# Heterogeneous Information Network Embedding for Mention Recommendation

**FENG YI[1], BO JIANG[2], AND JIANJUN WU[3]**

[1]School of Computer Science, Zhongshan Institute, University of Electronic Science and Technology of China, Zhongshan 528402, China
[2]Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China
[3]Beijing College of Politics and Law, Beijing 100090, China

Corresponding author: Bo Jiang (jiangbo@iie.ac.cn)

**ABSTRACT** Mention recommendation is the task of recommending the right candidate users in a message. Many works have been conducted on the problem of whom to mention. However, due to the sparsity and heterogeneous of mention data, none of them well solve the problem. The recent advances in network embedding representation learning provide an effective approach to model the sparsity and heterogeneous simultaneously in heterogeneous information network. To this end, we propose a novel Network Embedding Mention (NEM) recommendation model to recommend the right users in a message. NEM constructs a heterogeneous mention network based on different relationships among different entities. Then NEM learns a unified low dimensional embedding vector using random walk for users and messages by considering network structure and vertex content information. Finally, whom to mention is ranked by calculating the relevance scores from heterogeneous user and message embeddings. To evaluate the proposed method, we construct a large dataset and their corresponding social networks from a real-world social media platform. Through extensive experiments on real-world mention collection, we demonstrate that our proposed model outperforms the previous state-of-the-art methods in term of recommendation task.

**INDEX TERMS** Mention recommendation, network embedding, relation network, heterogeneous information.

## I. INTRODUCTION

With the exponential growth of of social networks such as Twitter and Weibo, a lot of information is created and spread by millions of users every day on these platforms. Furthermore, information flow among users is very easy to form large-scale cascade diffusion through social connections. This feature has attracted significant interests from different social application tasks, such as helping market promotion [1]–[3], influencing political election [4]–[6], and detecting fake news [7]–[9].

Social networks naturally consist of communities corresponding to certain social circles or interest groups. The propagation of information flow therefore is hindered due to a combined effect of structural trapping, as shown in Figure 1(a). Fortunately, social network offers a key function named *Mention*, which allows user to place other users

The associate editor coordinating the review of this manuscript and approving it for publication was Fabrizio Messina[ID].

in a message by the form of @*username*. For example, Figure 1(b) illustrates the examples of mention on social network, where the user @*David* post a message "Welcome to San Francisco, USA!" by mentioning another users @*Alice* and @*Bob*, and the user @*John* express a love to @*Lucy* with the message "@*Lucy* awe i love you too!!!! 1 am here!!!!!!". These mentioned users will be received a notification by the *Mentions* tab. From this, we can see mention can break through the structural trapping of social network and expand the scale of diffusion with the containing mention messages. On the other hand, mention also plays an important role in enhancing the strength of relationship. Therefore, it is important and necessary to explore whom to mention in a message on social networks.

Many studies on whom-to-mention in a message have been reported on social networks in recent decades. A simple but powerful strategy is to model the recommendation of whom-to-mention based on the influencing factors of mention behaviors using rank learning method [10]–[13].
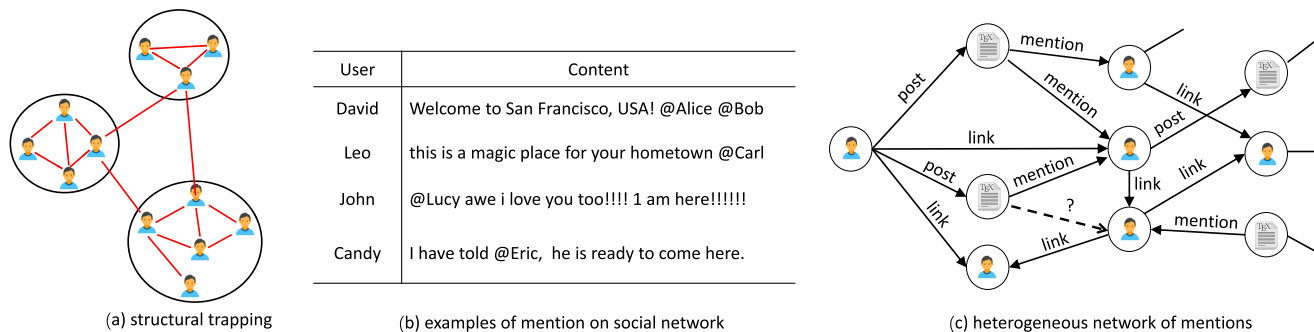
FIGURE 1. The observation and analysis of mention on social network.

Meanwhile, whom-to-mention can also be viewed as two classification problems on basis of link prediction algorithm [14]. Facing a large number of candidate users, mentioning the optimal users in the appropriate time is also an unbalance assignment problem [15]. Taken together, these studies suggest that the influencing factors of mention behaviors can be summarized as the content of message [11], [13], [16], the strength of social influence between users [13], [17], spatiotemporal information [11], [15], and the topic interests of user [17], [18]. However, none of the above models jointly consider multiple different object types (e.g., user, message and link relationship, etc.) in mention task.

The study of mention on social networks is based on the fusing heterogeneous data, including network structure, user profile and historical behaviors, as shown in Figure 1(c). From the figure, we can see the links capture the following relationships, the posts denote the publishing actions, and the mentions indicate the mentioning behaviors. This is a typical heterogeneous network through user, message and behaviors relationships. Besides, data sparsity is the common problem on social networks. It directly impedes the social models to achieve good results on specific tasks. To address these issues, network representation learning is proposed in the most recent research work. The network embedding algorithms not only encode the topological structure of each node, but also consider the rich information of node and alleviate the data sparsity [19]–[22].

Motivated by the network embedding, we propose a novel Network Embedding Mention recommendation model, called NEM, to recommend the right users in a message. This model first constructs a heterogeneous mention network based on following relationships, publishing actions and mention behaviors among different entities. Second, we utilize the network embedding model DeepWalk [19] to generate homogeneous, heterogeneous and correlation relationships, respectively. Third, in order to learn a unified low dimensional embedding vector for entities, we model these relationships from user and message dimensions by considering network structure and vertex content information. Finally, we calculate the relevance scores of the candidate users for a message to recommend the right users. We conduct extensive experiments on a social network dataset to evaluate the effectiveness

of our proposed model. The experiment results demonstrate that our proposed model outperforms the previous state-of-the-art methods in mention recommendations. This work makes the following three contributions:

- A heterogeneous mention network is constructed to model different entities such as author, message and user based on different relationships into a unified low dimensional embedding vector space. These heterogeneous information can be computed in the same embedding space.
- We propose a novel mention recommendation model based on deep random walk algorithm on the heterogeneous mention network. The approach can model both network structure and vertex content information. To the best of our knowledge, this is the first work for mention recommendation to exploit network embedding technique.
- Comprehensive experiments on a social network dataset clearly validate that our proposed model outperforms state-of-the-art methods, which proves the effectiveness of network embedding learning.

