Weighted Meta-Path Embedding Learning for Heterogenous Information Networks

Zhang Yongjun, Yang Xiaoping and Wang Liang

May 16, 2020

Weighted Meta-Path Embedding Learning for Heterogenous Information Networks

Yongjun Zhang, Xiaoping Yang, and Liang Wang

School of Information, Renmin University of China, Beijing 100872, China
{zyjun, yang, wangliang}@ruc.edu.cn

Abstract. A low-dimensional embedding can be easily applied in the downstream tasks for network mining and analysis. In the meantime, the popular models of random walk-based network embedding are viewed as the form of matrix factorization, whose computational cost is very expensive. Moreover, mapping different types of nodes into one metric space may result in incompatibility. To cope with the two challenges above, a weighted meta-path embedding framework (WMPE) is proposed in this paper. On one hand, a nearly-linear approximate embedding approach is leveraged to reduce the computational cost. On the other hand, the meta-path and its weight are learned to integrate the incompatible semantics in the form of weighted combination. Experiment results show that WMPE is effective and outperforms the state-of-the-art baselines on two real-world datasets.

Keywords: Heterogeneous Information Networks, Network Embedding, Representation Learning, Meta-path Learning.

1 Introduction

In the real world, a variety of entities and relationships can be abstracted and represented as heterogeneous information networks (HINs), on which data mining and analysis have raised increasing attention in the past decade due to the complex structures and rich semantic information. In the meantime, network embedding, mapping a high-dimensional and sparse network into a low-dimensional and dense space to learn the latent representation of nodes and edges while preserving the multiplex semantics, is extremely convenient for network analysis, e.g., classification [1], clustering [2], link prediction [3], and recommendation system [4].

Matrix factorization is widely utilized for network embedding. Transform network into the form of matrix and decompose it, so that each node can be represented as a distribution on the latent semantics. However, it is hardly suitable for large-scale networks due to the expensive computational cost. Inspired by Word2Vec algorithm, a series of random walk-based models [6-8] introduce neural network into the task of graph embedding, and achieve significant successes. Nevertheless, a recent study [9] shows that methods aforementioned can be viewed as asymptotically and implicitly process of matrix factoring in essence.

Heterogeneous information network brings not only the rich semantic information, but also the conflicts among variety of relationships, which is more complex and difficult to deal with. For example, a node is associated with two different types of nodes, which are irrelevant each other. It is hard to represent the sematic that a node is simultaneously close to two distant nodes in one metric space. Therefore, embedding different types of nodes and edges into a same feature space will lead to semantic incompatibility, which brings the special challenge for HINs embedding.

With the intention to solve the problems aforementioned, we propose a weighted meta-path embedding framework, referred to as WMPE, to learn the nodes representation for HINs. Firstly, bypassing the eigen-decomposition, an approximate commute embedding approach is utilized to embed the heterogeneous network into a low-dimensional metric space, which reduce the computational cost in nearly-linear time. Aiming at the existence of semantic incompatibility in HINs, a set of meta-paths are automatically generated. Furthermore, the weight of each meta-path is learned by optimizing the loss function with a small number of labeled nodes. At last, relatedness between nodes are obtained in the form of weighted combination of meta-paths atop node embeddings. In this way, we can preserve the different semantics specified by the generated meta-paths even in the presence of incompatibility.

The main contributions of this paper can be summarized as follows:

- Instead of eigen-decomposition, the nearly-linear approximate embedding approach is capable of applying in large-scale HINs.
- Without any domain knowledge or user guidance, meta-paths are automatically generated. We designed an optimization model to learn the importance of each meta-path.
- Extensive experiments demonstrate the capability of the proposed model, which outperforms the state-of-the-art baselines on two real-world datasets.

