \sigma\left(\frac{1}{K}\sum_{k=1}^{1}\sum_{v \in G_u} \alpha_{iv}^{k} \mathbf{W}^{k} H_f(v)\right) \\
\alpha_{iv} &= \frac{\exp\left(\mathrm{LeakyReLU}\left(\mathbf{a}^T\left[\mathbf{W}H_f(i)\|\mathbf{W}H_f(v)\right]\right)\right)}{\sum_{v \in G_u}\exp\left(\mathrm{LeakyReLU}\left(\mathbf{a}^T\left[\mathbf{W}H_f(i)\|\mathbf{W}H_f(v)\right]\right)\right)},
\end{aligned}
\tag{7}
$$

where $\mathbf{a}$ is the attention parameter, $\mathbf{a} \in R^{2d}$, $\mathbf{W} \in R^{d\times d}$ is model parameters, $\cdot^T$ represents transposition, $\|$ is the concatenation operation, α_{iv} indicates the importance of node i to node v, and K represents the number of heads.

3.3.4 Output Layer and Loss Function

As shown in **Figure 2c-3**, the full connection layer (FC layer [27]) was used to output the two-dimensional representation of each node, then the current ego-user result was taken out, the ground truth was compared with, and **formula 2** was optimized as the loss function used in our study.

4 EXPERIMENTS

In this section, extensive experiments were conducted with the aim of answering the following research questions:

- (RQ1) How does HetInf perform vs state-of-the-art baselines for influence prediction tasks?
- (RQ2) How do different components, for example, heterogeneous multi-attribute hidden layer aggregation module or semantic feature aggregation module, affect the model performance? How much performance gain is added to these modules?
- (RQ3) How do various hyper-parameters, for example, embedding dimension of keywords or the size of sampled heterogeneous neighbors set, impact the model performance?

4.1 Datasets

Following the previous studies [14], experiments on two public datasets were conducted to quantitatively evaluate our proposed model. The detailed statistics are presented in **Table 1**.

Digg [29]: The Digg dataset is a story collector, which contains the data of stories that were promoted to Digg home page within 1 month in 2009. For each story, this dataset collects the voting lists of all users, the voters' friendship links, and the timestamp of each vote. This dataset comes from the study in reference 30. In our experiment, we took the story as the "event" node and "voting" as the user action to build a heterogeneity graph. Due to the lack of text data in the dataset, the deep framework (**Figure 2c-2**) of semantic information was not used in this dataset.

Weibo [31]: Weibo is the most popular social networking platform in China. This dataset contains 3,000,000 original tweets and retweets and comments of the original tweets from September 28, 2012 to October 29, 2012. At the same time, the dataset also contains the follow relationship between users who participate in these tweets. The dataset comes from in the study in reference 32. In our experiment, we extracted hashtags as events and built the heterogeneous graph of users and events, and the behavior of users participating in events (comment or retweet) is regarded as user action.

4.2 Data Preparation

In view of the imbalance in the number of active neighbors proposed by DeepInf, we set a threshold $n > 5$ (n is the sum of the number of active users and active events).Therefore, less active observation samples are removed, and thus, the sample characteristics involved in training are significantly related to social influence [31]. In order to solve the problem of data skew, the down-sampling method was used to control the positive and negative ratio of samples at 1:3.

Compared with the previous study, the preparation has the following differences:

(1) For the choice of events, due to the shortcomings of the number of participants lack significance, we excluded some events with fewer participants and set the threshold of the number of participants to the top 30% of the distribution of the number of events so as to extract the total number of events in **Table 1**.
(2) In the Weibo dataset, we established the event–event relationship (edge) through the semantic correlation of events. Specifically, the historical text of each event was collected, the tweets text collection was sampled in the time window t, and then the semantic vectorization representation of the event was obtained by the par2vec [33] method. Then we calculated the cosine similarity between events; if the correlation threshold $n > 0.7$, the

TABLE 2 | Prediction performance of different methods on the two datasets (%).

Dataset	Model	AUC	Prec	Rec	F1
Weibo	SVM	77.11	43.27	70.79	53.71
	DeepInf	82.75	48.86	74.13	58.90
	MvInf	83.75	50.18	75.02	60.13
	HetInf-GCN	**85.10**	67.88	73.60	70.62
	HetInf-GAT	85.02	**68.12**	**74.03**	**70.95**
Digg	SVM	90.65	66.82	78.49	72.19
	DeepInf	88.97	68.80	73.49	71.21
	MlvInf	91.11	**70.35**	78.50	74.20
	HetInf-GCN	90.74	69.94	**79.75**	74.52
	HetInf-GAT	**92.03**	70.12	77.63	**74.68**

TABLE 3 | Relative gain in terms of F1 and Prec. against the best baseline (%).

