frontiers in Physics

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}.

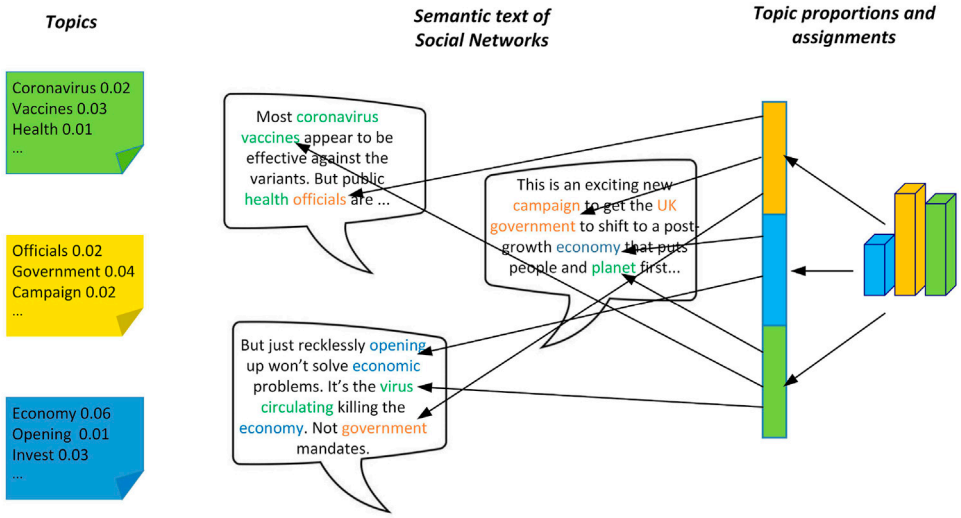

FIGURE 1 | LDA model for document generation process of semantic information published by users in social networks.

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 ith keyword in vector $\vec{w}$
$\vec{w}$	The vector of keywords
z_i	The ith 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

$$\begin{aligned} \vec{\alpha} &\rightarrow \vec{\theta}_d \rightarrow \vec{z} \\ \vec{\eta} &\rightarrow \vec{\beta}^{z_i} \rightarrow \vec{w} \end{aligned} \tag{1}$$

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:

$$\begin{aligned} &p\left(\vec{w}, \vec{z}\right) \propto p(\vec{w}, \vec{z}|\vec{\alpha}, \vec{\eta}) \\ &= p(\vec{z}|\vec{\alpha}) p(\vec{w}|\vec{z}, \vec{\eta}) = \prod_{d=1}^{|D|} \frac{\Delta(\vec{n}_d + \vec{\alpha})}{\Delta(\vec{\alpha})} \prod_{t=1}^{T} \frac{\Delta(\vec{n}_t + \vec{\eta})}{\Delta\left(\vec{\eta}\right)} \end{aligned} \tag{2}$$

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:

$$\begin{aligned} &p\left(z_i = o|\vec{z}_{\neg i}, \vec{w}\right) \propto p\left(z_i = o, w_i = y|\vec{z}_{\neg i}, \vec{w}_{\neg i}\right) \\ &= \int p\left(z_i = o, \vec{\theta}_d \mid \vec{w}_{\neg i}, \vec{z}_{\neg i}\right) p\left(w_i = y, \vec{\beta}_{z_i} \mid \vec{w}_{\neg i}, \vec{z}_{\neg i}\right) d\vec{\theta}_d d\vec{\beta}_{z_i} \\ &= \int p\left(z_i = o \mid \vec{\theta}_d\right) p\left(\vec{\theta}_d \mid \vec{w}_{\neg i}, \vec{z}_{\neg i}\right) p\left(w_i = y \mid \vec{\beta}_{z_i}\right) p\left(\vec{\beta}_{z_i} \mid \vec{w}_{\neg i}, \vec{z}_{\neg i}\right) d\vec{\theta}_d d\vec{\beta}_{z_i} \end{aligned} \tag{3}$$