2 Related Work

2.1 Meta-path of HIN

Meta-path describes the rich semantic via a sequence of relations amongst objects in HINs and using meta-paths to characterize the heterogeneity is distinct from that of homogeneous networks. The major line of work focusing on meta-path-based similarity or meta-path-based neighbors has been footstone for broad applications of heterogeneous information networks. PathSim [10] leverages the normalized number of reachable paths complying with the schema of meta-path to compute the similarity between a pair of nodes, and PathSelClus [11, 12] integrates a suite of meta-paths to extract network structure for objects clustering, where the paths are defined to be symmetrical and the relatedness is only measured between objects with the same type. In consideration of the asymmetric meta-paths, HeteSim [13] improves the method and focuses on relevance measure for objects with the same type or different types. Recently, some models [21-23] extract the meta-paths-level structural property and integrate other methods for HINs mining and analysis. However, the meta-paths are prepared in advance and treated equally, ignoring the fact that importance of the semantic information represented by corresponding meta-path is different from each other.

2.2 Network Embedding

Network embedding can be traced back to the usage of matrix factorization. Initially, matrix factorization is used to the dimension reduction, such as Multi-dimension Scaling (MDS) [14], Spectral Clustering [5], and Laplacian eigenmaps [15]. These methods transform the network into a matrix, and then a low-dimensional representation of the network can be obtained by matrix eigen-decomposition on each eigenvector. GraRep [16] decomposes k-step transition probability matrixes and integrates the k relationships to represent global characteristics of weighted network. HOPE [17] and M-NMF [18] capture network properties by depicting the n-th order proximity to construct relation matrixes for directed edges and macroscopic community structure respectively, where the optimizing process of matrix decomposition is quadratic time complexity at least. Therefore, the matrix factorization-based embedding is difficult to apply to large-scale networks.

Inspired by word embedding, the random walk-based models introduce neural network into network representation learning. DeepWalk [6] samples the network by depth-first random walk to generate nodes sequence, which can be served as "word-context" sentences for Word2Vec model to learn the nodes vectors. As an extension of DeepWalk, node2vec [8] designs a biased walk to explore the neighborhood of nodes in combination with BFS and DFS styles. LINE [7] defines the similarity of first-order proximity and second-order proximity to optimize objective function. However, reference [9] expounds the theoretical connections among these models of random walk-based network embedding, which can be induced to the form of matrix factorization.

In consideration of the diverse semantics amongst multiple nodes in heterogeneous information network, metapath2vec [19] generates the context sequences guided by meta-paths, and improves the Skip-Gram model to adapt to the negative sampling. In addition, HIN2Vec [20] takes nodes and their relationships specified in forms of meta-paths together as the input data for training, and also some efficient strategies are developed for data preparation. Recently, motivated by the great success of deep learning, some graph neural network models [21-23] are designed to extract "latent" semantics directly from a continuous distribution for HIN embedding. Reference [24] points out that, compared with homogenous networks, those methods above map multiple types of nodes and edges into one metric space, which will inevitably lead to incompatibility in heterogeneous information networks.

3 Preliminaries

In this section, we introduce some related concepts and formalize the notations.

DEFINITION 1. **Heterogeneous Information Network**. An information network is a directed graph $G = (V,\ E)$ with a node type mapping function $\phi: V \to A$ and a link type mapping function $\varphi: E \to R$, where each node $v \in V$ is mapped to one particular node type in A, i.e., $\phi(v) \in A$, and each edge $e \in E$ is mapped to one particular link type in R, i.e., $\varphi(e) \in R$. When $|A| > 1$or$|R| > 1$, the network is called a heterogeneous information network.

DEFINITION 2. **Network schema**. A network schema is denoted as $T_G = (A, R)$ to abstract the meta-information of the given HIN $G = (V, E)$ with node type mapping function $\phi: V \to A$ and a link type mapping function $\varphi: E \to R$.

DEFINITION 3. **Meta-path**. A meta-path is defined as a path schema in the form of $A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} \cdots \xrightarrow{R_t} A_{l+1}$, which describes a composite relation $R = R_1 \circ R_2 \circ \cdots \circ R_l$between type A_1 and A_{l+1}, where $\circ$ denotes the composition operator on relations.

DEFINITION 4. **Path weight matrix**. Given a meta-path $p: A_1 \to A_2 \to \cdots \to A_{l+1}$, the path weight matrix is defined as $M = W_{A_1A_2} \times W_{A_2A_3} \times \cdots \times W_{A_lA_{l+1}}$, where $W_{A_iA_j}$ is the adjacency matrix between the nodes of A_i and A_j. M_{ij} denotes the number of instances complying with the meta-path p from node $i \in A_1$ to node $j \in A_j$.