F1	Weibo	Digg	Prec	Weibo	Digg
MvInf	60.13	74.20	MvInf	50.18	70.35
HetInf	70.95	74.68	HetInf	68.12	70.12
Relative gain	17.9%	0.6%	Relative gain	35.7%	−0.3%

two events are semantically strongly correlated, and then we established the relation (edge) between events.

(3) For the extraction of node semantic features (**Figure 2c-2**), we fixed the number of keywords of each node (for example, $n = 20$). If the keyword samples of some nodes are insufficient ($n < 20$), we filled zeros to complete the vector to ensure the consistency of the input of the neural network.

4.3 Baselines

Support Vector Machine (SVM) [34]: A support vector machine (SVM) with linear kernel was used as the classification model. Specifically, the splicing of three features (including semantic features, topology features, and action status) was used as the input vector, and the problem was defined as two classification method.

DeepInf [14]: Our framework was compared with the influence network model based on the graph neural network, which constructs homogeneity subgraph based on user relationship and predicts user node action in the future.

MvInf [15]: Our framework was compared with the state-of-the-art graph neural network model MvInf, which introduces a multi-view structure based on DeepInf and uses the complementarity and consistency between different views to enhance learning performance. The difference is that our proposed model is based on the common influence of events and users.

HetInf and Its Variants: In the heterogeneous multi-attribute hidden layer aggregation module, different graph neural network frameworks were used to distinguish the two methods: HetInf–GCN and HetInf–GAT. Separately, HetInf–GAT uses the GAT [19] method as a method to fuse node features, mainly using attention mechanism to obtain the importance between nodes, while the HetInf–GCN method uses the GCN [27] framework to aggregate the node features and calculate the node influence by learning the node relationship of the subgraph.

4.4 Hyper-Parameter

In our proposed method, we used DeepWalk [24] to embed the node topology features; the restart probability of this method is 0.8, and the output vector length is 64 dimensions. We used the ReLU [35] as the activation function sigma (**Eq. 5**, **6**) and used the Adam [36] optimizer for training, with a learning rate of 0.005, and we set dropout = 0.5. We used 50, 25, and 25% of the instances for training, validation, and testing respectively; the batch size of all datasets was set to 256. In order to accommodate more nodes, we set the total number of nodes in the subgraph to 100 (including two different types of nodes).

4.5 Result and Analysis

4.5.1 (RQ1) Performance Analysis

How does HetInf perform vs state-of-the-art baselines of influence prediction methods? Will users take actions on the events in the future? What are the advantages of the proposed model compared with baseline? In order to answer question RQ1, we applied four indicators to compare with the previous state-of-the-art model (the same evaluation metrics as MvInf).

It should be noted that there are the following differences from the baseline method: in the semantic feature aggregation module, we used Word2Vec to embed each feature word into a vector with a dimension of 32. Specifically, the number of keywords for each node is 20. The output dimension of Bi-LSTM hidden layer is 128 and was used as GCN input (as shown in **Figure 2c-2**). The final output dimension of the GCN module is 128. In the topological feature aggregation module, the output dimension of DeepWalk is 128 and the state feature dimension was 2 (including action state and ego-state), so the GCN's input dimension is 130, and the output dimension of this module is 128 (similar to the DeepInf method). In the multi-attribute hidden layer aggregation module, for HetInf–GCN, we used two layers of GCN as the aggregation function of the module, in which the input layer of the first layer dimension is 256 and the output dimension of the second layer is 128. For HetInf–GAT, we used the GAT method, the input dimension is 256 and the output dimension is 128. Performance report of all models in **Table 2** and **Table 3** in which the best results are highlighted in bold.

(1) It can be seen from the results that in most cases, our proposed model is better than the baseline, especially in the accuracy and F1 value of microblog dataset (F1: 17.9%, Prec.: 35.7%), which proves that we have obtained the gain of accuracy after introducing heterogeneous networks and establishing event influence relations and verified the effectiveness of the proposed framework.

