Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Learning embeddings of a heterogeneous behavior network for potential behavior prediction*

Yue-yang WANG[‡1,2], Wei-hao JIANG[3], Shi-liang PU[3], Yue-ting ZHUANG[1]

[1]*College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China*

[2]*School of Big Data & Software Engineering, Chongqing University, Chongqing 401331, China*

[3]*Hikvision Research Institute, Hangzhou Hikvision Digital Technology Co., Ltd., Hangzhou 310051, China*

E-mail: yueyangw@zju.edu.cn; jiangweihao5@hikvision.com; pushiliang@hikvision.com; yzhuang@zju.edu.cn

Received Aug. 15, 2018; Revision accepted Jan. 31, 2019; Crosschecked Sept. 18, 2019; Published online Dec. 27, 2019

Abstract: Potential behavior prediction involves understanding the latent human behavior of specific groups, and can assist organizations in making strategic decisions. Progress in information technology has made it possible to acquire more and more data about human behavior. In this paper, we examine behavior data obtained in real-world scenarios as an information network composed of two types of objects (humans and actions) associated with various attributes and three types of relationships (human-human, human-action, and action-action), which we call the heterogeneous behavior network (HBN). To exploit the abundance and heterogeneity of the HBN, we propose a novel network embedding method, human-action-attribute-aware heterogeneous network embedding (a^{4}HNE), which jointly considers structural proximity, attribute resemblance, and heterogeneity fusion. Experiments on two real-world datasets show that this approach outperforms other similar methods on various heterogeneous information network mining tasks for potential behavior prediction.

Key words: Network embedding; Representation learning; Human behavior; Social networks; Heterogeneous information network; Attribute

https://doi.org/10.1631/FITEE.1800493 **CLC number:** TP39

1 Introduction

Exploring and analyzing the human behavior of specific groups, which is consistent and regular to some degree, can assist some organizations in making decisions. For example, governments explore terrorist behaviors to safeguard public security. Companies analyze behaviors of commercial spies to avoid business losses. Sellers get to know the requirements of enthusiasts in specific fields to achieve more profit. Human behavior is implicated in social actions that individuals have already performed, but these actions are difficult to observe in their entirety. Based on the insufficient information, predicting individuals who may take specific actions and further categorizing them into specific groups, which we call "potential behavior prediction," is a challenging task.

Let us illustrate the point using a practical scenario, in which a tourism website provides services for its members. Travel enthusiasts can start and organize traveling activities on the website and the website recommends activities to potential visitors. Here, a travel is an action, and the behavior is that a member group tends to join in a specific type of travel. However, the website cannot observe all actions of members. For example, some members may become friends after an activity and connect to each other offline for their next travels. How does the

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (Nos. U1509206, 61625107, and U1611461) and the Key Program of Zhejiang Province, China (No. 2015C01027)

ORCID: Yue-yang WANG, http://orcid.org/0000-0003-3210-0930

website precisely recommend interesting activities to members and divide them into groups based on incomplete action information? That is our topic.

Fortunately, with the rapid development of information technology, especially social networks, enormous and heterogeneous information from multiple perspectives can be considered to figure out an individual as part of a group. The characteristics of a member of the tourism website are also reflected in his/her registration information and photos or blogs he/she submits. The information is called the attribute, and can also promote similarity measurement and prediction accuracy to remedy the lack of action information. Thus, when we explore a real social system, the information contains attribute information of two types of objects, i.e., humans and actions, and also implicitly reflects three types of relationships, i.e., human-human, human-action, and action-action (the action-action relationship is the temporal continuity or spatial consistency of actions, like traveling activities in the same scenic spot). We formally define an information network, in which nodes represent objects with attributes and edges denote the relationships; we call it the heterogeneous behavior network (HBN). An example of an HBN is shown in Fig. 1. People's daily lifestyles are treated as actions, such as dining in a restaurant or meeting in a bar. Human attributes and action attributes are the consumer profile, which contains age, gender, occupation, etc., and the purchase history, which contains the type, amount, etc.

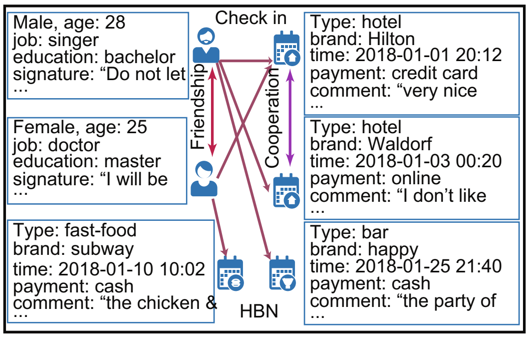

Fig. 1 An example of a heterogeneous behavior network (HBN). There are three types of links and two types of nodes associated with attributes

Our goal is to use an HBN constructed from practical information for potential behavior prediction. The most popular analysis methodology for information networks is network embedding, i.e., network representation learning, which embeds the networks into the low-dimensional space to directly measure the neighborhood similarities between nodes. However, in this application, the state-of-the-art network embedding methods suffer from limitations due to the following complexities of the HBN (Grover and Leskovec, 2016; Dong et al., 2017):

1. Semantics

The explicit edges and corresponding attributes in the network can reflect a semantic relationship between objects. For instance, friends of a member who likes hiking are not all interested in hiking, but a friend who has a similar age and occupation, and also checks into the same tourist hotel usually with the member has high chance of being a hiker.

2. Heterogeneity

There are two types of heterogeneity in this network: structural heterogeneity and attribute heterogeneity. The former refers to the multiple types of nodes and edges which should not be mixed. The latter refers to the multiple types of attributes that exist because the attributes come from cross-modal information.

3. High volume

The volume of the data obtained in the real world could be in tens of thousands or even billions.

In this study, we propose human-action-attribute-aware heterogeneous network embedding (a^{4}HNE), a novel method that is adequate for these characteristics of the HBN. a^{4}HNE learns the latent representation of humans and actions from the HBN by jointly considering the network structure, node attributes, and fusion of both. Specifically, to uniformly represent the cross-modal attributes of humans and actions, a^{4}HNE adopts the attribute encoder network to map them into a common space. The embedding representation yielded by a^{4}HNE can be used in downstream applications of potential behavior prediction, such as (1) predicting the individual in a group who may take a certain action, (2) classifying specific groups of humans, or (3) evaluating the relationship between humans and actions. Compared with other methods, a^{4}HNE can directly generate the representation of a new node without edges through the attribute encoder network.

We conduct experiments on two real-world datasets in different domains. One dataset comes from a practical application in a company for analyzing people's daily activities to predict the possibility of someone performing a certain action and classify them into specific groups. The other dataset is a public dataset about academic social networks for analyzing the research field of authors and papers, which is widely used to evaluate network embedding approaches. The effectiveness of our method is evaluated based on link prediction, node classification, and node clustering. The experimental results suggest that a^{4}HNE outperforms state-of-the-art methods. The robustness of our model is also demonstrated in various types of mining tasks. We analyze details about the proposed method including the cold-start problem, result visualization, and parameter sensitivity. Finally, we offer in-depth discussion of the differences of a^{4}HNE compared to the existing network embedding methods and what role the attribute plays in the HBN and more downstream applications of a^{4}HNE.

