Networks and Heterogeneous Media

http://www.aimspress.com/journal/nhm

NHM, 18(4): 1539–1574.
DOI: 10.3934/nhm.2023068
Received: 18 May 2023
Revised: 18 June 2023
Accepted: 05 July 2023
Published: 19 July 2023

Research article

Influence maximization in social networks using role-based embedding

Xu Gu[1], Zhibin Wang[1], Xiaoliang Chen[1,2,*], Peng Lu[2], Yajun Du[1] and Mingwei Tang[1]

[1] School of Computer and Software Engineering, Xihua University, Chengdu, 610039, China

[2] Department of Computer Science and Operations Research, University of Montreal, Montreal, QC H3C3J7, Canada

* **Correspondence:** Email: chexiaol@iro.umontreal.ca, chenxl@mail.xhu.edu.cn.

Abstract: Influence maximization (IM), a central issue in optimizing information diffusion on social platforms, aims to spread posts or comments more widely, rapidly, and efficiently. Existing studies primarily focus on the positive effects of incorporating heuristic calculations in IM approaches. However, heuristic models fail to consider the potential enhancements that can be achieved through network representation learning techniques. Some recent work is keen to use representation learning to deal with IM issues. However, few in-depth studies have explored the existing challenges in IM representation learning, specifically regarding the role characteristics and role representations. This paper highlights the potential advantages of combining heuristic computing and role embedding to solve IM problems. First, the method introduces role granularity classification to effectively categorize users into three distinct roles: opinion leaders, structural holes and normal nodes. This classification enables a deeper understanding of the dynamics of users within the network. Second, a novel role-based network embedding (RbNE) algorithm is proposed. By leveraging the concept of node roles, RbNE captures the similarity between nodes, allowing for a more accurate representation of the network structure. Finally, a superior IM approach, named RbneIM, is recommended. RbneIM combines heuristic computing and role embedding to establish a fusion-enhanced IM solution, resulting in an improved influence analysis process. Exploratory outcomes on six social network datasets indicate that the proposed approach outperforms state-of-the-art seeding algorithms in terms of maximizing influence. This finding highlights the effectiveness and efficacy of the proposed method in achieving higher levels of influence within social networks. The code is available at https://github.com/baiyazi/IM2.

Keywords: deep learning; social network; influence maximization; role embedding; node role division

1. Introduction

Social networking is currently sustaining the exchange of information among individuals, societies, and nations. An ever-increasing number of individuals tend to share their experiences and comments partially or entirely through online media platforms such as Weibo [1] and Facebook. The privacy features of social networks can remove communication barriers between individuals, allowing them to express themselves candidly and openly. Anyone can freely express and share their feelings [2], assess others' perspectives, and acknowledge supported opinions on social media platforms. Central to the entire communication discipline in social networks stems from viral marketing [3]. This strategic approach to information diffusion has been widely adopted in various domains, including product promotion [4], personalized recommendations [5], targeted advertising [6], the selection of influential users [7–10].

Information diffusion phenomena in social networks has brought about both immense convenience and potential threats in the dissemination of groundbreaking ideas. The perceptions of individuals within a social network have the ability to influence the commenting behavior and awareness of their neighboring users, leading to intermittent changes in the network topology. These individual perceptions, known as user influence, are critical for understanding user behavior, uncovering network propagation dynamics, and examining topology evolutions. The influence maximization (IM) problem, which was formulated by Kempe et al. [11], is a challenging issue that has been proven to be NP-hard. Cecilia et al. [12] discovered that examining the citizenship competencies plays an important role in a complex system like a society, and it is crucial to study their effects given the significance of these competences in shaping social systems. Within the domain of social network analysis, IM emerges as a pivotal undertaking involving carefully selecting a seed group within a given social network to maximize its influence on a broad spectrum of individuals. This optimization problem holds profound significance for the process of refining information diffusion strategies to accomplish diverse objectives, which can range from viral marketing and opinion-shaping to social mobilization. When executed adeptly, IM engenders remarkable enhancements in the efficiency and effectiveness of such campaigns. Notably, within the context of viral marketing, IM facilitates the identification of potential customers, thereby curtailing marketing costs and bolstering profits. Furthermore, IM plays a pivotal role in molding public opinion across various domains such as politics, health, and the environment, while also galvanizing individuals for social causes encompassing protests, donations, and petitions. By harnessing IM effectively, substantial dividends can be reaped, efficiently leveraging the power of social networks to accomplish a multitude of objectives. By employing diffusion cascades, it becomes possible to optimize the reach of influence for the chosen seed set [13, 14]. Identifying influential nodes within social networks offers invaluable insight into the underlying mechanisms that govern information diffusion phenomena, thereby informing effective strategies for message propagation. Moreover, the exploration of IM contributes to the development of novel algorithms and techniques that maximize the extent and impact of information dissemination in social networks.

An independent cascade (a stochastic diffusion model) is generally employed in IM to simulate the dissemination of information by seed nodes. The spread of influence is commonly measured in terms of the number of activated users. However, the majority of research in IM has focused on stochastic diffusion patterns, and few studies have explored the global-scale role approximation of social

network users. In reality, each user in a social network plays a specific role, whether as an opinion leader, a structural hole user, or an ordinary user. All of them contribute to the overall diffusion of information. This study takes into account the social reality that users with similar roles in a social network exhibit comparable behaviors and attributes. For instance, structural hole users can facilitate the exchange between two communities in a social network, while ordinary users typically receive information passively. By identifying user roles in a social network, researchers can better understand the mechanisms behind information diffusion phenomena and develop more accurate models for solving IM problems.

In recent years, several user role identification studies have recommended the knowledge contribution approach proposed in [15]. This approach identifies three user roles, namely givers, takers, and matchers, based on their knowledge contribution to disseminated information. Research context in light electric vehicle applications promotes the differentiation of more roles such as the vigilant user, passive collaborator, active decision-maker, and ambassador [16]. This four-dimensional role classification, determined by participation degrees, is particularly important in the service promotion of electric vehicles. Some studies [17, 18] have emphasized the importance of user role division in the information diffusion process. However, these studies have yet to consider the key factor of user roles in the existing IM research. Furthermore, network structure information has not been fully incorporated into studies on user role division.

The study delves into a novel network embedding algorithm that integrates user role information, thereby adding a fresh and invaluable perspective to the realm of efficient IM solutions. In particular, the proposed approach utilizes network embedding's benefits to incorporate similarities between users' global roles, and is hence termed as role-based network embedding (RbNE). This methodology entails the mapping of social network nodes into a fixed-dimensional space, whereby they are represented as low-dimensional vectors that capture both the structural and user role information. After calculating the embedding vectors of nodes, a novel logical propagation network can be constructed based on their similarities. Finally, a greedy heuristic algorithm is introduced to select a seed set of k nodes.

The main contributions of our work are enumerated as follows:

1. To address the existing gap in the literature where network structure information is not fully incorporated into role division analyses, we propose a novel user role division algorithm that incorporates both coarse-grained user role division (CGURD) and fine-grained user role division (FGURD). CGURD aims to divide users into different groups based on their overall contribution to the network, while FGURD focuses on identifying more precise user roles based on their relationships with neighboring nodes. By combining these two approaches, we can achieve a more comprehensive understanding of user roles in social networks, which can provide new insights into the IM problem.
2. Current research has not examined the effects of the user's global roles in efforts to solve the IM problem. A systematic understanding of how the global roles contribute to capturing the seed set with the most significant influence is still lacking. Therefore, this paper proposes a novel network embedding algorithm, entitled RbNE, to preserve the approximation between users' global roles. Subsequently, a greedy heuristic algorithm, named RbneIM, is developed to select the seed set $\mathcal{S}$. This algorithm considers the users' global roles as an essential factor in selecting the seed set and enables the identification of users who can maximize the influence spread in a social network.
3. Our study extensively evaluates the performance of RbneIM on four real-world datasets. The

experimental results indicate that our approach significantly outperforms the state-of-the-art methods, thereby demonstrating its superior performance in terms of solving the IM problem.

2. Related work

2.1. Social networks and linear threshold models

The task of IM was initially specified in social networks. Accordingly, this paper follows the same applied background. A social network can be represented by a two-tuple $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{v_1, v_2, ..., v_n\}$ and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V} = \{e_{ij}\}$ represent a set of n nodes and a set of edges between the nodes, respectively. An edge $e_{ij} = (v_i, v_j)$ indicates a potential relationship between nodes v_i and v_j, which is also associated with a weight $w_{ij} \geqslant 0$ of their connection strength. Table 1 summarizes the notations employed in this paper.

Table 1. Notations used in this paper.

Notation	Description of notation
$\mathcal{V}$	The set of nodes in a social network
$\mathcal{E}$	The set of edges in a social network
$\mathcal{A}$	User characteristic matrix $A_{\|V\|\times D}$
$\mathcal{S}$	Selected set of seed nodes, $S \subset \mathcal{V}$
$Nei(u)$	Direct neighbors of node u
Inf_{uv}	Local influence of node u on node v
$B(u)$	Betweenness centrality of node u
OLI_u	Opinion leader influence score of node u
$NC(u)$	Network constraint coefficient value of node u
$Sim(u, v)$	Relationship between node u and node v
$Pr(V \mid v)$	Probability of being activated by v's neighbors
$Pr(V \mid S)$	Probability of being activated by seed nodes $\mathcal{S}$

In social networks, many spreading processes can be modeled as complex chain reactions. Kempe et al. [19] introduced two probability diffusion models, namely independent cascade (IC) and linear threshold (LT), to explain these processes. A framework based on submodular functions is proposed to analyze the performance guarantees of algorithms for influence problems. Kempe et al. show that a greedy strategy can be within 63% of optimal for several classes of models, and they present computational experiments that demonstrate the superiority of their approximation algorithms over other node-selection heuristics. Among them, LT models are commonly involved as an extremely representative framework for understanding these mechanisms. This study evaluates the effectiveness of the seed node selection using the LT model [20]. We assume that a node $v \in \mathcal{V}$ of a social network G is influenced by each incoming neighbor weight $w \in [0, 1]$, which contributes to the idea behind the LT. An inactive user becomes active once the number of its neighbors reaches a certain threshold θ of active states. Specifically, each node v in a social network G has a threshold θ in the interval $[0, 1]$, representing the conditional value of the node v being activated. The activation of node v can be formalized as follows:

$$\sum_{u \in Nei(v)} w_{uv} \geqslant \theta \tag{2.1}$$

where $Nei(v)$ refers to the direct neighbors of node u. Specifically, in the LT model, each node has two attributes: threshold and weight. A node's threshold is a value between 0 and 1, indicating how many active neighbors it needs to become activated. The weight of a neighbor is also a value between 0 and 1, representing the neighbor's influence on the node. If the threshold of a node i is t and the neighbor set is N, then the node i will be activated under the following condition:

$$\sum (j \in N, \textit{the total weight of } j \textit{ when activated}) \geq t \tag{2.2}$$

where the total weight of j when activated refers to the sum of all edge weights connecting to node j.

2.2. *Influence maximization*

In social network analysis, IM is a crucial concept that involves selecting a subset of nodes or edges in a given network to maximize the impact of a particular objective. Individual characteristics and the effects between individuals are expressed in the form of social network topology. Hence, influence has both global and local scopes. A node with stronger global influence in a network has the ability to control the spread of information and behavior in the network. Moreover, a small subset of highly influential nodes in a social network can control the propagation of most of the information. Thus, selecting the correct seed nodes is crucial to achieving maximum impact in the spread of information in a social network. IM technology is useful for identifying critical nodes to maximize the impact of information diffusion throughout the network. Compared to a random selection of nodes, IM technology can predict and quantify influence, leading to more effective resource allocation and planning strategies. A node's influence on another node is considered local influence, and the more a node influences another node, the more the latter will follow and imitate the former's behavior in the social network. The process of defining node influence through local influence and network structure can yield better results, taking into account the requirements of different applications. Studies of dynamic social network nodes' influence has mostly been on static network topologies, examining users' influence or users' influence variation on static topologies over time.

The literature on IM problems can be traced back to a study by Domingos and Richardson [21]. Kempe et al. [19] then formulated the IM problem for the first time and presented two essential conclusions. First, IM issues can be modeled as a class of discrete optimization problems. Second, IM problems are NP-hard, which limits the development of existing IM approaches. Most of the current approaches rely on simple greedy calculations, traversing each node in the social network to calculate its marginal impact benefit. Nodes with larger marginal impact benefits are then included in the seed set $\mathcal{S}$. The formation process can be expressed as follows.

$$u \leftarrow arg \max_{u \in (\mathcal{V} \setminus \mathcal{S})} \sigma(\mathcal{S} \cup u) - \sigma(\mathcal{S}) \tag{2.3}$$

where the value of $\sigma(\mathcal{S})$ represents the propagation range of a node set $\mathcal{S}$, typically measured by the number of activated nodes. To construct the final seed set, greedy methods have been commonly employed to select the nodes with the highest influence benefit continuously. However, these methods have been criticized for their low efficiency and high time complexity, making them impractical for

large-scale networks. As a result, recent research has focused on improving the extensibility and efficiency of the IM problem.