(2) From the results of the Digg dataset, it can be seen that the heterogeneous graph neural network model with two types of nodes can also bring performance gain (F1: 0.6%, Prec.: 0.3%) (lack of semantic information of heterogeneous nodes), which proves that our proposed model can improve the prediction ability of user behavior only through heterogeneous social networks.

4.5.2 (RQ2) Ablation Analysis

HetInf is a deep learning model combining different modules, which calculates the influence between different nodes and

TABLE 4 | Statistics of the datasets.

Dataset	Model	F1	AUC
Weibo	Only-Topology	62.20	81.17
	Only-Semantic	57.89	74.33
	No-LSTM	54.33	72.65
	No-NN-3	64.48	83.31
	HetInf-GAT	70.95	85.10
	HetInf-GCN	70.62	85.02
Digg	No-NN-3	66.28	85.32
	HetInf-GAT	74.71	92.03
	HetInf-GCN	67.72	90.74

predicts user behavior by aggregating the embedding of different types of node attributes. To answer RQ2, we used Auc and F1 indicators as the standard for evaluating results, we conducted ablation studies to evaluate performances of several model variants which include:

1) No-NN-1 that cancels the LSTM method (NN-1) and then concatenates the vectors to embed the representation of the semantic feature to verify the impact of the semantic feature aggregation module on the results; 2) Only-Topology that uses heterogeneous topology encoding (C-2) to represent each node embedding (cancel C-1 module); 3) Only-Semantic that uses heterogeneous topology encoding (C-1) to represent each node embedding (cancel C-2 module); and 4) No-NN-3 that utilizes a FC layer to combine embeddings of different neighbor representation (replace NN-3). It should be noted that the Digg dataset lacks semantic information, so we only tested the results of 4) to verify the effectiveness of heterogeneous multi-attribute hidden layer aggregation module. The results of predicted AUC and F1 values are shown in **Table 4** and **Figure 3**.

(1) The performance of Only-Topology is better than that of Only-Semantic, indicating that the position of key nodes in the network is more influential (such as opinion leaders).
(2) The performance of Only-Semantic is better than that of No-lstm, indicating that Bi-LSTM–based content encoding is superior to “shallow” encoding like FC for capturing “deep” content feature interactions.
(3) HetInf (including GCN and GAT) is better than No-NN-3 shows that graph convolution network plays a role in capturing node type influence.
(4) HetInf–GAT is superior to HetInF–GCN, indicating that graph attention network can obtain more influential potential relationships than GCN method.

4.5.3 (RQ3) Hyper-Parameter Analysis

To answer question RQ3, we investigated how hyper-parameters affect the predictive performance of the model. We conducted a parameter analysis on the Weibo dataset and used F1 value as an evaluation indicator. Specifically, we tested the impact of three key parameters: 1) semantic attribute embedding dimension; 2) head for multi-head attention; and 3) number of keywords. The experimental results are shown in **Figure 4**.

(1) Semantic Attribute Embedding Dimension: As shown in **Figure 4A**, when the semantic attribute dimension d varies from 16 to 256, the overall evaluation indicator is increasing because more dimensions contain more information. However, when the dimension reaches 128, the performance begins to decline, which is likely due to the result of overfitting.
(2) Head for Multi-head Attention: Like DeepInf, we are concerned about the number of GAT heads in heterogeneous multi-attribute hidden layer. As shown in **Figure 4B**, the increase of heads brings benefits to performance, but after more than 8, the performance remains stable but it has a negative impact on efficiency.
(3) Number of Keywords: The feature words of network nodes represent the semantic bias of nodes, which directly affect the prediction results. As shown in **Figure 4C**, when the numbers changes, it means that the amount of semantic information of network nodes increases and the evaluation improves at the same time. However, when the number of keywords exceeds a certain value, it will bring down performance, which is likely due to the noise caused by sampling too many non-