4 Framework of Proposed WMPE

To cope with the expensive cost of matrix factorization in embedding process and incompatibility among heterogeneous semantics, we propose a weighted meta-path embedding framework, referred to as WMPE, to learn the low-dimension embedding representation for heterogeneous information networks. The framework of WMPE is shown in Fig.1, which includes in two phases. In the first phase, all the nodes in HIN are pretrained to learn the vectorized representation by a nearly-linear approximate embedding approach. In the second phase, a set of meta-paths are automatically generated, and the weight of each meta-path is learned by optimizing the loss function with a small number of labeled nodes. Afterward, the proposed WMPE framework is described in detail.

4.1 Approximate Commute Embedding

Instead of the eigen-decomposition, an approach of approximate commute embedding is applied to project the network into a low-dimension metric space for embedding learning, which can greatly reduce the computational complexity into nearly-linear time.

Given a graph and its transition probability matrix, commute distance is defined as the expected steps from node i to node j by random walk. The probability, to measure the distance between i and j, comprehensively considers all the reachable paths and reflects internal topological structure of the network.

A HIN can be denoted by a directed graph G with n notes and s edges. Given signed edge-vertex incidence matrix$B_{s\times n}$, defined as

$$B(e, v) = \begin{cases} 1, & \text{if } v \text{ is the head of } e, \\ -1, & \text{if } v \text{ is the tail of } e, \\ 0, & \text{otherwise,} \end{cases}$$

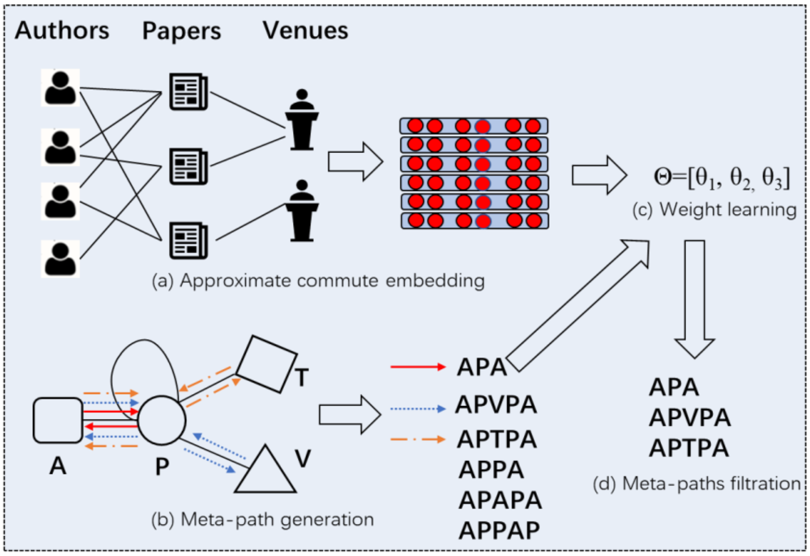


Fig. 1. An illustration of the WMPE to deal with the HINs

and diagonal matrix of edges weights $W_{s\times s}$, the Laplacian $L = B^T W B$ [25].

Therefore, the commute distance between node i and j can be written as:

$$\begin{aligned} c_{ij} &= V_G \left(e_i - e_j\right)^T L^+ \left(e_i - e_j\right) \\ &= V_G \left(e_i - e_j\right)^T L^+ B^T W B L^+ \left(e_i - e_j\right) \\ &= \left[\sqrt{V_G} W^{1/2} B L^+ \left(e_i - e_j\right)\right]^T \cdot \left[\sqrt{V_G} W^{1/2} B L^+ \left(e_i - e_j\right)\right] \end{aligned}$$

where $V_G = \sum w_{ij}$. That is, $\theta = \sqrt{V_G} W^{1/2} B L^+ \in \mathbb{R}^{s\times n}$ is a commute embedding for HIN G where c_{ij} is the squared Euclidean distance between the i-th and j-th column vectors in space θ. Since it takes $O(n^3)$ for the pseudo-inversion of L in $\theta = \sqrt{V_G} W^{1/2} B L^+$, an approximate commute embedding method is adopted more efficiently.