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:

$$\int p(z_i = o|\vec{\theta}_d) Dirichlet(\vec{\theta}_d|\vec{n}_{d,\neg i} + \vec{\alpha}) d\vec{\theta}_d \cdot \int p\left(w_i = y \mid \vec{\beta}_{z_i}\right)$$

$$Dirichlet\left(\vec{\beta}_{z_i}|\vec{n}_{t,\neg i} + \vec{\eta}\right) d\vec{\beta}_{z_i} = \frac{n^o_{d,\neg i} + \alpha_o}{\sum_{t=1}^{T} n^t_{d,\neg i} + \alpha_t}.$$

$$\frac{n^y_{k,\neg i} + \eta_y}{\sum_{h=1}^{H} n^h_{k,\neg i} + \eta_h} \Rightarrow p(z_i = o|\vec{w}, \vec{z}_{\neg i}) \propto \frac{n^o_{d,\neg i} + \alpha_o}{\sum_{t=1}^{T} n^t_{d,\neg i} + \alpha_t} \cdot \frac{n^y_{k,\neg i} + \eta_y}{\sum_{h=1}^{H} n^h_{k,\neg i} + \eta_h} \tag{5}$$

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 $y(h)$. $n^k_{d,\neg i}$ 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 ith row and jth column of the data matrix X, and $\theta^{z_j}_{d_i}$ denotes the probability that document d_i belongs to the jth topic. Therefore, when the probability of the jth topic in the topic distribution θ_{d_i} is zero, $X = 0$; when the probability of the jth 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 ith row and jth 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 ith row of A, which represents the similarity value between user i and other users. ε is the sparse adjustment factor. $\mathbf{1}$ 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:

$$\hat{a}_{i,j} = \begin{cases} \dfrac{c_{i,m+1} - c_{i,j}}{m c_{i,m+1} - \sum_{h=1}^{m} c_{i,h}} & j \leq m \\ 0 & j > m \end{cases} \tag{8}$$

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:

$$\begin{aligned} &\min_U \|A-U\|_\sigma \\ &s.t. \mathbf{1}^T\vec{u}_i = 1, u_{i,j} \geq 0, \mathrm{rank}\,(L) = n-k \end{aligned} \tag{11}$$

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:

$$\begin{aligned} &\sum_{i=1}^k \lambda_i(L) = \min \mathrm{Tr}\left(F^TLF\right) \\ &s.t. F^TF = I \end{aligned} \tag{13}$$

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:

$$\begin{aligned} &\min_{U,F} \|A-U\|_\sigma + \rho\mathrm{Tr}\left(F^TLF\right) \\ &s.t. \mathbf{1}^T\vec{u}_i = 1, u_{i,j} \geq 0, F^TF = I \end{aligned} \tag{14}$$

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 $\sum_{i,j} \frac{1}{2}\|\vec{f}_i - \vec{f}_j\|_2^2 s_{i,j} = \mathrm{Tr}\,(F^TLF)$. 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:

$$\begin{cases} \forall j, s_i\hat{u}_{i,j} - h_{i,j} - \hat{\varphi} - \hat{\xi}_j = 0 \\ \forall j, \hat{\varphi} \geq 0 \\ \forall j, \hat{\xi}_j \geq 0 \\ \forall j, \hat{u}_{i,j}\hat{\xi}_j = 0 \end{cases} \tag{19a}$$