The rest of this paper is organized as follows. In the next section, we review related work. In Section 3, we give several necessary definitions used in this paper and present the formal definition of the mention recommendation problem. In Section 4, we present a heterogeneous mention network embedding recommendation model for heterogeneous information network and its learning and inference procedures. Then in Section 5, we empirically evaluate our proposed method on a real-world dataset including a comparison to baseline methods along with their results. Finally in Section 6 we give a conclusion of this work.

## II. RELATED WORK

In this section, we review the related work from two aspects, network embedding methods and mention behavior modeling.

### A. NETWORK EMBEDDING METHODS

A great number of methods with network embedding have been proposed from different dimensions. For example, Wang *et al.* [23] use deep neural network to represent the nonlinear relations among nodes. Ou *et al.* [24] employ graph

embedding algorithm to preserve high-order proximities of large scale graphs and capture the asymmetric transitivity. Tu *et al.* [25] consider user preference and social influence to improve the accuracy of social recommendation. Besides, network node incorporating the text content and label information can boost the quality of network embedding representation and improve the learning performance. For instance, Yang *et al.* [26] incorporate text features of vertices into network representation learning. Tu *et al.* [27] learn vertex representations to reflect both their network structure and labeling information by jointly optimizing the max-margin classifier and the aimed social representation learning model. Tu *et al.* [28] also design context-aware embeddings for vertices with mutual attention mechanism according to the neighbors. Grover and Leskovec [29] propose a semi-supervised algorithm for learning continuous feature representations by using a second-order random walk approach to generate network neighborhoods for nodes in network. Besides, the rich interaction information such as retweet, reply also play an important role in learning node embedding representation. Tu *et al.* [30] model the interactions between vertices with translation mechanism to extract social relation in social networks. Ren *et al.* [31] combine the node2vec method and the incremental strategy to improve the efficiency and recommendation accuracy in the financial news recommendation task. Yin *et al.* [32] propose an attention mechanism-based graph neural network to predict links on a bipartite user–item graph using information propagation. In summary, network embedding indeed can alleviate data sparsity and improve the performance of node learning.

### B. MENTION BEHAVIOR MODELING

A number of studies have been proposed to analyze whom to mention in a message on social networks. Naturally, the problem can be viewed as the ranking task. Wang *et al.* [10] use ranking support vector regression to recommend the candidates with some features including user interest match, content-dependent user relationship and user influence. Tang *et al.* [11] use ranking support vector machine as the solution by utilizing content, social, location and time based features. To solve the mention overload problem, Zhou *et al.* [12] propose a personalized ranking model by considering multi-dimensional relations among users and tweets to generate the personalized mention list. Besides, Li *et al.* [13] utilize probabilistic factor graph model with mention relationship as edges and candidates as nodes to help users deal with overwhelmed information. Gong *et al.* [16] propose a topical translation-based method to predict the mentioned users by taking into consideration both content of microblog and histories of candidate users. Recently, neural network-based methods have been used for the problem. Huang *et al.* [18] design an end-to-end memory network architecture by incorporating users' interests with external memory. Ma *et al.* [33] propose a cross-attention memory network by using user's interests with external memory and the cross-attention mechanism to extract both textual and

visual information to improve the mention performance. Gui *et al.* [34] propose a cooperative multi-agent reinforcement learning method for mention recommendation by incorporating dozens of times as many historical tweets as previous approaches. These above mentioned models aim to find the right users to mention in a message for information diffusion. Whom to mention can also be cast as the other tasks. For instance, Jiang *et al.* [14] use link prediction to predict the mentioned users by using user, textual, link and temporal information features. Bao *et al.* [35] formulate it as a binary classification task by using three factors from structure, influence, and content. Ding *et al.* [15] model the task as an unbalance assignment problem using Hungarian method to find the optimal users.

However, existing approaches haven't yet been solved the problems well due to data sparsity and rich heterogeneous information. To the best of our knowledge, no existing studies have tackled the problem of whom to mention in a message using network embedding technique. Our study aims to leverage it.

## III. HETEROGENEOUS MENTION NETWORK
In this section, we first introduce basic concepts. Then we present notations used in the remaining of the paper.

### A. HETEROGENEOUS MENTION NETWORK
A heterogeneous mention network can be represented as $G = (V, E, W, C)$, where $V$ is a set of nodes, $E$ is a set of directed relationships between nodes, the relationships include following, post and mention. $W$ is a weight matrix that represents link weight among nodes, and $C$ denotes the set of meta information with nodes.

Figure 1(c) is an example for heterogeneous mention network on social network. The heterogeneous network not only includes three types of nodes (e.g., authors, messages and users), but also holds three types of links, where following denotes a user follow any others, and posting denotes a message is posted by the author, and mentioning denotes a user is mentioned in a message.

### B. META-PATH
Meta-path is defined as a path of linking two different nodes on network schema. In this paper, we formally define meta-path as $O_1 \xrightarrow{R_1} O_2 \xrightarrow{R_2} \cdots \xrightarrow{R_l} O_l$, where $O_l$ denotes the type of node, and $R_l$ denotes the type of relationship. Obviously, from the definition, $G$ is heterogeneous network.

The Figure 1(c) also shows different types of meta-paths. For example, $u \xrightarrow{follow} v$ represents user $u$ follows user $v$, $m \xrightarrow{mention} v$ represents message $m$ is mentioned user $v$ in the body, and $u \xrightarrow{follow} v \xrightarrow{post} m$ represents user $u$ receive message $m$ posted by user $v$. The different meta-paths with different linking nodes indicates different semantic relations and link networks. Table 1 describes the types of all relationships rules in our mention heterogeneous network.

**TABLE 1.** Relationship rules between vertexes.

| Types | Notation | Semantics |
|---|---|---|
| Link Relationship | $u \xrightarrow{follow} v$ | user $u$ is user $v$'s follower |
| | $u \xleftarrow{follow} v$ | users $u$ and $v$ are the mutual followers |
| | $u \xrightarrow{follow} v \xrightarrow{follow} g$ | user $u$ is a follower of user $g$'s follower |
| Action Relationship | $u \xrightarrow{post} m$ | user $u$ post a message $m$ |
| | $m \xrightarrow{mention} v$ | message $m$ mention user $v$ |
| | $u \xrightarrow{post} m \xrightarrow{mention} v$ | message $m$ posted by user $u$ mention user $v$ |
| Mix Relationship | $u \xrightarrow{follow} v \xrightarrow{post} m$ | user $u$ receive message $m$ posted by user $v$ |
| | $u \xrightarrow{follow} v \xrightarrow{post} m \xrightarrow{mention} g$ | user $u$ receive message $m$ posted by user $v$ containing user $g$ |

## IV. HETEROGENEOUS MENTION NETWORK EMBEDDING BASED MENTION RECOMMENDATION MODEL

In this section, we present a novel network embedding mention recommendation model, named NEM. This model is a unified learning framework based on random walk, which uses the network structure and vertex content information when learning vertex representations. We optimize the objective function by using hierarchical softmax, and rank the relevance scores of mention candidates.