The contributions are summarized as follows:

1. We propose a novel network embedding model that captures structural semantics, attribute semantics, and the interrelationships between them in the HBN for potential behavior prediction.

2. We use the attribute encoder network to learn the mapping of heterogeneous attributes into a common space, which can uniformly handle the cross-modal attributes of humans and actions.

3. We construct experiments for the tasks of link prediction, node classification, and node clustering on two real-world datasets. The results show that our model can achieve significant improvements over other methods, e.g., 11.7%–34.0% improvement over the best of the state-of-the-art methods in terms of area under the curve (AUC) for link prediction on the HIK dataset.

2 Related work

In this section, we review the related work related to potential behavior prediction network embedding techniques.

2.1 Potential behavior prediction

Potential behavior prediction can be roughly classified into two levels: action level and behavior level. By action-level prediction (regarding as a link prediction task), the unobserved or missing actions (activities) of individuals are predicted (Liben-Nowell and Kleinberg, 2007). For example, Lerman et al. (2012) proposed proximity metrics for link prediction and predicted the URL forwarding activity in social media. In behavior-level prediction, the action similarity and group division of humans are studied at a macroscopic level, which is similar to the classification task or clustering task. Yin et al. (2016) proposed a unified probabilistic generative model, user-community-geo-topic (UCGT), by incorporating the spatiotemporal data and semantic information.

In this study, we verify the proposed method in all the three tasks, i.e., link prediction, node classification, and node clustering, to prove that it is appropriate for both levels of potential behavior prediction.

2.2 Network embedding

Network embedding has been proved to be a highly effective method for mining tasks of information networks, e.g., link prediction (Vazquez et al., 2003; Backstrom and Leskovec, 2011), node classification (Sen et al., 2008; Bhagat et al., 2011), and node clustering (Ding et al., 2001). It has been applied in many fields (Ou et al., 2016; Tu et al., 2017b).

The initial works of network embedding are based mainly on matrix factorization (Koren, 2008; Tang and Liu, 2009; Ma et al., 2011), and have achieved some success but also suffered from high computational cost in handling large-scale data. With the development of deep learning, many researchers designed neural network based network embedding models to capture structural correlations in homogenous information networks (Perozzi et al., 2014; Tang et al., 2015a; Grover and Leskovec, 2016). struc2vec learns latent representations for the structural identity of nodes (Ribeiro et al., 2017). Yang et al. (2017) proposed a network embedding update algorithm to approximate higher-order proximities with a theoretical approximation bound.

Homogeneous information networks can be treated as a special case of heterogeneous information networks. For greater generalizability, some methods have been proposed to leverage the rich semantic meanings of different types of nodes and links in heterogeneous information networks (Chang et al., 2015; Tang et al., 2015b). Chen and Sun (2017) proposed a task-guided and path-augmented heterogeneous network embedding model for author identification. Due to the success of word2vec (Mikolov et al., 2013b) in natural language processing, meta-path based random walk methods have been brought into the heterogeneous information network (Chen and Wang, 2017; Dong et al., 2017; Huang and Mamoulis, 2017; Shi et al., 2019). However, the increasing number of node types in complex heterogeneous information networks will cause the combination explosion problem of meta paths.

The aforementioned research focuses on the structural semantics of information networks. However, in reality, information networks also involve attribute information. There are only a few researchers considering single attributes in homogenous information networks. Yang et al. (2015) introduced TADW (text-associated DeepWalk), improving matrix factorization based on DeepWalk with text information. Sun et al. (2016) proposed CENE (content-enhanced network embedding), leveraging the network structure and content information by treating the textual content as another type of node. Pan et al. (2016) presented TriDNR (tri-party deep network representation), exploiting network information with node-word correlation and label-word correspondence.

To the best of our knowledge, there is little research that can simultaneously address the three characteristics of the HBN, i.e., semantics, heterogeneity, and high volume. Zhang et al. (2018) proposed CARL (content-aware representation learning), which is based on random walks to handle both structural content and unstructured semantic content. Different from their work, we design a heterogeneous edge-sampling-based model, a^{4}HNE, which includes an attribute encoder network to map the heterogeneous cross-modal attribute information into a common space, to jointly learn the representations of attribute semantics and structural semantics of different types of nodes. Obviously, our proposed method is a deep learning method that can handle high-volume data.

3 Problem definition

In this section, we formally define the problem of network representation learning in heterogeneous behavior networks. First, the heterogeneous behavior network is defined.

Definition 1 (Heterogeneous behavior network) An HBN is defined as a graph $G=(V,E,C)$, where V is the set of vertices (nodes) representing objects with a mapping function $\phi: V \to T_V \in$ {human, action}, $E \subseteq V \times V$ is the set of edges (links) with associated weight w_{ij} representing the relationship between two vertices (v_i, v_j) with a mapping function $\psi: E \to T_E \in$ {human-human, human-action, action-action}, and $C=\{(v, \text{att})|v \in V, \text{att} \in A_{t_V}\}$ denotes the attributes of objects where A_{t_V} is the attribute domain of a vertex type $t_V \in T_V$.

Our goal is to learn the latent representations in an HBN. We define this problem as follows:

Definition 2 (Heterogeneous behavior network embedding) Given an HBN denoted as $G=(V,E,C)$, the aim of HBN embedding is to learn low-dimensional embeddings $\boldsymbol{X} \in \mathbb{R}^{|V|\times d}$, where $d \ll |V|$ is the embedding dimension.

Note that HBN embedding maps different types of nodes associated with attributes into a common latent space. This kind of embedding can be regarded as a feature vector of a node and benefits various HBN mining tasks, in which potential behavior prediction is required.

4 Our approach

In this section, we present our approach, a^{4}HNE, which jointly embeds the humans and actions associated with attributes in the HBN into a common space. To learn high-quality embeddings, three important points are considered, i.e., structural proximity which suggests that the embeddings of explicitly connected nodes are similar to each other, attribute resemblance which indicates that nodes with similar attributes tend to have similar embeddings, and heterogeneity fusion by which the heterogeneous structure and the attributes are fused. Fig. 2 illustrates the flowchart of the proposed approach.

4.1 Structural proximity

To learn embeddings that preserve structural proximity among nodes in different types of edges, we measure the conditional likelihood L_{sp} of every edge (v_i, v_j) between vertices v_i and v_j as

$$L_{\text{sp}}(v_i, v_j) = w_{ij} \log p(\boldsymbol{v}_i|\boldsymbol{v}_j). \tag{1}$$

Note that we use the normal font for vertices (nodes) and bold font for vectors if there is no ambiguity.

