



FB2vec: A Novel Representation Learning Model for Forwarding Behaviors on Online Social Networks

Li Ma¹, Mingding Liao¹, Xiaofeng Gao^{1(✉)}, Guoze Zhang², Qiang Yan²,
and Guihai Chen¹

¹ Shanghai Key Laboratory of Data Science, Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China

{mali-cs,uracil_forever}@sjtu.edu.cn, {gao-xf,gchen}@cs.sjtu.edu.cn

² Tencent, Shenzhen, China

{givenzhang,rolanyan}@tencent.com

Abstract. Representation learning in online social networks has been an important research task for better service, which targets at learning the low-dimensional vector representation for nodes in a network. There exists a kind of social network, which not only includes the topological structure and node attributes, but other information, such as the user behaviors. It is necessary to use these behaviors to learn the node representations. In this paper, we propose FB2vec to analyze forwarding behaviors and achieve better node representations. Moreover, an information intensity function based on the utility function is proposed to measure the possibility of forwarding behaviors. However, the intensity function can not reflect exact possibility value and only provide a relative intensity order. Therefore, we sample the intensity order pairs from datasets and train the intensity function to adapt original orders by an attribute-reserved siamese network. Extensive experiments demonstrate the effectiveness of FB2vec and the visualization of information intensity function indicates the rationality of FB2vec.

Keywords: User representation learning · Online social networks · Forwarding behaviors · Siamese networks

1 Introduction

Online social networks play an essential role in daily life with the rapid development of the internet and information technology. Network representation learning, which is to learn the low-dimension vector representations for network structure and node information [20], has shown great potential on the application of

This work was supported by the National Key R&D Program of China [2018YFB1004700]; the National Natural Science Foundation of China [61872238, 61972254]; the Tencent Joint Research Program, and the Open Project Program of Shanghai Key Laboratory of Data Science (No. 2020090600001).

© Springer Nature Switzerland AG 2021

F. Hutter et al. (Eds.): ECML PKDD 2020, LNAI 12457, pp. 216–231, 2021.

https://doi.org/10.1007/978-3-030-67658-2_13

online social networks. Most of the existing works of network representation learning for online social networks concentrate on analyzing the relationship of the network. However, the sparsity of the friendship network limits these applications. For example, Rochester used in [20], which is a Facebook network dataset, contains 4,563 nodes but only 161,404 links, which ignores the possibility that two nodes may be similar, but there is no link between them. However, there is such a specific network with a typical user behavior over online social networks. For example, the users from the online social networks WeChat could forward some articles into a group, although some of members of the group are not his or her friends. Compared with friendship networks, the networks with forwarding behaviors could be more dense because some of online social networks allow users to look up strangers' recommendation contents. So representation learning for forwarding behaviors is a potential research direction. However, few works pay attention to that because a forwarding behavior is a triple relationship and contains the information about forwarding amount, which makes it hard to be processed in a traditional network embedding model. Thus, the research on representation learning for forwarding behaviors is significant for data mining on online social networks.

Information dissemination is a common phenomenon in most of online social networks and the forwarding behavior is a form of it. In the process of information dissemination over social networks, a part of users generate contents, and the contents would be received and forwarded by other users, and then received by more users. Thus, forwarding behaviors could be described by three participants: a content generator, a content receiver and a content forwarder. There could be no direct link between a generator and a receiver, but a generator and a receiver could be connected by a forwarder. So the introduction of forwarding behaviors could break the limitation of friendship networks. Representation learning based on information dissemination could weaken the impact of the sparsity of friendship networks.

To get a better representation, many related works not only utilize the topological structure of the network, but the attributes of nodes, which could be beneficial for representation learning since they may crucially impact the connections and interactions with nodes [20]. However, these attribute-aware network embedding method could not be applied in the networks with information dissemination because there is more information. Information dissemination is affected by the content of information and the reactions of users to information, which could be measured by content influence and social influence correspondingly. The lack of researches on the information dissemination indicates excellent research prospects.

In this paper, we provide a representation learning model based on **F**orwarding **B**ehaviors called FB2vec. We focus on forwarding behaviors in the information dissemination process on online social networks and explore node embedding vectors to describe the information of users. In this model, users are classified into generators, forwarders and receivers, and they are encoded through multi-modal auto-encoder. In order to describe the forwarding behav-

iors of users, we innovatively design an information intensity function to measure the intensity of forwarding behaviors, which takes the embedding vectors of generators, forwards and receivers as input and takes the intensity of forwarding behaviors among users as output, and we use the siamese network to train the function. With the help of information intensity function, we can further promote the learning of embedding vectors of users. The experiments indicate that FB2vec outperforms baselines on the behavior prediction problem and similar user detection problem.

The main contributions of this paper can be summarized in these points:

- 1) We provide a novel model for representation learning for forwarding behaviors on online social networks (FB2vec) to utilize the additional dissemination information to promote the quality of the embedding vectors;
- 2) We propose a new method information intensity function, which combines the idea of network embedding and metric learning, to build the connection between embedding vectors and forwarding behaviors intensity;
- 3) Experiments prove that our model works better than state-of-the-art models both for behavior prediction task and similar user detection task, and parameter study shows that our model could measure the receivers' interest on forwarders.