### A. HETEROGENEOUS MENTION NETWORK EMBEDDING

The goal of NEM model is to learn a unified low dimensional embedding vector for user and message in the constructed heterogeneous mention network. Based on the network embedding representation, vertices sharing common edges or with similar text content are close in the embedding space, and can be performed the comparison and computation. In this paper, we divide the heterogeneous relationships among the vertices from heterogeneous mention network $G$ into three categories: homogeneous relationship, heterogeneous relationship and correlation relationship, as shown in Figure 2. In the following, we discuss different relationship modeling and then present heterogeneous mention network embedding.

### 1) HOMOGENEOUS RELATIONSHIP MODELING

Homogeneous network consists of the same type of target nodes, and the relationships among nodes are obtained from heterogeneous mention network based on user and message dimensions.

### a: USER-USER RELATIONSHIP MODELING

There are typically multiple paths both users in heterogeneous mention network. For example, user $A$ follows user $B$ indicates how much $A$ interests $B$ in the simple follow network, and user $A$ interacts user $B$ demonstrates how much $A$ knows $B$ in the interaction network. Therefore, user-user relationships need to be aggregated over these two paths.

According to the idea of DeepWalk [19], we construct a random walk generated from the heterogeneous mention network $G$. Specifically, we use a random walk path $s = \{u_1, u_2, \cdots, u_l\}$ to represent a sentence, where each vertex $u_i$

corresponds to a word in neural language models. We then utilize DeepWalk method to train Skip-Gram model on the generated random walk collection $S_u$ of users, and obtain a distributed vector representation for each user vertex. Given a vertex $u_i$ for all random walks $s \in S_u$, the objective function of user-user relationship modeling is defined as:

$$J_1 = \sum_{s \in S_u} \sum_{i=1}^{M} \sum_{t=-d, t \neq 0}^{d} logPr(u_{i+t}|u_i) \quad (1)$$

where $M$ is the number of users and $d$ is the length of contextual window with $u_i$. Given the current user vertex $u_i$, the probability of finding neighboring vertices $\{u_{i-d} : u_{i+d}\}$ is calculated as:

$$Pr(u_j|u_i) = \frac{exp(v_{u_j}^T \hat{v}_{u_i})}{\sum_{j=1}^{M} exp(v_{u_j}^T \hat{v}_{u_i})} \quad (2)$$

where $\hat{v}_{u_i}$ and $v_{u_j}$ are the input and output representation vectors of the vertices $u_i$ and $u_j$, respectively.

### b: MESSAGE-MESSAGE RELATIONSHIP MODELING

According to the results in [11], messages posted by different users with similar topic distributions are likely to mention the same users. Hence, we model message-message relationship to measure the strength of semantic dependence among message vertices. The distribution of topics satisfies the hypothesis that the messages similarities in observed spaces are consistent with the latent spaces. Meanwhile,

Similar to the user-user relationship modeling approach, the objective function of message-message relationship modeling is defined as:

$$J_2 = \sum_{s \in S_m} \sum_{j=1}^{N} \sum_{w=-d, w \neq 0}^{d} logPr(m_{j+w}|m_j) \quad (3)$$

where $N$ is the number of messages and $S_m$ is the generated random walk collection of messages. The probability of observing neighboring vertices $\{m_{j-d} : m_{j+d}\}$ given a message vertex $m_i$ is calculated using the softmax function as:

$$Pr(m_j|m_i) = \frac{exp(v_{m_j}^T \hat{v}_{m_i})}{\sum_{j=1}^{N} exp(v_{m_j}^T \hat{v}_{m_i})} \quad (4)$$

**(a) homogeneous relationships**    **(b) heterogeneous relationships**    **(c) correlation relationships**
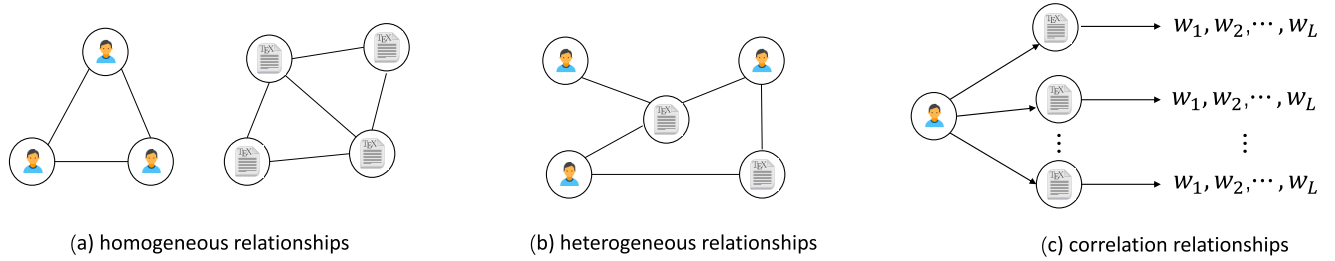
**FIGURE 2.** The category of mention relationships on social network. We classify the pairwise relationships among the vertices in the heterogeneous mention network G into three categories: homogeneous relationship, heterogeneous relationship and correlation relationship.

where $\hat{v}_{m_i}$ and $v_{m_j}$ are the input and output representation vectors of the vertices $v_i$ and $v_j$, respectively.

### 2) HETEROGENEOUS RELATIONSHIP MODELING

Similarly, heterogeneous network consists of the different type of target nodes, and the relationships are extracted from the interactions of users and messages in the heterogeneous network.

#### a: USER-MESSAGE RELATIONSHIP MODELING

There are two types of relationships used in the heterogeneous mention network that are referred to as posting and mentioning, respectively. Posting indicates a user post a message in social network, and mentioning represents a user is mentioned in the body of a message. We consider two relationships as a random walk path, and model user-message relationship to measure the strength of interactions among users and messages.

Based on these findings, we leverage the valuable relationship information between users and messages. In particular, we first collect all the messages posted by the same user. Then we use the user vector as input and simultaneously learn the input user vectors and output message vectors. It can be formalized by the following objective function:

$$J_3 = \sum_{i=1}^{M} \sum_{j=-d}^{d} logPr(m_j|u_i) \tag{5}$$

The probability of observing the message given a user $u_i$ can be calculated as:

$$Pr(m_j|u_i) = \frac{exp(v_{m_j}^T \hat{v}_{u_i})}{\sum_{j=1}^{N} exp(v_{m_j}^T \hat{v}_{u_i})} \tag{6}$$

### 3) CORRELATION RELATIONSHIP MODELING

Correlation network indicates the attribute information of the target nodes, and the relationships are extracted from the heterogeneous mention network on the basis of content-dimension.