For heterogeneous types of edges indicating human-action (E_{ha}), human-human (E_{hh}), and action-action (E_{aa}) relationships, the conditional

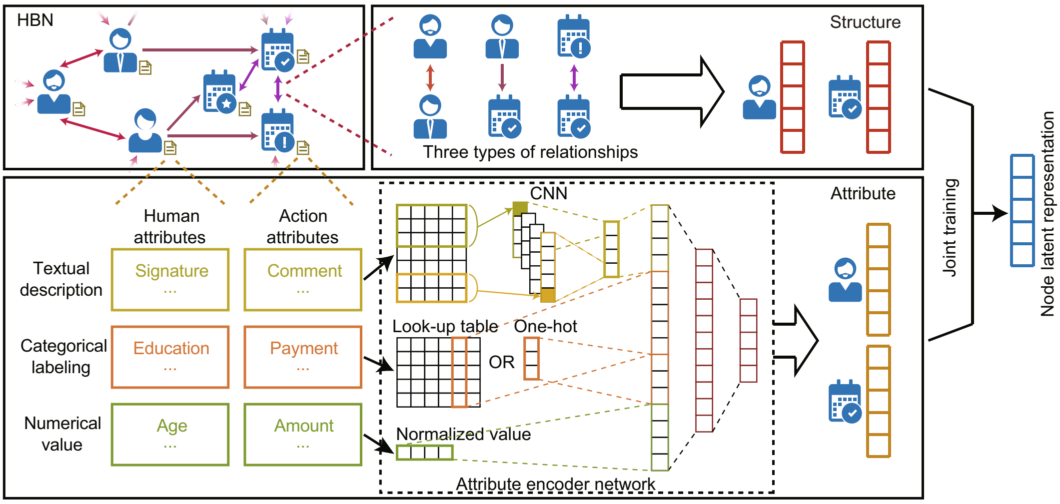

Fig. 2 Illustration flowchart of human-action-attribute-aware heterogeneous network embedding (a^{4}HNE) applied on the heterogeneous behavior network (HBN) in Fig. 1

probability of v_i generated by v_j is defined as

$$p(\boldsymbol{v}_i|\boldsymbol{v}_j) = \frac{\exp(\boldsymbol{v}_i^{\mathrm{T}} \cdot \boldsymbol{v}_j)}{\sum_{v_{i'} \in T} \exp(\boldsymbol{v}_{i'}^{\mathrm{T}} \cdot \boldsymbol{v}_j)}, \tag{2}$$

where $T = \{v_{i'}|\phi(v_{i'}) = \phi(v_i); v_{i'} \in V\}$ is the set of nodes that have the same type of v_i. Eq. (2) transforms the similarity of nodes into conditional probability through a softmax function. Because nodes v_i and v_j are more similar, the probability is higher.

4.2 Attribute resemblance

To uniformly process and use the cross-modal attributes, we aim to learn two mapping functions, $\boldsymbol{f}_{\mathrm{h}} : \mathbb{R}^m \mapsto \mathbb{R}^{d'}$ and $\boldsymbol{f}_{\mathrm{a}} : \mathbb{R}^n \mapsto \mathbb{R}^{d'}$, to map human attributes and action attributes into a common space. In this study, we present our design of an attribute encoder network to capture the attribute semantics of objects. Note that all the parameters of the network are jointly learned through gradient descent back propagation updating when optimizing the overall objective function. The attribute encoder network is described as follows:

To begin with, we categorize the attributes of nodes into three classes in general: textual, categorical, and numerical.

1. Textual attributes

For example, the natural-language description of a person is the textual attribute represented as a word sequence. We employ a convolutional neural network (CNN) to embed the textual attributes by taking the word sequence as the input. The CNN model has three layers. The first layer embeds words into word embeddings with a uniform size. The next layer performs convolutions over the vectors using multiple filter sizes. The last layer max-pools the result of the convolutional layer into a long feature vector regularized with a dropout operation.

2. Categorical attributes

The categorical attributes with a small number of categories (e.g., the gender) are embedded by one-hot encoding (Harris and Harris, 2010). However, for the attributes with a relatively large number of categories (e.g., the location), the one-hot encoding becomes too sparse; thus, we map them to dense representations that are suitable for neural networks.

3. Numerical attributes

The numerical attributes are normalized using a popular standardization that scales the values of attributes to lie between a given minimum value and a maximum value (e.g., between zero and one).

Following the results of the above steps, the attribute encoder network concatenates the potential expressions through a one-layer network. Then the concatenated vector expressions are connected to the multi-layer fully connected neural network, and the activation function is the rectified linear unit (ReLU) (Glorot et al., 2011). The final embedding $\boldsymbol{v}'$ learned by this network is treated as the attribute semantics of a vertex.

For $\boldsymbol{v}'$, we define the objective of attribute resemblance as follows:

$$L_{\mathrm{ap}}(v_i, v_j) = w_{ij} \log p(\boldsymbol{v}_i'|\boldsymbol{v}_j'). \tag{3}$$

In contrast with the structural proximity, which represents the structure information of $\boldsymbol{v}$ itself, attribute resemblance maximizes the conditional likelihood by the weight fitting of the attribute encoder network. It makes $\boldsymbol{v}'$ smoother in the attribute space; i.e., similar attributes tend to be represented as similar embeddings.

4.3 Heterogeneity fusion

To consider structure and attribute information comprehensively, the attributes of the nodes are assumed to be similar when their structural contexts are similar, and vice versa. We define the correlation objective between $\boldsymbol{v}$ and $\boldsymbol{v}'$ as follows:

$$L_{\mathrm{co}}(v_i, v_j) = w_{ij} \log p(\boldsymbol{v}_i' | \boldsymbol{v}_j) + w_{ij} \log p(\boldsymbol{v}_i | \boldsymbol{v}_j'). \tag{4}$$

With the above assumptions, we measure the conditional likelihood L of every edge (v_i, v_j) between vertices v_i and v_j as follows:

$$L(v_i, v_j) = L_{\mathrm{sp}}(v_i, v_j) + \beta_0 L_{\mathrm{ap}}(v_i, v_j) + \beta_1 L_{\mathrm{co}}(v_i, v_j), \tag{5}$$

where β_0 and β_1 are the parameters that control the weights of various parts.

For all types of edges, the overall heterogeneity fusion objective is defined as follows:

$$\mathcal{L} = \sum_{E_k} \sum_{(v_i, v_j) \in E_k} L(v_i, v_j), \tag{6}$$

where $E_k \in \{E_{\mathrm{hh}}, E_{\mathrm{ha}}, E_{\mathrm{aa}}\}$.

After jointly learning $\boldsymbol{v}$ and $\boldsymbol{v}'$, which maximizes the overall objective, we define a binary operator to obtain the final node embeddings as $\boldsymbol{x} = \boldsymbol{v} \circ \boldsymbol{v}'$. We consider several choices for the operator $\circ$, such as concatenation, average, former-only, and latter-only.

4.4 Model optimization

a^{4}HNE maximizes the conditional probabilities in $\mathcal{L}$. The optimizations of the conditional probability via softmax can be prohibitively expensive. Therefore, inspired by Mikolov et al. (2013a), we apply negative sampling to approximate the softmax likelihood function as follows:

$$\begin{aligned} \log p(\boldsymbol{v}_i | \boldsymbol{v}_j) \approx & \log \sigma(\boldsymbol{v}_i^{\mathrm{T}} \cdot \boldsymbol{v}_j) \\ & + \sum_{i=1}^{K} E_{z \sim P(v)} [\log \sigma(-\boldsymbol{v}_i^{\mathrm{T}} \cdot \boldsymbol{z})], \end{aligned} \tag{7}$$

where $\sigma(x) = 1/(1 - \exp(-x))$ is the sigmoid function, K is the number of negative samples, and z is a negative sample. $P(v) \propto d_v^{3/4}$ denotes the distribution of nodes and d_v is the out-degree of v.