Lemma 1 [25]. Given vectors $v_1, \dots, v_n \in R^{n\times s}$ and $\varepsilon > 0$, let $Q_{k_r\times s}$ be a random matrix where $Q(i,j) = \pm 1/\sqrt{k_r}$ with equal probability $k_r = O(\log n/\varepsilon^2)$. For any pair v_i, v_j, exist

$$(1-\varepsilon)\left\|v_i - v_j\right\|^2 \le \left\|Qv_i - Qv_j\right\|^2 \le (1+\varepsilon)\left\|v_i - v_j\right\|^2$$

with probability at least $1 - 1/n$.

Therefore, construct a matrix $Z_{k_r\times s} = \sqrt{V_G} Q W^{1/2} B L^+$ and we have:

$$(1-\varepsilon)c_{ij} \leq \left\|Z(e_i - e_j)\right\|^2 \leq (1+\varepsilon)c_{ij}$$

for $\forall v_i, v_j \in G$, with probability at least $1 - 1/n$ from Lemma 1. That is, $c_{ij} \approx \left\|Z(e_i - e_j)\right\|^2$ with an error ε. Due to the expensive computational cost of L^+, a nearly-linear time method ST-solver [26] is used instead. Let $Y = \sqrt{V_G}QW^{1/2}B$, and $Z = YL^+$ which is equivalent to $ZL = Y$. Then each z_i (the i-th row of Z) is computed by solving the equation $z_i L = y_i$ where y_i is the i-th row of Y, and the solution is denoted as $\tilde{z}_i$. Since $\|z_i - \tilde{z}_i\|_L \leq \varepsilon\|z_i\|_L$, we have:

$$(1-\varepsilon)^2 c_{ij} \leq \left\|\tilde{Z}(e_i - e_j)\right\|^2 \leq (1+\varepsilon)^2 c_{ij} \quad (1)$$

where $\tilde{Z}$ is the matrix consisting of $\tilde{z}_i$. Equation (1) indicates that $c_{ij} \approx \left\|\tilde{Z}(e_i - e_j)\right\|^2$ with the error ε^2 and $\tilde{Z}$ is the approximate commute embedding of G.

4.2 Meta-path Generation and Weight Learning

In order to measure the semantic-level relatedness of nodes, a set of meta-paths are generated automatically by a random walk strategy.

Given the start node type A_s and end node type A_e, the target is to find all the meta-paths in the form of $A_s - * - A_e$ within the maximum length L. On the network schema, start from node A_s, and randomly jump to the neighbor of current node for each step within L. Each time the node A_e is reached, the sequential passed nodes compose a meta-path. Repeating this process for the number of iterations, a set of candidate meta-paths are generated automatically. Note that, the start and end nodes can be of arbitrary types, not requiring the same type, and the path does not have to be symmetric. Fig.1(b) illustrates the process of meta-paths generation, the form is defined as $author - * - author$, with the maximum length 4. After the exploration on DBLP schema, a set of meta-paths are obtained including: APA, APPA, APPPA, APAPA, APCPA, and APTPA.

The contributions of different semantic relationships specified by the meta-paths are not equivalent to a certain target. In fact, the conflicting, irrelevant semantics will result in low performances in the following tasks. Also, more meta-paths will increase the computation cost. A proper way is to assign higher weights to the meta-paths that can promote the downstream tasks, and discard the irrelevant and conflicting meta-paths. Therefore, we design an objective function to transform the weight learning process of meta-paths into an optimization problem.

