MINE: A Method of Multi-Interaction Heterogeneous Information Network Embedding

**Dongjie Zhu[1], Yundong Sun[1], Xiaofang Li[2], Haiwen Du[3],
Rongning Qu[2], Pingping Yu[4, *], Xuefeng Piao[1], Russell Higgs[5] and Ning Cao[6]**

Abstract: Interactivity is the most significant feature of network data, especially in social networks. Existing network embedding methods have achieved remarkable results in learning network structure and node attributes, but do not pay attention to the multi-interaction between nodes, which limits the extraction and mining of potential deep interactions between nodes. To tackle the problem, we propose a method called Multi-Interaction heterogeneous information Network Embedding (MINE). Firstly, we introduced the multi-interactions heterogeneous information network and extracted complex heterogeneous relation sequences by the multi-interaction extraction algorithm. Secondly, we use a well-designed multi-relationship network fusion model based on the attention mechanism to fuse multiple interactional relationships. Finally, applying a multitasking model makes the learned vector contain richer semantic relationships. A large number of practical experiments prove that our proposed method outperforms existing methods on multiple data sets.

Keywords: Network embedding, network representation learning, interactive network, data mining.

1 Introduction

The development of mobile Internet has promoted the rise and prosperity of online social media. By 2018, Facebook's monthly active users have exceeded 2 billion [Hu, Li, Yang et al. (2019)], and Sina Weibo's monthly active users have reached 462 million [Cui,

---

[1] School of Computer Science and Technology, Harbin Institute of Technology, Weihai, 264209, China.

[2] Department of Mathematics, Harbin Institute of Technology, Weihai, 264209, China.

[3] School of Astronautics, Harbin Institute of Technology, Harbin, 150001, China.

[4] School of Information Science and Engineering, Hebei University of Science and Technology, Shijiazhuang, 050018, China.

[5] School of Mathematics and Statistics, University College Dublin, Dublin, D04 V1W8, Ireland.

[6] School of Internet of Things and Software Technology, Wuxi Vocational College of Science and Technology, Wuxi, 214028, China.

[*] Corresponding Author: Pingping Yu. Email: yppflx@hotmail.com.

Wang, Pei et al. (2018)]. A large amount of interactive data is continuously generated on social networks. This interactive data embodies the dynamic relationship between people and is a true portrayal of people's daily social life. To effectively mine and utilize this data is of great significance for public opinion analysis, recommendation system, community detection [Hu, Li, Yang et al. (2019); Xu, Wang, Wu et al. (2020)].

Traditional graph representation methods have the disadvantages of high algorithm complexity, difficulty in parallel processing and applying to current machine learning and deep learning models, which will no longer be applicable to large-scale information networks. Therefore, how to find an efficient network representation method, transforming the traditional network structure representation into a low-dimensional vector carrying potential information is an urgent problem to be solved [Wang, Cui and Zhu (2016)]. In order to solve the problems of traditional network representation, many experts have been committed to the new field of network embedding, which is to convert the traditional network structure representation into a low-dimensional node vector carrying potential information.

Initially, some classic methods, including the Laplacian feature map [Belkin and Niyogi (2002)], IsoMap [Tenenbaum, De Silva and Langford (2000)] were used to reduce the dimensionality of the graph structure data. This kind of methods first needs to construct the relation graph based on the extracted features using K-Nearst or other methods, and then transforms the relation graph into the low-dimensional vector space [Tang, Qu, Wang et al. (2015)]. Due to a large number of iterations and combination operations are required to construct the relationship graph, such methods are time-consuming and cannot be applied to large-scale network embedding. To deal with large-scale networks, Ahmed et al. [Ahmed, Shervashidze, Narayanamurthy et al. (2013)] proposed a novel decomposition technique to enable its dimensionality reduction processing to run on a decentralized system. However, this method was not originally designed for network embedding. Therefore, the resulting low-dimensional vector does not preserve the semantic relationship between network structure and nodes [Bourigault, Lagnier, Lamprier et al. (2014)]. Inspired by the natural language processing model Word2Vector Mikolov et al. [Mikolov, Chen, Corrado et al. (2013)], DeepWalk [Perozzi, Al-Rfou and Skiena (2014)] treats each node as a word, uses the Random Walk strategy to generate a series of node sequences for each node, then treats these "sequences" as input of the Word2Vector to get a low-dimensional vector representation of each node. Later, some improved methods based on DeepWalk were proposed. Node2Vec Grover et al. [Grover and Leskovec (2016)] improved the node sequence generation strategy, which is no longer a completely random strategy. It introduces the ideas of BFS and DFS and effectively fuses them to make the resulting sequence, which can better preserve the local and global information of the network structure. LINE Tang et al. [Tang, Qu, Wang et al. (2015)] introduces the concept of first-order and second-order proximity, which makes the model more comprehensive in the acquisition of structural information, and the method of negative sampling it introduced can accelerate the learning speed.