2 Related Work

Network embedding methods aim at learning the low-dimensional latent vector representation of nodes in a network, and these representations can be used as features for a wide range of tasks on graphs. Generally, network embedding models can be categorized into random walk model, edge-based model, and matrix factorization model.

Random walk models are exploited to capture structural relationships between nodes by the random walk. DeepWalk [11] is the pioneering work of random walk based network embedding, with the follow-up researches such as node2vec [4] and MMDW [17].

Matrix factorization methods represent the connections between network vertices in the form of a matrix and use matrix factorization to obtain the embedding vectors. As a successful example, APNE [5] incorporates structure, content and label information of the network simultaneously. [14] provides a unified framework for matrix factorization based model and analyzes DeepWalk, LINE, PTE and Node2vec as examples. [13] proposes the algorithm of large-scale network embedding as sparse matrix factorization.

Edge-based methods directly learn vertex representations from vertex-vertex connections to overcome the high time-consuming of random walk based models. LINE [16] first focus on first-order proximity and second-order proximity to learn the node representation, which achieves a much higher speed than node2vec. Graph Convolutional Networks indicates that edge-based models become a mature scheme [8] which reflects the influence between embedding vectors through links in networks. SDNE [19] provides an auto-encoder based

model to train the embedding vectors of two vertices of a link. TransNet [18] considers the embedding vectors of a link as the difference of embedding vectors of its vertices, which could be considered as an expansion of SDNE. DANE [2] designs a multi-modal autoencoder for joint training of structure network information and vertices attribute information, which is employed in our model. NECS [9] preserves the high-order proximity and incorporates the community structure in vertex representation learning. Especially, [1] showed the application potential of metric learning methods on network embedding, which inspires our model to employ metric learning methods to train the representation of forwarding behaviors.

3 Definition and Problem Statement

Information dissemination in the social networks is that a part of users generate contents, and the contents would be received and forwarded by other users, and then received by more users. So there are three views of users: generators, forwarders and receivers. More explanations of generators, forwarders, and receivers could be found in Sect. 5.1. Definition 1 elaborates the above phenomenon.

Definition 1 Information Dissemination. Let $G = (V, E)$ denotes the structure of a social network, where V is the set of users and $E \subset V \times V$ is the relationships between users. $D = \{(u^g, v^f, w^r, c)\}$ is the set of information dissemination behaviors, where (u^g, v^f, w^r, c) means that content c , generated by generator u^g , is transmitted by forwarder v^f and then influences receiver w^r .

The behaviors of receivers reflect the influence among users. In the view of the reason for forwarding, content influence and social influence are the main factors to promote users' forwarding behaviors. In other words, the receiver is affected by the generator, either because the receiver has interest on the content generated by the generator (content influence), or because the forwarders, the friends of the receiver (social influence) like the content. However, existing works of network embedding ignore the behaviors of users and fail to distinct content influence and social influence. Thus we define the representation learning for forwarding behaviors as Definition 2 to formulate the latent factors in these three views, inspired by traditional network embedding.

Definition 2 Representation Learning for Forwarding Behaviors. Let $P^g, P^f, P^r \in R^{|V| \times d}$ denote the generator embedding vector, forwarder embedding vector and receiver embedding vector correspondingly, where d is the length of embedding vector. Information intensity function $f(P_u^g, P_v^f, P_w^r)$ is defined to measure the comprehensive influence of the generator u and the forwarder v on the receiver w in the process of information dissemination and is used to the training of embedding vectors.

4 Behavior Embedding via Siamese Network

In this section, we propose FB2vec, a novel behavior embedding model based on metric learning and multi-modal autoencoder, which is illustrated in Fig. 1. Overall, there are two branches for a user where the first branch captures the topological features and the second one captures the profile features. These two kinds of features of generators, forwarders and receivers are mapped to the low-dimensional space by multi-modal autoencoder. Specially, we design a requirement aggregating function to integrate generator behavior embedding and forwarder behavior embedding, which is inspired by a utility function in economics. Then the information intensity function based on the requirement aggregating function could measure the information dissemination intensity with providing a considerable order instead of the exact value of the possibility of dissemination. Therefore, we sample the dissemination behavior into pairwise intensity order and employ an attribute-reserved siamese network to train the information intensity function and representations of users.