For a meta-path p_k, we build the meta-path relatedness of node pair (u, v) atop their approximate commute embeddings in the first phase as

$$sim_k(u,v) = \frac{exp(\,M_{uv}^k \cdot \mathrm{g}(u,v))}{\sum_{\phi(\tilde{v})=\phi(v)} exp(\,M_{u\tilde{v}}^k \cdot \mathrm{g}(u,\tilde{v})) + \sum_{\phi(\tilde{u})=\phi(u)} exp(\,M_{\tilde{u}v}^k \cdot \mathrm{g}(\tilde{u},v))},$$

where M_{uv}^k denotes the number of paths between u and v following meta-path p_k, and $\mathrm{g}(u,v)$ is the nodes closeness from u to v. Considering the existence of both undirected and directed relationships in heterogeneous networks, $g(u,v) \neq g(v,u)$, if the path between the two nodes contains a directed relation. Especially, we decompose the pre-trained node embedding into two sections $x = \begin{bmatrix} x^O \\ x^I \end{bmatrix}$, where x^O and x^I are two column vectors of the same dimension, and define nodes closeness on approximate commute embedding as

$$g(u,v) = \begin{cases} 2\, x_u^O \cdot x_v^I, & \text{directed from } u \text{ to } v \\ x_u^O \cdot x_v^I + x_v^O \cdot x_u^I, & \text{undirected} \end{cases},$$

where $\cdot$ denotes the inner-product.

on the contrary to the assumption that nodes of different types are independent each other, we hold the view that closely related nodes in a network are not only structurally but also semantically consistent. Therefore, taking all meta-paths into account, the lose function is designed to learn the weight of each meta-path as the follows:

$$L(\Theta) = \|1 - sign(u,v) \textstyle\sum_{k=1}^{K} \theta_k sim_k(u,v)\|^2 + \alpha\|\Theta\|^2$$

where $\Theta = \{\theta_1, \theta_2, \cdots, \theta_K\}$, θ_k is the importance of meta-path p_k, and α is the regularization parameter. The $sign()$ is an indicator function defined as

$$sign(u,v) = \begin{cases} 1, & u \text{ and } v \text{ have the same label} \\ -1, & \text{otherwise} \end{cases}.$$

The loss function is to maximize the importance of meta-path connecting nodes with same label and minimize that connecting nodes with different labels.

In order to minimize $L(\Theta)$, for each $k = 1,2,\cdots,K$, equate the partial derivative of loss function with respect to θ_k to zero. Iterating this process until the error converges, the importance of each meta-path is obtained. The meta-paths with negative or too small value will be discarded, which represent conflict and low correlative semantic meanings to the following tasks. Weights are computed by normalizing the importance values of reduced meta-paths. In this way, the weighted combination of independent semantics specified by meta-paths is to measure the similarities among nodes in heterogeneous information network, which can be easy-to-use for the downstream tasks, e.g., classification.

5 Experiments

5.1 Datasets

The proposed WMPE is evaluated on two real-world datasets: DBLP and IMDB.

DBLP is a bibliographical network in computer science, which includes four types of notes: paper (P), author (A), venue (V), and term (T). We construct two subsets of DBLP. DBLP_1 contains 240 venues (30 venues for each area), 189,378 authors, 217,764 papers and 263,332 terms in eight research areas. Because of the bias that a few authors publish most of the papers, DBLP_2 extracts 20 venues, 9,737 authors (publish at least five papers), 12,098 papers, and 9,936 terms in four research areas. To evaluate the methods, the research area is taken as ground truth.

IMDB is a movie network consisting of five types of notes: movie (M), actor (A), director (D), year (Y) and genre (G). There are 5000 movies, 2342 actors, 879 directors and 67 genres, where we use the movie genre as ground truth.

5.2 Baselines and Evaluation Metrics

We compare the proposed WMPE framework with the state-of-the-art baseline models: DeepWalk, LINE, HIN2Vec and metpath2vec. DeepWalk and LINE are designed for homogeneous network embedding. HIN2Vec c and metpath2vec aim at heterogeneous information network, but all nodes and relationships are mapped into the same feature space.

Furthermore, the two popular metrics: Accuracy (Acc) and macro F1-score (F1) are adopted to assess the classification results.

5.3 Meta-path Filtration

A set of meta-paths are generated and the corresponding importance values are also learned respectively on DBLP and IMDB. The results are shown in Table 1, and the preserved meta-paths are indicated in bold.

Table 1. Results of meta-path generation and the importance.