With the development of BP Neural Networks Zhao et al. [Zhao, Zhang, Shi et al. (2019)], some network embedding methods based on deep learning have been proposed. Chang et al. [Chang, Han, Tang et al. (2015)] introduces heterogeneous information into

the network, including texts, images and makes full use of the feature extraction ability of deep learning to achieve state-of-the-art performance.

The existing network embedding methods have achieved great success in the fields of node classification, link prediction, recommendation system, community discovery. But there are still the following shortcomings:

Most focus on isomorphic interactions. Different relationships are not treated differently. However, the relationship between network nodes in reality is intricate. The simple static network relationship cannot fully reflect the rich network relationship between nodes, and the complex dynamic interaction between nodes will be the key to mine the potential relationship of the nodes. Take the Twitter network as an example. Users have relationships of following, tweet forwarding, tweet commenting, and liking. Different relationships represent different semantic relationships and ignoring the differences between these relationships will lose a lot of potentially important information, as shown in Fig. 1.

There is no distinction between the importance of different interactions and nodes. Not all relationships should be valued, and some relationships may have a negative effect on node representation learning. At the same time, the importance that can be reflected by positive interactions is different [Chen, Wang, Wang et al. (2019)]. Indiscriminate consideration of all node relationships will introduce some noise, affecting the effect of network embedding.

Learn and optimize network representations with a single task. Most existing network embedding methods only learn and optimize for a certain task or target, such as node attribute similarity and node label classification. These methods may perform better after being optimized for their respective tasks [Zhu, Wang, You et al. (2019)]. However, the rich relationship between nodes cannot be fully explored, so that the learned network node representation is not well suited for other downstream applications.

To tackle the problem, we propose a method of multi-interaction heterogeneous information network embedding (MINE), which can effectively capture different interactions between different nodes and effectively combine them to improve the effect of network embedding representation learning. The intuition behind MINE is that different interactions represent different semantic relationships in low-dimensional vector space, and cross-pattern interactions can mine deep potential semantic relationships (see Fig. 1). More importantly, the different interactions between nodes play different roles in the final network embedding representation.
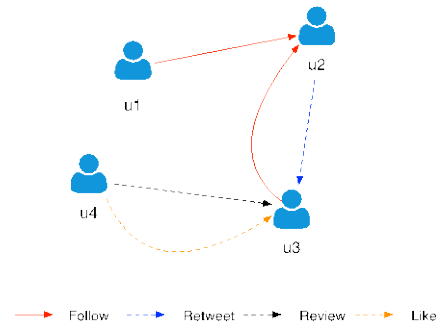
Figure 1: Different interactions between users, u_1 follows u_2, u_3 forwards u_2's tweets, u_4 review on the forwarded tweets by u_3, although u_1 and u_4 have no direct relationship, but u_1 can also finds u_4 cross multiple interactions. So there is a potential interaction between u_1 and u_4

Based on the above intuition, we first separate the multi-interaction network into a network of various single interaction patterns according to the characteristic interaction relationship model and learn the respective node representations separately. Next, we fuse the representations of the various interactive network nodes by different weights. Inspired by the Attention mechanism, we propose an attention strategy for calculating the weights of different nodes in different interaction patterns. Finally, we use multi-tasking to optimize representation learning.