Most research on IM is based on traditional propagation models such as IC and LT [20] and their variations. However, some studies have shown that the IC and LT models may not accurately approximate influence. To address this issue, Oriedi et al. [22] proposed a selective breadth-first traversal algorithm that efficiently generates an optimal seed set for IM. According to their argument, using models like the IC and LT models may result in an incorrect influence estimation. The authors have proposed an algorithm to create the best seed set for maximizing influence. They have tested their method using real data and proved that it is better than traditional IM algorithms. Oriedi et al. have effectively developed a more precise approach for modeling social network influence. Similarly, Sun et al. [23] introduced the self-Aactivation IC (SAIC) model that incorporates self-activation as an additional factor in influence propagation, where nodes can be self-activated and selected as seeds. They characterized two optimization problems arising from self-activation: preemptive IM (PIM) and boosted PIM (BPIM). Specifically, the PIM problem involves identifying nodes that can reach the most number of nodes before other self-activated nodes if self-activated. In contrast, the BPIM problem aims to select seeds that are guaranteed to reach the most number of nodes before other self-activated nodes. They proposed scalable algorithms for both PIM and BPIM to address these challenges and assessed their approximation guarantees. The results of their study indicate that the algorithms perform much better than baseline methods, particularly for the PIM problem and the BPIM problem when there are varying self-activation behaviors among nodes.

Guo and Wu [24] investigated adaptive influence maximization with multiple activations problems, which take into account that not all users are willing to become influencers in the seed set. The researchers addressed a problem wherein each user is connected with a probability of activation as a seed, allowing for multiple triggers. To mathematically model this scenario, Guo and Wu have proposed a novel concept called adaptive-dr-submodularity, defined on the domain of an integer lattice, to maximize an adaptive monotone and dr-submodular functions while satisfying the expected knapsack constraint. This problem has not been previously investigated in existing studies, necessitating a comprehensive exploration of its approximability. They have developed a strategy that combines an adaptive greedy policy with sampling techniques to tackle the challenge of estimating expected influence spread while maintaining the approximation ratio and reducing time complexity. Other related work can be found in [25–28]. Luo et al. [25] have proposed the iterative competitive opinion maximization model, which aims to maximize the total opinions in competitive scenarios by combining user opinions and rival strategies. Unlike existing IM approaches, this model effectively suppresses the propagation of negative opinions and identifies optimal responses to opponents' seed node choices. The authors employ an iterative inference algorithm based on the greedy strategy to reduce computational complexity and achieve optimal outcomes. Zhang and Zhang [26] investigated the computational complexity of IM and analyzed the approximation guarantee of the greedy algorithm within the generalized model. Their research introduces a coordination game model that offers a game-theoretic perspective on IM. This model extends existing frameworks such as the majority vote model and the LT model. Furthermore, the incorporation of strategies to improve the algorithm's performance represents a significant contribution to the existing body of literature. However, as mentioned in the introduction section, every user in a social network plays a role in disseminating information. The previous IM studies mentioned above ignore the global-scale role

approximation of users in the network. Liu et al. [27] introduced CONE, an active learning framework designed to address the estimation of user opinions in multi-round campaigns involving influence propagation. Their methodological approach to modeling user preference data is notable for their ability to handle scenarios in which prior knowledge of user opinions is unavailable. This approach holds practical implications, particularly in viral marketing, and including precision advertising and reputation management domains. Banerjee et al. [28] have presented a pioneering model, termed UIC, to overcome the existing constraints in the literature. The UIC model stands out by integrating users' economic factors into their product adoption and purchase decisions, aiming to maximize social welfare and foster customer loyalty within the network. Additionally, the authors shed light on the underexplored realm of complementary items, which has received limited scrutiny in previous studies on multiple items.

Several recent studies on IM have effectively utilized deep learning techniques to identify and evaluate user influence in social networks [29–36]. These studies have shown promising results in the area of improving the performance of IM algorithms. Keikha et al. [29] have presented a novel methodology to tackle the challenge of IM on interconnected networks, employing deep learning techniques. Their proposed algorithm harnesses the power of deep learning for feature engineering, allowing for the preservation of both local and global structural information. By showcasing monotonicity and submodularity, the algorithm provides an assurance of an optimal solution. Notably, this study pioneers the utilization of network embedding to address the IM problem, marking a significant advancement in the field. Zhan et al. [30] proposed a general framework called NE-IM that leverages representation learning to address computational cost and improve stability. NE-IM contains two components: structure-based embedding and feature-based embedding. Their work incorporates heterogeneous information in IM models and applies representation learning to improve the efficiency and accuracy of IM models. Tian et al. [31] proposed two topic-aware social influence propagation models based on IC and LT models and developed a deep influence evaluation model to evaluate the user influence under different circumstances. They encoded the feature of each node by a vector, which enabled them to construct a solution efficiently without considering the complex graph structure. Their network learns a generalized heuristic framework to solve the NP-hard TIM problem using meta-learning, without requiring specialized knowledge and improving advertising injections. Li et al. [32] have presented a framework aimed at maximizing market influence in the USA domestic air passenger transportation market by adjusting flight frequencies. They used neural networks to predict market influence while considering several features such as air carrier performance features and transportation network features. They integrated neural networks to predict market influence and developed an adaptive gradient ascent method for solving the nonlinear optimization problem in flight frequency optimization. Zhang et al. [33] designed a network dynamic GCN to extract the in-depth structural information of social networks for IM. The proposed algorithm utilizes a leader fake labeling mechanism to generate node labels that are helpful for seed node selection during training. Finally, a heuristic method based on the Mahalanobis distance was developed to select influential seed nodes with learned node representations. Li et al. [34] suggested a Gaussian propagation model based on social networks and a multi-dimensional space modeling approach for propagation simulation. Their approach uses an improved CELF algorithm to accelerate the IM algorithm and evaluate the proposed technique based on theoretical proofs. Li et al. [35] then proposed a new approach to IM in social networks that takes into account multi-dimensional characteristics such as user emotions and

group features. Specifically, Li et al. defined user emotion power and cluster credibility as measures of the interaction effects of individual emotions and proposed a potential influence user discovery algorithm based on an emotion aggregation mechanism to locate seed candidate sets. Li et al. [36] proposed a novel adaptive agent-based evolutionary approach to solve the IM problem in dynamic and large-scale social networks. A key component of the proposed approach is an adaptive solution optimizer that drives the evolutionary process and adapts candidate solutions dynamically. Motivated by the success of these works, our paper aims to take the next step and integrate the global role information of users in the final embedding vector using network embedding. By incorporating this additional information, our proposed method will further enhance the accuracy and effectiveness of IM in social networks.

2.3. Network representation learning

Network representation learning, also known as network embedding [37], has garnered significant attention in recent years. This technology aims to transform a network's features into a low-dimensional continuous representation matrix while retaining the network structure and inherent properties. Recent advances in deep learning have enabled researchers to generate node embeddings through social network analysis techniques, such as DeepWalk [38], LINE [39], and node2vec [40]. These techniques utilize a prearranged random walk strategy to construct a corpus that displays the connections between the network components while preserving the characteristics of the network structure. The SkipGram model [41] is employed to acquire the node vector representation of a network. This model uses the context of words to identify the underlying relationships between nodes in the network. After an extensive process, the low-dimensional embedding representations of the nodes in the network are established, allowing a greater understanding of the network structure and the underlying relationships between nodes. Although these random walk methods have been proven to achieve better performance in network embedding, they ignore nodes' global structure and properties. Analyzing these global structures and node characteristics is essential to understanding the network accurately. Consequently, a few investigators have started to integrate network exploration techniques with other node properties. For example, Keikha et al. [42] devised a network embedding algorithm, community aware random walk for network embedding (CARE), that aims to conserve the local neighborhood and community information of a network while maintaining its global structure.

Drawing on prior literature, we innovatively utilized random walks to extract the embedding matrix of the target network. Diverging from previous studies, our novel approach places emphasis on incorporating the user's global role information. This integration enables a comprehensive representation of roles and their local neighbors within the network. As a result, our methodology provides an enhanced perspective of the user's network position and their potential associations with other users.

3. Methodology

3.1. Analysis of user role division

Information dissemination is a complex process due to the dynamic influence of one user on another [43]. The structural attributes of users in a social network reflect their roles in different

communities. In this context, the primary challenge is to understand how the network structure affects the dissemination of information in a role-divided scenario. Most of the existing random walk methods based on the network structure only consider the influence of the direct domain nodes in a network, such as edge propagation probabilities between two nodes, while ignoring the roles of users in the network. Users with similar roles in the network tend to have similar structural attributes, and previous research methods have not accurately captured this feature.

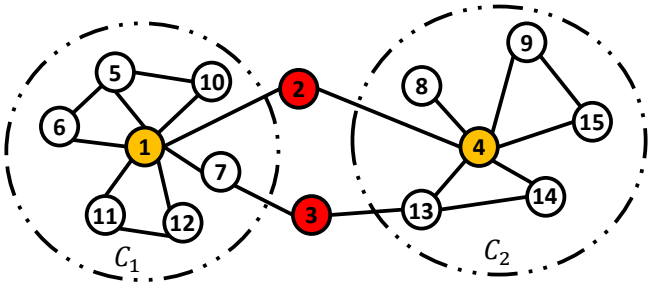


Figure 1. Case network.

Figure 1 shows a classic social network scenario. Each node in the network represents a user, and the connection table between the nodes indicates the relationship between the users. In addition, the shared colors (yellow or red) in the figure imply that these nodes have similar global roles. Two communities, labeled as C_1 and C_2, are also depicted in the figure, each having its own opinion leaders (yellow nodes 1 and 4) that usually have similar attributes, such as higher node degrees. Red nodes 2 and 3 span multiple communities and typically play a critical role in the exchange of community information. Such red users are generally called structural hole nodes in a social network. This example highlights two essential aspects of following users on social networks:

1. Similar user roles usually have similar attributes;
2. Different user roles have distinct functions in the exchange of information.

The global role information of users in the network plays a vital role in information dissemination. In the traditional random walk sampling process, first-order or high-order neighbor nodes are considered, but the similarity of users with similar global roles is overlooked. In network representation learning, we hope that similar nodes in the network will eventually have similar vector representations. Therefore, users with comparable global roles should have corresponding vector representations. However, conventional or biased random walks cannot accurately approximate the user's global role. This paper addresses this limitation by incorporating the user's global role into the traditional random walk process. By sampling from the training corpus, we obtain the vector representation of each node. In the upcoming research, the aim is to investigate the role division of users in the network. The problem will be approached from two perspectives. First, the focus will be on CGURD. Second, the issue of user role division will be addressed in greater detail.

In a given social network $G(V, E)$, it is possible to represent its attributes or structural characteristics using a matrix A of dimensions $|V| \times D$, where D represents the embedded dimension in $A_{|V|\times D}$. Users in the network with similar attributes or structural characteristics are expected to belong to the same role set. This study aims to map each user V_i to its corresponding role R_j in the set of user roles $R = R_1, R_2, ..., R_K$. It is assumed that user roles in the network can be classified into K categories

where K is significantly smaller than $|V|$, i.e., $k \ll |V|$. Our aim is to determine a mapping function $\phi : V_i \to R_k$ that can map each user to its role R based on its attributes or structural information.

In network representation learning, the concepts of coarse-grained and fine-grained refer to two distinct levels of abstraction concerning the network graph. Coarse-grained clustering aims to consolidate nodes to maximize their similarity within groups while minimizing the similarity between groups. This method typically produces larger clusters consisting of nodes with similar properties or roles in the network. Conversely, fine-grained clustering aims to group nodes with highly specific features or roles, resulting in smaller clusters with nodes possessing more precise properties or roles within the network.

Coarse-grained clustering typically yields smaller clusters comprising nodes with more specific properties or roles, facilitating the identification of larger-scale patterns and communities in the network. This approach is especially advantageous when computational efficiency is a priority. In contrast, fine-grained clustering focuses on identifying highly specific patterns or roles within the network. As a result, this approach may generate a larger number of clusters and require more computational resources. Nevertheless, the fine-grained approach offers valuable insights into the intricate structural properties and relationships present in the network.