	Meta-path	Length	Importance
DBLP	**author-paper-author (APA)**	2	0.13
	author-paper-paper-author (APPA)	3	0.09
	author-paper-paper-paper-author (APPPA)	4	0.05
	author-paper-author-paper-author (APAPA)	4	0.11
	author-paper-term-paper-author (APTPA)	4	0.31
	author-paper-venue-paper-author (APVPA)	4	0.78
IMDB	**movie-actor-movie (MAM)**	2	0.17
	movie-director-movie (MDM)	2	0.44
	movie-year-movie (MYM)	2	-0.30
	movie-actor- movie-director-movie (MAMDM)	4	0.03
	movie-actor-movie-year-movie (MAMYM)	4	-0.46
	movie-director-movie -year-movie (MDMYM)	4	-0.42

For DBLP dataset, we set the maximum length as 4， and the meta-paths are generated in the form of $author - * - author$, in which both the start node type and end node type are authors. The order of importance for each meta-path in DBLP is as follow: APVPA > APTPA >APA>APAPA>APPA>APPPA. It turns out that, compared with other relations, the authors publishing papers at the same conference are more likely to be in the same field.

For IMDB dataset, we set the maximum length as 4， and the meta-paths are generated in the form of $movie - * - movie$. Since the meta-paths are undirected, reciprocal paths are filtrated away. We note that all the importance values of the meta-path including the type of year (Y) are negative, which means there is no connection between the year a film is made and its genre. Therefore, those meta-paths will disturb the following tasks.

5.4 Results of Classification

We conduct a classification experiment using KNN classifier to verify the effectiveness of the algorithm. The embeddings of nodes in networks are represented by 128-dimension vectors for all the methods. Randomly select 2%, 4% and 6% of the labelled nodes for weight learning respectively. Due to the random initialization, we repeat the baselines 10 times and report the average performances in Table 2, in which a larger value implies a better effect.

As can be seen, on the whole, WMPE outperforms the other baselines on three datasets for two metrics, which validate the effectiveness of our proposed model. In terms of datasets, all methods on DBLP_2 achieve best results. We conclude that DBLP_2, compared with DBLP_1, discards large numbers of authors who publish only a few papers to void the sparsity of network, which is instrumental for classification. The worst overall performances are on IMDB. One explanation could be that an actor/actress or director does not exactly associated with a certain movie genre，which can be confirmed from weights of the meta-paths including the type actor. Compared to the best performances of baselines on DBLP_1, a large-scale and biased network, WMPE achieves the improvements of 5.55% Acc, and 6.31% F1 over metapath2vec.

For baselines, the observations in table 2 show that the results of heterogeneous network embedding models (HIN2Vec and metpath2vec) are generally better than that of the homogeneous (DeepWalk and LINE). DeepWalk and LINE cannot use the meta-paths and the results will not change, although the labelled nodes are increasing. Since meta-paths are introduced to represent the semantic information among different types of nodes. Besides, because of discarding the conflicting and irrelevant meta-paths and assigning higher weights to the effective meta-paths, our method WMPE is superior to all the methods above.

5.5 Impact of Different Meta-Paths

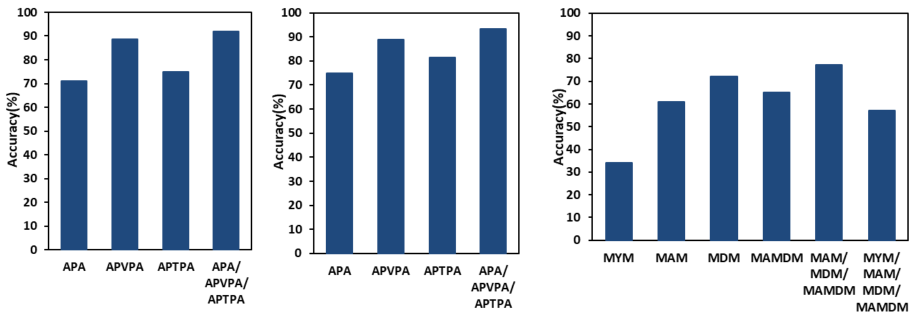
We also conduct an experiment to analyze the effect of different meta-paths and the results are given in Fig. 2.