To the best of our knowledge, this article is the first network embedding method to fuse multi-modal interactive networks. Our contributions can be summarized as follows:

1) We introduce multi-interactive heterogeneous information network (MIHIN) representation method and extract complex heterogeneous relations sequences by multi-interactive extraction algorithms.

2) We carefully designed a multi-relational network fusion model based on attention mechanism to fuse multiple relationships.

3) We propose a multi-tasking model network embedding model, making the learned vector cover richer semantic relationships.

4) Through a lot of practical experiments, we have proved that our proposed method has superiority in interactive prediction.
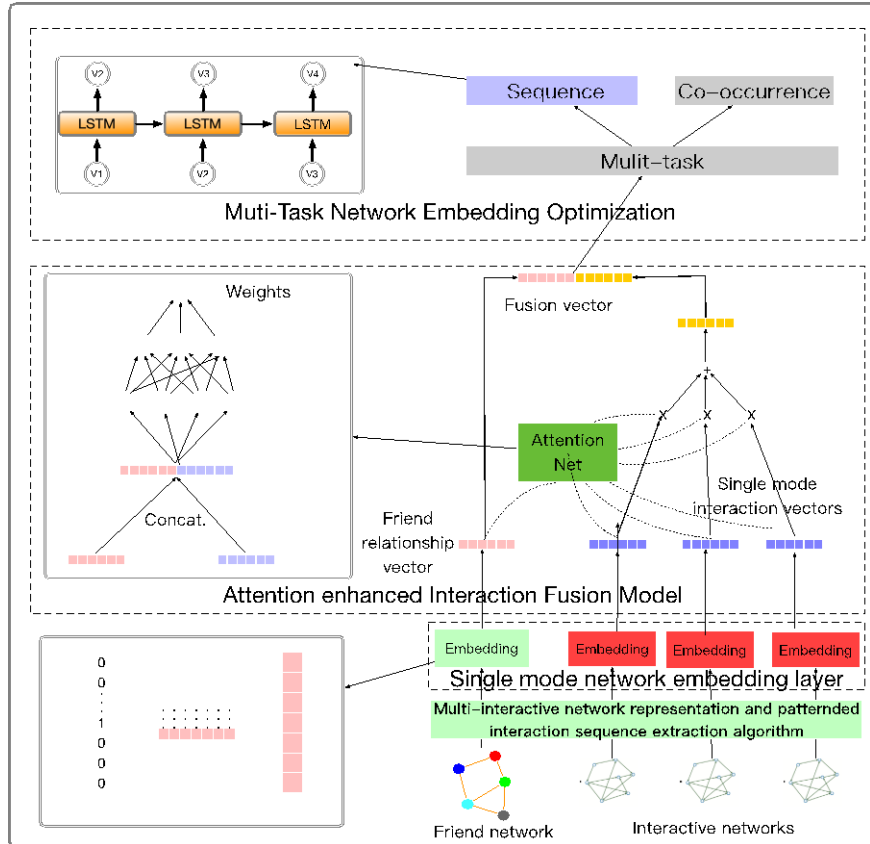
2 The design of MINE

Figure 2: The overall framework of MINE

The overall framework of MINE is shown in Fig. 2. From bottom to top, there are multi-interactive network representation and interaction mode extraction algorithm, single mode network embedding layer, attention-enhanced interaction fusion model and multi-task network embedding optimization layer.

The input of the entire model is the sub-network edgelist of various interaction patterns; the multi-interactive network presentation method is responsible for combining the sub-networks of the various modes as a hybrid interactive network and the sequences of nodes in a specific pattern are extracted by the multi-interaction sequence extraction algorithm (in Section 2.1); the single mode network embedding layer is responsible for mapping the node's one-hot coding to a low dimensional vector. To better explore the interaction between users in different patterns and the influence of different patterns in social networks, the multi- network embedding layer based on Attention mechanism is used to fuse multi-mode network vector representations (in Section 2.2); Finally, the multitasking layer is used to improve the feature extraction capabilities of the entire network model and the ability to represent embedded vector embedding (in Section 2.3). Next, we will detail each part of the model.