3.1.1. CGURD

We can intuitively express the simple mapping of vector A_i by using either its cumulative sum or average value. Specifically, this can be expressed as follows:

$$\begin{aligned} \phi(x) &= R(x_1 + x_2 + x_3 + ... + x_D) \\ or &= R((x_1 + x_2 + x_3 + ... + x_D)/D) \end{aligned} \tag{3.1}$$

The vector $x = [A_{i1}, A_{i2}, A_{i3}, ..., A_{iD}]$ is obtained from the matrix A, where $R(x)$ represents the specific user role of x. This means that when a vector value x is input, its corresponding role category is output. It should be noted that if the matrix A represents the structural characteristics of users, such as the adjacency matrix of social network G, then the sum of features for each user represents the degree of its node. Intuitively, the mapping relationship described above is divided based on the node degree of users.

Unfortunately, Eq (3.1) is not interpretable if the matrix A represents the attribute characteristics of the nodes. Therefore, we need an alternative approach to address this issue. In this paper, we introduce the Non-negative matrix factorization (NMF) algorithm to handle this problem. The process of obtaining user roles from the user characteristic matrix can be represented in Figure 2:

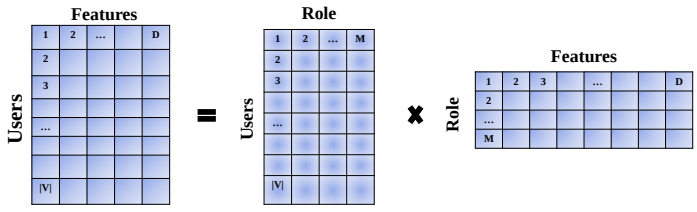


Figure 2. Diagram of NMF decomposition of user characteristic matrix A.

Based on the concept of NMF matrix decomposition, the dimensionality of the user characteristic

matrix $A_{|V|\times D}$ can be reduced using the following iterative formula:

$$A_{|V|\times D} \approx R_{|V|\times M} \times F_{M\times D} \tag{3.2}$$

In the NMF algorithm, we utilize the search matrix $R_{|V|\times M} = [r_1\ r_2\ ...\ r_M] \in \mathcal{R}+_{|V|\times M}$ and the coefficient matrix $F_{M\times D} = [f_1\ f_2\ ...\ f_D] \in \mathcal{R}+_{M\times D}$ to reduce the dimensionality of the user characteristic matrix $A_{|V|\times D}$. Among them, M is the number of basis vectors and is often much smaller than $|V|$ or D, i.e., $M \ll |V|, M \ll D$. In this paper, the matrix R is regarded as a user role matrix, where each row represents the role feature vector to which the user belongs. On the other hand, the F matrix represents the probability of each role to which each user belongs. We aim to minimize the loss function:

$$\begin{gathered}\mathcal{L}(R,F) = \arg\min_{R,F}(\frac{1}{2}\|A - RF\|^2 + \mathcal{R}(R,F)) \\ s.t.\ R, F \geq 0\end{gathered} \tag{3.3}$$

The final user role matrix R is obtained after minimizing the loss function with a regularization penalty $\mathcal{R}(R,F)$. Afterward, the matrix $R \in \mathcal{R}^{|V|\times M}$ is partitioned into K disjoint sets of nodes $V_1, V_2, ..., V_K$ by solving the k-means objective as follows:

$$\min_{V_i, i\in[1,K]} \sum_{j=1}^{K} \sum_{r_i\in V_j} \|r_i - c_j\|^2, \text{where } c_j = \frac{\sum_{u_i\in V_j} r_i}{|V_j|} \tag{3.4}$$

In the process described above, the user role division algorithm can be described as Algorithm 1.

3.1.2. FGURD

To analyze the characteristics of user roles during information dissemination in online social networks, this study classifies all users into three categories: opinion leaders, spanner holes, and ordinary users. The definition and primary properties of a node are illustrated in Figure 1. using a simple example.

Opinion leader: This refers to the minority of individuals at the core of the network who serve as a crucial source of information and influence within the community, capable of shaping the attitudes of the majority. As illustrated in the figure above, the yellow node is located at the center of the community to which it belongs, representing an opinion leader.

Structural hole users: Structural hole users are in a key position in the network but differ from opinion leader nodes in terms of high influence. They significantly impact the depth and breadth of information dissemination by acting as bridges between different communities. As demonstrated in the above figure, the red nodes represent such users.

Ordinary users: In the context of online social networks, ordinary users refer to those who do not possess the characteristics of structural holes or belong to a group of opinion leaders. Despite not having a central position in the network, ordinary users represent the majority of users and are considered edge users. They play a crucial role in information dissemination. In the presented figure, the white nodes depict ordinary users, emphasizing their significance as an integral part of the network. Although ordinary user nodes may not directly affect the global structure and evolution of the network

Algorithm 1 CGURD(A, K, max_iter)

Input:

The matrix of user's characteristic $A \in \mathcal{R}^{|V|\times D}$
Number of user role types K
The number of algorithm iterations $iter$

Output:

User role partition dictionary $RoleDic$

Initialize dictionary $RoleDic$ to empty
Initialize the value of M to satisfy $M \ll |V|, M \ll D$
Perform SVD decomposition on initial matrix A
Randomly initialize $R \in \mathcal{R}^{|V|\times M}$, $F \in \mathcal{R}^{M\times D}$
for $iter$ **in** range(1, max_iter + 1) **do**
    if $\mathcal{L}(R, F) \geq 1e-4$ **then**
        Update matrix R by gradient descent
        Update matrix F by gradient descent
    else
        break
    end if
end for
Perform k-means clustering by applying Eq (3.4).
Get node label and map to $RoleDic$
return $RoleDic$

like influential user nodes, they are also an indispensable part of the social network. Ordinary users nodes play the following roles in the benefits of social networks:

- Provide content: Ordinary users can provide rich and diverse content to social networks, attract more users to join the network and increase the value of the whole social network.
- Spreading information: Ordinary users can spread information and opinions by their own behavior, so they can help content and opinion diffusion by spreading and reposting even if they have no influence.
- Guide the diffusion of the network: Ordinary users establish their own social relationships, improve their exposure rate, and then attract more ordinary users like them to join the network, thus contributing to the prosperity and development of social networks.

Online social network users exhibit several significant traits, prompting the search for a viable approach to differentiate user roles based on these attributes.

(1) Opinion leader influence

The present study focuses on identifying opinion leaders within the network by analyzing both the local and global characteristics of users and studying each node's influence. In the information dissemination process, the local influence that node u has on node v is determined by two main factors. First, the influence of node u itself is typically evaluated by its degree within the social network.

Second, the number of mutual friends of node u that have an influence on node v which can be measured by the Jaccard coefficient.

$$Inf_{uv} = \alpha_1 D(u) + \alpha_2 \frac{|Nei(u) \cap Nei(v)|}{|Nei(u) \cup Nei(v)|} \tag{3.5}$$

where, Inf_{uv} represents the local influence of node u on node v. $D(u)$ is an expression related to 1-hop neighbors of user u, which is applied to measure the local influence of user u. $Nei(u)$ is the set of direct neighbors of u. Jaccard's coefficient $\frac{|Nei(u)\cap Nei(v)|}{|Nei(u)\cup Nei(v)|}$ is a widely-used measure to estimate the mutual friends of nodes u and v, and it is adopted to calculate the local influence of nodes u and v. The balance-parameters α_1 and α_2 satisfy $\alpha_1 + \alpha_2 = 1$. The formula to calculate $D(i)$ is as follows:

$$D(i) = \frac{Nei(i)}{\sum_{K \in G} Nei(k)} \tag{3.6}$$

To calculate the influence value for node i on its neighbors, the influence-gathering equation is used. This equation takes into account the local influence of the node i on node v and the influence of each node by other nodes. After obtaining the local influence of a node, we can calculate the influence value for node i on its neighbors using Eq (3.7):

$$Inf_i = \frac{1}{|N(i)|} \sum_{v \in N(i)} Inf_{vi} \tag{3.7}$$

where $N(i)$ is the neighbor set of node i. In terms of the global influence of nodes in social networks, this study focuses on the betweenness centrality of nodes.

$$\begin{aligned} B(u) &= \sum_{v,k,u \in V, v \neq k \neq u} \frac{P_{vuk}}{P_{vk}} \\ \hat{B}(u) &= \frac{B(u)}{\sum_{v \neq u} \sum_{k \neq u,v} \frac{P_{vuk}}{P_{vk}}} \end{aligned} \tag{3.8}$$

Equation (3.8) defines the betweenness centrality of a node, which is used to determine the global influence of nodes in social networks. Here, P_{vk} represents the number of shortest paths between two nodes v and k, while P_{vuk} represents the number of shortest paths between nodes v and k passing through node u. $\hat{B}(u)$ stands for the normalized global influence. Finally, we can obtain the betweenness centrality of node u.

This study combines local and global structural information to obtain the influence of node i in the entire network.

$$OLI_u = \beta_1 Inf_u + \beta_2 B(u) \tag{3.9}$$

where OLI_u represents the opinion leader influence OLI score of node u.

In order to achieve a balance between global and local influences in the calculation of OLI_u, the balance parameters β_1 and β_2 are utilized. These parameters are designed to adjust the relative weight of each factor, and they are subject to the constraint that their sum must equal 1. By combining global and local structural information, the opinion leader index of each node can be determined. The process of identifying opinion leaders in social networks is described in Algorithm 2.

As presented in Table 2, we employed diverse methods to assess the influence of nodes in the case network, which is illustrated in Figure 1.

Algorithm 2 OLI(G, α_1, α_2, β_1, β_2, k)

Input:

Network: $G(V, E)$
Node local influence balance parameters: α_1 and α_2
OLI balance parameters: β_1 and β_2
Number of nodes selected: k

Output:

Top k opinion leader set: $OLIsList$

Initialize $OLIsList$ to empty
Initialize the list $OLIsVal$ to store the OLI value of each node
for u **in** V **do**
    Get $D(u)$ by using Eq (3.6)
    Initialize temporary variable val to zero
    Calculate the betweenness centrality of nodes in network by using Eq (3.8)
    for v **in** $Nei(u)$ **do**
        Get by equation $\frac{|Nei(u)\cap Nei(v)|}{|Nei(u)\cup Nei(v)|}$ and store into $temp$
        $val = val + temp$
    end for
    Calculate $Inf_u = \frac{1}{\sum_{k\in G} len(Nei(k))} * val$ (reference Eq (3.7))
    Get betweenness centrality of node u and store into $B(u)$
    Get OLI_u by using Eq (3.9)
    Append OLI_u to $OLIsVal$
end for
Sort the node by score
Get top k nodes
return $OLIsList$

To evaluate the effectiveness of our proposed approach, we compared it against several well-known centrality measures, including degree centrality (DC), betweenness centrality (BC), closeness centrality (CC), and eigenvector centrality (EC). The final row of Table 2 displays the sum of values for each method. It is worth noting that the sum of the respective columns for each method is different. In order to visualize the data more intuitively, we employed a stacked line chart to demonstrate the trends in the different node measurement methods within the case network, as illustrated in Figure 3.

The network structure depicted in Figure 1 reveals that node 1 and node 4 have higher centrality, which is consistent with the trends illustrated in Figure 3 for all methods. Figure 3 indicates that the proposed OLI method exhibits a similar trend as the other methods, but with a higher degree of discrimination. Therefore, compared to the other methods evaluated in Table 2, the method proposed in this paper performs better. For the case network structure of Figure 1, we set α_1 and α_2 to 0.8 and 0.2, respectively, and β_1 and β_2 to 0.5 each. Since the network structure is small, this study emphasizes the node's own influence when computing the local influence of the node. The local and global structural information of the nodes are integrated, and the same weight is assigned to the node's local influence and global influence to calculate the final OLI.

Table 2. A case study to compare the calculation results for OLI.

