Relation Structure-Aware Heterogeneous Information Network Embedding

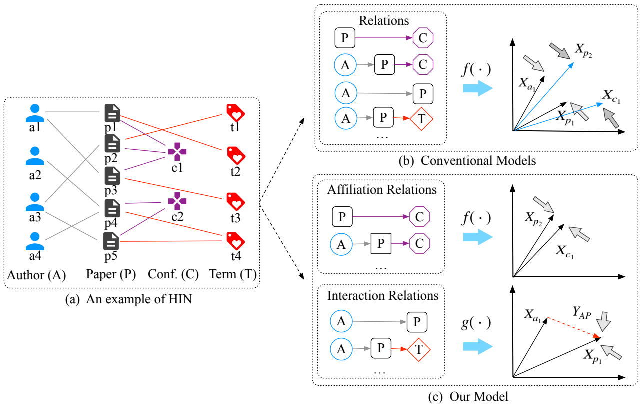
Yuanfu Lu,[1] Chuan Shi,[1*] Linmei Hu,[1] Zhiyuan Liu[2]
[1]Beijing University of Posts and Telecommunications, Beijing, China
[2]Tsinghua University, Beijing, China
{luyuanfu, shichuan, hulinmei}@bupt.edu.cn, liuzy@tsinghua.edu.cn

Abstract

Heterogeneous information network (HIN) embedding aims to embed multiple types of nodes into a low-dimensional space. Although most existing HIN embedding methods consider heterogeneous relations in HINs, they usually employ one single model for all relations without distinction, which inevitably restricts the capability of network embedding. In this paper, we take the structural characteristics of heterogeneous relations into consideration and propose a novel Relation structure-aware Heterogeneous Information Network Embedding model (RHINE). By exploring the real-world networks with thorough mathematical analysis, we present two structure-related measures which can consistently distinguish heterogeneous relations into two categories: Affiliation Relations (ARs) and Interaction Relations (IRs). To respect the distinctive characteristics of relations, in our RHINE, we propose different models specifically tailored to handle ARs and IRs, which can better capture the structures and semantics of the networks. At last, we combine and optimize these models in a unified and elegant manner. Extensive experiments on three real-world datasets demonstrate that our model significantly outperforms the state-of-the-art methods in various tasks, including node clustering, link prediction, and node classification.

1 Introduction

Network embedding has shed a light on the analysis of networks as it is effective to learn the latent features that encode the properties of a network (Cui et al. 2018; Cai, Zheng, and Chang 2018). Although the state-of-the-arts (Perozzi, Al-Rfou, and Skiena 2014; Grover and Leskovec 2016; Tang et al. 2015; Wang, Cui, and Zhu 2016) have achieved promising performance in many data mining tasks, most of them focus on homogeneous networks, which only contain one single type of nodes and edges. In reality, many networks are usually with multiple types of nodes and edges, widely known as heterogeneous information networks (HINs) (Sun et al. 2011; Shi et al. 2017). Taking the DBLP network for example, as shown in Figure 1(a), it contains four types of nodes: Author (A), Paper (P), Conference (C) and Term (T), and multiple types of relations: writing/written relations, and publish/published relations, etc. In addition, there are composite relations represented by meta-paths (Sun et al. 2011) such as *APA* (co-author relation) and *APC* (authors write papers published in conferences), which are widely used to exploit rich semantics in HINs. Thus, compared to homogeneous networks, HINs fuse more information and contain richer semantics. Directly applying traditional homogeneous models to embed HINs will inevitably lead to reduced performance in downstream tasks.



(a) An example of HIN

(b) Conventional Models

(c) Our Model

Figure 1: The illustration of an HIN and the comparison between conventional methods and our method (non-differentiated relations v.s. differentiated relations).

To model the heterogeneity of networks, several attempts have been done on HIN embedding. For example, some models employ meta-path based random walk to generate node sequences for optimizing the similarity between nodes (Shang et al. 2016; Dong, Chawla, and Swami 2017; Fu, Lee, and Lei 2017). Some methods decompose the HIN into simple networks and then optimize the proximity between nodes in each sub-network (Tang, Qu, and Mei 2015; Xu et al. 2017; Shi et al. 2018). There are also some neural network based methods that learn non-linear mapping functions for HIN embedding (Chang et al. 2015; Wang et al. 2018; Han et al. 2018). Although these methods consider the heterogeneity of networks, they usually have an assumption that one single model can handle all relations and nodes, through keeping the representations of two nodes

[*] Corresponding author: Chuan Shi (shichuan@bupt.edu.cn).

close to each other, as illustrated in Figure 1(b).

However, various relations in an HIN have significantly different structural characteristics, which should be handled with different models. Let's see a toy example in Figure 1(a). The relations in the network include atomic relations (e.g., *AP* and *PC*) and composite relations (e.g., *APA* and *APC*). Intuitively, *AP* relation and *PC* relation reveal rather different characteristics in structure. That is, some authors write some papers in the *AP* relation, which shows a peer-to-peer structure. While that many papers are published in one conference in the *PC* relation reveals the structure of one-centered-by-another. Similarly, *APA* and *APC* indicate peer-to-peer and one-centered-by-another structures respectively. The intuitive examples clearly illustrate that relations in an HIN indeed have different structural characteristics.