The objective in Eq. (6) is jointly trained and the detailed algorithm is shown in Algorithm 1. To avoid the weight incompatibility of the edges between different types of nodes, we alternatively sample from sets of edges constructed by different relationships. The time complexity is $\mathcal{O}(|E|d)$, so a^{4}HNE is scalable.

Algorithm 1 a^{4}HNE

Input: heterogeneous behavior network $G = (V, E, C)$, number of epochs γ

Output: latent node embeddings $\boldsymbol{X} \in \mathbb{R}^{|V| \times d}$

Initialize the parameters of $\boldsymbol{f}_{\mathrm{h}} \in \mathbb{R}^{m \times d'}$, $\boldsymbol{f}_{\mathrm{a}} \in \mathbb{R}^{n \times d'}$, textual and categorical attribute lookup tables randomly
for $i = 1$ to γ **do**
  $\mathcal{O} = \mathrm{shuffle}(E)$
  for $j = 1$ to $\max(|E_{\mathrm{hh}}|, |E_{\mathrm{ha}}|, |E_{\mathrm{aa}}|)$ **do**
    for $E_k \in \{E_{\mathrm{hh}}, E_{\mathrm{ha}}, E_{\mathrm{aa}}\}$ **do**
      Sample an edge from E_k, and draw K negative edges
      Calculate the corresponding loss
      Update parameters and embeddings by stochastic gradient descent (SGD)
    end for
  end for
end for
for $\boldsymbol{v} \in V$ **do**
  Calculate $\boldsymbol{x} = \boldsymbol{v} \circ \boldsymbol{v}'$ to construct $\boldsymbol{X}$
end for

---

5 Experimental setup

5.1 Datasets

1. HIK

We construct the HBN from a real-world dataset provided by Hikvision (Hangzhou Hikvision Digital Technology Co., Ltd.). For commercial reasons, we cannot clarify the actual meaning of the real system, so we explain the objects, relationships, and attributes by mapping them into the tourism website illustrated in Section 1. The human and the action in the HBN stand for the user (i.e., the member of the website) and the travel activity, respectively. We construct the human-action relationship when a user participates in a travel activity and the human-human relationship when users follow each other on the website. The weight of the human-human edge is the communication frequency of the pair of users. There are no direct relationships between activities in the HIK dataset, so we skip the action-action relationship. The user has 10 types of attributes (gender, age, address, occupation, etc.) and the activity has 14 types of attributes (time, location, activity type, duration, initiator, etc.). The label of the tourists is treated as the ground truth of the classification tasks.

2. AMiner

We use the AMiner dataset (https://aminer.org/aminernetwork) (Tang et al., 2008) to analyze the research field of authors and papers. In the HBN, we construct the human-action relationship when authors publish papers, treat the co-author relationship as the human-human relationship, and treat the citation relationship as the action-action relationship. The author attribute information contains research interests (t), the count of published papers (pc), the total number of citations (cn), the H-index (hi), the P-index with equal A-index (pi), and the P-index with unequal A-index (upi) (Stallings et al., 2013). The paper attribute information contains the title and the abstract. The weight of the edges is binary. We choose all the papers from the most popular venue in eight research fields (*IEEE Trans. Parall. Distrib. Syst.*, STOC, *IEEE Commun. Mag.*, *ACM Trans. Graph.*, CHI, ACL, CVPR, and WWW) defined by Dong et al. (2017) and select all the relative authors who published these papers. There are 13 553 papers and 16 604 authors with labels in the filtered dataset.

The detailed statistics of these two datasets are shown in Table 1.

5.2 Compared algorithms

The proposed method a^{4}HNE and two degenerated variations, i.e., a^{4}HNE-NH (no heterogeneity) and a^{4}HNE-NC (no correlation), are compared with several recent embedding methods as follows:

Doc2Vec (Le and Mikolov, 2014) extends word2vec (Mikolov et al., 2013b) to learn document representations. We use Doc2Vec pre-trained by the English Wikipedia distributed bag of words (DBOW) model as the attribute-only method to directly learn embeddings from the textual attribute information of nodes.

DeepWalk (Perozzi et al., 2014) performs random walks over networks and learns low-dimensional node embeddings. node2vec (Grover and Leskovec, 2016) proposes a biased random walk algorithm based on DeepWalk to efficiently explore die neighborhoods. LINE (Tang et al., 2015a) uses both the first- and second-order proximities to learn node embeddings in large-scale networks. These methods can handle only homogeneous information networks without attributes. Hence, we ignore the node type and attributes in the HBN for these methods. Specifically, for node2vec, we choose its parameters as $p = 2$, $q = 0.5$ and $p = 0.5$, $q = 2$.

CANE (Tu et al., 2017a) leverages structure information and textural attribute information to learn context-aware embeddings for nodes with a mutual attention mechanism for homogeneous information networks. This method requires homogeneous information networks with textual attributes. We use the research interests of authors and the title of papers in the AMiner dataset as the textual attributes.

PTE (Tang et al., 2015b) extends LINE to learn predictive text embedding through heterogeneous text networks. Metapath2vec++ (Dong et al., 2017) is a meta-path based random walk algorithm with a heterogeneous skip-gram model to perform node embeddings. They are typical heterogeneous network embedding methods that do not consider attributes. Specifically, for PTE, we construct several bipartite networks (co-author, author2paper, and citation for

Table 1 Statistics of datasets

Dataset	Node type	Number of labels	Number of attribute types	$\|V\|$	$\|E\|$	$\|E_{\mathrm{hh}}\|$	$\|E_{\mathrm{ha}}\|$	$\|E_{\mathrm{aa}}\|$
HIK	Human	2	10	4863	21 191	5807	15 384	–
	Action	–	14	11 723				
AMiner	Human	8	6	16 604	124 776	62 115	31 263	31 398
	Action	8	2	13 553				

$|V|$: number of vertices; $|E|$: number of edges; $|E_{\mathrm{hh}}|$: number of human-human edges; $|E_{\mathrm{ha}}|$: number of human-action edges; $|E_{\mathrm{aa}}|$: number of action-action edges

AMiner; user-user and user-activity for HIK) and restrain it as an unsupervised embedding method.

a^{4}HNE-NH is a variation of the proposed a^{4}HNE that does not consider heterogeneity. We generate homogeneous sub-networks by just preserving the same type of attributes for nodes. For the HIK dataset, we use only the categorical and numerical attributes of users and remove the activities. For the AMiner dataset, we use the same homogeneous sub-network as CANE.

a^{4}HNE-NC is a variation of the proposed a^{4}HNE. We set $\beta_1 = 0$ so that the correlation of structure and attribute is not considered.

5.3 Parameter settings

For fair comparison, we try different parameters for all embedding methods and report the best performance in experiments. The parameter settings are as follows: The dimension of textual embedding is set to 300, as used in Doc2Vec embedding. For node embeddings, the dimension is set as 128 by default in the AMiner dataset and 32 in the HIK dataset. The other default settings include: walk length $l = 80$, number of walks per node $w = 10$ for random walk based methods; hyper-parameters $\alpha = 0.7, \beta = 0.1, \gamma = 0.1$ in CANE and the same corresponding weights in a^{4}HNE-NH; number of negative samples $K = 1$, as used in CANE; learning rate $r_1 = 0.001$; operator "$\circ$" is a concatenation. Fianlly, all the embedding vectors are normalized.