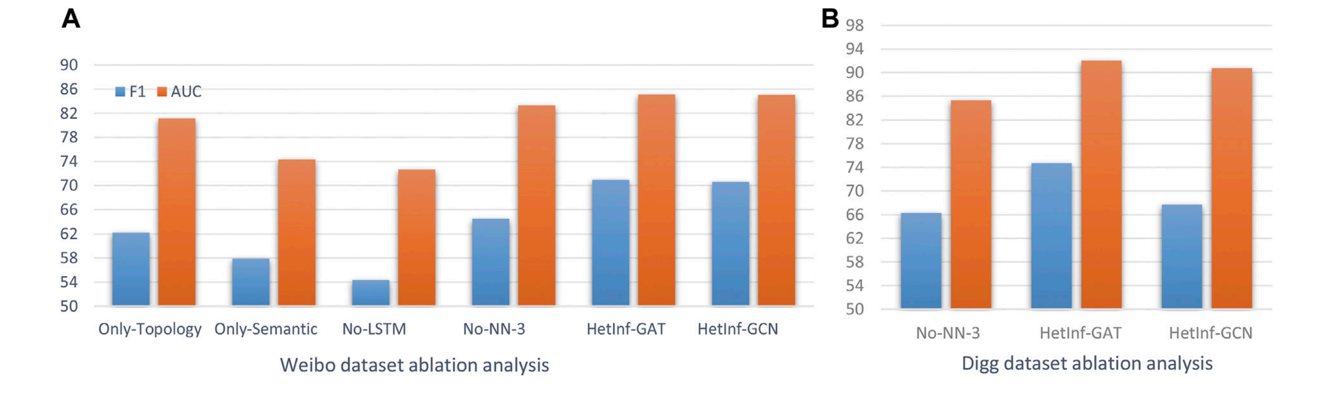

FIGURE 3 | Performances of variant proposed models **(A)** eibo dataset ablation analysis. **(B)** Digg dataset ablation analysis.

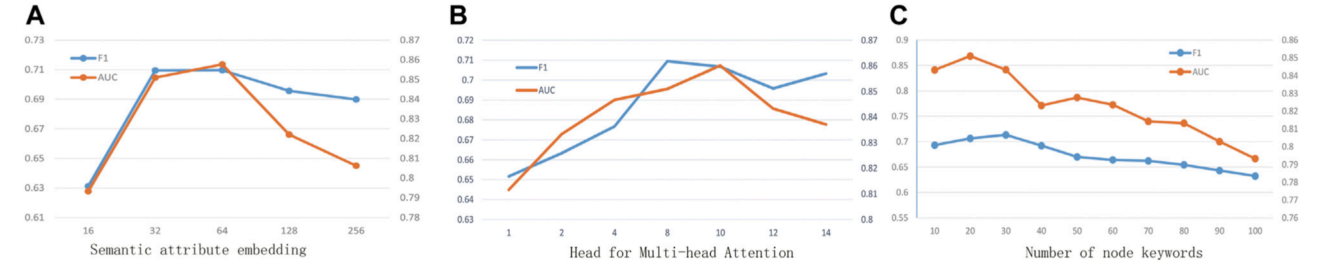

FIGURE 4 | Parameter analysis **(A)** Semantic attribute embedding. **(B)** Head for multi-head attention. **(C)** Number of node Keywords.

significant keywords. As can be seen from the figure, it is best to control the feature words between 20 and 40.

5 RELATED WORK

5.1 Social Influence Prediction

Social influence prediction is a fundamental problem in a social network analysis, which supports downstream tasks. At the micro-perspective [37,38], this problem is mainly modeled by analyzing user relationships. There are many different research directions, such as a user interaction influence analysis [39,40], network structure diversity analysis [41,42], topic influence analysis [10], and influence maximization [43]. Specifically, in the study in reference 44, the existence of social influence was proved by quantitative analysis of mutual influence. The study in reference 31 proposes social local network concepts, using user interaction and network structure to predict user behavior. The study in reference 45 uses topic level influence to model user influence. The study in reference 46 introduces a topic-level influence propagation and aggregation algorithm to derive the indirect influence between nodes. In recent years, with the continuous progress of deep learning, many studies have introduced deep learning into social influence prediction to improve the prediction performance. A popular deep learning method is [14], which provides an end-to-end framework to predict social influence by learning the potential features of users. The study in reference 15 has improved the study in reference 14 to enhance feature representation and result accuracy with a multi-view model. The study in reference 47 proposes a social influence prediction model NNMInf based on neural network multi-label classification. The study in reference 48 introduces a deep neural network framework witch simulate social influence and predict human behavior. Compared with traditional methods, these deep learning models show better learning performance.

5.2 Heterogeneous Graph Neural Network

In recent years, we have identified a huge development of the graph neural network in deep learning technology [27,49], and the state-of-the-art model GAT [19], which represents the method of depth learning-based graphical representation as the graph neural network (GNN), the main idea is as follows: the first step is to calculate the feature representation of neighbor nodes, and the second step is to aggregate neighbors through message passing mechanism to obtain the feature representation of nodes [50].