It is non-trivial to consider different structural characteristics of relations for HIN embedding, due to the following challenges: (1) How to distinguish the structural characteristics of relations in an HIN? Various relations (atomic relations or meta-paths) with different structures are involved in an HIN. Quantitative and explainable criteria are desired to explore the structural characteristics of relations and distinguish them. (2) How to capture the distinctive structural characteristics of different categories of relations? Since the various relations have different structures, modeling them with one single model may lead to some loss of information. We need to specifically design appropriate models which are able to capture their distinctive characteristics. (3) The different models for the differentiated relations should be easily and smoothly combined to ensure simple optimization in a unified manner.

In this paper, we present a novel model for HIN embedding, named **R**elation structure-aware **HIN E**mbedding (**RHINE**). In specific, we first explore the structural characteristics of relations in HINs with thorough mathematical analysis, and present two structure-related measures which can consistently distinguish the various relations into two categories: Affiliation Relations (ARs) with one-centered-by-another structures and Interaction Relations (IRs) with peer-to-peer structures. In order to capture the distinctive structural characteristics of the relations, we then propose two specifically designed models. For ARs where the nodes share similar properties (Yang and Leskovec 2012), we calculate Euclidean distance as the proximity between nodes, so as to make the nodes directly close in the low-dimensional space. On the other hand, for IRs which bridge two compatible nodes, we model them as translations between the nodes. Since the two models are consistent in terms of mathematical form, they can be optimized in a unified and elegant way.

It is worthwhile to highlight our contributions as follows:

- To the best of our knowledge, we make the first attempt to explore the different structural characteristics of relations in HINs and present two structure-related criteria which can consistently distinguish heterogeneous relations into ARs and IRs.
- We propose a novel relation structure-aware HIN embedding model (RHINE), which fully respects the distinctive structural characteristics of ARs and IRs by exploiting appropriate models and combining them in a unified and elegant manner.
- We conduct comprehensive experiments to evaluate the performance of our model. Experimental results demonstrate that our model significantly outperforms state-of-the-art network embedding models in various tasks.

2 Related Work

Recently, network embedding has attracted considerable attention. Inspired by word2vec (Mikolov et al. 2013), random walk based methods (Perozzi, Al-Rfou, and Skiena 2014; Grover and Leskovec 2016) have been proposed to learn representations of networks by the skip-gram model. After that, several models are designed to better preserve network properties (Tang et al. 2015; Ou et al. 2016; Ribeiro, Saverese, and Figueiredo 2017). Besides, there are some deep neural network based models for network embedding (Wang, Cui, and Zhu 2016; Cao, Lu, and Xu 2016). However, all the aforementioned methods focus only on learning the representations of homogeneous networks.

Different from homogeneous networks, HINs consist of multiple types of nodes and edges. Several attempts have been done on HIN embedding and achieved promising performance in various tasks (Tang, Qu, and Mei 2015; Shang et al. 2016; Fu, Lee, and Lei 2017; Wang et al. 2018; Shi et al. 2018). PTE (Tang, Qu, and Mei 2015) decomposes an HIN to a set of bipartite networks and then performs network embedding individually. ESim (Shang et al. 2016) utilizes user-defined meta-paths as guidance to learn node embeddings. Metapath2vec (Dong, Chawla, and Swami 2017) combines meta-path based random walks and skip-gram model for HIN embedding. HIN2Vec (Fu, Lee, and Lei 2017) learns the embeddings of HINs by conducting multiple prediction training tasks jointly. HERec (Shi et al. 2018) filters the node sequences with type constraints and thus captures the semantics of HINs.

All the above-mentioned models deal with all relations in HINs with one single model, neglecting differentiated structures of relations. In this paper, we explore and distinguish the structural characteristics of relations with quantitative analysis. For relations with distinct structural characteristics, we propose to handle them with specifically designed models.

3 Preliminaries

In this section, we introduce some basic concepts and formalize the problem of HIN embedding.

Definition 1. *Heterogeneous Information Network (HIN).* An HIN is defined as a graph $G = (V, E, T, \phi, \varphi)$, in which V and E are the sets of nodes and edges, respectively. Each node v and edge e are associated with their type mapping functions $\phi : V \rightarrow T_V$ and $\varphi : E \rightarrow T_E$, respectively. T_V and T_E denote the sets of node and edge types, where $|T_V| + |T_E| > 2$, and $T = T_V \cup T_E$.

Definition 2. *Meta-path.* A meta-path $m \in M$ is defined as a sequence of node types t_{v_i} or edge types t_{e_j} in the form

Table 1: Statistics of the three datasets. t_u denotes the type of node u, $\langle u, r, v\rangle$ is a node-relation triple.