6 Quantitative analysis

6.1 Link prediction

In link prediction, we conduct experiments on the two datasets keeping α (from 30% to 90%) edges chosen randomly from each type of relationship and then predict the missing edges. Note that there are very few edges when $\alpha \leq 30\%$, resulting in the cold-start problem which will be explained in Section 6.4. Table 2 shows the performance of link prediction with the AUC (Hanley and McNeil, 1982) values.

From this table, we have the following observations:

1. a^{4}HNE consistently achieves significant improvement compared to other methods. This indicates that a^{4}HNE can capture the semantics of nodes and is effective in link prediction.

Table 2 Results of link prediction on the AMiner and HIK datasets

Method	Results of link prediction on AMiner			
	α=30%	α=50%	α=70%	α=90%
Doc2Vec	0.6211	0.6199	0.6185	0.6112
DeepWalk	0.8966	0.9484	0.9660	0.9732
node2vec (p=2, q=0.5)	0.8948	0.9465	0.9664	0.9736
node2vec (p=0.5, q=2)	0.8947	0.9479	0.9665	0.9746
LINE	0.8976	0.9492	0.9620	0.9668
CANE	0.9065	0.9345	0.9439	0.9446
PTE	0.7429	0.8917	0.9444	0.9614
Metapath2vec++	0.8133	0.8610	0.8813	0.8870
a^{4}HNE-NH	0.9357	0.9598	0.9684	0.9718
a^{4}HNE-NC	0.9008	0.9580	0.9705	0.9741
a^{4}HNE	**0.9383**	**0.9659**	**0.9730**	**0.9747**

Method	Results of link prediction on HIK			
	α=30%	α=50%	α=70%	α=90%
DeepWalk	0.6004	0.6581	0.7336	0.7690
node2vec (p=2, q=0.5)	0.5995	0.6786	0.7363	0.7740
node2vec (p=0.5, q=2)	0.5963	0.6709	0.7220	0.7575
LINE	0.5952	0.6640	0.7157	0.7583
PTE	0.5885	0.6572	0.7148	0.7592
Metapath2vec++	0.5043	0.5401	0.5391	0.5459
a^{4}HNE-NC	0.6843	0.7089	0.7220	0.7960
a^{4}HNE	**0.8048**	**0.8335**	**0.8465**	**0.8645**

The best results are in bold

2. With the increasing number of missing edges, i.e., the decreasing α, a^{4}HNE achieves more improvements than other methods. The possible reason is that a^{4}HNE can remedy the insufficiency of structure information using the attributes.

3. Specifically, for the HIK dataset, a^{4}HNE outperforms the best of the other methods by 11.7%–34.0% in terms of AUC. It is perhaps because the attribute encoder network could exploit rich cross-modal attributes, which is difficult when used in other methods.

4. The correlation of the structure and attribute is important and is derived from the good performance of a^{4}HNE over a^{4}HNE-NC with a relative gain of up to 17.6% on the HIK dataset.

6.2 Node classification

We evaluate the network embeddings in the classification task. A logistic regression classifier fed by the embeddings of all labeled nodes is employed. We vary the training ratio (T_R) on the two datasets from 30% to 90% and the rest of nodes are used for testing. For each T_{R}, the classification experiments are repeated independently 10 times. The averaged micro-F_1 measures are reported in Table 3.

Table 3 Results of classification on the HIK and AMiner datasets

Dataset	Method	Results of classification			
		30%	50%	70%	90%
HIK (user)	DeepWalk	0.6026	0.6075	0.6085	0.6119
	node2vec (p=2, q=0.5)	0.6146	0.6281	0.6321	0.6194
	node2vec (p=0.5, q=2)	0.5858	0.5979	0.5982	0.6071
	LINE	0.5849	0.5971	0.5915	0.5952
	PTE	0.6015	0.6069	0.6097	0.6125
	Metapath2vec++	0.6063	0.6246	0.6347	0.6310
	a^{4}HNE-NH	0.6331	0.6392	0.6397	0.6406
	a^{4}HNE-NC	0.6071	0.6082	0.6252	0.6440
	a^{4}HNE	**0.6534**	**0.6654**	**0.6715**	**0.6833**
AMiner (author)	Doc2Vec	0.6507	0.6674	0.6787	0.6785
	DeepWalk	0.7251	0.7316	0.7338	0.7378
	node2vec (p=2, q=0.5)	0.7243	0.7312	0.7312	0.7307
	node2vec (p=0.5, q=2)	0.7237	0.7320	0.7334	0.7344
	LINE	0.7285	0.7314	0.7347	0.7371
	CANE	0.7871	0.7920	0.7933	0.7971
	PTE	0.7670	0.7712	0.7749	0.7799
	Metapath2vec++	0.7589	0.7653	0.7676	0.7655
	a^{4}HNE-NH	0.7674	0.7736	0.7746	0.7802
	a^{4}HNE-NC	0.7777	0.7846	0.7876	0.7868
	a^{4}HNE	**0.7958**	**0.7968**	**0.8027**	**0.7996**
AMiner (paper)	Doc2Vec	0.6652	0.6811	0.6938	0.6939
	DeepWalk	0.8956	0.9003	0.9016	0.9015
	node2vec (p=2, q=0.5)	0.8986	0.9013	0.9043	0.9089
	node2vec (p=0.5, q=2)	0.8981	0.9023	0.9034	0.9061
	LINE	0.8826	0.8869	0.8901	0.8914
	CANE	0.9206	0.9250	0.9249	0.9268
	PTE	0.9342	0.9372	0.9390	0.9411
	Metapath2vec++	0.9349	0.9386	0.9402	0.9389
	a^{4}HNE-NH	0.9147	0.9185	0.9206	0.9242
	a^{4}HNE-NC	0.9155	0.9201	0.9231	0.9276
	a^{4}HNE	**0.9434**	**0.9462**	**0.9469**	**0.9502**

The best results are in bold

A general observation from the results is that a^{4}HNE achieves the best performance on the two datasets for classification. The constant gain achieved by a^{4}HNE over a^{4}HNE-NH is around 2.4%–3.7% for authors and around 2.8%–3.1% for papers on the AMiner dataset. Furthermore, we can see the support for the attributes and heterogeneity. On one hand, a^{4}HNE-NH already outperforms all the baselines on HIK due to the importance of the attributes; on the other hand, a^{4}HNE gains further improvement over a^{4}HNE-NH on both datasets by leveraging heterogeneity. Also, the significance of correlation is backed up by a^{4}HNE to a^{4}HNE-NC with a relative improvement of at least 2.3% for authors and 3.1% for papers.

6.3 Node clustering

The latent representations learned by embedding methods can also be used for the node clustering task. The learned embeddings are input in the k-means algorithm to cluster the nodes and the results are evaluated by the same label used for the classification task in terms of normalized mutual information (NMI) (Sun et al., 2011). All clustering experiments are conducted 10 times and the average performance is reported.