2.1 Multi-interaction network representation method and multi-interaction sequence extraction algorithm
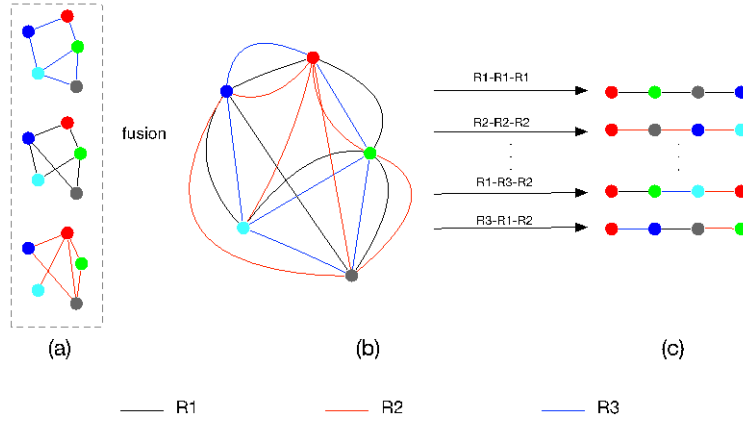

Figure 3: Schematic diagram of hybrid interactive sequence extraction

To better explore the complex interactions existing in the real social network, we first fuse the various single interactive sub-networks as shown in Fig. 3(a) to obtain multi-interactions heterogeneous information network (MIHIN) as shown in Fig. 3(b). It can be seen that there may be multiple edges of different colors between two nodes in the MIHIN network, that is, different interaction relationships may exist between the two nodes. Inspired by metapath2vec [Dong, Chawla and Swami (2017)], we use the multi-pattern path matching node sequence generation algorithm to generate hybrid interactive sequences. The difference is that metapath2vec extracts different types of nodes, but in our research, there are different types of interactions (as shown in Fig. 3(c)). Specifically, for a multi-interactive heterogeneous information network and an interaction pattern, the transition probability is as follows:

$$p(v^{i+1} \mid v^i, p, v^{i-1}) = \begin{cases} \alpha_{p,q}(v^{i-1}, v^{i+1}) & (v^i, v^{i+1}) \in E, (v^i, v^{i+1}) = R_i \\ 0 & (v^i, v^{i+1}) \in E, (v^i, v^{i+1}) \neq R_i \\ 0 & (v^i, v^{i+1}) \notin E \end{cases} \quad (1)$$

It can be seen that the sequence of walking strictly follows the relationship pattern, and the walk is performed according to the pattern circularly:

$$p(v^{i+1} \mid v^i) = p(v^{(i+1)\%l} \mid v^{i\%l}), \, if \, i \geq l \quad (2)$$

$\alpha_{p,q}(v^{i-1}, v^{i+1})$ is derived from Eq. (3). It defines the walk strategy under the condition that the relationship of the ith step of the relationship mode is satisfied, and the corresponding schematic diagram is as shown in Fig. 4. Direction and distance are added

to make it better to capture the bi-directional distance relationship between nodes [Zhu, Sun, Cao et al. (2019)].

$$\alpha_{p,q}(v_{i-1},v_{i+1}) = \begin{cases} p & d(v^{i-1},v^{i+1})=0 \\ 1 & d(v^{i-1},v^{i+1})=1 \\ q & d(v^{i-1},v^{i+1})=2 \end{cases} \tag{3}$$

where $d(v_{i-1},v_{i+1})$ is the bi-directional distance between v^{i+1} and v^{i-1}, p represents the probability of returning to the previous node, and q represents the probability of moving away from the previous node.
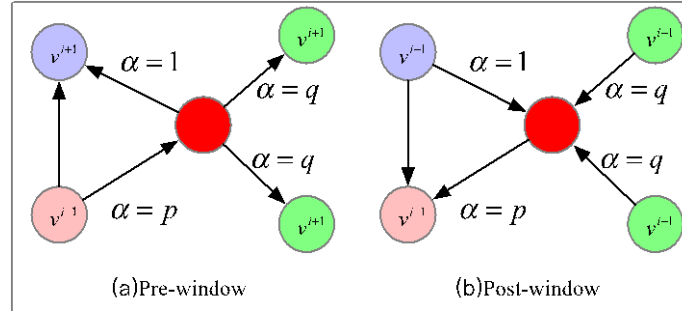