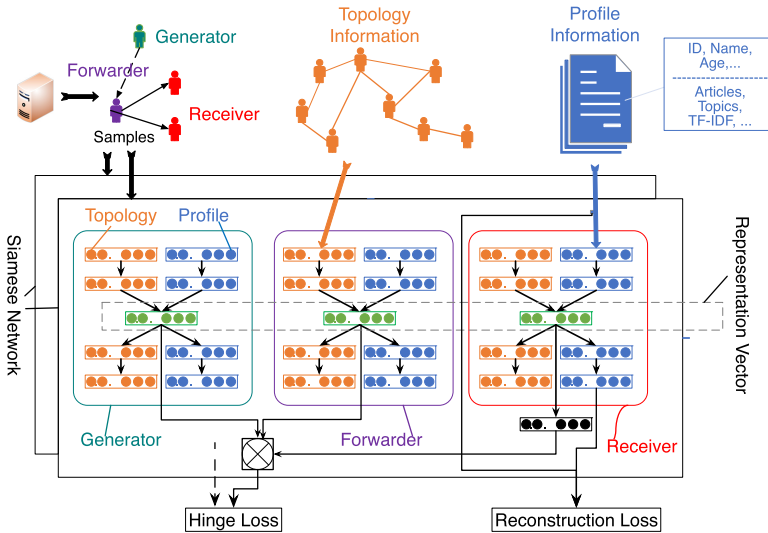


Fig. 1. Illustration of FB2vec

4.1 Attribute Preservation

The basic work for behavior embedding is to capture and preserve the topological structure of and attributes of users, which is solved by a multi-modal autoencoder. The two branches in Fig. 1 for each user (generator, forwarder or receiver) form a multi-modal autoencoder. All generators share the same autoencoder \mathcal{AE}^g parameters. Similarly, one autoencoder \mathcal{AE}^f is employed for all forwarders and

\mathcal{AE}^r for all receivers. Let x_u be the topological structure features of u and y_u be the profile features of u . Take \mathcal{AE}^g as an example. There are K layers in \mathcal{AE}^g in Eq. (1)–(3). Equation (1) represents the multi-layer neural network for an encoder.

$$\begin{aligned}
 h_{x,u,(1)}^g &= \sigma(W_{x,(1)}^g x_u + b_{x,(1)}^g), \\
 &\dots, \\
 h_{x,u,(K)}^g &= \sigma(W_{x,(K)}^g h_{x,u,(K-1)}^g + b_{x,(K)}^{(g)}), \\
 h_{y,u,(1)}^g &= \sigma(W_{y,(1)}^g y_u + b_{y,(1)}^g), \\
 &\dots, \\
 h_{y,u,(K)}^g &= \sigma(W_{y,(K)}^g h_{y,u,(K-1)}^g + b_{y,(K)}^{(g)}).
 \end{aligned} \tag{1}$$

Equation (2) maps topological structure features and attribute features of generator u into the embedding vectors.

$$P_u^g = \sigma(W^g [h_{x,u,(K)}^g, h_{y,u,(K)}^g] + b^g) \tag{2}$$

Equation (3) reconstructs features from embedding vector.

$$\begin{aligned}
 h_{x,u,(-K)}^g &= \sigma(W_{x,(-K)}^g P_u^g + b_{x,(-K)}^g), \\
 &\dots, \\
 h_{x,u,(-1)}^g &= \sigma(W_{x,(-1)}^g h_{x,u,(-2)}^g + b_{x,(-1)}^g), \\
 h_{y,u,(-K)}^g &= \sigma(W_{y,(-K)}^g P_u^g + b_{y,(-K)}^g), \\
 &\dots, \\
 h_{y,u,(-1)}^g &= \sigma(W_{y,(-1)}^g h_{y,u,(-2)}^g + b_{y,(-1)}^g)
 \end{aligned} \tag{3}$$

Here, $\theta_g = \{W_{x,(k)}^g, W_{y,(k)}^g, b_{x,(k)}^g, b_{y,(k)}^g\}$ are the model parameters. $\sigma(\cdot)$ denotes the non-linear activation function. The model parameters can be learned by minimizing the reconstruct error as Eq. (4), where \otimes is the Hadamard product. Considering the possible sparsity of features, the number of non-zero elements in x or y may be far less than the number of zero elements. Thus, we impose a penalty function ϕ to make the autoencoder concentrate on the non-zero elements. When the features are sparse, $\phi(x) = |x|$, otherwise, $\phi(x) \equiv 1$.

$$\mathcal{L}^g = \sum_{u \in V} \|\hat{x}_u^g - x_u^g\| \otimes \phi(x_u^g) + \|\hat{y}_u^g - y_u^g\| \otimes \phi(y_u^g) \tag{4}$$

Similarly, model parameters θ_f for the forwarder autoencoder and model parameters θ_r for the receiver autoencoder can be learned by minimizing the reconstruct error \mathcal{L}^f and \mathcal{L}^r , where x_v, y_v is the topological structure features and the profile features of forwarder v , x_w, y_w is the topological structure features and the profile features of receiver w :

$$\mathcal{L}^f = \sum_{v \in V} \|\hat{x}_v^f - x_v^f\| \otimes \phi(x_v^f) + \|\hat{y}_v^f - y_v^f\| \otimes \phi(y_v^f) \tag{5}$$

$$\mathcal{L}^r = \sum_{w \in V} \|(\hat{x}_w^r - x_w^r) \otimes \phi(x_w^r)\| + \|(\hat{y}_w^r - y_w^r) \otimes \phi(y_w^r)\| \quad (6)$$

Generally, FB2vec could make use of other kinds of neural networks such as convolutional neural networks for image features and recurrent neural networks for text features easily in the multi-modal autoencoder, which makes FB2vec become a general framework for any social networks.