As shown in Fig. 2(a) and Fig. 2(b), the effect of the experiment is consistent with the importance of the meta-paths that the meta-path with the highest weight gets the best results. This suggests that the relation of publishing papers in common venues is more sensitive than the relations of sharing common terms and co-author to research areas.

For IMDB dataset, shown in Fig. 2(c), we add a meta-path MYM to compare with the preserved ones. The accuracy of MDM is better than that of MAM. That is, directors, compared with actors/actresses, are more likely to have their inherent genres. We note that, increasing meta-paths can improve the accuracy in general, but it does not mean the more, the better. A possible reason is that some meta-paths may contain noisy or conflicting information, e.g., MYM.

Table 2. Performance comparison of classification. The best results are indicated in bold.

Method	Metric	DBLP_1			DBLP_2			IMDB		
		2%	4%	6%	2%	4%	6%	2%	4%	6%
DeepWalk	Acc	65.38	65.38	65.38	74.32	74.32	74.32	53.20	53.20	53.20
	F1	62.71	62.71	62.71	70.10	70.10	70.10	52.02	52.02	52.02
LINE	Acc	81.74	81.74	81.74	85.23	85.23	85.23	68.27	68.27	68.27
	F1	80.43	80.43	80.43	81.82	81.82	81.82	65.54	65.54	65.54
HIN2Vec	Acc	83.61	84.67	84.73	90.09	91.87	93.04	72.45	72.68	73.26
	F1	84.27	85.40	85.62	88.34	89.34	90.73	71.78	71.88	72.09
metapath2vec	Acc	86.93	86.21	87.64	**93.73**	**95.49**	95.55	72.56	72.56	73.42
	F1	83.57	84.07	86.32	91.65	91.64	91.89	70.30	70.30	71.35
WMPE	Acc	**90.76**	**91.73**	**91.89**	93.57	94.29	**95.58**	**74.24**	**75.09**	**78.22**
	F1	**89.01**	**90.38**	**91.23**	**92.10**	**92.66**	**93.21**	**74.93**	**76.32**	**77.06**



(a)DBLP_1 (b)DBLP_2 (c)IMDB

Fig. 2. Impact of meta-paths on different datasets

5.6 Parameter k_r

Parameter k_r is related to the error of approximate commuting embedding. A proper value can not only ensure the classification effectiveness, but also reduce the learning time of network embedding. Fig. 3 illustrates the influence of k_r on classification accuracy. With the increase of k_r, the accuracy curve is rising, but tend to be smooth. According to the curve, k_r is set to 60, 50 and 40 on dataset DBLP_1, DBLP_2 and IMDB respectively.

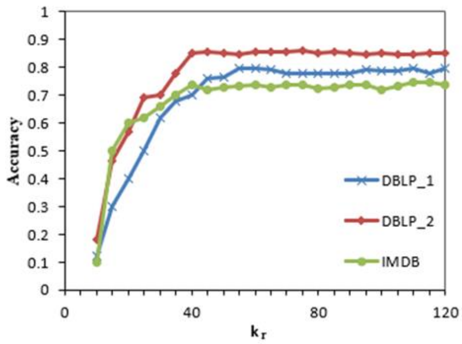


Fig. 3. Classification results with different k_r

6 Conclusion

This paper presents a weighted meta-path embedding learning for heterogenous information networks, called WMPE, which can easily integrate incompatible semantics by a weighted combination of the effective meta-paths. Also, a nearly-linear approximate approach reduces the time complexity in embedding process. Experimental results on two real-world datasets prove that the proposed WMPE is effective and feasible.