#### a: USER-CONTENT CORRELATION MODELING

User interests could impact the overall effectiveness of mention recommender. Meanwhile, the interests of a user can be inferred by user generated contents. Therefore, we also model user interests on basis of the content of message. Based on the assumption that similar users have similar patterns of mention behavior in observe space and in hidden space, Our observation also find that users with similar topic interests are likely to be mentioned in the same message.

Specifically, we collect all the text content associated with messages which are posted by one user. Texts are represented by bag-of-words model. Finally, the objective function is formulated as

$$J_4 = \sum_{i=1}^{M} \sum_{j=-d}^{d} logPr(w_j|u_i) \tag{7}$$

Similarly, the probability of observing the words given a user $u_i$ is defined as:

$$Pr(w_j|u_i) = \frac{exp(v_{w_j}^T \hat{v}_{u_i})}{\sum_{w=1}^{L} exp(v_w^T \hat{v}_{u_i})} \tag{8}$$

where $\hat{v}_{w_j}$ is the output representation of word $w_j$ and $L$ is the number of distinct words in the whole heterogeneous information network.

#### b: MESSAGE-CONTENT CORRELATION MODELING

The goal of mention is to expand information spread. The message contents directly influence diffusion effects. Thus, the correlation between message and content should be considered.

Based on the above observations, we collect all the text content corresponded to one message. Then the message-content correlation relationship can be modeled as the contextual information of words within a message. The objective function is achieved by maximizing the following log-likelihood function:

$$J_5 = \sum_{i=1}^{N} \sum_{j=-d}^{d} logPr(w_j|m_i) \tag{9}$$

The probability of observing contextual words given vertex $m_i$ is defined as:

$$Pr(w_j|m_i) = \frac{exp(v_{w_j}^T \hat{v}_{m_i})}{\sum_{w=1}^{L} exp(v_w^T \hat{v}_{m_i})} \tag{10}$$

## B. HETEROGENEOUS MENTION NETWORK EMBEDDING
In this section, we explain the proposed MEN model, which is used to jointly capture the network structure and the content information associated with each vertex in the network.

### c: MODEL ENSEMBLE
We incorporate homogeneous relationships, and heterogeneous relationships, as well as correlation relationships into an integrated framework. The objective of the integrated framework is to maximize the following log likelihood function as

$$
\begin{aligned}
L = {} & \lambda \sum_{s \in S_u} \sum_{i=1}^{M} \sum_{t=-d, t \neq 0}^{d} logPr(u_{i+t}|u_i) \\
& + \lambda \sum_{s \in S_m} \sum_{j=1}^{N} \sum_{w=-d, w \neq 0}^{d} logPr(m_{j+w}|m_j) \\
& + \lambda \sum_{i=1}^{M} \sum_{j=-d}^{d} logPr(m_j|u_i) \\
& + (1 - \lambda) \sum_{i=1}^{M} \sum_{j=-d}^{d} logPr(w_j|u_i) \\
& + (1 - \lambda) \sum_{j=1}^{N} \sum_{j=-d}^{d} logPr(w_j|m_j)
\end{aligned}
\tag{11}
$$

where $\lambda$ is a tunable parameter to balance the weight strength between network structure information and text content information, $d$ is the window size of sequence and $w_j$ is the $j$-th word in a contextual window.

Notice that the first three terms in Eq. (11) denote mutual reinforced information between users and messages, the last two terms in Eq. (11) indicate the text information and user information will jointly affect $\hat{v}_{w_j}$, which represents the output representation of word $w_j$. In turn, the word vector representation can further propagate back to influence the input representation of $u_i$ and $m_j$ in the network. As a result, the vertex representation (i.e., the input vectors of vertices) will be enhanced by both network structure and content information.

### d: MODEL OPTIMIZATION
The stochastic gradient descent can directly be used to solve Eq. (4.11), but the process of computing the gradient in getting conditional probability in Eq. (4.2), Eq. (4.4), Eq. (4.6), Eq. (4.8) and Eq. (4.10) is time-consuming and expensive cost. This is because the softmax probability of all words need to be calculated. To reduce computational complexity, hierarchical softmax [36] model can be used to build three Huffman trees. Specifically, user vertices can be considered as leaves, and message vertices as leaves as well as distinct words as leaves. Therefore, given a target vertex $u_i$, then we can compute as follows:

$$
Pr(u_k|u_i) = \prod_{k=1}^{c} Pr(s_{2k}|u_i)
\tag{12}
$$

where $(s_{20} \rightarrow s_{21} \rightarrow \cdots \rightarrow s_{2c})$ represents the path from the root $s_{20}$ to each leaf $s_{2c}$ for vertex $u_i$.

Meanwhile, $P(s_{2k}|u_i)$ can be further modeled by a binary classifier, which is defined as:

$$
P(s_{2k}|u_i) = \sigma(v_{u_i}^T \hat{v}_{v_{s_{2k}}})
\tag{13}
$$

where $\sigma(x)$ is the sigmoid function and $\hat{v}_{v_{s_{2k}}}$ is the representation of tree vertex $s_{2k}$'s parent. We can also use the technique to compute conditional probability in Eq. (4), Eq. (6), Eq. (8) and Eq. (10).

### e: NETWORKING EMBEDDING MENTION RECOMMENDATION
We finally consider the task of mention recommendation as a ranking problem over a set of candidate users $p_i \in P$ ($i = 1, 2, \cdots, L$). The proposed model aims to return the top ranked candidate users by measuring the relevance scores between user and message. Specifically, given a coming message $m$ posted by $u$, the relevance scores is calculated based on heterogeneous embedding into vectors as

$$
r_q = V_m v_m^T + V_u v_u^T
\tag{14}
$$

where $V_m = [v_{m_1}; v_{m_2}; \cdots; v_{m_N}]$ is the vector representation of training messages, $v_m$ is the vector representation of the coming message $m$, $V_u = [v_{u_1}; v_{u_2}; \cdots; v_{u_M}]$ is the vector representation of users related to training messages, $v_u$ is the vector representation of user $u$. Once the training is finished, the candidates are ranked by the relevance scores. Finally, we obtain the final user list for mention recommendation based on the top ranked. Algorithm 1 summarizes the whole process that determines the training users relevance scores associated with the given message.

## V. EXPERIMENTS AND ANALYSIS
### A. DATASET AND METRICS
#### 1) DATA DESCRIPTION
Weibo is one of the most popular social network platforms in China like Twitter, which allows users to follow each other, and mentions any users in a message. In this paper, we use a publicly available Weibo dataset [37], which consists of user profile, tweet and the snapshot of network structure, etc. Particularly, lots of tweets contain mention behaviors, which is more suitable for experimental datasets.