Table 4 shows the node clustering results as measured by NMI on both the HIK and AMiner datasets. The proposed method has significantly more impact on clustering than other methods. With the combination of heterogeneity and cross-modal attributes, a^{4}HNE has a more encouraging improvement than the simplified a^{4}HNE-NH and a^{4}HNE-NC, which agrees with the classification results.

Table 4 Results of clustering on the HIK and AMiner datasets

Method	NMI value for clustering		
	HIK (user)	AMiner (author)	AMiner (paper)
Doc2Vec	–	0.2977	0.2251
DeepWalk	0.0192	0.3791	0.6001
node2vec (p=2, q=0.5)	0.0159	0.3852	0.6238
node2vec (p=0.5, q=2)	0.0184	0.3783	0.6245
LINE	0.0006	0.3792	0.5275
CANE	–	0.4726	0.6684
PTE	0.0194	0.4358	0.6769
Metapath2vec++	0.0301	0.4514	0.7014
a^{4}HNE-NH	0.0352	0.4564	0.6493
a^{4}HNE-NC	0.0312	0.4641	0.6921
a^{4}HNE	**0.0379**	**0.4992**	**0.7203**

The best results are in bold

6.4 Cold-start problem

The cold-start problem occurs when there are few edges between nodes in networks. We study the performance by varying cold-start degrees and repeating the link prediction experiments on the

AMiner dataset, setting α from 5% to 25%. Because CANE cannot be applied on the HIK dataset, we conduct cold-start analysis only on the AMiner dataset to show the promotion by the attribute. The results are shown in Fig. 3. Specifically, with the same α, the results of DeepWalk, node2vec, and LINE do not yield significant differences. For convenience, we show only the DeepWalk results for comparison.

As shown in Fig. 3, both a^{4}HNE and CANE gain greater improvements over the structure-only methods for lower α. So, in the cold-start scenario, the attribute information is valuable for remedying the insufficiency of the network structure information. In particular, a^{4}HNE achieves the best results when $\alpha \geq 15\%$ but CANE performs better when $\alpha = 5\%$ or 10%. We believe the reason is that the a^{4}HNE attribute encoder network for cross-modal attributes is more complex than the CNN of CANE for only textual attributes. Thus, the attribute encoder network does not fully converge when sampling on the small edge set. It is also demonstrated by the higher performance improvement rate of a^{4}HNE versus CANE with the increase of α.

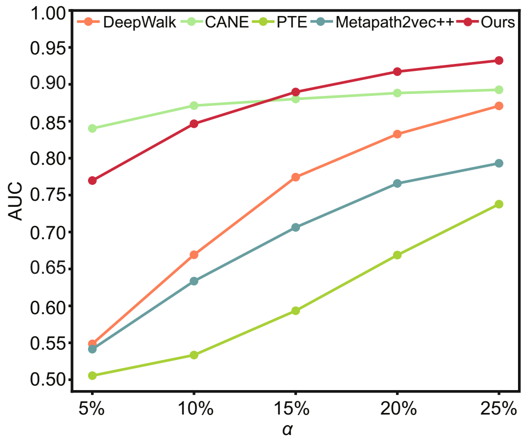

Fig. 3 Results of cold-start prediction on the AMiner dataset
References to color refer to the online version of this figure

7 Qualitative analysis via visualization

Visualization, which shows the layout of a network on a two-dimensional (2D) space, can provide some intuition about the performance of network embeddings. We project the embeddings learned by different methods to a 2D space using the t-SNE package (van der Maaten and Hinton, 2008) for authors and papers in the AMiner dataset (Fig. 4). We show the visualization of a^{4}HNE and two compared algorithms with better results than baselines on the classification task, i.e., PTE and CANE.

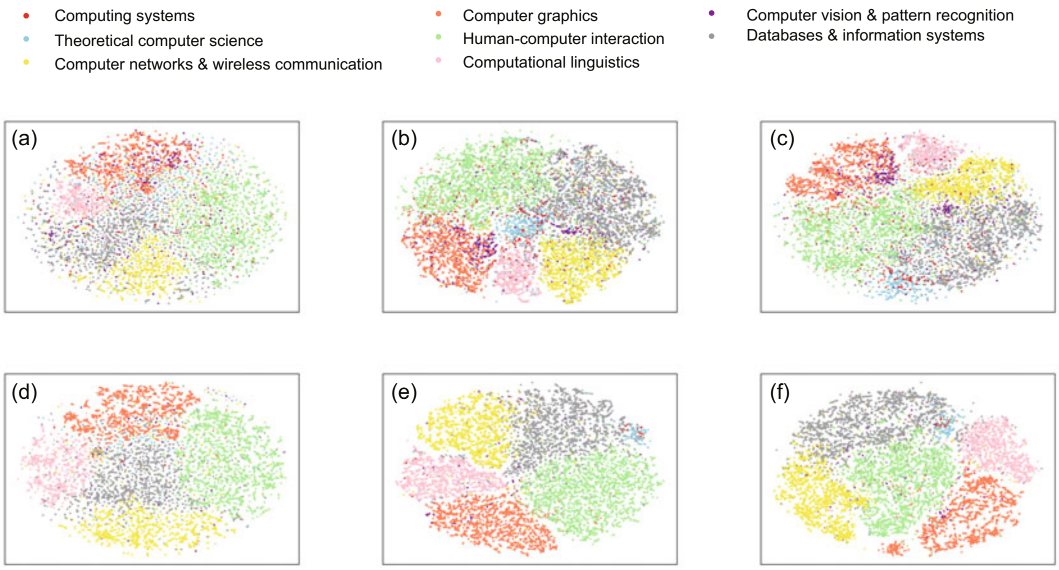

Fig. 4 Visualization results on the AMiner dataset. PTE (a), CANE (b), and a^{4}HNE (c) are results for authors; PTE (d), CANE (e), and a^{4}HNE (f) are results for papers. References to color refer to the online version of this figure

For the author results on the AMiner dataset in Figs. 4a–4c, PTE can basically identify human-computer interaction (green), computer networks & wireless communication (yellow), computational linguistics (pink), databases & information systems (gray), and computer graphics (coral). However, it mixes the computing systems (red), theoretical computer science (blue), and computer vision & pattern recognition (purple). CANE and a^{4}HNE perform better because the nodes with the same colors are distributed more closely, especially for blue and purple points. Moreover, compared to CANE which maps the blue points to the center area and thus intersects with all the other areas, a^{4}HNE distinguishes the blue points from coral, pink, and yellow points, satisfying the real scenarios that theoretical computer science has little intersection with the three groups (computer graphics, computational linguistics, and computer networks & wireless communication). Figs. 4d–4f show that the visualization of papers on the AMiner dataset leads to similar results. Particularly, we can see that the boundaries of all the groups labeled by a^{4}HNE (Fig. 4f) are clearer than those of CANE (Fig. 4e). Thus, a^{4}HNE's ability to learn meaningful embeddings for real-world networks is intuitively demonstrated.