Recently, the heterogeneous graph neural network has become the main branch of GNN. The main task is to learn the representation of heterogeneous nodes on the graph neural network, so as to adapt to the downstream tasks based on heterogeneous networks. The study in reference 13 realizes node representation of heterogeneous networks by aggregating features of different types of nodes in stages. The study in reference 51 proposes a heterogeneous graph neural network based on hierarchical attention, including node level attention and semantic level attention. Node level attention aims to learn the importance between nodes and their neighbors based on meta-paths, while semantic level attention can learn the importance of different meta-paths. The study in reference 52 proposes a heterogeneous graph neural network method for subgraphs, which trains a classifier to learn the neighbor average features of the random sampling graph of the relational "metagraph." The MAGNN [53] model which contains the node content transformation to encapsulate input node attributes, the intra-meta-path aggregation to incorporate intermediate semantic nodes, and the inter-meta-path aggregation to combine messages from multiple meta-paths. GTN [54], which generates new graph structures by identifying useful connections between unconnected nodes on the original graph, can learn effective node embeddings on the new graphs in an end-to-end fashion. HGNN-AC [55] based on reference 53 proposed a general framework for heterogeneous graph neural network *via* Attribute Completion, including pre-learning of topological embedding and attribute completion with attention mechanism. These heterogeneous graph neural network representation methods enhance the representation ability of nodes and provide a more practical idea for downstream tasks.

6 CONCLUSION

In this study, we studied the problem of influence prediction based on a heterogeneous neural network, introduced a novel model HetInf that combines three neural network modules models to jointly infer the interaction between events and users in heterogeneous networks, and predicted the future behavior of network users. The local sampling method of heterogeneous networks was improved to capture the law of information dissemination, so as to obtain a more realistic user influence subgraph. Experimental results show that the influence prediction model can benefit from the heterogeneous network as well as joint learning embedding of users and events. In general, the empirical studies verify the effectiveness of our proposed model compared to the baseline methods.

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

LG contributed to the core idea of the experiment design and analysis results under the guidance of BZ. HW assisted in experiment code and experiment analysis. HZ and ZZ analyzed the comparative experiment. BZ supervised the research, provided financial support, and provided the financial support and experimental equipment. BZ is the corresponding author. All authors discussed the results and contributed to the final manuscript.

FUNDING

This work is supported by the National Key Research and Development Program of China (2018YFC0831703) and the National Natural Science Foundation of China (Nos. 61732022, 61732004, 62072131, and 61672020).