4.2 Information Dissemination Intensity

Information dissemination intensity is expected to evaluate the intensity of influence on the receiver w under the joint action of the generator u and the forwarder v . Obviously, if the influence is higher, the higher the frequency of forwarding behavior (u, v, w, c) is. So we design a function, called information intensity function, that takes the embedding vectors of generators, forwarders and receivers as input, and outputs the information dissemination intensity. Then FB2vec trains the output of information dissemination intensity to match the real data, which in turn can promote the optimization of embedding vectors of generators, forwarders and receivers. In other words, the design of information intensity function is a fake task, which aims to help the training of users' representation.

Since the reactions of receivers are dependent on the source of information P_u^g (content requirement) and the reactions of their friends P_v^f (social requirement) in information dissemination, the information intensity function $f(P_u^g; P_v^f; P_w^r)$ should integrate the hidden influence factors of generator u and forwarder v into the context factors and calculate them with the hidden factors of receiver w . Assuming that the receiver w pursues the maximization of content requirement and social requirement as a rational individual, so we design the requirement aggregating function as Eq. (7) inspired by a kind of utility function [3], where P_1 refers to P_u^g , P_2 refers to P_v^f , β is the utility parameter and \otimes is the Hadamard product. With the help of requirement aggregating function, the information dissemination intensity in the triple relationship could be measured. The information intensity function f is defined as Eq. (8), where $A \in R^{d \times d}$ is the embedding combination matrix. We have known that the reactions of users are dependent on the source of information P_u^g and the reactions of their friends P_v^f . We intuitively know that the distance of P_u^g and P_w^r reflects the intimacy between generators and receivers and the distance of P_v^f and P_w^r reflects the intimacy between forwarders and receivers. The choice of the utility function is reasonable because the reactions of a receiver depends on the two options (content and social requirement) with best utility.

$$g(P_1, P_2, \beta) = P_1^\beta \otimes P_2^{1-\beta} \quad (7)$$

$$f(P_u^g, P_v^f, P_w^r) = QAQ^T \text{ where } Q = g(P_u^g, P_v^f, \beta(P_w^r)) - P_w^r \quad (8)$$

$$f(P_u^g, P_v^f, P_w^r) = \|g(P_u^g, P_v^f, \beta(P_w^r))B - P_w^r B\|^2 \quad (9)$$

In further discussion, information intensity function is equivalent to the Euclidean distance of linear transform of $g(P_u^g, P_v^f)$ and P_w^r , because Eq. (8)

can be transformed into Eq. (9) when A is semi-positive and $A = BB^T$. The platform-aware parameter A (as well as B) could be trained on different social networks, which indicates the application potential of FB2vec on different social networks with information dissemination and guarantees the generality of FB2vec. For a better description of users' patterns, β is replaced as the user-aware parameter $\beta(P_w^r)$ in $[0, 1]$, which is defined as the output of a multi-layer neural network with the input of the corresponding receiver behavior embedding vector P_w^r .

However, the utility function used in the Eq. (7) could only represent the order of preference factors instead of the exact value of latent factors [3], so the information intensity function in FB2vec is not suitable to measure the exact value, which makes it challenging to employ the ordinary network embedding model for parameter training. To solve the training problem, we explore the pairwise intensity order through a siamese network.

4.3 Pairwise Intensity Order

In the process of information dissemination, the users' behaviors are described as (u^g, v^f, w^v, c) in Definition 2. Without loss of generality, the behaviors of the same u^g, v^f and w^v are aggregated into $l(e_{u,v,w})$ where $e_{u,v,w}$ is short for the behavior (u^g, v^f, w^v, c) and $l(\cdot)$ is the number of $e_{u,v,w}$.

The information intensity function may not accurately describe the specific quantity of forwarding behaviors, but only the relative size relationship of quantity as it depends on the utility function. Therefore, we train the relationship between $l(e_{u,v,w})$ and the number of actual forwarding behaviors. That means the multi-modal autoencoder of generators, forwarders and receivers is used twice in each training, which forms a siamese network. FB2vec inputs two behaviors at the same time, one is the data extracted from the dataset, and the other is the negative sampling data for the same data, which is similar to the negative sampling in network embedding.

We randomly sample the generators, forwarders and receivers for each item $e_{u,v,w}$ to generate the confident negative samples. The sampling procedure is summarized as Algorithm 1. In detail, there are four categories of negative samples: negative samples nw^r for the receiver w^r , negative samples nv^f for the forwarder v^f , negative samples nu^g for the generator u^g and the forwarder v^f when $u^g = v^f$, and negative samples nu^g for the generator u^g when $u^g \neq v^f$. For example, the negative receiver sample nw^r means that the intensity from the generator u^g and the forwarder v^f to the receiver w^r is much larger than intensity from the u^g and v^f to nw^r . The negative samples for the generator u^g should be categorized into two classes by whether the forwarder is also the generator. Negative samples have to maintain consistency if the forwarder is also the generator.