Node	DC	BC	CC	EC	OLI
1	0.5000	0.6040	0.5380	0.5790	0.3830
2	0.1430	0.3850	0.5190	0.2540	0.2120
3	0.1430	0.0769	0.4000	0.1090	0.0585
4	0.4290	0.5110	0.5000	0.2510	0.3250
5	0.2140	0.0055	0.3780	0.3510	0.0577
6	0.1430	0.0000	0.3680	0.2840	0.0388
7	0.1430	0.1040	0.4120	0.2100	0.0722
8	0.0714	0.0000	0.3410	0.0767	0.0100
9	0.1430	0.0000	0.3500	0.1100	0.0438
10	0.1430	0.0000	0.3680	0.2840	0.0388
11	0.1430	0.0000	0.3680	0.2550	0.0429
12	0.1430	0.0000	0.3680	0.2550	0.0429
13	0.2140	0.0934	0.3890	0.1470	0.0892
14	0.1430	0.0000	0.3680	0.1220	0.0396
15	0.1430	0.0000	0.3500	0.1100	0.0438
sum	2.8584	1.7798	6.0170	3.3977	1.4982

(2) Structural hole score

Burt's theory of structural holes [44] explains the competitive relationships in social networks. In the realm of social networks, it is a common occurrence for individuals with similar professional or personal interests to seek each other out and form tight-knit communities. The ties between these groups, however, tend to be comparatively sparse. In network parlance, nodes that serve as inter-group conduits, known as "structural holes", play a crucial role in facilitating the exchange of information across community boundaries. As shown in Figure 1, nodes 2 and 3 act as bridges for communication between two communities. The ability of a node to utilize structural holes is measured by the network constraint coefficient, as shown in Eq (3.10). A smaller network constraint coefficient indicates a

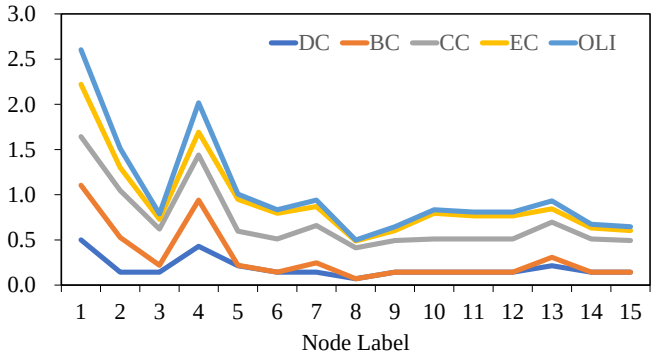


Figure 3. Stacked line graph of the changing trend of various node measurement methods for the case network.

greater possibility of structural holes, which can be beneficial for information dissemination.

The network constraint coefficient value for node u is denoted by $NC(u)$:

$$NC(u) = \sum_{v \in Nei(u)} (w_{uv} + \sum_{k \in Nei(v)} w_{vk} w_{ku})^2, \quad (k, u, v \in V and\ k \neq u, v) \tag{3.10}$$

where, w_{uv} represents the ratio of the energy invested by node u to maintain the relationship with node v to the total energy invested by node u, as shown in Eq (3.11).

$$w_{uv} = \frac{weight_{uv}}{\sum_{k \in Nei(u)} weight_{uk}} \tag{3.11}$$

Equation (3.10) is applied to calculate the network constraint coefficient value for all nodes in the network. In an unweighted graph, the edge weight $weight_{uv}$ is equal to 1 if there is a connecting edge between node u and node v, and it represents the weight value of the edge from node u to node v otherwise. After calculating the network constraint coefficient values for all nodes, we can identify the first k nodes with smaller values as the target structural hole nodes.

(3) Ordinary nodes

The selection thresholds for opinion leaders and structural hole users in the network are determined as γ_1 and γ_2, respectively, based on the findings of Wu et al. [45]. According to their study, only 1% of users in a network are considered as opinion leaders or structural hole users. However, they play a crucial role in creating or participating in 50% of the links in the network. To obtain the set of ordinary user nodes in the network, Equation 3.12 is utilized.

$$Or = V - Op - St, (|Op| = |V| * \gamma_1\ ,\ |St| = |V| * \gamma_2) \tag{3.12}$$

where Or represents a collection of ordinary nodes, Op represents a collection of opinion leader nodes, St represents a collection of structural hole nodes, and V represents all nodes in the network. Finally, the process of FGURD of nodes in the network can be described by Algorithm 3.

3.2. Role-based random walk for embedding

In the previous section, the process of identifying user roles in social networks was discussed. To obtain network embedding representations of users that preserve the local structure and global role approximations, a random walk-based network embedding approach is adopted, as illustrated in Figure 4. First, the role of nodes in the network is calculated using either Algorithm 1 or Algorithm 3. Second, the first-order or second-order local approximation of the node is captured by the random walk of the topological structure, and the approximation of the global role of the node is preserved through the random walk of the node role. By combining all random walks into one corpus, node embeddings can be learned by training the SkipGram model with negative sampling [46]. The SkipGram model can predict the conditional probability of co-occurrence among words within a fixed window size, and maximizing this probability allows the model to obtain vector representations of words in the corpus.

Algorithm 3 FGURD(G, α_1, α_2, β_1, β_2, γ_1, γ_2)

Input:

Network: $G(V, E)$

Node local influence balance-parameters: α_1 and α_2

OLI balance-parameters: β_1 and β_2

Threshold for the number of opinion leaders and structural hole users: γ_1 and γ_2

Output:

User role partition dictionary *RoleDic*

Initialize dictionary *RoleDic* to empty
$OLNumber = |V| * \gamma_1$
$SHNumber = |V| * \gamma_2$
Get *OLNumber* opinion leaders in the network via Algorithm 1.
Initialize the list $SHsList$ to empty
for u **in** V **do**
    Calculate network constraint coefficient value by Eq (3.10) and Eq (3.11).
    Add node network constraint coefficient value to $SHsList$
end for
Sort $SHsList$ in reverse order; select $SHNumber$ nodes
Get ordinary nodes by Eq (3.12).
Map the node to the corresponding role and add them to *RoleDic*
return *RoleDic*

The objective of the SkipGram model is to maximize the average log probability of a sequence of training words w_1, w_2, w_3, ..., w_T, as shown in the equation above.

$$\max \frac{1}{T} \sum_{t=1}^{T} \sum_{-win \le j \le win, j \neq 0} logp(w_{t+j} \mid w_t) \tag{3.13}$$

where the parameter *win* represents a predefined window size, with a larger value leading to more additional training examples and potentially higher accuracy. The softmax function is employed to estimate the probability distribution of $p(w_{t+j} \mid w_t)$, which is defined as follows:

$$p(w_O \mid w_I) = \frac{\exp\left(v_{w_O}^{'T} v_{w_I}\right)}{\sum_{w=1}^{W} \exp\left(v_w^{'T} v_{w_I}\right)} \tag{3.14}$$

where the word representations of "input" and "output" data are denoted as v_w and $v_w^{'}$, respectively. W is the total number of words in the vocabulary.

In the context of network embedding, the random walk algorithm is commonly used to generate sequences of nodes. Starting from an initial node $u \in G$, the algorithm randomly selects a neighboring node and moves to it. This process is repeated for a predefined number of steps. A method for network embedding called Role-based Random Walk Network Embedding (RbNE) has been developed, and its pseudocode is presented in Algorithm 4. RbNE takes a social network as input and outputs low-dimensional representations for each node in the network, where nodes with similar roles will have similar representations. To obtain the node representations in RbNE, each node in the network is first

divided into roles using either Algorithm 1 or Algorithm 3, which correspond to CGURD and FGURD, respectively. Then, the role of the node is used to perform random walk simulation, resulting in the node's representation. The random walk network embedding methods that are divided by user roles in Algorithm 1 and Algorithm 2 are named RbNE-CG and RbNE-FG, respectively. The parameter settings for the algorithms are detailed in the corresponding sections.

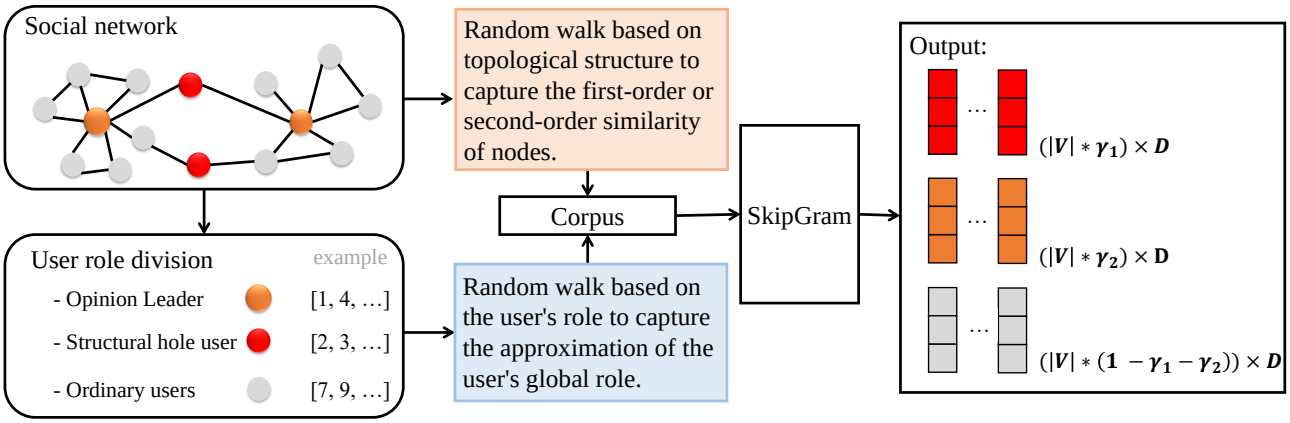


Figure 4. Addition of user role random walks to train the network embedding.

Algorithm 4 RbNE(G, *len*,*number*, *win*, d)

Input:

Social network: $G(V, E)$
Maximum length of random walk: *len*
Number of random walks: *number*
Window length: *win*
Final representation size: d

Output:

Matrix of node representations: $\Phi \in R^{|V| \times d}$

1: Initialize walks to empty
2: Divide user roles through Algorithm 1 or Algorithm 3, and store the results in *RoleDic*
3: **for** node u **in** V **do**
4: Get the list of roles $Roles_u$ = $RoleDic[u]$
5: $walk_1$ = TraditionalRandomWalk(G, u, *len*)
6: $walk_2$ = RolebasedWalk(G, u, $Roles_u$, *len*)
7: Append $walk_1$ to *walks*
8: Append $walk_2$ to *walks*
9: **end for**
10: SkipGram(Φ, *walks*, *win*)
11: **return** Φ

The traditional random walk method (line 5, Algorithm 4 captures the local approximation of nodes, similar to Deepwalk and node2vec. In contrast, the random walk method based on node role

(line 6, Algorithm 4 captures the global node approximation. As a result, the final training corpus *walks* contains both the local neighbor approximation relationship and the global role approximation relationship of each node.

Embedding techniques in network analysis entail the inclusion of neighboring nodes, degrees, labels, and other relevant attributes to impart specifications onto the individual nodes within the network. This approach unveils valuable associations and connections between nodes, ultimately amplifying the efficacy of both network analysis and machine learning methodologies.

3.3. IM with node embeddings

The representations of all nodes in the network G are now available and are denoted as $\Phi \in R^{|V|\times d}$. To measure the relationship between any two nodes, the cosine similarity of their representation vectors can be calculated directly using the following equation:

$$Sim(u, v) = \frac{\Phi_u \cdot \Phi_v}{\|\Phi_u\| \times \|\Phi_v\|} \tag{3.15}$$

The relationship score between two nodes u and v is defined as $Sim(u, v)$ in our approach. A higher score indicates a higher probability of node u influencing node v. Thus, a new logically structured network called a propagation probability network can be constructed, where the similarity between two nodes is determined by Eq (3.15). To simplify the network, a similarity threshold θ is introduced. Only when their similarity score is greater than θ is a logical connection edge added between two nodes u and v. Hence, the adjacency matrix representation of the connection strength between any two nodes in the network can be described as follows:

$$p_{u,v} = \begin{cases} Sim(u, v), & if\ Sim(u, v) \geqslant \theta \\ 0, & otherwise \end{cases} \tag{3.16}$$

where θ represents the hyper parameter and θ belongs to the interval $(0, 1)$. By applying the weight calculation method, we can obtain the desired new logical structure network. In this structure, $p_{u,v}$ denotes the probability of information propagation from node u to node v. Assuming the independence of influence probabilities among users, the probability of node i being activated by its neighbor $Nei(i)$ can be computed using the propagation probability of its friends:

$$Pr(V \mid v) = 1 - \prod_{u \in Nei(v)} (1 - p_{u,v}) \tag{3.17}$$

Similarly, the total influence spread of all non-seed nodes under the influence of the seed node set $\mathcal{S}$ can be quantified for each vertex $u \in V$:

$$\begin{aligned} Pr(V \mid \mathcal{S}) &= \frac{1}{|V|} \sum_{u \in V} \left[1 - \prod_{v \in \mathcal{S}} (1 - p_{u,v}) \right] \\ &= 1 - \frac{1}{|V|} \sum_{u \in V} \prod_{v \in \mathcal{S}} (1 - p_{u,v}) \end{aligned} \tag{3.18}$$

The objective of the IM task is to increase the number of activated nodes influenced by the seed node set $\mathcal{S}$, which is equivalent to maximizing the value of $Pr(V \mid \mathcal{S})$. Therefore, our optimization goal can be formulated as follows:

$$\begin{aligned}\arg\max_{\mathcal{S}}(Pr(V \mid \mathcal{S})) &= \arg\max_{\mathcal{S}}\left[1 - \frac{1}{|V|}\sum_{u\in V}\prod_{v\in\mathcal{S}}(1 - p_{u,v})\right] \\ &= \arg\min_{\mathcal{S}}\left[\frac{1}{|V|}\sum_{u\in V}\prod_{v\in\mathcal{S}}(1 - p_{u,v})\right]\end{aligned} \tag{3.19}$$

As the direct optimization of the optimization goal is not feasible, a greedy heuristic algorithm is utilized in this study. Specifically, for undirected networks, a Connected components [19] type of heuristic is employed to compute the score for each node; and subsequently, the k nodes with the highest scores are chosen. Algorithm 5. presents the three-step procedure, which includes the following:

1. Calculation of the similarity between each user utilizing the network embedding matrix Φ and construction of a new logical network structure matrix A (lines 2–3).
2. Random deletion of edges according to their weights to acquire connected components in the network (lines 11–20).
3. Assign each node a weight value based on the number of its neighbors and select the k nodes with the highest weight values as the seed node set (lines 21–28).