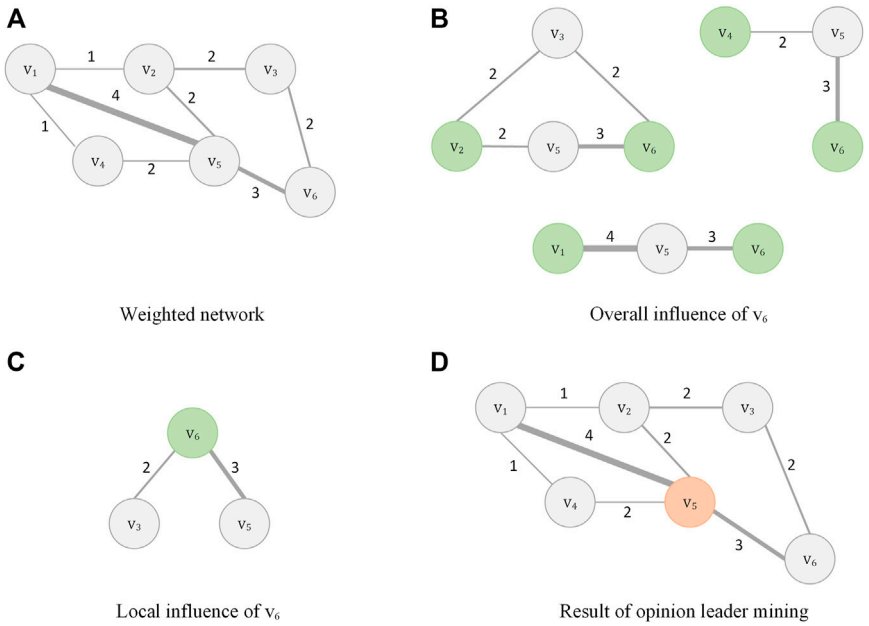

FIGURE 2 | An example of online opinion leader mining method based on global and local influence, **(A)** is weighted network. **(B)** is overall influence of v_6, **(C)** is local influence of v_6, **(D)** is result of opinion leader mining.

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 ξ^*. As can be seen, $\hat{\xi}^*$ is determined by solving the root finding problem when $f(\hat{\xi}^*) = 0$. Since $\xi^* \geq 0$, $f'(\xi^*) \leq 0$ 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:

$$\begin{aligned} &\min_F \mathrm{Tr}(F^T LF) \\ &s.t. F^T F = I \end{aligned} \tag{24}$$

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 ρ, σ.
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 θ_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:*

$$M(v_i) = \left\{v_j | v_j \in V, \exists e_{v_i,v_j} \in E\right\}, v_i \in V \tag{25}$$

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:*

$$PN\left(v_i, v_j\right) = \left\{v_k \in V, v_k = M(v_i) \cap M\left(v_j\right)\right\}, v_i, v_j \in V \tag{26}$$

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:*

$$\bar{d}(v_i) = \frac{\sum_{v_j = M(v_i)} d\left(v_j\right)}{d(v_i)} \tag{30}$$

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:*

$$\bar{\omega}(v_i) = \frac{\sum_{v_j=M(v_i)} \omega(v_j)}{\omega(v_i)} \tag{31}$$

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:

$$Total(v_i) = d(v_i) \sum_{v_j \in V} pow\left(B, \left|PN(v_i, v_j)\right|\right) \tag{34}$$

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:

$$Total'(v_i) = \omega(v_i) \sum_{v_j \in V} pow\left(B, \left|PN(v_i, v_j)\right|\right) \tag{35}$$

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:

$$Local(v_i) = \sum_{v_j=M(v_i)} C_D(v_j) p(v_j) \tag{36}$$

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:

$$Local'(v_i) = \sum_{v_j=M(v_i)} C_D^{\omega}(v_j) p'(v_j) \tag{37}$$

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

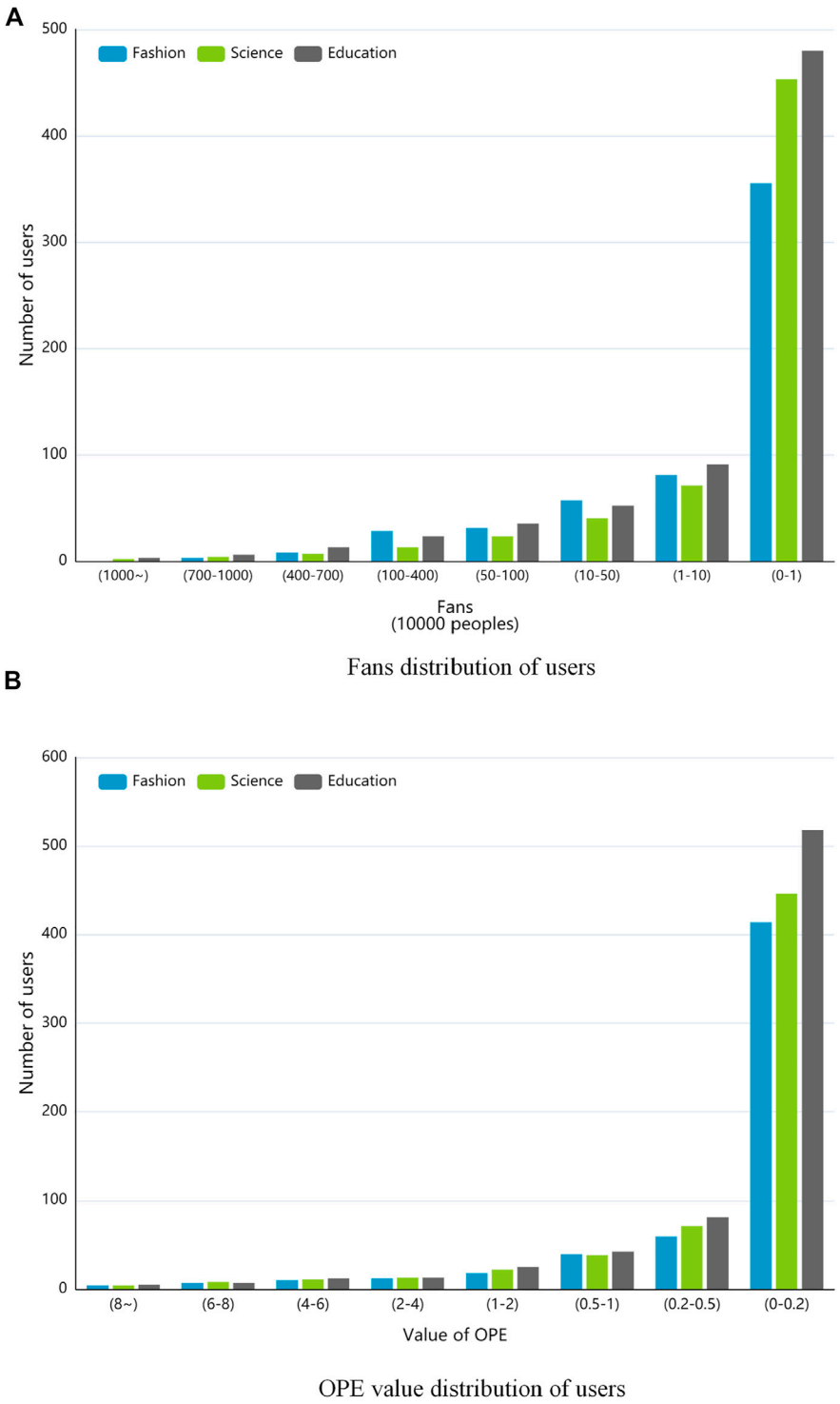

FIGURE 3 | Distribution of fans and *OPE* in micro-blog dataset, **(A)** is fans distribution of users. **(B)** is *OPE* value distribution of users.