Algorithm 1: Sampling intensity order relationship

Input: Graph G , Dissemination Behaviors D
Output: Sample Result $S[\cdot]$ for each $e_{u,v,w}$

- 1 Aggregate (u^g, v^f, w^r, c) into $l(e_{u,v,w})$;
- 2 **foreach** $e_{u,v,w}$ **do**
- 3 Randomly sample several nw^r 's where $l(u^g, v^f, nw^r) \ll l(e_{u,v,w})$;
- 4 $S[e_{u,v,w}].add((u^g, v^f, nw^r))$;
- 5 Randomly sample several nv^f 's where $l(u^g, nv^f, w^r) \ll l(e_{u,v,w})$;
- 6 $S[e_{u,v,w}].add((u^g, nv^f, w^r))$;
- 7 **if** $u^g = v^f$ **then**
- 8 Random sample several nu^g 's where $l(nu^g, nu^g, w^r) \ll l(e_{u,v,w})$;
- 9 $S[e_{u,v,w}].add((nu^g, nu^g, w^r))$;
- 10 **else**
- 11 Random sample several nu^g 's where $l(nu^g, v^f, w^r) \ll l(e_{u,v,w})$;
- 12 $S[e_{u,v,w}].add((nu^g, v^f, w^r))$;
- 13 **return** S ;

4.4 Model Optimization

After the sampling procedure, there are several negative samples for each $e_{u,v,w} = (u^g, v^f, w^r, c)$. We employ the large margin strategy to distinguish each $e_{u,v,w}$ with their negative samples [21]. Large margin strategy only focuses on the relative distance between positive and negative items, which perfectly meets the requirement of FB2vec with the utility function. Therefore, the aim is to minimize Eq. (10), where m is the margin. The object function enforces each $f(e)$ to become smaller than $f(ne) - m$.

$$\mathcal{L}^{infl} = \sum_e \sum_{ne \in S[e]} \min(f(e) - f(ne) + m, 0) \quad (10)$$

Based on the mathematical information intensity function and pairwise intensity order, we propose a novel representation learning model FB2vec for forwarding behaviors with preservation of the topological and profile features on online social networks. The whole object function of FB2vec can be written as Eq. (11) where L^{reg} is the L_2 regularizer for all parameters, α and γ are the weighting factors. The object function is minimized by the root mean square propagation gradient method [7] to optimize the representation learning model.

$$\mathcal{L} = \mathcal{L}^{infl} + \alpha(\mathcal{L}^g + \mathcal{L}^f + \mathcal{L}^r) + \gamma L^{reg} \quad (11)$$

5 Experiments

To comprehensively evaluate the proposed FB2vec, we conduct experiments to show the efficiency of behavior embedding vectors in aspect of the behavior prediction problem and the similar user detection problem.

5.1 Experiment Setup

Dataset Description. We apply our model to a commercial dataset *WeChat Article* and a public social network retweets datasets *Sina Weibo* [22]. The details of datasets are the following:

- *WeChat Article*: This dataset is collected from Tencent WeChat, which is composed of the articles published by Official Accounts, the forwarding and reading behaviors of general users accounts. Official Accounts are special accounts for publishing contents and building dynamic services. In WeChat, only Official Accounts could be generators. A forwarder could forward the articles into WeChat groups or “Moments” to recommend them to its friends. These friends would see the brief introduction of articles and view the whole article if interested. If friends read the whole article, they would be treated as receivers.
- *Sina Weibo*: This dataset crawled from *Sina Weibo*, which is the largest online social network in China, contains about 177 thousand users and 23 million retweet behavior [22]. In *Sina Weibo*, generators, forwarders and receivers are all general users. When a user posts a tweet, it becomes a generator, and then other users would forward it as forwarders. If a user forwards the retweets from forwarders or the tweets from generators, it would be treated as a receiver.

We clean the data and extract dense forwarding subgraphs which are sampled through random walk methods for convenient model comparison. Through preprocessing, *WeChat Article* contains 90 thousand users, 40 thousand Official Accounts and 1.54 million forwarding behaviors. Weibo contains 250 thousand users and 422 thousand forwarding behaviors. We split 70% of forwarding behaviors as the training set.

Implementation Details. We use Tensorflow to implement the program. The profile features used in Weibo dataset is the verified status, gender, and tweet counts of users. The profile features used in *WeChat Article* are the one-hot vector of related article classes. For example, the profile features of a forwarder are the classes of articles it forwards. The topological features are 128 embedding vectors generated by DeepWalk [11].

The parameters of the model are set by preliminary experiments. The weighting factor α and γ are set to guarantee the same order of magnitude of the different components object function. For *Sina Weibo* dataset, we set $\alpha = 3 \times 10^{-2}$, $\gamma = 10^{-5}$, $d = 64$. For *WeChat Article* dataset, we set $\alpha = 3 \times 10^{-5}$, $\gamma = 10^{-5}$, $d = 64$. The negative sample number for each behavior is 24 (6 samples for each sample type) on two datasets. The neural network used in multi-modal autoencoder and used for β are all 3 layers, the node number of the hidden layer is 50 for network features, 25 for profile features and 10 for β . The training batch number is 2560. The maximum number of iteration step is set as 40.

Evaluation Protocol. We deploy FB2vec and baselines to *Behavior Prediction Problem* and *Similarity User Detection Problem*, and choose several comparative methods, including some significant or state-of-the-art models for network embedding and forwarding prediction, to evaluate the performance of FB2vec. Four standard metrics Precision, Recall, F1 score and AUC (Area Under Curve) [12] are employed to measure the performance.