By implementing this heuristic algorithm, the seed node set $\mathcal{S}$ can be effectively selected.

3.4. Algorithmic complexity discussion

The proposed RbneIM algorithm (Algorithm 5) has a time complexity of $O(R * |V| * k)$, where R and k are both constants. The outer loop runs for a constant number of iterations R, while the inner loop traverses the network and the current cropped subgraph, which has a constant size k. In contrast, the RbNE algorithm (Algorithm 4) has greater time complexity determined by the most expensive of its three parts. First, Algorithm 3 is called to perform role division, which has a time complexity of $O(|V| * n)$, where n is the largest number of node neighbors in the network $|Nei(i)|$. Second, in the sampling process, the algorithm iterates according to the predefined sampling length *len* (constant) and randomly adds nodes to the sampling sequence according to predefined rules, which takes $O(|V| * len)$ time. Finally, the SkipGram algorithm has a time complexity of $O(|V|)$. Therefore, the time complexity of the RbNE algorithm is $O(|V|) * (O(|V| * n) + O(|V| * len) + O(|V|))$. Since n, k, and *len* are all constants, the final time complexity of the proposed RbneIM algorithm is $O(|V|^2)$. Several baseline methods utilize matrix operations, but this approach can lead to memory insufficiency when dealing with large-scale graphs. In contrast, the RbneIM method employs heuristic algorithms that significantly reduce the computational complexity, resulting in strong scalability.

Algorithm 5 RbneIM(G, Φ, θ, P, R, k)

Input:

Social network: $G(V, E)$
Embedding matrix of G: $embedding \in R^{|V|\times d}$
Hyper parameter of connecting edges: θ
Propagation probability under Independent Cascade: P
Number of iterations: R

Number of seed nodes: k

Output:

Selected node seed list: $Seeds$

Initialize $Seeds$ to empty
Calculate the cosine similarity matrix A of each node in the embedding matrix Φ via Eq (3.15)
According to Eq (3.16), $A[A < \theta] = 0$ processing to get logical network structure L
Initialize node's score dict $score = \{0 : 0, 1 : 0, 2 : 0, ...\}$
for $i = 1...R$, **do**
    $G' = deepcopy(L)$
    Randomly select blocked edges by $(1 - P)^{p_{ij}}$ and remove them from G'
    Initialize connected components dict $ccDict$ to empty
    Initialize node's visited list $vis = [false, ...]$
    Initialize the count variable $index = 0$
    for $node \in G'$ **do**
        if not $vis[node]$ **then**
            $vis[node] = true$
            $ccDict[index++] = [node]$
            $nodes = G'.neighbors(node)$
            for $nei \in nodes$ **do**
                if not $vis[nei]$ **then**
                    $vis[nei] = true$
                    $ccDict[index].append(nei)$
                    $nodes.append(G'.neighbors(nei))$
                end if
            end for
        end if
        Sort $ccDict$ according to size
        for $component \in ccDict$ **do**
            $nodes = ccDict.values()$
            $temp\_score = \frac{1}{\sqrt{len(nodes)}}$
            for $node \in nodes$ **do**
                $score[node] += temp\_score$
            end for
        end for
    end for
end for
Select k nodes with the smallest score from $score$
return $Seeds$

4. Experimental evaluation

This section begins by presenting the social network dataset and parameter settings employed in this study. Subsequently, the baseline algorithm used is briefly introduced, followed by an analysis of

the experimental findings.

4.1. Datasets

This study utilized six public real-world datasets to provide varying-sized networks. This aimed to assess the feasibility and effectiveness of the proposed IM method. Table 3 presents a comprehensive overview of the datasets. The datasets were carefully selected based on their diversity, which includes different types of social networks, ranging from online social networks to co-authorship networks. Moreover, the datasets contain a varying number of nodes and edges, ranging from small-scale networks to large-scale networks, thus providing a diverse range of networks for our analysis. By using such diverse datasets, we aim to evaluate the performance of our proposed method under different network settings, which can help to enhance the generalizability of our findings.

Table 3. Statistics for the datasets used in the experiments, including the number of nodes (#Node) and the number of edges (#Edge).

Dataset	#Node	#Edge
Dolphins	62	161
Facebook_Caltech36	769	16662
NetScience	1591	5880
Cora	2710	5430
Ca-GrQc	4158	26850
Facebook-Government	7057	89428

(i) **Dolphins**[*] [47]. The dataset used in this study is an undirected social network consisting of 62 dolphins living in a community off of Doubtful Sound, New Zealand. The dataset encompasses frequent associations between the dolphins in the form of links between them.

(ii) **Facebook_Caltech36**[†] [48]. A social friendship network extracted from Facebook consisting of people as nodes, with edges representing friendship ties.

(iii) **NetScience**[‡] [49]. The NetScience dataset is a co-authorship network that involves scientists working on network theory and experiments. A visual representation of the largest component of this network can be accessed via the URL.

(iv) **Cora**[§]. The Cora dataset is a collection of machine learning papers, and it includes the citation relationships between them. These relationships are used to construct the network topology for this dataset.

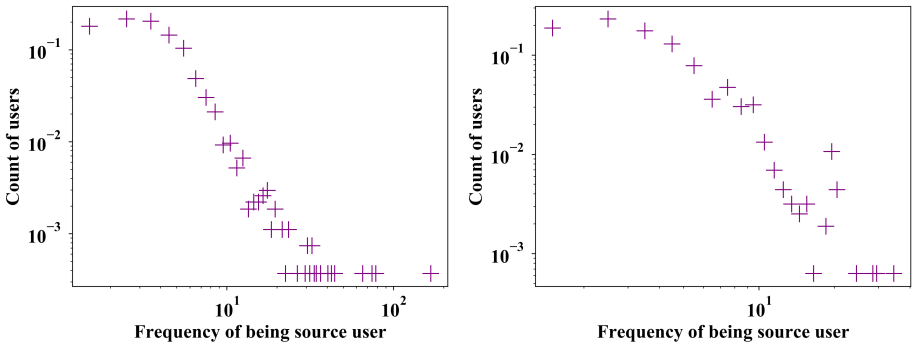
(v) **Ca-GrQc**[¶] [50]. The Ca-GrQc dataset is a collaboration network of arXiv General Relativity and Quantum Cosmology. It is derived from the e-print arXiv and includes scientific collaborations between author papers submitted to the General Relativity and Quantum Cosmology categories. The dataset covers papers submitted between January 1993 and April 2003.

(vi) **Facebook-Government**[‖] [51].The data collection process involved gathering information on the

[*]http://www-personal.umich.edu/ mejn/netdata/
[†]https://networkrepository.com/socfb-Caltech36.php
[‡]http://www-personal.umich.edu/ mejn/centrality/
[§]https://linqs.soe.ucsc.edu/data
[¶]http://snap.stanford.edu/data/ca-GrQc.html
[‖]https://networkrepository.com/fb-pages-government.php

Facebook pages of politicians in November 2017. The resulting network is represented as nodes, which correspond to the politician pages, and edges, which indicate mutual relationships between them.

We have plotted the frequency distribution of user node degrees to characterize the Cora and NetScience datasets.



(a) Cora **(b)** NetScience

Figure 5. Node degree distribution of users on Cora and NetScience. The x axis represents the node degree of users, and the y axis represents the number of such users (valued as loglog).

The user node degree distributions for the Cora and NetScience datasets are presented in Figure 5, indicating a power-law distribution. This suggests that certain users are more susceptible to influence from their social connections.

4.1.1. Baseline methods

This paper introduces three typical initial ranking methods and a state-of-the-art IM method to evaluate the comparative performance of the proposed RbneIM method.

(i) **Random**: Nodes are initially ranked randomly.
(ii) **Degree centrality** [52]. Degree centrality measures the influence of a node based on the number of its neighbors, with nodes having higher degrees being considered as more influential.
(iii) **Betweenness centrality** [53]. The betweenness centrality measures the extent to which a node acts as a bridge along the shortest paths between other nodes. A node with higher betweenness centrality has a greater number of shortest paths passing through it. The betweenness of node u is calculated by Eq (3.8).
(iv) **Pagerank centrality** [54]. The PageRank centrality measures the importance of a node based on the structure of the network. It was originally created to evaluate the importance of web pages by using their link structures. Since then, it has been applied in various fields, such as social network analysis, link prediction and recommendation analysis.
(v) **DeepIM** [29]. The DeepIM algorithm is the first to employ deep learning techniques to solve the IM problem. It uses the CARE algorithm [42] to learn node embeddings. Cosine similarity is employed to measure similarity between nodes, k similar nodes are recorded for each node, and a set of seed nodes is selected through statistical analysis.

(vi) **GCNIM** [33].The research contributes to the field of social network analysis by proposing a new technique that overcomes the limitations of traditional algorithms and deep learning-based approaches while achieving high performance and efficiency in the area of seed set identification for IM tasks.

(vii) **ABEM** [36]. The approach utilizes agent-based modeling and genetic algorithms to effectively address the complex task of selecting key influencers in a distributed environment. By leveraging these techniques, the approach identifies users' influence capability and optimizes the influencer set's selection. This innovative solution tackles the challenge of capturing real-time user and diffusion features, enabling the accurate and efficient identification of key influencers.

It is noteworthy that this paper presents two algorithms, namely CGURD (Algorithm 1) and FGURD (Algorithm 3), for role division. The role division outcome will have an impact on the sampling outcome of the final random walk process, leading to a different embedding representation vector of the node under the two algorithms. Therefore, the use of these two node partitioning algorithms will ultimately influence the selection of seed nodes. In this paper, the RbneIM algorithm is executed using Algorithm 1. and Algorithm 3. for node division, resulting in RbneIM-CG and RbneIM-FG, respectively.

(i) **RbneIM-CG**: This model utilizes the CGURD algorithm to determine the global role of users in the network. The RbNE algorithm is then employed to perform the sampling of the training corpus. The final selection of seed nodes is accomplished through RbneIM.

(ii) **RbneIM-FG**: Compared with the RbneIM-CG model, only the user role division algorithm is different.

4.2. Analysis and comparison

This section presents an analysis of the key techniques proposed in this paper and compares them with existing approaches to demonstrate the feasibility of the proposed approach. IM is the foundation for introducing and understanding influence dissemination within social networks.

(i) Our proposed methodology presents several advantages over existing deep learning IM methods. It leverages network embedding techniques to assign attribute values to user nodes, allowing for a more comprehensive analysis of user influence. Unlike other methods that primarily focus on a single global factor or the node's own attributes, our approach also takes into account the influence factor between users, providing a more nuanced understanding of influence dynamics in social networks. Additionally, our methodology considers attributes at multiple levels of granularity, enabling a more fine-grained analysis and capturing the diverse factors that contribute to user influence.

(ii) Our study presents an innovative algorithm for user role division, integrating both CGURD and FGURD to offer a comprehensive and refined approach. The CGURD component of our algorithm focuses on classifying users into distinct groups based on their overall network contribution, allowing for a broader understanding of user roles. In contrast, the FGURD aspect concentrates on analyzing the relationships between users and their adjacent nodes to identify more specific and localized user roles. By combining CGURD and FGURD, our algorithm provides a robust and precise user role division strategy that captures the intricacies of user

dynamics in the network. Furthermore, the methods evaluated in this paper employ various network embedding techniques, as outlined in Table 4.

Table 4. A detailed comparison of the model proposed in this paper and other studies.

Methods	Node attributes	Nodes selected
DeepIM	DeepIM preserves both global and local macro properties of user nodes by utilizing network embedding techniques of deep learning methods.	Random walk is used to assign values to nodes.
ABEM	Represents users as autonomous and proactive agents that possess the ability to communicate with their neighbors, extract information from the local environment, and estimate their influence capacity. ABEM utilizes agent-based modeling to identify potential influencers in changing real-world networks.	The search scope of ABEM is continually updated through user agents, which is a task efficiently executed by the proposed algorithms. These algorithms maintain the existing potential influencers while concurrently modifying parts of the solutions.
GCNIM	A network dynamic GCN with adaptive layers according to different network scales was designed to obtain the information representation of node position influence.	This method incorporates a leader fake labeling mechanism that automates the generation of node labels to facilitate the selection of seed nodes during model training.
Ours	The global location influence (Algorithm 1) attribute and the influence attribute between nodes (Algorithm 3)	Our method encompasses both the local neighbor approximation relationship and the global role approximation relationship for every node (Algorithm 4).