8 Parameter sensitivity

In this section, we investigate the performance of three mining tasks with regard to parameter dimension d on both HIK and AMiner datasets. In Fig. 5, we can see that the performance remains stable, only slightly influenced by the number of attribute types. For the small number of attribute types on the AMiner dataset, excessively large dimension causes a little performance loss. The optimum results can be achieved when $d = 256$ on the HIK dataset and $d = 128$ on the AMiner dataset.

We also analyze the two hyper parameters (β_0 and β_1) of our model, where β_0 is the weight of attribute resemblance in Eq. (3), and β_1 denotes the weight of correlation in Eq. (4). Table 5 shows the performance of classification on the HIK dataset as the other results have similar trends. We can see that choosing $\beta_1/\beta_0 = 10$ can achieve a better performance of a^{4}HNE on the HIK dataset.

9 Discussion

9.1 Importance of attribute information

Experiments and applications of a^{4}HNE certainly prove that the attribute information is significant for performance improvement of network embedding. Specifically, for an information network with insufficient structure, i.e., implicit edge semantics or few edges, the effect of attribute information cannot be ignored. Specifically, it involves three aspects:

1. As the experiments show, the boost brought by a^{4}HNE on the AMiner dataset is less than that on the HIK dataset. By a careful investigation, we think that the relationships in the AMiner dataset can be used to predict behaviors directly, which

Table 5 Hyper-parameter analysis on the HIK dataset ($T_R = 50\%$)

Weight	Results of hyper-parameter analysis			
	β_1=0.01	β_1=0.1	β_1=1	β_1=10
β_0=0.01	0.6354	**0.6490**	0.6420	0.6375
β_0=0.1	0.6470	0.6404	**0.6654**	0.6553
β_0=1	0.6172	0.6473	0.6504	**0.6561**
β_0=10	0.6175	0.6257	0.6383	0.6471

The best results are in bold

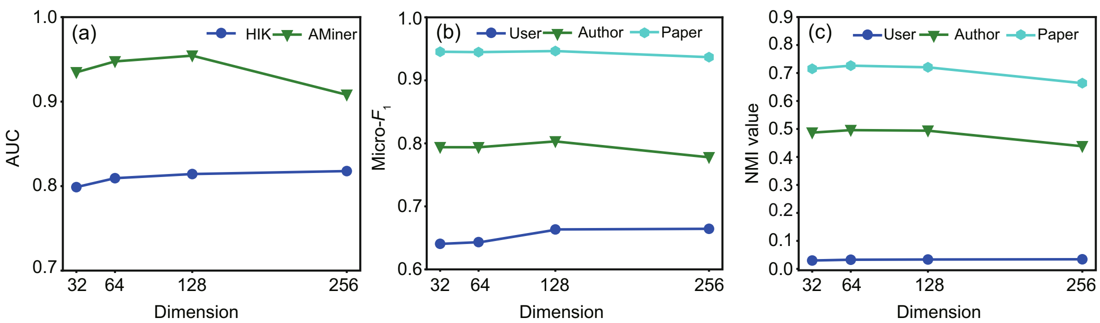

Fig. 5 Parameter sensitivity study: (a) link prediction, $\alpha = 50\%$; (b) classification, T_R=50%; (c) clustering

can be handled by all baseline methods. However, the edge in the HIK dataset has no such explicit semantics. For example, two authors linked by a co-author edge in the AMiner dataset, meaning that the authors have collaborated on a paper, they may have the same research field. Users in the HIK dataset who follow each other on the website does not mean that they have the same travel interests. The implicit attributes like age and occupation play an important role in predicting behaviors.

2. In the cold-start scenario in which the information network contains few edges, both a^{4}HNE and CANE, which consider the attribute information, notably outperform the structure-only methods.

3. The attribute information can be used to directly generate the representation of the "new comer." We conduct a case study to show this issue. We randomly select some nodes in the network and remove all their edges to eliminate the structure information. Then we put their attributes into the trained attribute encoder network and use the latter-only operator $\circ$ to generate their embeddings. As a result, these nodes can be classified into the correct group by the trained logistic regression classifier. An example is shown in Fig. 6. A new author C without edges is classified in the same research field as A and B (human-computer interaction) based on similar attributes.

In conclusion, we suggest that researchers should consider more about the attribute information when constructing information networks besides the HBN for human potential behavior prediction.

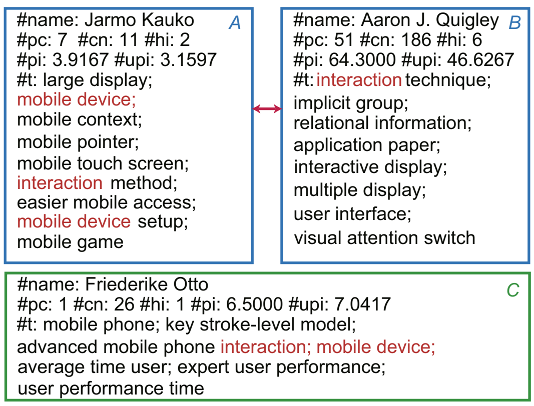

Fig. 6 An example of using only attribute information to generate the representation of a "new comer." There exists a co-author relationship between author *A* and author *B*. Author *C* is a new author who is not in the training dataset

9.2 Downstream applications

In previous sections, we explained applications of the proposed a^{4}HNE by two real normal human behavior social systems, i.e., the tourism website and the academic network. As a matter of fact, in some scenarios of bad behaviors, the action is more difficult to observe due to individuals' deliberate concealment. Thus, the structure information in their HBN is more insufficient and a^{4}HNE can play a greater role (the HIK dataset actually involves one kind of bad behavior).

For example, it is crucial for an unsecured micro-credit loan platform to know whether a user will repay a loan. The registered user of the platform can be treated as human in the HBN and the action stands for repaying a loan. The profile of a user includes gender, age, education, income, etc., and the attributes of a loan include the amount of money, currency, deadline, interest, etc. The relationship of two users is derived from the communication and intersection between them, like the call logs. However, the communication and intersection information is not easily accessible for the platform. So, a^{4}HNE is a better choice to make most of the attributes.

Moreover, a^{4}HNE is a network embedding method and the representation generated by it can be used in various downstream applications, such as recommendation and anomaly detection in social computing.

10 Conclusions and future work

We have studied the problem of using social information to predict potential human behavior, i.e., predicting potential individuals who may take specific actions (link prediction) and further categorizing them into specific groups (classification or clustering). To this end, we have first constructed the HBN from mass humans and actions information, and then proposed a^{4}HNE, a novel network embedding method that handles the HBN. a^{4}HNE embeds nodes in the HBN into a common space that makes full use of all the relationships, attributes, and types of nodes. Extensive experiments demonstrated that a^{4}HNE is effective and robust for various heterogeneous information network mining tasks for human potential behavior prediction. In future work, we plan to generalize our model for more sophisticated heterogeneous information networks in which there exist more than one link indicating a variety of relationships between node pairs.

Compliance with ethics guidelines

Yue-yang WANG, Wei-hao JIANG, Shi-liang PU, and Yue-ting ZHUANG declare that they have no conflict of interest.

References

Backstrom L, Leskovec J, 2011. Supervised random walks: predicting and recommending links in social networks. Proc 4th ACM Int Conf on Web Search and Data Mining, p.635-644.