Figure 4: Schematic diagram of single-step context walk strategy

2.2 Attention-enhanced interaction fusion model

In the previous section, we obtained a node vector representation for each interaction pattern. To distinguish the contribution of different interaction patterns to the final node vector representation, one basic approach is to set different weights for each mode. The interaction pattern that has a greater impact should be set to a larger ratio. First of all, it is difficult to determine which interaction patterns contribute more to the final variable vector representation. Second, the setting of the specific ratio is a challenge task, and it is difficult to find the best ratio of the final merged fusion representation.

To solve the above problems, we introduced the Attention mechanism to learn these weights autonomously according to the target tasks. The ratio of each pattern is defined as follows:

$$\alpha_i^r = \frac{\exp(w_i^r \odot x_i^r)}{\sum_{r' \in R} \exp(w_i^{r'} \odot x_i^{r'})} \tag{4}$$

where x_i^r is the presentation of v_i in the interaction pattern of r, w_i^r is the corresponding ratio learned by Attention Net. In particular, we keep original vector X^f of follow relation pattern The learning of the specific ratio is performed by taking X^f and the node vector representations in various interaction patterns as inputs. Finally, the fused vector is represented as:

$$x_i = x_i^f + \sum_{r \in R, r \neq f} \alpha_i^r \odot x_i^r \tag{5}$$

where x_i is the final fusion vector representation of v_i, $x_i^f \in X^f$ is the representation of v_i of follow relation pattern. α_i^r can be continuously updated and adjusted to achieve optimal values as the entire model is trained.

2.3 Multi-task network embedding optimization method

In the previous section, we get the final multi-interaction fusion vector representation of each node, which contains rich semantic information in each interaction pattern. In this part, we design a multi-task learning model, which uses the fusion vector representation of nodes as a shared input and uses each task to learn together to better mine the potential semantic information. In the multi-tasking learning of this paper, the main task is used to evaluate and optimize the co-occurrence frequency of nodes within the specified window size of the node sequence. Therefore, it is necessary to minimize the objective function defined by Eq. (6).

$$O_1 = \sum_{v_j \in C_{v_i}^f} -\log p(v_i, v_j) \tag{6}$$

where $C_{v_i}^f$ is the context node set of node v_i in the follow network. $p(v_i, v_j)$ is the co-occurrence frequency of nodes v_i and v_j. The auxiliary task is to predict the sequence of nodes in each interaction mode through the LSTM network, and the sequence refers to a sequence obtained by a specific pattern interaction sequence extraction algorithm. The input is the previous node, and the target is the next node in the sequence, as shown in Fig. 5. For each interactive pattern network, the goal of this task is to maximize the probability of the next node:

$$p(X^{t+1} | X^t) \tag{7}$$

where $X^t = (x^1, x^2, ..., x^t)$ and $X^{t+1} = (x^2, x^3, ..., x^{t+1})$ We add the SoftMax function as the final output after the output of the model, see Eq. (8).

$$O_2' = soft\max(x_i) = \frac{e^{x_i}}{\sum_i e^{x_i}} \tag{8}$$

For each sequence, the cross-loss function is used to measure the loss of the model, see Eq. (9).
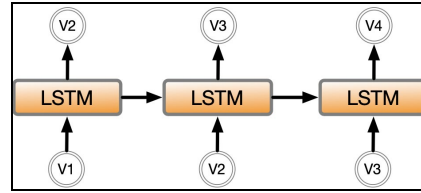
$$O_2 = -\sum y \ln O_2' \tag{9}$$

Figure 5: The diagram of interaction node sequence prediction

3 Experiment

In this section, we will validate the performance of the model through interactive prediction tasks on a variety of practical network datasets. To verify the performance of the model, we use the accuracy indicator to evaluate the model.