4.3. Parameter setting

The experiments conducted in this study used default values for various parameters mentioned in the paper. Specifically, the node local influence balance parameters α_1 and α_2 in Eq (3.15) were set to 0.8 and 0.2, respectively. The OLI score balance parameters β_1 and β_2 in Eq (3.5) were set to 0.5 and 0.5, respectively. In Eq (3.12), the thresholds γ_1 and γ_2 for opinion leaders and structural hole nodes were set to 0.2 and 0.1, respectively. Other parameters used in the experiments were set as follows: random walk length $len = 80$, random walk number $number = 10$ and embedding matrix dimension $d = 256$. The SkipGram training window size was set to $win = 5$, and the negative sampling frequency and learning rate were both set to 0.025. The threshold $\theta = 0.5$ was used to establish a connection edge between any two nodes of the new logical network in Eq (3.16). In the RbneIM algorithm, the propagation probability of the IC model was set to 0.5, and the number of algorithm iterations $R = 20$. The non-default value parameters for each dataset are shown in Table 5.

Table 5. Non-default values of parameters applied in experiments for six datasets.

Dataset	θ	*len*	γ_1	γ_2
Dolphins	0.7	140	0.14	0.16
Facebook_Caltech36	0.9	80	0.18	0.12
NetScience	0.5	80	0.12	0.18
Cora	0.5	20	0.18	0.12
Ca-GrQc	0.1	40	0.12	0.18
Facebook-Government	0.7	60	0.16	0.14

All methods were implemented using Python 3, and the experiments were performed on a Windows OS with AMD Ryzen 5 3500U, 2.10 GHz CPU and 16 GB memory. Details of our software and hardware environments were as follows: Windows 11, Python ver. 3.6.6, NumPy ver. 1.19.2, NetworkX ver. 2.1, Gensim ver. 3.8.3, Pandas ver. 0.24.2, Matplotlib ver. 2.2.3.

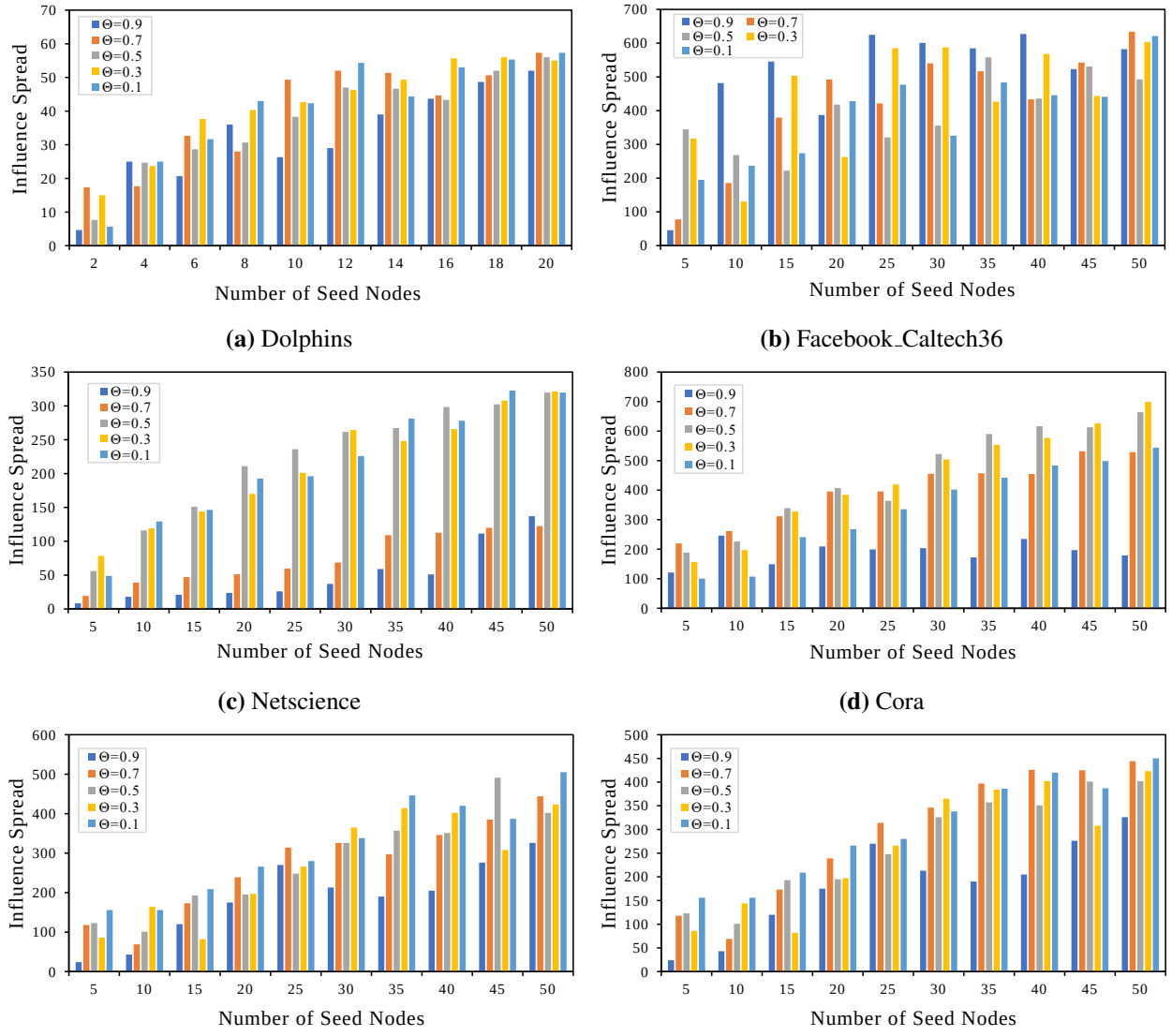
5. Results

In Eq (3.16), a threshold parameter θ was defined to create a new logical network structure, which directly affects the performance of the RbneIM model. Therefore, the analysis of θ parameters is performed first. Seed node sizes were selected based on the dataset scale, ranging from 2 to 20 with a stride of 2 for the Dolphins dataset, and from 5 to 50 with a stride of 5 for the remaining datasets. Experimental results are displayed in Figure 6, where θ was chosen as a value between 0.1 to 0.9 with a step size of 0.2. The figure shows that different datasets have different optimal θ values. The influence spread was considered for various numbers of seed nodes, and a counting method was used to evaluate the pros and cons of each θ in the current dataset. The best performing θ was then selected for each dataset value. Finally, the chosen values for θ were 0.7, 0.9, 0.5, 0.5, 0.1 and 0.7 for the Dolphins, Facebook_Caltech36, Netscience, Cora, CA-GrQc and Facebook-Government datasets, respectively.

We present a comprehensive comparison of the influence spread achieved by different algorithms, namely random, DC, BC, PageRank centrality, DeepIM and RbneIM, utilizing the LT model. The corresponding results are illustrated in Figure 7 for six distinct networks. Upon examining smaller datasets, as depicted in Figure 7a for Dolphins and Figure 7b for Facebook_Caltech36, we notice that the influence spread generated by various models exhibit similar outcomes. However, our proposed RbneIM method maintained its superior effectiveness. As the dataset size increased, both DeepIM and RbneIM consistently outperformed the other approaches by a substantial margin, with RbneIM exhibiting the highest level of performance. Experiments on multiple datasets demonstrated the superior performance of the proposed RbneIM method.

Table 6. The selection ratio experiment for opinion leaders and structural hole nodes, where the horizontal axis is γ_1, and the vertical axis is six datasets respectively.

Dataset	0.02	0.04	0.06	0.08	0.10	0.12	0.14	0.16	0.18	0.20
Dolphins	51.4	51.7	51.8	51.1	51.1	50.5	**51.92**	51.2	46.2	50.3
Facebook_Caltech36	567.3	568.4	570.6	558.1	554.4	576.5	578.9	580.9	**585.4**	574.2
NetScience	303.6	305.2	305.9	310.3	312.5	**317.2**	312.8	310.3	303.6	310.4
Cora	1343.0	1260.6	1297.7	1320.8	1322.0	1342.3	1342.9	1368.5	**1381.7**	1368.1
Ca-GrQc	1009.2	1024.1	1069.3	1096.2	1107.3	**1202.3**	1188.1	1172.2	1158.8	1103.4
Facebook-Government	3603.6	3374.1	3033.1	3112.9	3277.4	3312.3	3557.0	**3642.8**	3413.6	3376.1

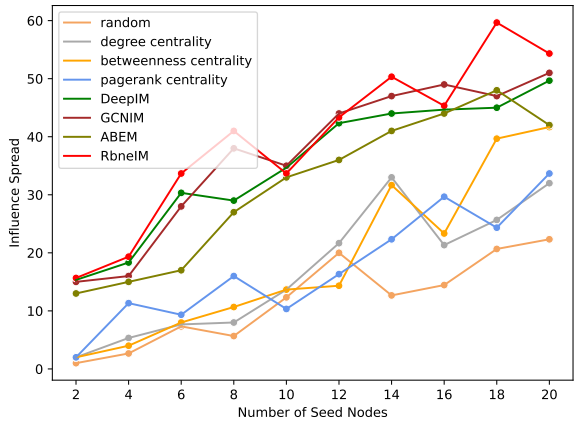


(a) Dolphins **(b)** Facebook_Caltech36

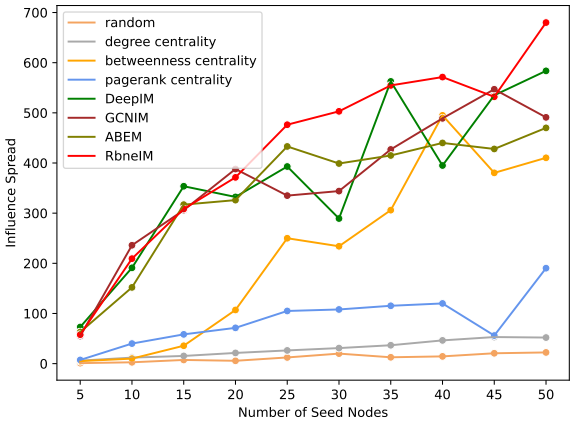
(c) Netscience **(d)** Cora

(e) Ca-GrQc **(f)** Facebook-Government

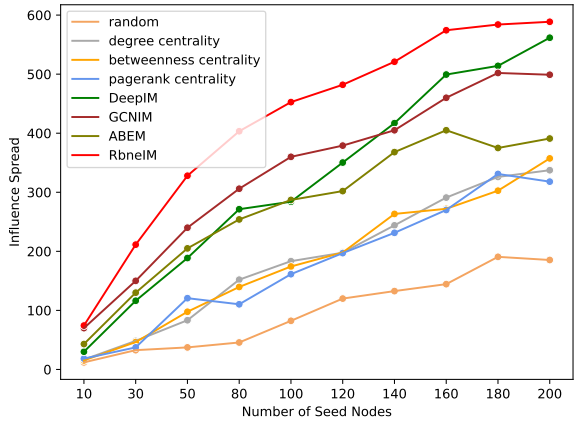
Figure 6. Effects of different values of θ (see Eq (3.16)) in the RbneIM algorithm on different datasets.