Behavior Prediction Problem is to predict whether a user w is a receiver in the context of the generator u and the forwarder v . In *Sina Weibo* dataset, a user is a receiver if it forwards a tweet generated by u and forwarded by v . In *WeChat Article* dataset, a user is a receiver if it reads the whole article generated by u and forwarded by v . The problem is similar to the famous forwarding prediction problem [10]. We select 50 thousand triples (u, v, w) from the testing dataset as positive forwarding item and sample randomly 50 thousand triples (u, v, w) which does not appear in datasets as negative forwarding item. FB2vec solves the *Behavior Prediction Problem* by calculating the information intensity $f(\cdot)$ for each triple (u, v, w) . If the information intensity is smaller than a threshold, w is a receiver in the context of the generator u and the forwarder v . The threshold is set as the median number of all information intensity in test data.

Similarity User Detection Problem is required to judge whether the given two users are similar or not, which is a common problem solved by network embedding methods. In our experiments, the similarity is defined by the ratio of common receivers as Eq. (12), where $D(u)$ means the set of all receivers of the generator u . We select 50 thousand pairs of generators by uniform sampling and select 50 thousand pairs of generators with at least one common receivers. Then generator pairs with the top 50% similarity are labeled as similar user pairs and the other 50% pairs are labeled as unsimilar user pairs. FB2vec solves *Similarity User Detection Problem* by computing the distance of generator embedding vector as Eq. (13). If the distance is smaller than a threshold, the pair of generators is predicted as a similar user pair. The threshold is set as the median number of all distances in test data. The threshold used in network embedding baseline models is set in the same way.

$$Sim(u_1, u_2) = \frac{2|D(u_1) \cap D(u_2)|}{|D(u_1)| + |D(u_2)|} \quad (12)$$

$$Dist(u_1, u_2) = -\|P_{u_1}^g B - P_{u_2}^g B\|^2 \quad (13)$$

The baseline models of the two tasks are DeepWalk, LINE, DANE and SDNE. The parameters for these four compared models are set as either the default settings suggested by the authors or tuned to find the best settings. Specially, we set all embedding vectors length as 128. Since network embedding model can not handle the triple relationship (u, v, w) , we mask the generator u in training and testing datasets and apply network embedding models on the network connected between the receivers and their corresponding forwarders. Then network embedding models predict the links between forwarders and receivers.

We add another two baseline models, PMF [15] and OCCF [6], for behavior prediction. PMF is the basic one and OCCF is the start-of-the-art one consider-

ing node attributes. Both two models are matrix factorization model since other kinds of models are hard to implement because of the lack of available codes and the difficulty of feature requirements for most of the feature-based models. These models are to predict whether a message would be forwarded. We mask the forwarders in datasets and apply them to predict whether the contents of generators would be accepted by the receiver w .

5.2 Experiment Result

Behavior Prediction. We organize experimental results of FB2vec and other comparative methods for behavior prediction in Table 1. In the comparison of baselines, FB2vec outperforms on two datasets. In the *WeChat Article* dataset, FB2vec increases F1-score by from 3.9% to 33.5% and increases AUC by from 6.5% to 16.9%. In the *Sina Weibo* dataset, FB2vec increases F1-score by from 0.2% to 28.0% and increases AUC from 4.4% to 23.6%. The performances of all models on the *Sina Weibo* dataset is much worse than that on the *WeChat Article* because the network of the *WeChat Article* is much denser.

Table 1. Comparison results: behavior prediction

Model	WeChat				Weibo			
	Precision	Recall	F1-score	AUC	Precision	Recall	F1-score	AUC
DeepWalk	0.791	0.772	0.781	0.861	0.551	0.577	0.562	0.627
LINE	0.701	0.714	0.707	0.847	0.552	0.505	0.528	0.593
SDNE	0.882	0.857	0.869	0.927	0.602	0.622	0.611	0.668
DANE	0.911	0.905	0.908	0.907	0.644	0.621	0.638	0.700
PMF	0.729	0.761	0.745	0.845	0.562	0.597	0.578	0.655
OCCF	0.897	0.916	0.907	0.921	0.658	0.691	0.674	0.702
FB2vec	0.940	0.946	0.944	0.988	0.676	0.676	0.676	0.733

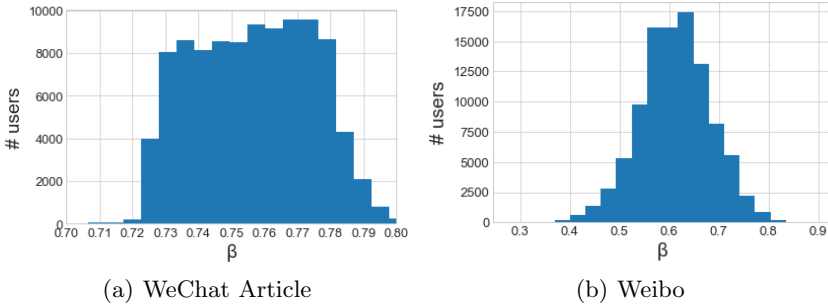
Similar User Detection. The experimental results of FB2vec and other comparative methods for similar user detection are shown in Table 2. It shows that FB2vec outperforms on two datasets. In the *WeChat* dataset, FB2vec increases F1-score by from 4.1% to 41.4% and increases AUC by from 1.9% to 31.5%. In the *Sina Weibo* dataset, FB2vec increases F1-score by from 0.4% to 31.1% and increases AUC by from 1.1% to 8.5%.