3.1 Datasets

We used the following open source real social network datasets from different platforms in our experiments.

● YouTube Dataset. A multi-dimensional network consists of various type of interactions which can be used to study shared communities among heterogeneous interactions. This is the data set crawled on Dec, 2008 from YouTube which is a video sharing site where various interactions occur between users. It contains five interactive pattern networks, we have selected the following:

$A_{11}(c)$: the contact network between the 15088 users;

$A_{12}(f)$: the number of shared friends between two users in the 848, 003 contacts;

$A_{13}(s)$: the number of shared subscriptions between two users;

$A_{14}(v)$: the number of shared favorite videos.

● Higgs Twitter Dataset. The Higgs dataset has been built after monitoring the spreading processes on Twitter during the announcement of the discovery of a new particle with the features of the elusive Higgs boson on 4th July 2012. The messages posted in Twitter about this discovery between 1st and 7th July 2012 are considered. It contains four interactive pattern networks：

$A_{21}(f)$: friends/followers social relationships among users.

$A_{22}(r)$: re-tweeting (retweet network);

$A_{23}(c)$: comment network to existing tweets;

$A_{24}(m)$: mentioning (mention network) other users;

Detailed statistics for each data set are shown in Tab. 1. As can be seen from the table, the scale of each network is different, and the average degree of nodes is also different, which fully reflects the unbalanced distribution of nodes in real social networks and can better verify the performance of the model in the real environment.

Table 1: Statistics of datasets

Networks	$|V|$	$|E|$	Avg. degree
A_{11}	13723	76765	11.19
A_{12}	13242	1940806	293.13
A_{13}	11765	5574249	947.60
A_{14}	13160	3797635	577.14
A_{21}	456626	14855842	32.53
A_{22}	256491	328132	1.28
A_{23}	38918	32523	0.84
A_{24}	116408	150818	1.30

3.2 Experiment setup

For the three benchmark methods DeepWalk, LINE, and Node2Vec, because they cannot handle heterogeneous interactions, we use the constructed multi-interactive heterogeneous information network as its input and treat all edges as equivalent types. For all network embedding methods, we set different parameters according to different data set sizes and characteristics, but in each data set, the parameters are same. For the relatively small dataset Higgs Twitter, we set the output dimension d to 128, and since the shortest path length of its network structure is only 9, the sequence length is set to 9; for the larger datasets Tencent Weibo and YouTube, we set the output dimension to 256 and the sequence length to 20. The learning rate is uniformly set to 0.001, the window size is set to 7, and the optimal p and q are found by grid search.

For our proposed model, we also need to set the interaction patterns. For the YouTube dataset, we choose the friend relationship pattern f and three interaction modes: $p_1 = c - s : u_1 \xleftrightarrow{c} u_2 \xleftrightarrow{s} u_3$ (Used to find people who have a common subscription with u_3 who interact closely with u_2); $p_2 = c - v : u_1 \xleftrightarrow{c} u_2 \xleftrightarrow{v} u_3$ (To find people who share a hobby with u_2 who interact closely with u_1); $p_3 = s - v : u_1 \xleftrightarrow{s} u_2 \xleftrightarrow{v} u_3$ (used to find people who share a hobby with u_2 who have a common subscription with u_1). For the Twitter dataset, we choose the friend relationship f and three interaction patterns: $p_1 = r - m : u_1 \xleftrightarrow{r} u_2 \xleftrightarrow{m} u_3$ (used to find people person mentioned by u_2 who frequently retweets $u_1's$ tweet); $p_2 = r - c : u_1 \xleftrightarrow{r} u_2 \xleftrightarrow{c} u_3$ (used to find people who are commented by u_2 who often retweets $u_1's$ tweet); $p_3 = c - m : u_1 \xleftrightarrow{c} u_2 \xleftrightarrow{m} u_3$ (used to find people who are mentioned by u_2 who often comments $u_1's$ tweet).