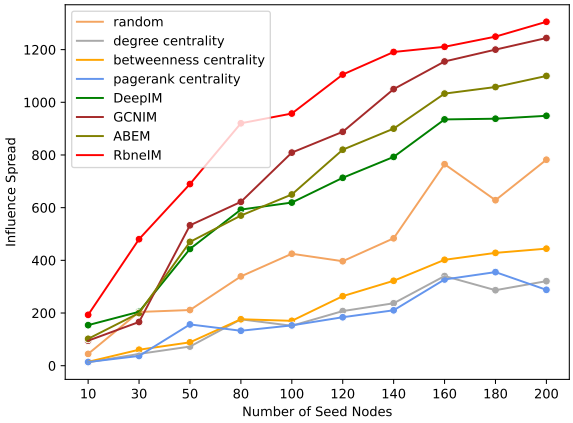
(a) Dolphins



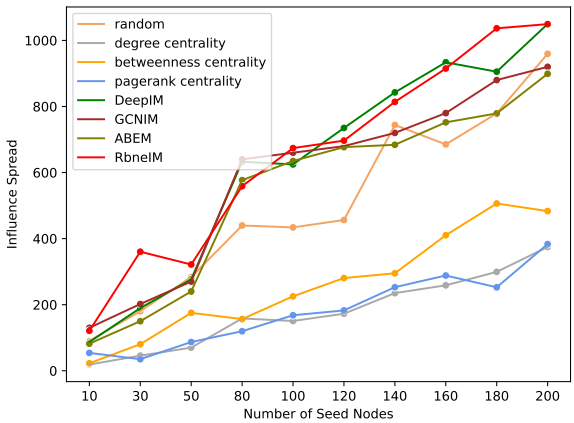
(b) Facebook_Caltech36



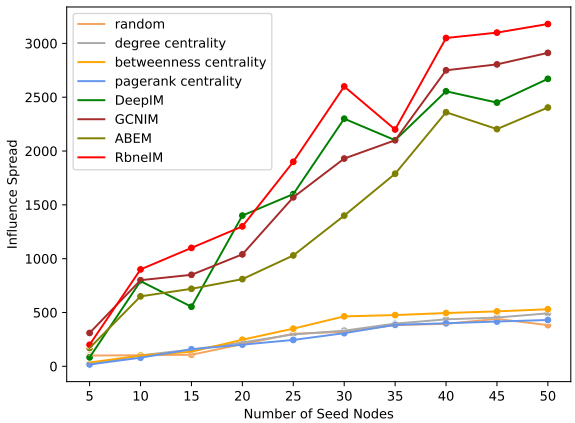
(c) Netscience



(d) Cora



(e) Ca-GrQc



(f) Facebook-Government

Figure 7. Comparison of the simulation results for LT models with various baseline methods and different numbers of seed sets.

Additionally, the size of the seed nodes selected for each dataset varied due to the differences in dataset size. For example, in the Dolphins network, the seed node size $[2, 4, 6, ...]$ was selected, with

Table 7. Average of experimental results of different seed node selection algorithms on six networks.

Dataset	Random	Degree	Betweenness	Pagerank	DeepIM	RbneIM
Dolphins	13.6	17.1	19.0	17.4	33.6	39.6
Facebook_Caltech36	25.2	41.7	258.6	111.3	370.9	429.3
NetScience	122.8	187.9	186.6	179.4	320.2	421.9
Cora	118.5	185.1	237.1	185.8	634.2	930.2
Ca-GrQc	86.7	178.5	263.4	182.4	614.9	664.8
Facebook-Government	242.7	341.3	459.6	297.7	1795.3	2037.5

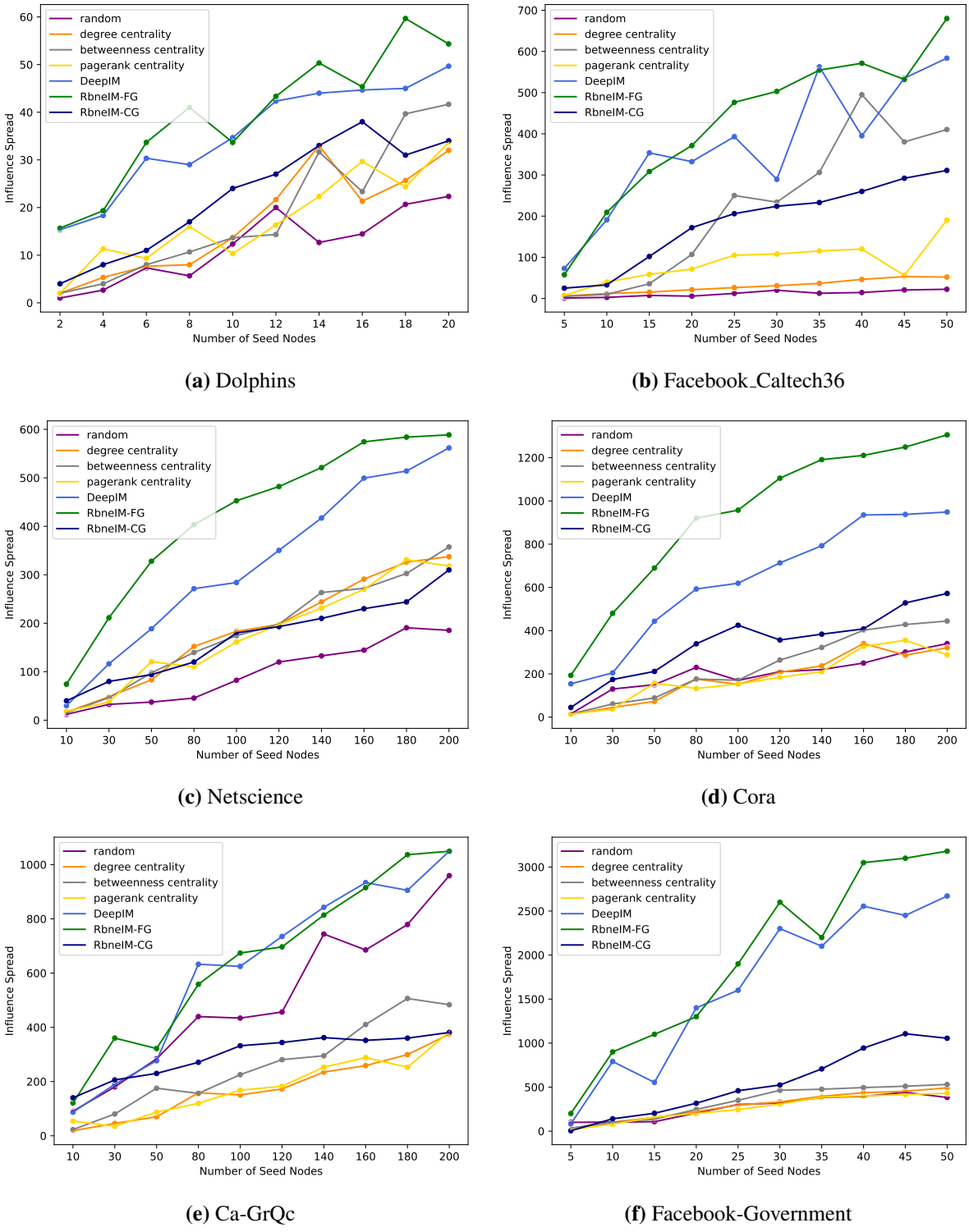
a maximum of 20 nodes selected for the seed node set. For Facebook_Caltech36, a maximum of 50 nodes were selected, and for the remaining datasets, up to 200 nodes were selected. The average results of LT simulations for each of the six networks, with varying numbers of seed nodes, are reported in Table 7.

The average influence diffusion of each algorithm on six datasets with different seed node numbers, taken from 20 experiments was calculated in Table 7. The results indicate that our proposed RbneIM algorithm outperforms the baseline algorithms, particularly Random and DC. Random selects the seed node randomly from any network node, while DC centrality uses the number of neighboring nodes in one hop. BC and PageRank centrality calculate the number of shortest paths through a node and the importance of its links, respectively. DeepIM takes into account the node community structure factor in the network embedding and calculates node similarity to select seed nodes. However, these baseline algorithms do not consider users' global role similarity, which reduces their efficiency in selecting seed nodes.

As the proposed network embedding algorithm relies on a sequence of random walk sampling, an analysis was conducted to examine the impact of different random walk lengths on the RnbeIM model. To this end, the experiment involved selecting random walk lengths *len* ranging from 20 to 200 with a step size of 20; the analysis results are presented in Table 8. Seed node sets were determined based on the size of the datasets, with a size of 20 for the Dolphins dataset and 50 for the remaining datasets. Corresponding to the step size, the number of seed node sets was calculated, and 200 rounds of LT model propagation simulation were carried out. The final results represent the average value, with one decimal place reserved. The analysis in Table 8 highlights that different step lengths of various datasets have a notable impact on the experimental outcomes. The optimal result is identified in bold font in the table, and its corresponding random walk length was selected as the parameter value on this dataset. Table 5 presents the selected parameter values.

Table 8. Effect of different random walk lengths *len* on the experimental results.

Dataset	20	40	60	80	100	120	140	160	180	200
Dolphins	53.5	56.8	53.3	54.8	55.1	56.0	**60.1**	53.3	53.5	53.0
Facebook_Caltech36	587.8	621.8	612.7	**624.3**	595.7	577.4	581.4	595.8	561.7	560.0
NetScience	615.8	591.4	606.6	**617.7**	612.4	593.1	597.7	596.3	608.0	596.7
Cora	**1336.6**	1317.0	1315.8	1329.1	1315.7	1310.5	1295.8	1311.6	1266.4	1254.9
Ca-GrQc	1081.7	**1200.4**	1060.4	1114.4	1098.9	952.0	1081.2	1021.3	988.7	986.0
Facebook-Government	1130.6	2671.5	**3530.8**	2836.7	2482.8	2280.4	2258.7	2252.8	1944.6	2100.3



(a) Dolphins

(b) Facebook_Caltech36

(c) Netscience

(d) Cora

(e) Ca-GrQc

(f) Facebook-Government

Figure 8. Comparative experiments of RbneIM-FG and RbneIM-CG on six datasets.

In addition, a comparison was made between the RbneIM-FG and RbneIM-CG models proposed in this paper on six datasets. It is noteworthy that in the previous experiments, the RbneIM model used the FGURD (Algorithm 3.) algorithm by default to divide user roles in the network, which is

the RbneIM-FG model. The experimental results are presented in Figure 8, which shows that the RbneIM-FG model outperforms the RbneIM-CG model.

The results clearly demonstrate that the use of a global influence algorithm alone to select the seed set yields extremely low propagation efficiency. However, by leveraging the proposed FGURD algorithm to identify different influence roles, the efficiency of influence propagation is significantly improved. The observed discrepancy in experimental outcomes underscores the inadequacy of relying solely on global influence. The proposed model combining global and local information for network embedding achieved the best results on the six datasets, serving as a confirmation of the method's effectiveness on IM issues.

The paper sets the thresholds for opinion leaders and structural hole nodes as γ_1 and γ_2 (as described in Eq (3.12), respectively. The values of γ_1 and γ_2 are set to 0.2 and 0.1, respectively, in the parameter setting section. It should be noted that these values may vary depending on the network structure. Here, we assumed that 70% of the nodes in any social network are ordinary nodes. To explore the impact of varying these two parameters, this study included an experiment for which the result is presented in Table 6. The number of seed nodes was set to a fixed value for each dataset, with different divisions made based on the dataset size. For example, the Dolphins dataset was set to 20, while the Facebook_Caltech36 and Netscience datasets were set to 50. The remaining three datasets were set to 200. We varied the value of γ_1 from 0.02 to 0.2 with a step size of 0.02, corresponding to $\gamma_2 = 0.3 - \gamma_1$. The optimal results are displayed in bold font in Table 6, and the corresponding x-axis value was selected as the value of γ_1 for the dataset, with $0.3 - \gamma_1$ being the value of γ_2. The selected parameter values are shown in Table 5.

6. Conclusion and future work

The present paper introduces a novel network embedding algorithm, named RbNE, for social networks; it incorporates users' global roles into the embedding process. The proposed RbNE approach merges the CGURD and FGURD methods, aiming at gathering both the overall contribution of the user to the network and the relationships between the user and its neighboring nodes. This results in a more comprehensive representation of the user's global role and approximate user information. Building on this embedding method, we propose a greedy heuristic algorithm, RbneIM, to solve the IM problem by fully integrating the global role information and filtering out the seed set.

Previous studies have encountered challenges in effectively integrating both local and global information concerning users in social networks. A notable limitation has been the neglect of the IM problems' sensitivity to the global roles of users. Additionally, there is a lack of comprehensive understanding regarding the potential contribution of global roles in identifying seed sets that exhibit substantial influence. To address these gaps, this paper proposes the RbNEIM approach, which considers users' global role as a crucial criterion in selecting seed sets and identifying users with the potential to maximize their impact on social networks. We evaluated RbNEIM on six popular social network datasets and compared its performance with state-of-the-art methods and recent baselines. The results demonstrate that our proposed method outperforms existing techniques, highlighting its superior performance in terms of solving IM problems. In future work, we will explore the optimization potential of graph neural networks and attention mechanisms to further enhance the performance of RbNEIM.

Experimental analysis reveals the following: (1) RbNEIM can combine global and local information for network embedding, and it can comprehensively maintain the approximation between the user's global roles; (2) By integrating heuristic calculation and role embedding methods, RbNEIM can significantly improve the performance of the IM problem by considering the user's global role as an essential factor in selecting the seed set and identifying users who can spread the maximum influence in the social network; (3) The proposed method is robust to hyperparameter tuning. The insights gained from this study have the potential to advance the development of future social networks and IM problems.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This work is supported by the Science and Technology Program of Sichuan Province (Grant nos. 2023YFS0424, 2022YFG0378) and the National Natural Science Foundation (Grant nos. 61902324, 11426179, and 61872298).