In order to achieve high-quality results, we preprocess the dataset by extracting all interactions represented via @ symbol. Since publisher may mention multiple users in a message, we can extract multiple mention instances. Besides, we prepare our dataset by sampling users whose number of historical interactions are between 10 and 1,000 as publishers, to avoid the influences of inactive users and extremely active spammers. Table 2 summarizes the detailed information of the used dataset.

#### 2) EVALUATION METRICS
We use Precision, Recall and F-Score to evaluate the performance of user recommendation methods for the highest

---

**Algorithm 1** The Learning Algorithm for NEM Model

---

**Require:** The heterogeneous mention network $G = (V, E, W, C)$, the user $A$, the coming messages and all the candidates, expected number of dimension of the vector representation $k$, window size $d$, iteration number $T$, the walk length of Random Walk $l$ and the number of the recommended user $U$.

**Ensure:** Recommendation mentioned user list.

 1: Generate random walk collection $S_u$ from the user and authors of the training messages, generate random walk collection $S_m$ from the coming message and training messages;

 2: Generate a user binary tree $T_u$, generate a message binary tree $T_m$ and generate a vocabulary binary tree $T_w$;

 3: Get the initial input vector representation $v_u$, $v_m$ and output vector representation $\hat{v}_u$, $\hat{v}_m$ for the user and each author of the training message, the coming message and all the training messages, respectively;

 4: Get the initial output vector representation $\hat{v}_{w_j}$ for each word $w_j \in W$

 5: **for** $iter = 1, 2, 3, \cdots, T$ **do**

 6:     Fix $\hat{v}_{w_j}$, solve Eq.(11) to update $v_u$, $v_m$, $\hat{v}_u$, $\hat{v}_m$;

 7:     Fix $\hat{v}_u$, $\hat{v}_m$, solve Eq.(11) to update $v_u$, $v_m$, $\hat{v}_{w_j}$;

 8: **end for**

 9: Calculate the relevance score $r_q$ for the given message and rank the candidate mentioned users according to $r_q$;

10: Select top ranking $Q$ candidate users as recommendation list.

---

| Dataset | #Users | #Messages | #Relations | #Mentions |
|---------|--------|-----------|------------|-----------|
| Weibo   | 11,925 | 208,274   | 5,368,253  | 2,382,052 |

ranked result. To evaluate whether candidates with the top $n$ highest scores contain the correct result, we use Hits@3 and Hits@5. Since the rank of the results is usually important, we also use the Mean Reciprocal Rank (MRR) metrics to evaluate the rank of the recommended results.

### B. BASELINE METHODS

We compare the proposed model with the following state-of-the-art methods:

- **Frequency Descending (FD)**: The ranked list depends on the frequency ranks of the candidates in the history. A candidate user mentioned with a higher frequency by the author would have a higher rank in the recommendation list.
- **WTM**: This method uses support vector regression as ranking algorithm by measuring user interest match, content-dependent user relationship and user influence [10].
- **PMPR**: This model considers the mention recommendation as a probabilistic ranking problem to find the maximal possibility candidate by using probabilistic

factor graph model in the heterogeneous social network [13].
- **CAR**: This method uses a ranking support vector machine model by considering content, social, location and time based features to recommend the target users [11].
- **A-UUTTM**: This method considers not only the content of a microblog but also the histories of candidates based on translation-based model [16].
- **AU-HMNN**: The model proposed in this paper incorporates the textual information of query tweets and the history interests of the author and candidate users. The history interests encoder is a memory network architecture with a hierarchical attention mechanism [18].

In our experiments, all of the above baseline methods are set with default parameter settings. For example, the number of mention in CAR is 4, A-UUTTM set $\mu = 0.8$ and the topic number to 30, AU-HMNN fix the number of hops to 6 and the number of embedding dimensions to 300. For our model, we empirically set $\lambda = 0.2$, and the embedding dimension to 30, and the window size to 2, and the length of random walk to 20, and the number of iterations to 100.

### C. COMPARISON WITH OTHER NETWORK EMBEDDING APPROACHES

To evaluate the performance of heterogeneous information network embedding based recommendation approach, we compare it with the other four embedding based recommendation approaches:

- **DeepWalk**: It learns paper network representation by utilizing network structure information [19].
- **LINE**: It preserves local and global network structure to learn paper network representation [20].
- **Node2Vec**: It embeds variable length of text into a fixed length distributed vector using neural network models [29].
- **Author2Vec**: It embeds variable length of text into a fixed length distributed vector using neural network models [38].
- **TriDNR**: It simultaneously considers paper network structure and paper vertex content to learn paper network representation [39].

In this paper, the following default parameter values are used in our experiments: In DeepWalk model, we set the number of walks *numwalk=80*, the length of walk *legwalk=40*, the dimension of the embedding *embedding=128*. For Line model, the dimension of the embedding is 100, the number of negative samples is 5, the starting value of the learning rate is 0.025. In Node2Vec model, we set the length of walk to 80, the number of walks to 10, the upper bound of number of neighbors to 30. Meanwhile, we set the dimension of the author embedding to 100, the size of window to 10 in Author2Vec method. For TriDNR, the weight of text is 0.8, the number of feature is 100, and the random state is 2.