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

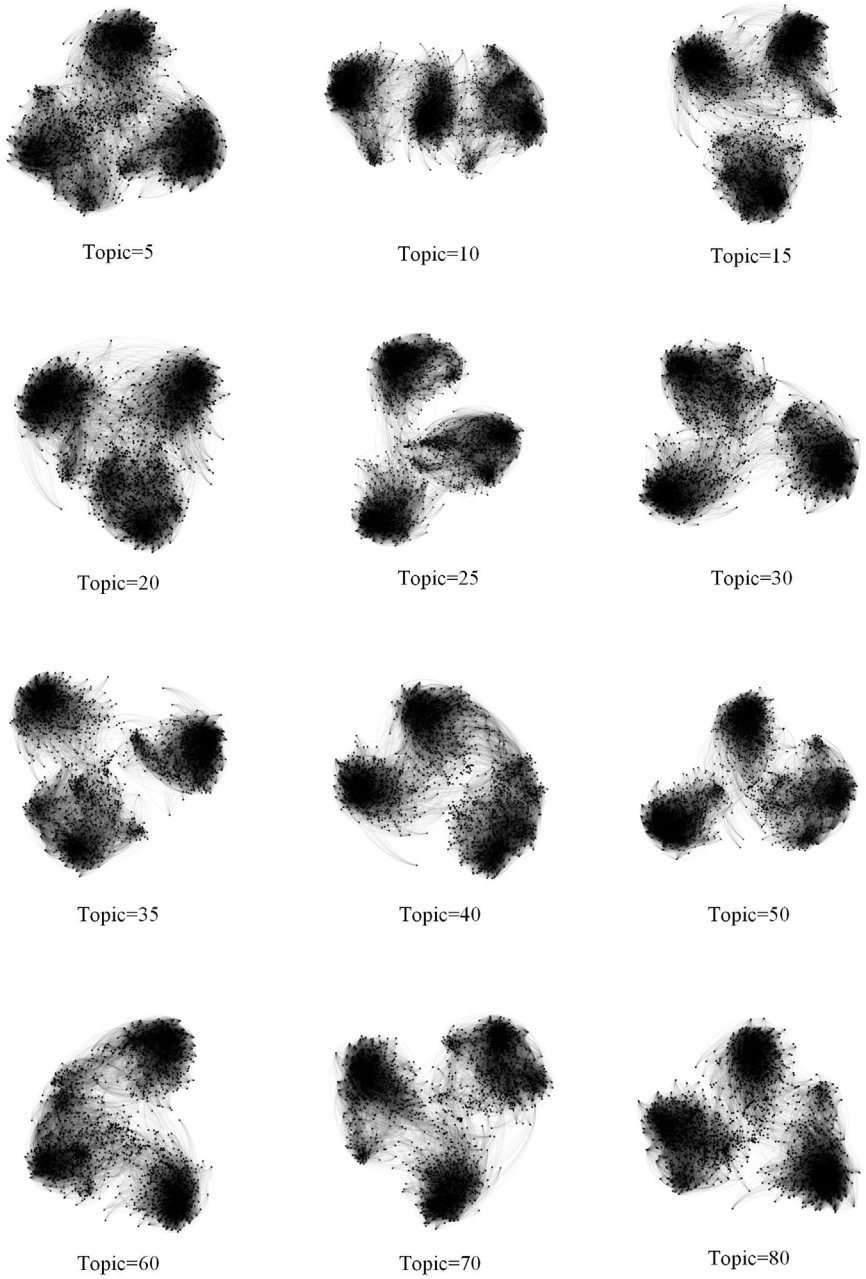

FIGURE 4 | Relationship network diagram of different topic numbers.

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 (τ = 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

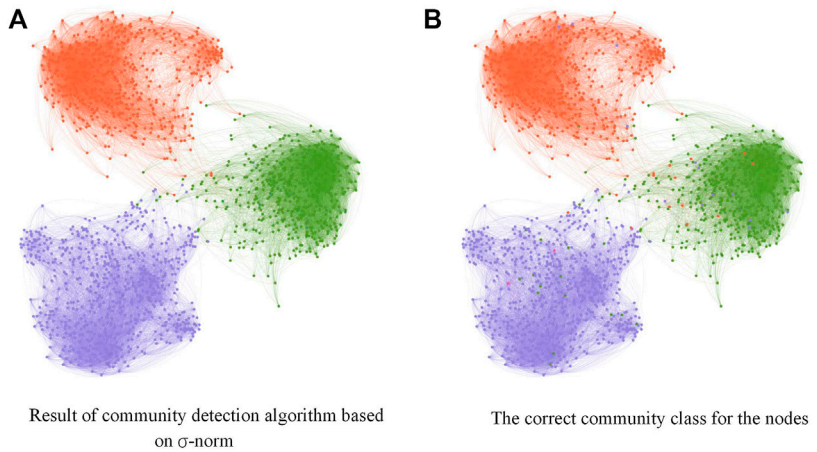

FIGURE 5 | The results of community detection, **(A)** is the result of community detection algorithm based on σ-norm, **(B)** is the correct community class for all nodes.