Conflict of interest

The authors declare that there is no conflict of interest.

References

1. J. Gu, G. Li, N. D. Vo, J. J. Jung, Contextual Word2Vec model for understanding chinese out of vocabularies on online social media, *Int. J. Semant. Web. Inf. Syst.*, **18** (2022), 1–14. https://doi.org/10.4018/ijswis.309428
2. G. Manal, Social media data for the conservation of historic urban landscapes: Prospects and challenges, in *Culture and Computing. Design Thinking and Cultural Computing* (eds. M. Rauterberg), Springer, (2021), 209–223. https://doi.org/10.1007/978-3-030-77431-8_13
3. J. Zhao, L. Yang, X. Yang, Maximum profit of viral marketing: An optimal control approach, in *Proceedings of the 2019 4th International Conference on Mathematics and Artificial Intelligence*, Association for Computing Machinery, (2019), 209–214. https://doi.org/10.1145/3325730.3325767
4. D. Pedro, R. Matt, Mining the network value of customers, in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, Association for Computing Machinery, (2001), 57–66. https://doi.org/10.1145/502512.502525
5. X. Song, B. L. Tseng, C. Lin, M. Sun, Personalized recommendation driven by information flow, in *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, Association for Computing Machinery, (2006), 509–516. https://doi.org/10.1145/1148170.1148258

6. Y. Li, D. Zhang, K. Tan, Real-time targeted influence maximization for online advertisements, *Proc. VLDB Endow.*, **8** (2015), 1070–1081. https://doi.org/10.14778/2794367.2794376
7. L. Simone, M. Diego, R. Giuseppe, M. Maurizio, Mining micro-influencers from social media posts, in *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, Association for Computing Machinery, (2020), 867–874. https://doi.org/10.1145/3341105.3373954
8. X. Zhou, S. Li, Z. Li, W. Li, Information diffusion across cyber-physical-social systems in smart city: a survey, *Neurocomputing*, **444** (2021), 203–213. https://doi.org/10.1016/j.neucom.2020.08.089
9. V. Soroush, M. Mostafa, R. Deb, Rumor gauge: predicting the veracity of rumors on Twitter, *ACM Trans. Knowl. Discov. Data*, **11** (2017), 1–36. https://doi.org/10.1145/3070644
10. S. R. Sahoo, B. B. Gupta, Multiple features based approach for automatic fake news detection on social networks using deep learning, *Appl. Soft Comput.*, **100** (2021), 106983. https://doi.org/10.1016/j.asoc.2020.106983
11. D. Kempe, J. Kleinberg, E. Tardos, Maximizing the spread of influence through a social network, in *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Association for Computing Machinery, (2003), 137–146. https://doi.org/10.1145/956750.956769
12. A. G. Cecilia, B. Manuel, T. M. Valentina, An agent-based social simulation for citizenship competences and conflict resolution styles, *Int. J. Semant. Web Inf. Syst.*, **18** (2022), 1–23. https://doi.org/10.4018/IJSWIS.306749
13. Y. Rong, Q. Zhu, H. Cheng, A model-free approach to infer the diffusion network from event cascade, in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, Association for Computing Machinery, (2016), 1653–1662. https://doi.org/10.1145/2983323.2983718
14. S. Galhotra, A. Arora, S. Virinchi, S. Roy, Asim: A scalable algorithm for influence maximization under the independent cascade model, in *Proceedings of the 24th International Conference on World Wide Web*, Association for Computing Machinery, (2015), 35–36. https://doi.org/10.1145/2740908.2742725
15. A. Cetto, M. Klier, A. Richter, J. F. Zolitschka, "Thanks for sharing"—Identifying users' roles based on knowledge contribution in Enterprise Social Networks, *Comput. Net.*, **135** (2018), 275–288. https://doi.org/10.1016/j.comnet.2018.02.012
16. L. Sopjani, J. J. Stier, S. Ritzén, M. Hesselgren, P. Georén, Involving users and user roles in the transition to sustainable mobility systems: The case of light electric vehicle sharing in Sweden, *Transp. Res. Part D: Transp. Environ.*, **71** (2019), 207–221. https://doi.org/10.1016/j.trd.2018.12.011
17. L. B. Jeppesen, K. Laursen, The role of lead users in knowledge sharing, *Res. Policy*, **38** (2009), 1582–1589. https://doi.org/10.1016/j.respol.2009.09.002
18. I. Singh, N. Kumar, S. K. G., T. Sharma, V. Kumar, S. Singhal, Database intrusion detection using role and user behavior based risk assessment, *J. Inf. Secur. Appl.*, **55** (2020), 102654. https://doi.org/10.1016/j.jisa.2020.102654

19. D. Kempe, J. M. Kleinberg, E. Tardos, Maximizing the spread of influence through a social network, in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, Association for Computing Machinery, (2003), 137–146. https://doi.org/10.1145/956750.956769

20. P. Shakarian, A. Bhatnagar, A. Aleali, E. Shaabani, R. Guo, The independent cascade and linear threshold models, in *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, Association for Computing Machinery, (2015), 177–184. https://doi.org/10.1007/978-3-319-23105-1_4

21. M. Richardson, P. Domingos, Mining knowledge-sharing sites for viral marketing, in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, Association for Computing Machinery, (2002), 61–70. https://doi.org/10.1145/775047.775057

22. D. Oriedi, C. de Runz, Z. Guessoum, A. A. Nyongesa, Influence maximization through user interaction modeling, in *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, Association for Computing Machinery, (2020), 1888–1890. https://doi.org/10.1145/3386901.3388999

23. L. Sun, A. Chen, P. S. Yu, W. Chen, Influence maximization with spontaneous user adoption, in *Proceedings of the 13th International Conference on Web Search and Data Mining*, Association for Computing Machinery, (2020), 573–581. https://doi.org/10.1145/3336191.3371785

24. J. Guo, W. Wu, Adaptive influence maximization: if influential node unwilling to be the seed, *ACM Trans. Knowl. Discov. Data*, **15** (2021), 1–23. https://doi.org/10.1145/3447396

25. J. Luo, X. Liu, X. Kong, Competitive opinion maximization in social networks, in *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, Association for Computing Machinery, (2019), 250–257. https://doi.org/10.1145/3341161.3342899

26. Y. Zhang, Y. Zhang, Top-K influential nodes in social networks: A game perspective, in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Association for Computing Machinery, (2017), 1029–1032. https://doi.org/10.1145/3077136.3080709

27. X. Liu, X. Kong, P. S. Yu, Active ppinion maximization in social networks, *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Association for Computing Machinery, (2018), 1840–1849. https://doi.org/10.1145/3219819.3220061

28. P. Banerjee, W. Chen, L. V.S. Lakshmanan, Maximizing welfare in social networks under a utility driven influence diffusion model, in *Proceedings of the 2019 International Conference on Management of Data*, Association for Computing Machinery, (2019), 1078–1095. https://doi.org/10.1145/3299869.3319879

29. M. M. Keikha, M. Rahgozar, M. Asadpour, M. F. Abdollahi, Influence maximization across heterogeneous interconnected networks based on deep learning, *Expert Syst. Appl.*, **140** (2020). https://doi.org/10.1016/j.eswa.2019.112905

30. Q. Zhan, W. Zhuo, Y. Liu, Social influence maximization for public health campaigns, *IEEE Access*, **7** (2019), 151252–151260. https://doi.org/10.1109/ACCESS.2019.2946391

31. S. Tian, S. Mo, L. Wang, Z. Peng, Deep reinforcement learning-based approach to tackle topic-aware influence maximization, *Data Sci. Engineer.*, **5** (2020), 1–11. https://doi.org/10.1007/s41019-020-00117-1

32. D. Li, J. Liu, J. Jeon, S. Hong, T. Le, D. Lee, et al., Large-scale data-rriven airline market influence maximization, in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, Association for Computing Machinery, (2021), 914–924. https://doi.org/10.1145/3447548.3467365

33. C. Zhang, W. Li, D. Wei, Y. Liu, Z. Li, Network dynamic GCN influence maximization algorithm with leader fake labeling mechanism, *IEEE Trans. Comput. Soc. Syst.*, (2022), 1–9. https://doi.org/10.1109/TCSS.2022.3193583

34. W. Li, Z. Li, A. M. Luvembe, C. Yang, Influence maximization algorithm based on Gaussian propagation model, *Inf. Sci.*, **568** (2021), 386–402. https://doi.org/10.1016/j.ins.2021.04.061

35. W. Li, Y. Li, W. Liu, C. Wang, An influence maximization method based on crowd emotion under an emotion-based attribute social network, *Inf. Process. Manage.*, **59** (2022), 102818. https://doi.org/10.1016/j.ipm.2021.102818

36. W. Li, Y. Hu, C. Jiang, S. Wu, Q. Bai, E. M. K. Lai, ABEM: an adaptive agent-based evolutionary approach for influence maximization in dynamic social networks, *Appl. Soft Comput.*, **136** (2023), 110062. https://doi.org/10.1016/j.asoc.2023.110062

37. H. Cai, V. W. Zheng, K. C. Chang, A comprehensive survey of graph embedding: Problems, techniques, and applications, *IEEE Trans. Knowl. Data Engineer.*, **30** (2018), 1616–1637. https://doi.org/10.1109/TKDE.2018.2807452

38. B. Perozzi, R. Al-Rfou, S. Skiena, Deepwalk: online learning of social representations, in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, Association for Computing Machinery, (2014), 701–710. https://doi.org/10.1145/2623330.2623732

39. J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, Q. Mei, Line: large-scale information network embedding, in *Proceedings of the 24th international conference on World Wide Web*, Association for Computing Machinery, (2015), 1067–1077. https://doi.org/10.1145/2736277.2741093

40. A. Grover, J. Leskovec, node2vec: scalable feature learning for networks, in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, Association for Computing Machinery, (2016), 855–864. https://doi.org/10.1145/2939672.2939754

41. C. McCormick, Word2vec tutorial-the skip-gram model, 2016. Available from: http://mccormickml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model.

42. M. M. Keikha, M. Rahgozar, M. Asadpour, Community aware random walk for network embedding, *Knowledge-Based Syst.*, **148** (2018), 47–54. https://doi.org/10.1016/j.knosys.2018.02.028

43. T. Lou, J. Tang, Mining structural hole spanners through information diffusion in social networks, in *Proceedings of the 22nd international conference on World Wide Web*, Association for Computing Machinery, (2013), 825–836. https://doi.org/10.1145/2488388.2488461

44. R. S. Burt, Structural holes and good ideas, *Am. J. Soc.*, **110** (2004), 349–399. https://doi.org/10.1086/421787
45. S. Wu, J. M. Hofman, W. A. Mason, D. J. Watts, Who says what to whom on Twitter, in *Proceedings of the 20th International Conference on World Wide Web, WWW 2011*, Association for Computing Machinery, (2011). https://doi.org/10.1145/1963405.1963504
46. T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, 2013. Available from: http://arxiv.org/abs/1301.3781.
47. D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Slooten, S. M. Dawson, The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations, *Behav. Ecol. Soc.*, **54** (2003), 396–405. https://doi.org/10.1007/s00265-003-0651-y
48. A. L. Traud, P. J. Mucha, M. A. Porter, Social structure of Facebook networks, *Phys. A*, **391** (2012), 4165–4180. https://arxiv.org/1102.2166
49. M. J. Newman, Finding community structure in networks using the eigenvectors of matrices, *Phys. Rev. E*, **74** (2006), 036104. https://arxiv.org/abs/physics/0605087v3
50. J. Leskovec, J. Kleinberg, C. Faloutsos, Graph evolution: Densification and shrinking diameters, *ACM Trans. Knowl. Discovery Data*, **1** (2007). https://doi.org/10.1145/1217299.1217301
51. B. Rozemberczki, R. Davies, R. Sarkar, C. Sutton, Gemsec: graph embedding with self clustering, in *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2019*, Association for Computing Machinery, (2019), 65–72. https://doi.org/10.1145/3341161.3342890
52. J. Zhang, Y. Luo, Degree centrality, betweenness centrality, and closeness centrality in social network, in *Proceedings of the 2017 2nd International Conference on Modelling, Simulation and Applied Mathematics (MSAM2017)*, **132** (2017), 300–303. https://doi.org/10.2991/msam-17.2017.68
53. M. E. J. Newman, A measure of betweenness centrality based on random walks, *Soc. Net.*, **27** (2005), 39–54. https://doi.org/10.1016/j.socnet.2004.11.009
54. D. F. Gleich, PageRank beyond the Web, *SIAM Rev.*, **57** (2015), 321–363. https://doi.org/10.1137/140976649

AIMS Press