**TABLE 3.** Comparison of mention results with Precision, Recall, F-Score, MRR, Hits@3 and Hits@5 metrics.

| Category | Method | Precision | Recall | F-Score | MRR | Hits@3 | Hits@5 |
|---|---|---|---|---|---|---|---|
| Baselines | FD | 0.365 | 0.381 | 0.373 | 0.471 | 0.469 | 0.477 |
| | WTM | 0.543 | 0.566 | 0.554 | 0.541 | 0.545 | 0.557 |
| | PMPR | 0.636 | 0.640 | 0.638 | 0.635 | 0.648 | 0.657 |
| | CAR | 0.690 | 0.696 | 0.693 | 0.697 | 0.711 | 0.737 |
| | A-UUTTM | 0.712 | 0.728 | 0.720 | 0.719 | 0.726 | 0.748 |
| | AU-HMNN | 0.742 | 0.750 | 0.746 | 0.726 | 0.743 | 0.751 |
| | CROMA | 0.751 | 0.765 | 0.758 | 0.758 | 0.752 | 0.758 |
| | NEM | **0.763** | **0.779** | **0.771** | **0.775** | **0.760** | **0.767** |
| Embeddings | DeepWalk | 0.609 | 0.604 | 0.606 | 0.618 | 0.649 | 0.663 |
| | LINE | 0.646 | 0.635 | 0.640 | 0.651 | 0.665 | 0.695 |
| | Node2Vec | 0.668 | 0.664 | 0.666 | 0.677 | 0.707 | 0.712 |
| | Author2Vec | 0.702 | 0.685 | 0.693 | 0.706 | 0.711 | 0.721 |
| | TriDNR | 0.710 | 0.696 | 0.703 | 0.714 | 0.724 | 0.733 |
| Features | NEM-UU | 0.637 | 0.633 | 0.635 | 0.646 | 0.673 | 0.679 |
| | NEM-MM | 0.669 | 0.653 | 0.660 | 0.673 | 0.678 | 0.687 |
| | NEM-UM | 0.617 | 0.606 | 0.611 | 0.621 | 0.634 | 0.663 |
| | NEM-UC | 0.653 | 0.649 | 0.650 | 0.661 | 0.688 | 0.694 |
| | NEM-MC | 0.676 | 0.664 | 0.670 | 0.680 | 0.690 | 0.698 |
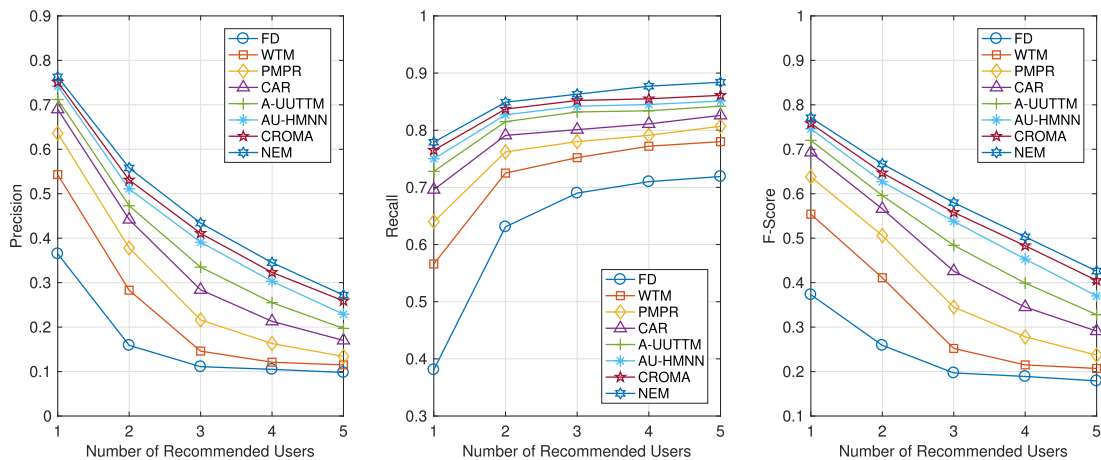


**FIGURE 3.** Precision, recall, and F-Score in different number of recommended users.

## D. PERFORMANCE AND ANALYSIS

In this section, we compare the recommendation performance of all baseline methods and variants in Table 3, and show the performance changes on the different number of recommended users in Figure 3.

### 1) PERFORMANCE COMPARISON WITH OTHER BASELINE METHODS

Table 3 with baselines category shows the comparisons of the proposed method with the state-of-the-art methods. From the results, we can draw the following observations: (1) Our proposed model outperforms the baseline methods. Furthermore, the advantages of our approach are more obvious as the number of mention increases. (2) AU-HMNN can achieve better the performance of mention recommendation than most of conventional methods (e.g., CAR, PMPR, WTM, FD) when using an end-to-end memory network. (3) A-UUTTM performs obviously better than the other methods, which demonstrates that the topic interests is an important factor

for mention. (4) CAR has a obviously better performance than the aforementioned baselines. (5) The reinforcement learning method like CROMA for mention recommendation is a good research direction. The experimental results show that considering a lot of historical tweets can capture the mention behaviors successfully. In particular, the best results of our proposed model for MRR and Hits@5 are relatively greater performance compared with other methods. Hence, by incorporating homogeneous relationships, and heterogeneous relationships as well as correlation relationships, our proposed model indeed performs well in mention recommendation based on network embedding technique.

### 2) OTHER NETWORK EMBEDDING APPROACHES FOR EVALUATION

We perform the different network embedding methods in heterogeneous mention network, and the experimental results are shown in Table 3 with embeddings category. From the comparison results, we can conclude DeepWalk-based

and LINE-based mention recommendation methods perform poorly in all the comparison results. The main reason is both approaches only consider the structure features of sparse vertices, and ignore the node information. Node2Vec achieves better performance than DeepWalk-based and LINE-based methods. An explicit explanation is Node2Vec designs a smart search strategies for nodes. As results illustrate, Author2Vec shows better performance than all above works. This is because the author nodes hold the rich information comparing to network structure. Compared with all above methods with the topological information and the author information, TriDNR-based mention recommendation approach shows the best performance when exploiting inter-node relationships and node-content correlation to learn an optimal representation for each node. Finally, we can see that the proposed NEM approach outperforms all the other network embedding approaches by utilizing homogeneous relationships and heterogeneous relationships as well as correlation relationships with network embedding representation.

### 3) RESULT COMPARISON ON DIFFERENT RELATIONSHIP FEATURES

We implement the different variants of our model to show the effectiveness of our proposed factors in Table 3 with features category:

- **NEM-UU**: Eliminating the effect of user-user relationship modeling by removing the first term in Eq. (4.11).
- **NEM-MM**: Eliminating the effect of message-message relationship modeling by removing the second term in Eq. (4.11).
- **NEM-UM**: Eliminating the effect of user-message relationship modeling by removing the third term in Eq. (4.11).
- **NEM-UC**: Eliminating the effect of user-content correlation modeling by removing the forth term in Eq. (4.11).
- **NEM-MC**: Eliminating the effect of message-content correlation modeling by removing the fifth term in Eq. (4.11).

From the Table 3 with features category, we can conclude that (1) The performance of NEM degrades when eliminating any of those relationship features. (2) Furthermore, we can observe that basically the order of importance for all features is NEM-UM > NEM-UU > NEM-UC > NEM-MM > NEM-MC. More specifically, user-message relationship modeling is the most important feature, which indicates that the topical match degree between the coming message and target users is the most significant factor when occurring a mention. Second, user-user relationship modeling recommendation performs better than the other three. The result indicates that the strength of social influence among users affects future interaction behaviors. Third, performance for message-message relationship modeling recommendation is relatively poor. The primary reason is that message on social network dataset is highly sparse. In summary, we can observe

that the topic interests of user and the strength of history interaction are the primary factors in the mention recommend task.

### 4) EFFECT OF MENTION COUNT

Figure 3 shows the performance of our model and the baseline models with different numbers of recommended users, varying from 1 to 5. From the figure, we can see that (1) Our proposed model achieves consistently better performance than the other methods with different number of recommendations. (2) With the number of recommended users increasing, Precision decreases and Recall increases gradually, indicating while the number of mentions in a message are neither too much nor too little, the recommendation achieves the best performance. It is a reasonable explanation that each message to be mentioned with a small number of users (e.g., one to three persons) would be viewed as a intimate chat with close friends, and a spam otherwise. (3) We also observe that the best performance of mention recommendation is obtained in term of F-Score when recommending the top one user.

### E. PARAMETER SENSITIVITY

We analysis the influence of several critical parameters of the proposed model in the following perspectives.