Parameter Study. β is the utility parameter to represent the weight of generator factors (content influence) and forwarder factors (social influence). Users are more inclined to content influence when β approaches 0, while users are more inclined to social influence when β approaches 1. Figure 2 shows the number of

Table 2. Comparison results: similar user detection

Model	WeChat				Weibo			
	Precision	Recall	F1-score	AUC	Precision	Recall	F1-score	AUC
DeepWalk	0.653	0.667	0.659	0.738	0.600	0.603	0.601	0.703
LINE	0.582	0.599	0.591	0.700	0.523	0.531	0.527	0.701
DANE	0.791	0.815	0.803	0.903	0.684	0.692	0.688	0.752
SDNE	0.768	0.752	0.760	0.882	0.661	0.653	0.657	0.701
FB2vec	0.831	0.840	0.836	0.921	0.701	0.680	0.691	0.761

users with the distribution of β . In *Sina Weibo*, most of β ranges from 0.4 to 0.8 with a normal distribution. However, most of β in *WeChat Article* dataset ranges from 0.72 to 0.79, which is approximate to uniform distribution. These results indicate that content is more important than social influence in both two datasets. Moreover, the pattern of user behavior in WeChat is more similar than that in *Sina Weibo*.

**Fig. 2.** Distribution of β

Here, we visualize the information dissemination intensity index by three Official Accounts on *WeChat Article* dataset in Fig. 3, 4 and 5 as examples. The reason to visualize it on WeChat datasets instead of Weibo datasets is that Official Account product information specially and professionally and they tend to attract a specific group. We call these three Official Accounts as NewsAccount, MilitaryAccount, and EntertainAccount¹, because they are the famous Official Accounts with more than one million followers in the news section, military section and entertainment section correspondingly. The NewsAccount is the official accounts operated by one of the most famous Chinese newspaper, focusing on news of Chinese society and culture. MilitaryAccount is operated by the biggest official Chinese military newspaper. EntertainAccount is one of

¹ Their names are hidden because of the requirement of Tencent Company.

the biggest Official Accounts for anecdotes of entertainer, which are popular among women. Figure 3, 4 and 5 show their information intensity distribution. The x-axis shows age and the y-axis shows information intensity. Note that the lower information intensity represents higher attraction. The dots in the lower left quarter in Fig. 4(a) means that EntertainAccount attracts women and the younger more. Similarly, MilitaryAccount attracts men and the old more indicated by Fig. 5(b), and the preference of people with different ages and genders to NewsAccount is similar. The above phenomenon is matched with prior knowledge of these Official Accounts. Therefore, FB2vec satisfies the expectation of information dissemination intensity measure.