3.3 Experiment result

After setting the parameters, we use the node embedding results as input to perform the node interaction prediction task to evaluate the performance of our proposed model and compare it with the benchmark model. The node interaction prediction task we defined refers to predicting whether two nodes have the interaction relationship set in our experiment. It is worth noting that the interaction relationship between two nodes does not mean that there is a direct connection between the two nodes. For example, and in Fig. 1 have no directly connected edges, but they have an interactive relationship between Follow-Retweet-Review and Follow-Retweet-Like.

On each dataset, we selected different proportions of nodes as training data for training, and the rest as validation sets. Each experiment is the result of an average of 10 different sampling nodes. Tab. 2 shows the performance of each model under two datasets. As can be seen from the table, the models we propose are better than the benchmark model. Among them, LINE gets the worst result. The main reason is that LINE only pays attention to first-order proximity and second-order proximity. For two nodes that are far away, it cannot learn the semantic relationship. Therefore, it cannot Properly predict the potential long-distance interaction.

Table 2: Accuracy performance comparison for interaction prediction with different node ratio for train

Dataset	Algorithm	20%	40%	60%	80%
YouTube	DeepWalk	**0.364**	0.478	0.711	0.756
	Node2Vec	0.267	0.437	0.694	0.775
	LINE	0.231	0.395	0.662	0.689
	MINE	0.358	**0.543**	**0.781**	**0.821**
Twitter	DeepWalk	0.342	0.401	0.572	0.698
	Node2Vec	0.312	0.423	0.587	0.721
	LINE	0.289	0.381	0.506	0.631
	MINE	**0.348**	**0.441**	**0.629**	**0.779**

3.4 Parameter sensitivity analysis

In this section we explore and analyze the sensitivity of our proposed model to different parameters, including the dimension of the network embedding (d), the number of trainings (N), and the length of the sequence (L).

For the hyperparameter d, we set it to 16, 32, 64, 128, 256, 512 respectively; the sequence length increases from 10 to 50; the training times increases from 50 to 1000, and the change of accuracy is observed.
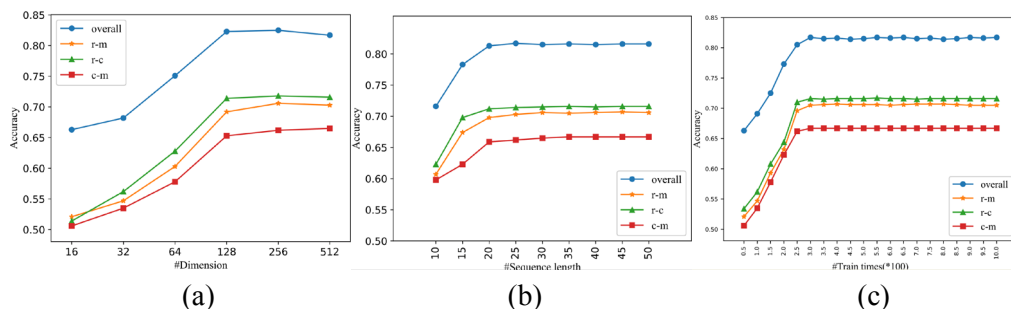
Figure 5: (a), (b), (c) show the effect of node dimension, sequence length and number of training on model performance respectively

Fig. 5(a) shows the trend of the accuracy values for various interaction patterns on the Twitter dataset as d is changed. We can see that when d exceeds 128 dimensions, the accuracy will be stable, which proves that our model can capture the potential mixed interaction information between nodes with low latent embedded vectors. Similar results are shown in Figs. 5(b) and 5(c). When the sequence length is greater than 20, our model also obtains stable and good results. For the training times, the model can convergence just be trained after 300 times, which reflects the efficiency of our model.