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

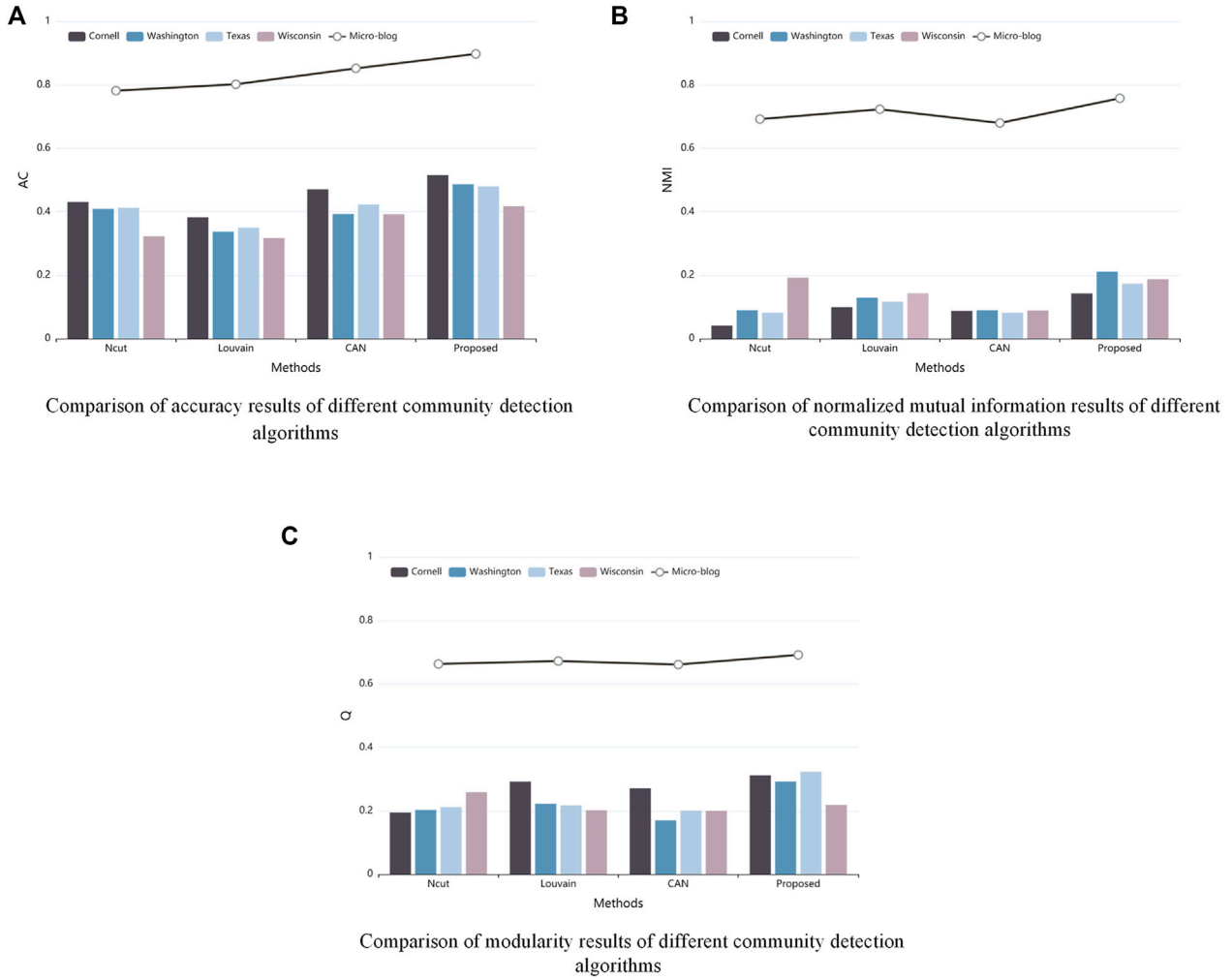

FIGURE 6 | Comparative analysis of WebKB dataset and Micro-blog dataset based on AC, NMI and Q, **(A)** is comparison of accuracy results for different community detection algorithms, **(B)** is comparison of normalized mutual information results for different community detection algorithms, **(C)** 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