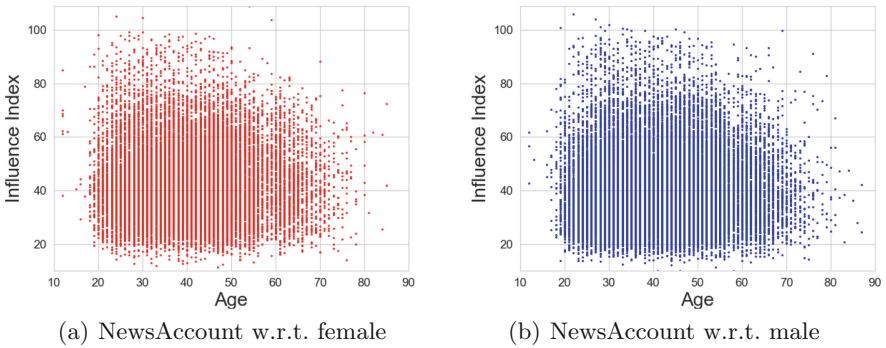


Fig. 3. Information index distribution of NewsAccount

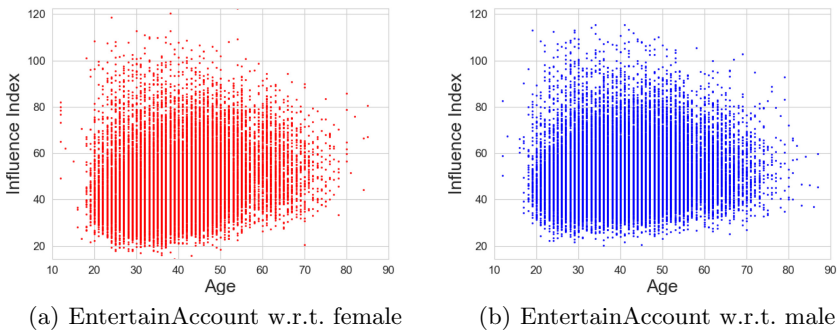


Fig. 4. Information index distribution of EntertainAccount

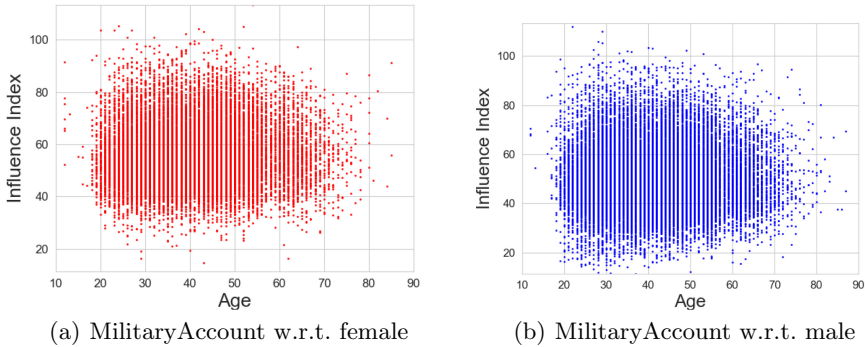


Fig. 5. Information index distribution of MilitaryAccount

6 Conclusion

In this paper, we aim to establish a general representation learning model for forwarding behaviors on online social networks. We propose a novel concept of behavior embedding, which provides a new research idea of user representation analysis. Then we design a general framework FB2vec to generate behavior embedding vectors, which combines the ideas of network embedding and metric learning. Especially, FB2vec defines the information dissemination intensity function with the help of economic utility function and we train FB2vec by adapting the intensity order with real data. The experimental results on two datasets *Sina Weibo* and *WeChat Article* show that FB2vec outperforms baselines for different tasks. The parameter study proves that the FB2vec coincides with the prior knowledge of representation of behaviors.

References

1. Chen, H., Yin, H., Wang, W., Wang, H., Nguyen, Q.V.H., Li, X.: PME: projected metric embedding on heterogeneous networks for link prediction. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), pp. 1177–1186 (2018)
2. Gao, H., Huang, H.: Deep attributed network embedding. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI), pp. 3364–3370 (2018)
3. Gerard, D.: Representation of a preference ordering by a numerical function. *Decis. Process.* **3**(1), 159–165 (1954)
4. Grover, A., Leskovec, J.: node2vec: scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), pp. 855–864 (2016)
5. Guo, J., Xu, L., Huang, X., Chen, E.: Enhancing network embedding with auxiliary information: an explicit matrix factorization perspective. In: Pei, J., Manolopoulos, Y., Sadiq, S., Li, J. (eds.) DASFAA 2018. LNCS, vol. 10827, pp. 3–19. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-91452-7_1

6. Jiang, B., et al.: Retweeting behavior prediction based on one-class collaborative filtering in social networks. In: Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR), pp. 977–980 (2016)
7. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. In: 3rd International Conference on Learning Representations (ICLR), pp. 1–15 (2015)
8. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: 5th International Conference on Learning Representations (ICLR), pp. 1–14 (2017)
9. Li, Y., Wang, Y., Zhang, T., Zhang, J., Chang, Y.: Learning network embedding with community structural information. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI), pp. 2937–2943 (2019)
10. Liu, Y., Zhao, J., Xiao, Y.: C-RBFNN: a user retweet behavior prediction method for hotspot topics based on improved RBF neural network. *Neurocomputing* **275**, 733–746 (2018)
11. Perozzi, B., Al-Rfou, R., Skiena, S.: DeepWalk: online learning of social representations. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), pp. 701–710 (2014)
12. Powers, D.M.: Evaluation: from precision, recall and f-measure to ROC, informedness, markedness and correlation. *J. Mach. Learn. Technol.* **2**(1), 37–63 (2011)
13. Qiu, J., et al.: NetSMF: large-scale network embedding as sparse matrix factorization. In: The World Wide Web Conference (WWW), pp. 1509–1520 (2019)
14. Qiu, J., Dong, Y., Ma, H., Li, J., Wang, K., Tang, J.: Network embedding as matrix factorization: unifying DeepWalk, LINE, PTE, and Node2vec. In: Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining (WSDM), pp. 459–467 (2018)
15. Salakhutdinov, R., Mnih, A.: Probabilistic matrix factorization. In: Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems (NeurIPS), pp. 1257–1264 (2007)
16. Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., Mei, Q.: LINE: large-scale information network embedding. In: Proceedings of the 24th International Conference on World Wide Web (WWW), pp. 1067–1077 (2015)
17. Tu, C., Zhang, W., Liu, Z., Sun, M.: Max-margin DeepWalk: discriminative learning of network representation. In: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI), pp. 3889–3895 (2016)
18. Tu, C., Zhang, Z., Liu, Z., Sun, M.: TransNet: translation-based network representation learning for social relation extraction. In: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI), pp. 2864–2870 (2017)
19. Wang, D., Cui, P., Zhu, W.: Structural deep network embedding. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), pp. 1225–1234 (2016)
20. Wang, H., et al.: A united approach to learning sparse attributed network embedding. In: IEEE International Conference on Data Mining (ICDM), pp. 557–566 (2018)
21. Weinberger, K.Q., Saul, L.K.: Distance metric learning for large margin nearest neighbor classification. *J. Mach. Learn. Res.* **10**, 207–244 (2009)
22. Zhang, J., Tang, J., Li, J., Liu, Y., Xing, C.: Who influenced you? Predicting retweet via social influence locality. *ACM Trans. Knowl. Discov. Data (TKDD)* **9**(3), 25:1–25:26 (2015)