Bhagat S, Cormode G, Muthukrishnan S, 2011. Node classification in social networks. In: Aggarwal C (Ed.), Social Network Data Analytics. Springer, Boston, p.115-148. https://doi.org/10.1007/978-1-4419-8462-3_5

Chang SY, Han W, Tang JL, et al., 2015. Heterogeneous network embedding via deep architectures. Proc 21st ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining, p.119-128. https://doi.org/10.1145/2783258.2783296

Chen T, Sun YZ, 2017. Task-guided and path-augmented heterogeneous network embedding for author identification. Proc 10th ACM Int Conf on Web Search and Data Mining, p.295-304. https://doi.org/10.1145/3018661.3018735

Chen YX, Wang CG, 2017. HINE: heterogeneous information network embedding. Int Conf on Database Systems for Advanced Applications, p.180-195. https://doi.org/10.1007/978-3-319-55753-3_12

Ding CHQ, He XF, Zha HY, et al., 2001. A min-max cut algorithm for graph partitioning and data clustering. Proc IEEE Int Conf on Data Mining, p.107-114.

Dong YX, Chawla N, Swami A, 2017. metapath2vec: scalable representation learning for heterogeneous networks. Proc 23rd ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining, p.135-144. https://doi.org/10.1145/3097983.3098036

Glorot X, Bordes A, Bengio Y, 2011. Deep sparse rectifier neural networks. Proc 14th Int Conf on Artificial Intelligence and Statistics, p.315-323.

Grover A, Leskovec J, 2016. node2vec: scalable feature learning for networks. Proc 22nd ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining, p.855-864. https://doi.org/10.1145/2939672.2939754

Hanley JA, McNeil BJ, 1982. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, 143(1):29-36. https://doi.org/10.1148/radiology.143.1.7063747

Harris DM, Harris S, 2010. Digital Design and Computer Architecture. Morgan Kaufmann, Amsterdam.

Huang ZP, Mamoulis N, 2017. Heterogeneous information network embedding for meta path based proximity. https://arxiv.org/abs/1701.05291

Koren Y, 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model. Proc 14th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining, p.426-434. https://doi.org/10.1145/1401890.1401944

Le Q, Mikolov T, 2014. Distributed representations of sentences and documents. Proc 31st Int Conf on Machine Learning, p.1188-1196.

Lerman K, Intagorn S, Kang JH, et al., 2012. Using proximity to predict activity in social networks. Proc 21st Int Conf on World Wide Web, p.555-556. https://doi.org/10.1145/2187980.2188124

Liben-Nowell D, Kleinberg J, 2007. The link-prediction problem for social networks. *J Am Soc Inform Sci Technol*, 58(7):1019-1031. https://doi.org/10.1002/asi.20591

Ma H, Zhou DY, Liu C, et al., 2011. Recommender systems with social regularization. Proc 4th ACM Int Conf on Web Search and Data Mining, p.287-296. https://doi.org/10.1145/1935826.1935877

Mikolov T, Sutskever I, Chen K, et al., 2013a. Distributed representations of words and phrases and their compositionality. Proc 26th Int Conf on Neural Information Processing Systems, p.3111-3119.

Mikolov T, Chen K, Corrado G, et al., 2013b. Efficient estimation of word representations in vector space. https://arxiv.org/abs/1301.3781

Ou MD, Cui P, Pei J, et al., 2016. Asymmetric transitivity preserving graph embedding. Proc 22nd ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining, p.1105-1114. https://doi.org/10.1145/2939672.2939751

Pan SR, Wu J, Zhu XQ, et al., 2016. Tri-party deep network representation. Proc 25th Int Joint Conf on Artificial Intelligence, p.1895-1901.

Perozzi B, Al-Rfou R, Skiena S, 2014. DeepWalk: online learning of social representations. Proc 20th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining, p.701-710. https://doi.org/10.1145/2623330.2623732

Ribeiro LFR, Saverese PHP, Figueiredo DR, 2017. struc2vec: learning node representations from structural identity. Proc 23rd ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining, p.385-394. https://doi.org/10.1145/3097983.3098061

Sen P, Namata G, Bilgic M, et al., 2008. Collective classification in network data. *AI Mag*, 29(3):93-106.

Shi C, Hu BB, Zhao WX, et al., 2019. Heterogeneous information network embedding for recommendation. *IEEE Trans Knowl Data Eng*, 31(2):357-370. https://doi.org/10.1109/TKDE.2018.2833443

Stallings J, Vance E, Yang JS, et al., 2013. Determining scientific impact using a collaboration index. *PNAS*, 110(24):9680-9685. https://doi.org/10.1073/pnas.1220184110

Sun XF, Guo J, Ding X, et al., 2016. A general framework for content-enhanced network representation learning. https://arxiv.org/abs/1610.02906

Sun Y, Han J, Yan X, et al., 2011. PathSim: meta path-based top-k similarity search in heterogeneous information networks. *Proc VLDB Endowm*, 4(11):992-1003.

Tang J, Zhang J, Yao LM, et al., 2008. ArnetMiner: extraction and mining of academic social networks. Proc 14th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining, p.990-998. https://doi.org/10.1145/1401890.1402008

Tang J, Qu M, Wang MZ, et al., 2015a. LINE: large-scale information network embedding. Proc 24th Int Conf on World Wide Web, p.1067-1077.
https://doi.org/10.1145/2736277.2741093

Tang J, Qu M, Mei QZ, 2015b. PTE: predictive text embedding through large-scale heterogeneous text networks. Proc 21st ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining, p.1165-1174.
https://doi.org/10.1145/2783258.2783307

Tang L, Liu H, 2009. Relational learning via latent social dimensions. Proc 15th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining, p.817-826.
https://doi.org/10.1145/1557019.1557109

Tu C, Liu H, Liu Z, et al., 2017a. CANE: context-aware network embedding for relation modeling. Proc 55th Annual Meeting of the Association for Computational Linguistics, p.1722-1731.

Tu C, Zhang Z, Liu Z, et al., 2017b. TransNet: translation-based network representation learning for social relation extraction. Proc 26th Int Joint Conf on Artificial Intelligence, p.2864-2870.

van der Maaten L, Hinton G, 2008. Visualizing data using t-SNE. *J Mach Learn Res*, 9:2579-2605.

Vazquez A, Flammini A, Maritan A, et al., 2003. Global protein function prediction from protein-protein interaction networks. *Nat Biotechnol*, 21:697-700.
https://doi.org/10.1038/nbt825

Yang C, Liu Z, Zhao D, et al., 2015. Network representation learning with rich text information. Proc 24th Int Conf on Artificial Intelligence, p.2111-2117.

Yang C, Sun M, Liu Z, et al., 2017. Fast network embedding enhancement via high order proximity approximation. Proc 26th Int Joint Conf on Artificial Intelligence, p.3894-3900.

Yin HZ, Hu ZT, Zhou XF, et al., 2016. Discovering interpretable geo-social communities for user behavior prediction. IEEE 32nd Int Conf on Data Engineering, p.942-953.
https://doi.org/10.1109/ICDE.2016.7498303

Zhang CX, Swami A, Chawla NV, 2018. CARL: content-aware representation learning for heterogeneous networks. https://arxiv.org/abs/1805.04983