### 1) EFFECT OF λ

The parameter $\lambda$ is a tunable weight to balance the strength of the network structure and message content. Here, we report the experimental results of our proposed model when $\lambda$ varies from 0.0 to 1.0, as shown in Figure 4. From the figure, we observe that our model obtains the best performance when $\lambda = 0.2$. Furthermore, when the weight parameter $\lambda$ is close to 0, the network topology cannot affect the final result. When $\lambda$ is sufficiently large, the performance decrease with a larger number of sparsity network structure.
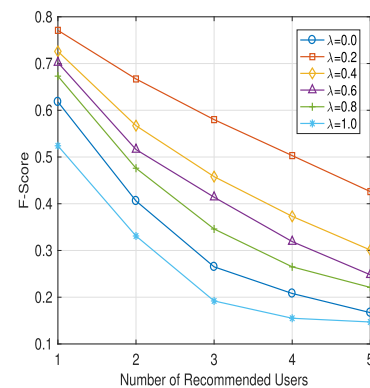


**FIGURE 4.** The effect of *d*.

### 2) EFFECT OF EMBEDDING DIMENSION *k*

The embedding dimension indicates the power of vector representation for vertices. Figure 5 plots the performance of our proposed model with various values of *k*. From the
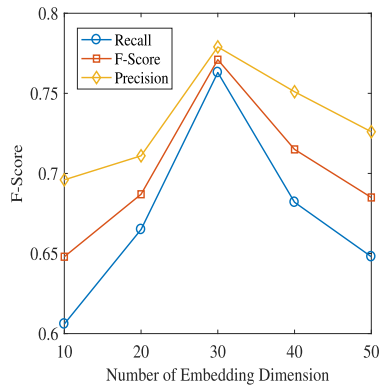
**FIGURE 5.** The effect of *k*.

figure, we can observe that the plot first gradually rises and then drops gradually along with the increase of *k* gradually. In particular, it is clear that the best performance is achieved when *k* is around 30. Therefore, we choose $k = 30$ as the latent embedding dimension.

### 3) EFFECT OF WINDOW SIZE *d*
The window size decide the prediction power of contextual word with vertices. Figure 6 plots the performance of our proposed model with various values of *d*. From the figure, we can observe that the plot gradually rises with the increase of window size *d* gradually, and F-Score becomes stable when $d = 2$. Considering the calculation effect and time efficiency, we choose $d = 2$ as the window size.
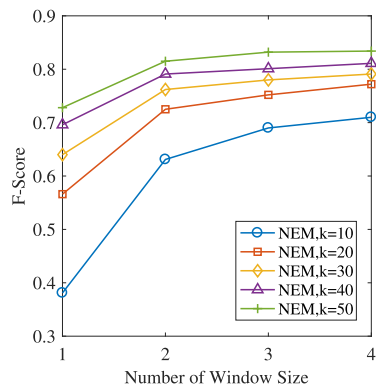


**FIGURE 6.** The effect of *d*.

### 4) EFFECT OF LENGTH OF RANDOM WALK *l*
We analyze the contribution of different length of random walk on mention recommendation. Specifically, we respectively set the length of random walk to {5,10,20,50} into our proposed model and evaluate the increase of the recommendation performance. Table 4 shows the accuracy of different lengths on our proposed model. From the table, we can observe clear increase on the performance when adding the length of random walk. In particular, $l = 20$ indicates that our proposed model performs the best performance.

**TABLE 4.** Performances of our proposed model with the different lengths of random walk.

| Length of RW | Precision | Recall | F-Score |
|---|---|---|---|
| 5 | 0.572 | 0.773 | 0.659 |
| 10 | 0.703 | 0.753 | 0.723 |
| 20 | **0.763** | **0.779** | **0.771** |
| 50 | 0.722 | 0.743 | 0.732 |

As increasing the length of random walk, the performance of our proposed model will further boost. It is observed that it is better to set 20 length with an acceptable time cost.

## VI. CONCLUSION
In this paper, we propose a novel network embedding mention recommendation model to recommend the right users in a message by incorporating homogeneous relationships, and heterogeneous relationships as well as correlation relationships. Based on network embedding, we use these mention relationship models to construct the objective function under the framework of random walk. To demonstrate the effective of the proposed model, we construct extensive experiments. The experimental results reveal that our proposed method can outperform the state-of-the-art baseline methods. As future work, we will consider how to apply graph neural network methods to better incorporate the embeddings of multiple social contexts.

## REFERENCES
[1] J. Tang, S. Wang, X. Hu, D. Yin, Y. Bi, Y. Chang, and H. Liu, "Recommendation with social dimensions," in *AAAI*, vol. 2016, pp. 251–257.
[2] L. Tang and E. Y. Liu, "Joint user-entity representation learning for event recommendation in social network," in *Proc. IEEE 33rd Int. Conf. Data Eng. (ICDE)*, Apr. 2017, pp. 271–280.
[3] X. Wang, S. C. H. Hoi, M. Ester, J. Bu, and C. Chen, "Learning personalized preference of strong and weak ties for social recommendation," in *Proc. 26th Int. Conf. World Wide Web*, Apr. 2017, pp. 1601–1610.
[4] R. Buettner and K. Buettner, "A systematic literature review of Twitter research from a socio-political revolution perspective," in *Proc. 49th Hawaii Int. Conf. Syst. Sci. (HICSS)*, Jan. 2016, pp. 2206–2215.
[5] K. Johnson, D. Jin, and D. Goldwasser, "Leveraging behavioral and social information for weakly supervised collective classification of political discourse on Twitter," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2017, pp. 741–752.
[6] J. Hemsley and S. Jackson, "Political issues that spread: Understanding retweet behavior during the 2016 US presidential election," in *Proc. 9th Int. Conf. Social Media Soc.*, 2018, pp. 305–309.
[7] J. Ma, W. Gao, P. Mitra, S. Kwon, B. J. Jansen, K.-F. Wong, and M. Cha, "Detecting rumors from microblogs with recurrent neural networks," in *Proc. IJCAI*, 2016, pp. 3818–3824.
[8] M. Lukasik, P. K. Srijith, D. Vu, K. Bontcheva, A. Zubiaga, and T. Cohn, "Hawkes processes for continuous time sequence classification: An application to rumour stance classification in Twitter," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, vol. 2, 2016, pp. 393–398.
[9] S. Kwon, M. Cha, and K. Jung, "Rumor detection over varying time windows," *PLoS ONE*, vol. 12, no. 1, 2017, Art. no. e0168344.
[10] B. Wang, C. Wang, J. Bu, C. Chen, W. V. Zhang, D. Cai, and X. He, "Whom to mention: Expand the diffusion of tweets by@ recommendation on micro-Blogging systems," in *Proc. 22nd Int. Conf. World Wide Web*, 2013, pp. 1331–1340.
[11] L. Tang, Z. Ni, H. Xiong, and H. Zhu, "Locating targets through mention in Twitter," *World Wide Web*, vol. 18, no. 4, pp. 1019–1049, Jul. 2015.