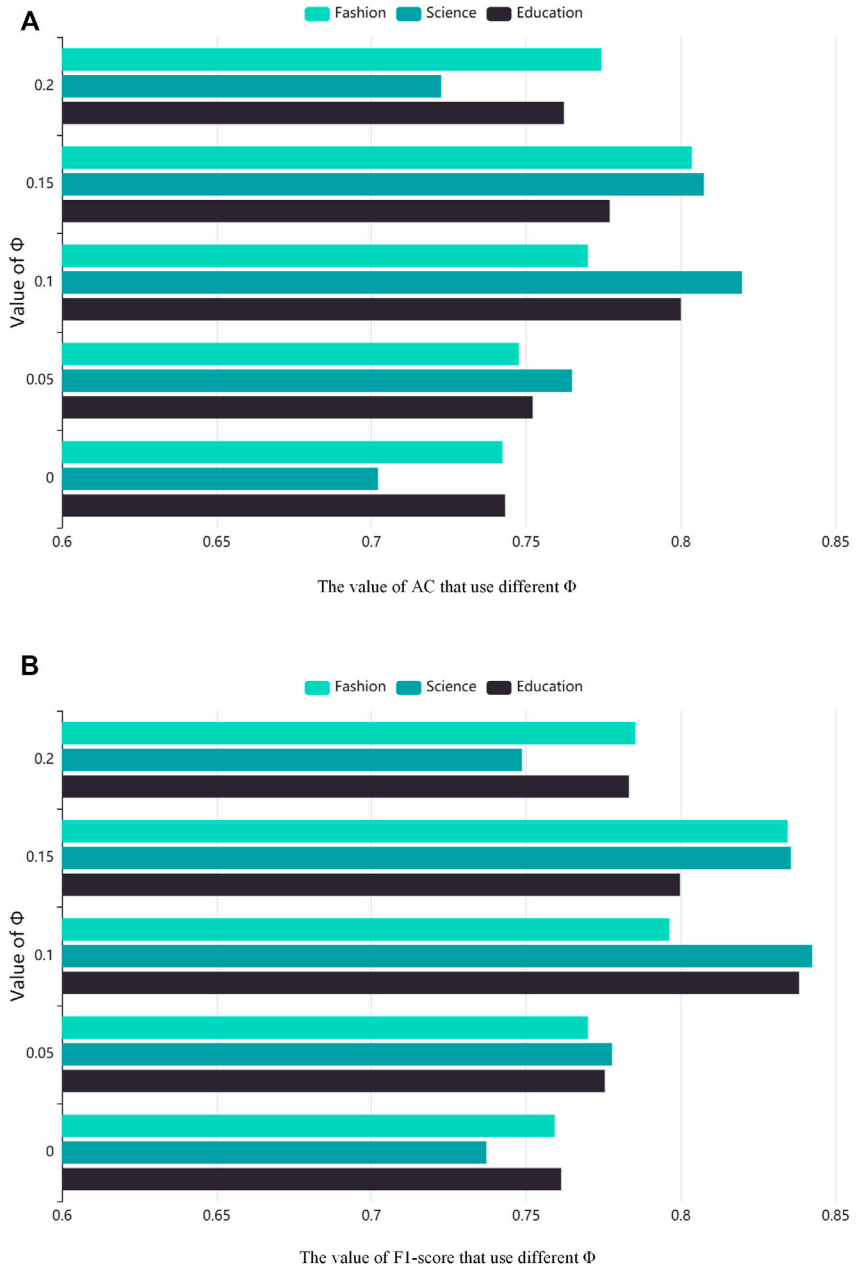

FIGURE 7 | Accuracy and F1-score analysis of opinion leaders mining with different similarity thresholds ϕ, **(A)** is the value of AC that use different ϕ, **(B)** is the value of F1-score that use different ϕ.