4 Conclusion and discussion

In this paper, in order to solve the problem of multi-interaction embedding between users in social networks, we propose the MINE model. Firstly, we introduce the multi-interactions heterogeneous information network and extract complex heterogeneous relation sequences by the multi-interaction extraction algorithm. Secondly, we exploit a well-designed multi-relationship network fusion model based on the attention mechanism to fuse multiple interactional relationships. Finally, applying a multitasking model makes the learned vector cover richer semantic relationships. A large number of practical experiments prove that our proposed method outperforms existing methods on multiple data sets. However, our model also has some shortcomings. For example, when analyzing interactions, we do not consider the chronological order of interactions; we do not introduce textual emotion information to better explore the relationship between users. These are our future work.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

Ahmed, A.; Shervashidze, N.; Narayanamurthy, S.; Josifovski, V.; Smola, A. J. (2013): Distributed large-scale natural graph factorization. *Proceedings of the 22nd International Conference on World Wide Web*, pp. 37-48.

Belkin, M.; Niyogi, P. (2002): Laplacian eigenmaps and spectral techniques for embedding and clustering. *Advances in Neural Information Processing Systems*, pp. 585-591.

Bourigault, S.; Lagnier, C.; Lamprier, S.; Denoyer, L.; Gallinari, P. (2014): Learning social network embeddings for predicting information diffusion. *Proceedings of the 7th ACM International Conference on Web Search and Data Mining*, pp. 393-402.

Chang, S.; Han, W.; Tang, J.; Qi, G. J.; Aggarwal, C. C. et al. (2015): Heterogeneous network embedding via deep architectures. *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 119-128.

Chen, H.; Wang, L.; Wang, S.; Luo, D.; Huang, W. et al. (2019): Label aware graph convolutional network-not all edges deserve your attention. ArXiv Preprint, vol. 1907, no. 04707.

Cui, P.; Wang, X.; Pei, J.; Zhu, W. (2018): A survey on network embedding. *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 5, pp. 833-852.

Dong, Y.; Chawla, N. V.; Swami, A. (2017): Metapath2vec: scalable representation learning for heterogeneous networks. *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 135-144.

Grover, A.; Leskovec, J. (2016): Node2vec: scalable feature learning for networks. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 855-864.

Hu, F.; Li, Z.; Yang, C.; Jiang, Y. (2019): A graph-based approach to detecting tourist movement patterns using social media data. *Cartography and Geographic Information Science*, vol. 46, no. 1, pp. 368-382.

Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. (2013): Efficient estimation of word representations in vector space. ArXiv Preprint, vol. 1301, pp. 3781.

Perozzi, B.; Al-Rfou, R.; Skiena, S. (2014): Deepwalk: online learning of social representations. *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 701-710.

Tang, J.; Qu, M.; Wang, M.; Zhang, M.; Yan, J. et al. (2015): Line: large-scale information network embedding. *Proceedings of the 24th International Conference on World Wide Web*, pp. 1067-1077.

Tenenbaum, J. B.; De Silva, V.; Langford, J. C. (2000): A global geometric framework for nonlinear dimensionality reduction. *Science*, vol. 290, no. 5500, pp. 2319-2323.

Wang, D.; Cui, P.; Zhu, W. (2016): Structural deep network embedding. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1225-1234.

Xu, J.; Wang, A.; Wu, J.; Wang, C.; Wang, R. et al. (2020): SPCSS: social network based privacy-preserving criminal suspects sensing. *IEEE Transactions on Computational Social Systems*, vol. 7, no. 1, pp. 261-274.

Zhao, G.; Zhang, Y.; Shi, Y.; Lan, H.; Yang, Q. (2019): The application of BP neural networks to analysis the national vulnerability. *Computers, Materials & Continua*, vol. 58, no. 2, pp. 421-436.

Zhu, D.; Wang, Y.; You, C.; Qiu, J.; Cao, N. et al. (2019): MMLUP: multi-source & multi-task learning for user profiles in social network. *Computers, Materials & Continua*.

Zhu, D.; Sun, Y.; Cao, N.; Qiao, X.; Xu, M. et al. (2019): BDNE: a method of bi-directional distance network embedding. *International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery IEEE*. pp. 158-161.