REFERENCES

1. Gui H, Liu J, Tao F, Jiang M, Norick B, Kaplan L, et al. Embedding Learning with Events in Heterogeneous Information Networks. *IEEE Trans Knowl Data Eng* (2017) 29:2428–41. doi:10.1109/TKDE.2017.2733530
2. Mehrizi MA, Corò F, Cruciani E, D'Angelo G. Election Control through Social Influence with Unknown Preferences. In: International Computing and Combinatorics Conference. New York, NY: Springer (2020). p. 397–410. doi:10.1007/978-3-030-58150-3_32
3. Dholakia UM, Bagozzi RP, Pearo LK. A Social Influence Model of Consumer Participation in Network- and Small-Group-Based Virtual Communities. *Int J Res marketing* (2004) 21:241–63. doi:10.1016/j.ijresmar.2003.12.004
4. Kim YA, Srivastava J. Impact of Social Influence in E-Commerce Decision Making. vol. 258 of ACM International Conference Proceeding Series. In: ML Gini, RJ Kauffman, D Sarppo, C Dellarocas, F Dignum, editors. Proceedings of the 9th International Conference on Electronic Commerce: The Wireless World of Electronic Commerce, 2007; August 19-22, 2007; Minneapolis, MN, USA. University of MinnesotaACM (2007). p. 293–302. doi:10.1145/1282100.1282157
5. Ye M, Liu X, Lee W. Exploring Social Influence for Recommendation: a Generative Model Approach. In: WR Hersh, J Callan, Y Maarek, M Sanderson, editors. The 35th International ACM SIGIR conference on research and development in Information Retrieval, SIGIR '12; August 12-16, 2012; Portland, OR, USA. ACM (2012). p. 671–80.
6. Hosni AIE, Li K, Ahmad S. Minimizing Rumor Influence in Multiplex Online Social Networks Based on Human Individual and Social Behaviors. *Inf Sci* (2020) 512:1458–80. doi:10.1016/j.ins.2019.10.063
7. Zhou X, Wu B, Jin Q. User Role Identification Based on Social Behavior and Networking Analysis for Information Dissemination. *Future Generation Comput Syst* (2019) 96:639–48. doi:10.1016/j.future.2017.04.043
8. Li S, Zhao D, Wu X, Tian Z, Li A, Wang Z. Functional Immunization of Networks Based on Message Passing. *Appl Math Comput* (2020) 366:124728. doi:10.1016/j.amc.2019.124728
9. Liu L, Tang J, Han J, Yang S. Learning Influence from Heterogeneous Social Networks. *Data Min Knowl Disc* (2012) 25:511–44. doi:10.1007/s10618-012-0252-3
10. Barbieri N, Bonchi F, Manco G. Topic-Aware Social Influence Propagation Models. *Knowl Inf Syst* (2013) 37:555–84. doi:10.1007/s10115-013-0646-6
11. Zhang Z-K, Liu C, Zhan X-X, Lu X, Zhang C-X, Zhang Y-C. Dynamics of Information Diffusion and its Applications on Complex Networks. *Phys Rep* (2016) 651:1–34. doi:10.1016/j.physrep.2016.07.002
12. 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
13. Zhang C, Song D, Huang C, Swami A, Chawla NV. Heterogeneous Graph Neural Network. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, August 4–8, 2019 (2019) p. 793–803. doi:10.1145/3292500.3330961
14. Qiu J, Tang J, Ma H, Dong Y, Wang K, Tang J. Deepinf: Social Influence Prediction with Deep Learning. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Halifax, NS, August 13–17, 2017 (2018). p. 2110–9.
15. Xu H, Jiang B, Ding C. Mvinf: Social Influence Prediction with Multi-View Graph Attention Learning. *Cogn Comput* (2021) 1–7. doi:10.1007/s12559-021-09822-z
16. Calais Guerra PH, Veloso A, Meira W, Jr, Almeida V. From Bias to Opinion: A Transfer-Learning Approach to Real-Time Sentiment Analysis. In: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, San Diego, CA, August 21–24, 2011 (2011). p. 150–8.
17. Tong H, Faloutsos C, Pan J. Fast Random Walk with Restart and its Applications. In: Proceedings of the 6th IEEE International Conference on Data Mining (ICDM 2006); 18-22 December 2006; Hong Kong, China. IEEE Computer Society (2006) p. 613–22. doi:10.1109/icdm.2006.70
18. Goldberg Y, Levy O. *word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method*. CoRR abs/1402.3722 (2014).
19. Veličković P, Cucurull G, Casanova A, Romero A, Lio P, Bengio Y. *Graph Attention Networks*. arXiv preprint arXiv:1710.10903 (2017).
20. Niepert M, Ahmed M, Kutzkov K. Learning Convolutional Neural Networks for Graphs. vol. 48 of JMLR Workshop and Conference Proceedings. In: M Balcan KQ Weinberger, editors. Proceedings of the 33nd International Conference on Machine Learning, ICML 2016; June 19-24, 2016; New York City, NY, USA. JMLR.org (2016). p. 2014–23.
21. Duan Y, Chen Z, Wei F, Zhou M, Shum H. Twitter Topic Summarization by Ranking Tweets Using Social Influence and Content Quality. In: M Kay C Boitet, editors. COLING 2012, 24th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers; 8-15 December 2012; Mumbai, India. Indian Institute of Technology Bombay (2012). p. 763–80.
22. Zheng J, Cai F, Chen H. Incorporating Scenario Knowledge into A Unified Fine-Tuning Architecture for Event Representation. In: J Huang, Y Chang, X Cheng, J Kamps, V Murdock, J Wen, et al. editors. Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020; July 25-30, 2020; Virtual Event, China. ACM (2020) p. 249–58. doi:10.1145/3397271.3401173
23. Ramos J. Using Tf-Idf to Determine Word Relevance in Document Queries. *Proc first instructional Conf machine Learn (Citeseer)* (2003) 242:29–48.
24. Perozzi B, Al-Rfou R, Skiena S. Deepwalk: Online Learning of Social Representations. In: SA Macskassy, C Perlich, J Leskovec, W Wang,

R Ghani, editors. The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14; August 24 - 27, 2014; New York, NY, USA. ACM (2014). p. 701–10.