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

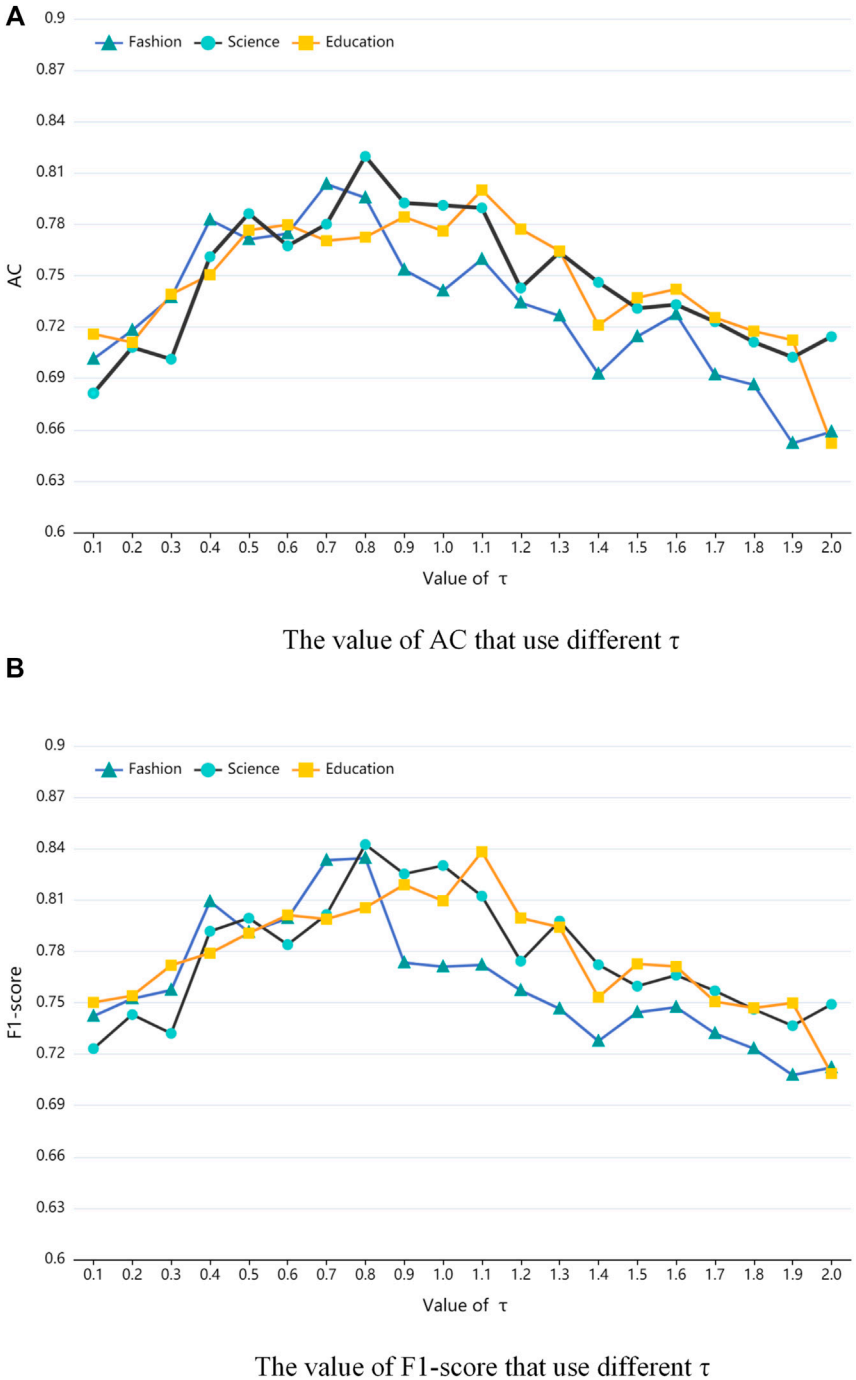

FIGURE 8 | Accuracy and F1-score analysis of opinion leaders mining with different positive tuning parameters τ, **(A)** is the value of AC that use different τ, **(B)** is the value of F1-score that use different τ.

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.