References

1. Lee, G., Kang, S., Whang, J.: Hyperlink classification via structured graph embedding. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1017–1020 (2019).
2. Fan, S., Wang, X., Shi, C., et al.: One2multi graph autoencoder for multi-view graph clustering. In: WWW '20: Proceedings of The Web Conference 2020, pp.3070–3076 (2020).
3. Rosso, P., Yang, D., Philippe, C.M.: Beyond triplets: hyper-relational knowledge graph embedding for link prediction. In: WWW '20: Proceedings of The Web Conference 2020, pp 1885–1896 (2020).
4. Wen, Y., Guo, L., Chen, Z., et al.: Network embedding based recommendation method in social networks. In: Companion of the Web Conference 2018, pp. 11–12 (2018).
5. Ng, A. Y., Jordan, M. I., Weiss, Y.: On spectral clustering: analysis and an algorithm. In: Advances in Neural Information Processing Systems, pp. 849–856 (2002).
6. 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, pp. 701–710 (2014).
7. Tang, J., Qu, M., Wang, M., et al.: LINE: Large-scale information network embedding. In: Proceedings of 24th International Conference on World Wide Web, pp. 1067–1077 (2015).
8. 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, pp. 855–864 (2016).

9. Qiu, J., Dong, Y. Ma, H., et al.: Network embedding as matrix factorization: unifying DeepWalk, LINE, PTE, and node2vec. In: Proceedings of the 11th ACM International Conference on Web Search and Data Mining, pp. 459–467 (2018).
10. Sun, Y., Han, J., Yan, X., et al.: Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. In: Proceedings of the VLDB Endowment, pp. 992-1003 (2011).
11. Sun, Y., Norick, B., Han, J., et al.: Integrating meta-path selection with user-guided object clustering in heterogeneous information networks. In: Proceedings of 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 723-724 (2012).
12. Yu, X., Sun, Y., Norick, B., et al.: User guided entity similarity search using meta-path selection in heterogeneous information networks. In: Proceedings of the 21st ACM International Conference on Information and Knowledge Management, pp. 2025-2029 (2012).
13. Shi, C., Kong, X., Huang, Y., et al.: HeteSim: A general framework for relevance measure in heterogeneous networks. In: IEEE Transactions on Knowledge and Data Engineering, pp. 2479-2492 (2014).
14. Agarwal, A., Phillips, J. M. and Venkatasubramanian, S.: Universal multi-dimensional scaling. In: Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 1149–1158 (2010).
15. Belkin, M. and Niyogi, P.: Laplacian eigenmaps and spectral techniques for embedding and clustering. In: Advances in Neural Information Processing Systems, pp. 585–591 (2002).
16. Cao, S., Lu, W., and Xu, Q.: GraRep: Learning graph representations with global structural information. In: Proceedings of the 24th ACM International Conference on Information and Knowledge Management, pp. 891–900 (2015).
17. Ou, M. , Cui, P., Pei, et al.: Asymmetric transitivity preserving graph embedding. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1105–1114 (2016).
18. Wang, X., Cui, P., Wang, J., et al.: Community preserving network embedding. In: 31st AAAI Conference on Artificial Intelligence, pp. 203–209 (2017).
19. Dong, Y., Chawla, N. V. and Swami, A.: metapath2vec: Scalable representation learning for heterogeneous networks. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 135–144 (2017).
20. Fu, T., Lee, W., and Lei, Z.: HIN2Vec: Explore meta-paths in heterogeneous information networks for representation learning. In: Proceedings of the 26th ACM on Conference on Information and Knowledge Management, pp. 1797–1806 (2017).
21. Hu, Z., Dong, Y., Wang, K., et al.: Heterogeneous graph transformer. In: Proceedings of the World Wide Web Conference, (2020).
22. Wang, X., Ji, H., Shi, C., et al.: Heterogeneous graph attention network. In: Proceedings of the World Wide Web Conference, pp. 2022–2032 (2019).
23. Hu, B., Fang, Y., and Shi, C.: Adversarial Learning on Heterogeneous Information Networks. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 120–129 (2019).
24. Shi, Y., Zhu, Q., Gou, F., et al.: Easing embedding learning by comprehensive transcription of heterogeneous information networks. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 2190–2199 (2018).
25. Khoa, N. L. D., Chawla, S.: Large scale spectral clustering using resistance distance and Spielman-Teng solvers. In: International Conference on Discovery Science, pp. 7–21 (2012).
26. Spielman, D. A., Srivastava, N.: raph sparsification by effective resistances. In: Proceedings of the 40th annual ACM Symposium on Theory of Computing, pp. 563–568 (2008).