25. Beamer S, Asanovic K, Patterson D. Direction-Optimizing Breadth-First Search. In: SC'12: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis, Salt Lake City, UT, November 11–15, 201. IEEE (2012). p. 1–10. doi:10.1109/sc.2012.50
26. Huang Z, Xu W, Yu K. *Bidirectional Lstm-Crf Models for Sequence Tagging*. arXiv preprint arXiv:1508.01991 (2015).
27. Kipf TN, Welling M. Semi-Supervised Classification with Graph Convolutional Networks. Conference Track Proceedings (OpenReview.net). In: 5th International Conference on Learning Representations; April 24-26, 2017; Toulon, France. ICLR 2017 (2017).
28. Ugander J, Backstrom L, Marlow C, Kleinberg J. Structural Diversity in Social Contagion. *Proc Natl Acad Sci* (2012) 109:5962–6. doi:10.1073/pnas.1116502109
29. Lerman K, Ghosh R, Surachawala T. *Social Contagion: An Empirical Study of Information Spread on Digg and Twitter Follower Graphs*. CoRR abs/1202.3162 (2012).
30. Hogg T, Lerman K. Social Dynamics of Digg. *EPJ Data Sci* (2012) 1:5. doi:10.1140/epjds5
31. Zhang J, Tang J, Li J, Liu Y, Xing C. Who Influenced You? Predicting Retweet via Social Influence Locality. *ACM Trans Knowl Discov Data* (2015) 9:1–26. doi:10.1145/2700398
32. Zhang J, Liu B, Tang J, Chen T, Li J. Social Influence Locality for Modeling Retweeting Behaviors. In: F Rossi, editor. IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence; August 3-9, 2013; Beijing, China. IJCAI/AAAI (2013). p. 2761–7.
33. Le QV, Mikolov T. Distributed Representations of Sentences and Documents. vol. 32 of JMLR Workshop and Conference Proceedings. In: Proceedings of the 31th International Conference on Machine Learning, ICML 2014; 21-26 June 2014; Beijing, China. JMLR.org (2014). p. 1188–96.
34. Fan R-E, Chang K-W, Hsieh C-J, Wang X-R, Lin C-J. Liblinear: A Library for Large Linear Classification. *J machine Learn Res* (2008) 9:1871–4. doi:10.1145/1390681.1442794
35. Agarap AF. *Deep Learning Using Rectified Linear Units (Relu)*. arXiv preprint arXiv:1803.08375 (2018).
36. Zhang Z. Improved Adam Optimizer for Deep Neural Networks. In: 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS). IEEE (2018). p. 1–2. doi:10.1109/iwqos.2018.8624183
37. Gao L, Liu Y, Zhuang H, Wang H, Zhou B, Li A. Public Opinion Early Warning Agent Model: A Deep Learning cascade Virality Prediction Model Based on Multi-Feature Fusion. *Front Neurorobot* (2021) 15:674322. doi:10.3389/fnbot.2021.674322
38. Li M, Wang X, Gao K, Zhang S. A Survey on Information Diffusion in Online Social Networks: Models and Methods. *Information* (2017) 8:118. doi:10.3390/info8040118
39. Lee RK-W, Hoang T-A, Lim E-P. Discovering Hidden Topical Hubs and Authorities in Online Social Networks. In: M Ester D Pedreschi, editors. Proceedings of the 2018 SIAM International Conference on Data Mining, SDM 2018; May 3-5, 2018; San Diego, CA, USA. San Diego Marriott Mission ValleySIAM (2018). p. 378–86. doi:10.1137/1.9781611975321.43
40. Cha M, Haddadi H, Benevenuto F, Gummadi PK. In: WW Cohen S Gosling, editors. Measuring user influence in twitter: The million follower fallacy. In Proceedings of the Fourth International Conference on Weblogs and Social Media, ICWSM 2010; May 23-26, 2010; Washington, DC, USA. The AAAI Press (2010).
41. Qiu J, Chen Q, Dong Y, Zhang J, Yang H, Ding M, et al. GCC: Graph Contrastive Coding for Graph Neural Network Pre-training. In: R Gupta, Y Liu, J Tang, BA Prakash, editors. KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining; August 23-27, 2020; Virtual Event, CA, USA. ACM (2020). p. 1150–60.