[12] G. Zhou, L. Yu, C.-X. Zhang, C. Liu, Z.-K. Zhang, and J. Zhang, "A novel approach for generating personalized mention list on micro-blogging system," in *Proc. IEEE Int. Conf. Data Mining Workshop (ICDMW)*, Nov. 2015, pp. 1368–1374.

[13] Q. Li, D. Song, L. Liao, and L. Liu, "Personalized mention probabilistic ranking–recommendation on mention behavior of heterogeneous social network," in *Proc. Int. Conf. Web-Age Inf. Manage.* Springer, 2015, pp. 41–52.

[14] B. Jiang, Y. Sha, and L. Wang, "Predicting user mention behavior in social networks," in *Natural Language Processing and Chinese Computing*. Springer, 2015, pp. 146–158.

[15] Z. Ding, X. Zou, Y. Li, S. He, J. Cheng, F. Qiao, and H. Wang, "Mentioning the optimal users in the appropriate time on twitter," in *Asia-Pacific Web Conf.* Springer, 2016, pp. 464–468.

[16] Y. Gong, Q. Zhang, X. Sun, and X. Huang, "Who will you@?" in *Proc. 24th ACM Int. Conf. Inf. Knowl. Manage.*, 2015, pp. 533–542.

[17] S. Pramanik, Q. Wang, M. Danisch, S. Bandi, A. Kumar, J.-L. Guillaume, and B. Mitra, "On the role of mentions on tweet virality," in *Proc. IEEE Int. Conf. Data Sci. Adv. Analytics (DSAA)*, Oct. 2016, pp. 204–213.

[18] H. Huang, Q. Zhang, and X. Huang, "Mention recommendation for Twitter with End-to-end memory network," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 1872–1878.

[19] B. Perozzi, R. Al-Rfou, and S. Skiena, "DeepWalk: Online learning of social representations," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2014, pp. 701–710.

[20] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," in *Proc. 24th Int. Conf. World Wide Web*, 2015, pp. 1067–1077.

[21] T.-Y. Fu, W.-C. Lee, and Z. Lei, "HIN2Vec: Explore meta-paths in heterogeneous information networks for representation learning," in *Proc. ACM on Conf. Inf. Knowl. Manage.*, 2017, pp. 1797–1806.

[22] Y. Dong, N. V. Chawla, and A. Swami, "metapath2vec: Scalable representation learning for heterogeneous networks," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2017, pp. 135–144.

[23] D. Wang, P. Cui, and W. Zhu, "Structural deep network embedding," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 1225–1234.

[24] M. Ou, P. Cui, J. Pei, Z. Zhang, and W. Zhu, "Asymmetric transitivity preserving graph embedding," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 1105–1114.

[25] C. Tu, X. Zeng, H. Wang, Z. Zhang, Z. Liu, M. Sun, B. Zhang, and L. Lin, "A unified framework for community detection and network representation learning," 2016, *arXiv:1611.06645*. [Online]. Available: http://arxiv.org/abs/1611.06645

[26] C. Yang, Z. Liu, D. Zhao, M. Sun, and E. Y. Chang, "Network representation learning with rich text information," in *Proc. 24th Int. Joint Conf. Artif. Intell.*, 2015, pp. 2111–2117.

[27] C. Tu, W. Zhang, Z. Liu, and M. Sun, "Max-margin deepwalk: Discriminative learning of network representation," in *Proc. IJCAI*, 2016, pp. 3889–3895.

[28] C. Tu, H. Liu, Z. Liu, and M. Sun, "CANE: Context-aware network embedding for relation modeling," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2017, pp. 1722–1731.

[29] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2016, pp. 855–864.

[30] C. Tu, Z. Zhang, Z. Liu, and M. Sun, "TransNet: Translation-based network representation learning for social relation extraction," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017.

[31] J. Ren, J. Long, and Z. Xu, "Financial news recommendation based on graph embeddings," *Decis. Support Syst.*, vol. 125, Oct. 2019, Art. no. 113115.

[32] R. Yin, K. Li, G. Zhang, and J. Lu, "A deeper graph neural network for recommender systems," *Knowl.-Based Syst.*, vol. 185, Dec. 2019, Art. no. 105020.

[33] R. Ma, Q. Zhang, J. Wang, L. Cui, and X. Huang, "Mention recommendation for multimodal microblog with cross-attention memory network," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jun. 2018, pp. 195–204.

[34] T. Gui, P. Liu, Q. Zhang, L. Zhu, M. Peng, Y. Zhou, and X. Huang, "Mention recommendation in Twitter with cooperative multi-agent reinforcement learning," in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2019, pp. 535–544, doi: 10.1145/3331184.3331237.

[35] P. Bao, H.-W. Shen, J. Huang, and H. Chen, "Mention effect in information diffusion on a micro-blogging network," *PLoS ONE*, vol. 13, no. 3, 2018, Art. no. e0194192.

[36] F. Morin and Y. Bengio, "Hierarchical probabilistic neural network language model," in *Proc. Aistats*, vol. 5, 2005, pp. 246–252.

[37] J. Zhang, J. Tang, J. Li, Y. Liu, and C. Xing, "Who influenced you? Predicting retweet via social influence locality," *ACM Trans. Knowl. Discovery from Data*, vol. 9, no. 3, pp. 1–26, Apr. 2015.

[38] S. Ganguly, M. Gupta, V. Varma, and V. Pudi, "Author2Vec: Learning author representations by combining content and link information," in *Proc. 25th Int. Conf. Companion World Wide Web*, 2016, pp. 49–50.

[39] S. Pan, J. Wu, X. Zhu, C. Zhang, and Y. Wang, "Tri-party deep network representation," in *Proc. 25th Int. Joint Conf. Artif. Intell.*, 2016, pp. 1895–1901.

**FENG YI** received the M.S. and Ph.D. degrees from the University of Chinese Academy of Sciences, China. He is currently an Assistant Professor with the School of Computer Science, University of Electronic Science and Technology of China, Zhongshan Institute. His main research interests include data mining and machine learning.

**BO JIANG** received the Ph.D. degree in computer science from the University of Chinese Academy of Sciences, China, where he is currently an Associate Researcher with Institute of Information Engineering, Chinese Academy of Sciences, China. His main research interests include data mining, knowledge graph, social network mining, and recommendation system. He has served as a Reviewer for the IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS and Elsevier.

**JIANJUN WU** received the Ph.D. degree in computer science from the University of Chinese Academy of Sciences, China. He is currently an Associate Professor with the Computer Science College, Beijing College of Politics and Law. His main research interests include social network mining, data mining, and machine learning.

• • •