42. 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
43. Tian S, Mo S, Wang L, Peng Z. Deep Reinforcement Learning-Based Approach to Tackle Topic-Aware Influence Maximization. *Data Sci Eng* (2020) 5:1–11. doi:10.1007/s41019-020-00117-1
44. Singla P, Richardson M. Yes, There Is a Correlation: - from Social Networks to Personal Behavior on the Web. In: J Huai, R Chen, H Hon, Y Liu, W Ma, A Tomkins, et al. editors. Proceedings of the 17th International Conference on World Wide Web, WWW 2008; April 21-25, 2008; Beijing, China. ACM (2008). p. 655–64.
45. Hu J, Meng K, Chen X, Lin C, Huang J. Analysis of Influence Maximization in Large-Scale Social Networks. *SIGMETRICS Perform Eval Rev* (2014) 41:78–81. doi:10.1145/2627534.2627559
46. Liu L, Tang J, Han J, Jiang M, Yang S. Mining Topic-Level Influence in Heterogeneous Networks. In: J Huang, N Koudas, GJF Jones, X Wu, K Collins-Thompson, A An, editors. Proceedings of the 19th ACM Conference on Information and Knowledge Management, CIKM 2010; October 26-30, 2010; Toronto, Ontario, Canada. ACM (2010). p. 199–208. doi:10.1145/1871437.1871467
47. Wang X, Guo Z, Wang X, Liu S, Jing W, Liu Y. Nnmlinf: Social Influence Prediction with Neural Network Multi-Label Classification. In: Proceedings of the ACM Turing Celebration Conference-China (2019). p. 1–5.
48. Luceri L, Braun T, Giordano S. Social Influence (Deep) Learning for Human Behavior Prediction. In: S Cornelius, K Coronges, B Gonçalves, R Sinatra, A Vespignani, editors. *International Workshop on Complex Networks*. Cham: Springer (2018). p. 261–9. doi:10.1007/978-3-319-73198-8_22
49. Hamilton WL, Ying Z, Leskovec J. Inductive Representation Learning on Large Graphs. In: I Guyon, U von Luxburg, S Bengio, HM Wallach, R Fergus, SVN Vishwanathan, et al. editors. Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017; December 4-9, 2017; Long Beach, CA, USA (2017). p. 1024–34.
50. Wu Z, Pan S, Chen F, Long G, Zhang C, Yu PS. A Comprehensive Survey on Graph Neural Networks. *IEEE Trans Neural Netw Learn Syst.* (2021) 32:4–24. doi:10.1109/tnnls.2020.2978386
51. Wang X, Ji H, Shi C, Wang B, Ye Y, Cui P, et al. Heterogeneous Graph Attention Network. In: L Liu, RW White, A Mantrach, F Silvestri, JJ McAuley, R Baeza-Yates, et al. editors. The World Wide Web Conference, WWW 2019; May 13-17, 2019; San Francisco, CA, USA. ACM (2019). p. 2022–32. doi:10.1145/3308558.3313562
52. Yu L, Shen J, Li J, Lerer A. *Scalable Graph Neural Networks for Heterogeneous Graphs*. CoRR abs/2011.09679 (2020).
53. Fu X, Zhang J, Meng Z, King I. MAGNN: Metapath Aggregated Graph Neural Network for Heterogeneous Graph Embedding. In: Y Huang, I King, T Liu, M van Steen, editors. WWW '20: The Web Conference 2020; April 20-24, 2020; Taipei, Taiwan. ACM/IW3C2 (2020). p. 2331–41. doi:10.1145/3366423.3380297
54. Yun S, Jeong M, Kim R, Kang J, Kim HJ. Graph Transformer Networks. In: HM Wallach, H Larochelle, A Beygelzimer, F d'Alché-Buc, EB Fox, R Garnett, editors. Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019; December 8-14, 2019; Vancouver, BC, Canada (2019). p. 11960–70.
55. Jin D, Huo C, Liang C, Yang L. Heterogeneous Graph Neural Network via Attribute Completion. In: J Leskovec, M Grobelnik, M Najork, J Tang, L Zia, editors. WWW '21: The Web Conference 2021; April 19-23, 2021; Virtual Event/Ljubljana, Slovenia. ACM/IW3C2 (2021). p. 391–400. doi:10.1145/3442381.3449914

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.

Copyright © 2022 Gao, Wang, Zhang, Zhuang and Zhou. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.