Datasets	Nodes	Number of Nodes	Relations ($t_u \sim t_v$)	Number of Relations	Avg. Degree of t_u	Avg. Degree of t_v	Measures $D(r)$	Measures $S(r)$	Relation Category
DBLP	Term (T)	8,811	PC	14,376	1.0	718.8	718.8	0.05	AR
	Paper (P)	14,376	APC	24,495	2.9	2089.7	720.6	0.085	AR
	Author (A)	14,475	AP	41,794	2.8	2.9	1.0	0.0002	IR
	Conference (C)	20	PT	88,683	6.2	10.7	1.7	0.0007	IR
			APT	260,605	18.0	29.6	1.6	0.002	IR
Yelp	User (U)	1,286	BR	2,614	1.0	1307.0	1307.0	0.5	AR
	Service (S)	2	BS	2,614	1.0	1307.0	1307.0	0.5	AR
	Business (B)	2,614	BL	2,614	1.0	290.4	290.4	0.1	AR
	Star Level (L)	9	UB	30,838	23.9	11.8	2.0	0.009	IR
	Reservation (R)	2	BUB	528,332	405.3	405.3	1.0	0.07	IR
AMiner	Paper (P)	127,623	PC	127,623	1.0	1263.6	1264.6	0.01	AR
	Author (A)	164,472	APC	232,659	2.2	3515.6	1598.0	0.01	AR
	Reference (R)	147,251	AP	355,072	2.2	2.8	1.3	0.00002	IR
	Conference (C)	101	PR	392,519	3.1	2.7	1.1	0.00002	IR
			APR	1,084,287	7.1	7.9	1.1	0.00004	IR

of $t_{v_1} \xrightarrow{t_{e_1}} t_{v_2} ... \xrightarrow{t_{e_l}} t_{v_{l+1}}$ (abbreviated as $t_{v_1} t_{v_2} ... t_{v_l+1}$), which describes a composite relation between v_1 and v_{l+1}.

Definition 3. *Node-Relation Triple*. In an HIN G, relations R include atomic relations (e.g., links) and composite relations (e.g., meta-paths). A node-relation triple $\langle u, r, v\rangle \in P$, describes that two nodes u and v are connected by a relation $r \in R$. Here P represents the set of all node-relation triples.

Example 1. For example, as shown in Figure 1(a), $\langle a_2, APC, c_2\rangle$ is a node-relation triple, meaning that a_1 writes a paper published in c_2.

Definition 4. *Heterogeneous Information Network Embedding*. Given an HIN $G = (V, E, T, \phi, \varphi)$, the goal of HIN embedding is to develop a mapping function $f : V \rightarrow \mathbb{R}^d$ that projects each node $v \in V$ to a low-dimensional vector in $\mathbb{R}^d$, where $d \ll |V|$.

4 Structural Characteristics of Relations

In this section, we first describe three real-world HINs and analyze the structural characteristics of relations in HINs. Then we present two structure-related measures which can consistently distinguish various relations quantitatively.

4.1 Dataset Description

Before analyzing the structural characteristics of relations, we first briefly introduce three datasets used in this paper, including DBLP[1], Yelp[2] and AMiner[3](Tang et al. 2008). The detailed statistics of these datasets are illustrated in Table 1.

DBLP is an academic network, which contains four types of nodes: author (A), paper (P), conference (C) and term (T). We extract node-relation triples based on the set of relations $\{$*AP, PC, PT, APC, APT*$\}$. Yelp is a social network, which contains five types of nodes: user (U), business (B), reservation (R), service (S) and star level (L). We consider the relations $\{$*BR, BS, BL, UB, BUB*$\}$. AMiner is also an academic network, which contains four types of nodes, including author (A), paper (P), conference (C) and reference (R). We consider the relations $\{$*AP, PC, PR, APC, APR*$\}$. Notice that we can actually analyze all the relations based on meta-paths. However, not all meta-paths have a positive effect on embeddings (Sun et al. 2013). Hence, following previous works (Shang et al. 2016; Dong, Chawla, and Swami 2017), we choose the important and meaningful meta-paths.

4.2 Affiliation Relations and Interaction Relations

In order to explore the structural characteristics of relations, we present mathematical analysis on the above datasets.

Since the degree of nodes can well reflect the structures of networks (Wasserman and Faust 1994), we define a degree-based measure $D(r)$ to explore the distinction of various relations in an HIN. Specifically, we compare the average degrees of two types of nodes connected with the relation r, via dividing the larger one by the smaller one ($D(r) \geq 1$). Formally, given a relation r with nodes u and v (i.e., node relation triple $\langle u, r, v\rangle$), t_u and t_v are the node types of u and v, we define $D(r)$ as follows:

$$D(r) = \frac{\max[\bar{d}_{t_u}, \bar{d}_{t_v}]}{\min[\bar{d}_{t_u}, \bar{d}_{t_v}]}, \qquad (1)$$

where $\bar{d}_{t_u}$ and $\bar{d}_{t_v}$ are the average degrees of nodes of the types t_u and t_v respectively.

A large value of $D(r)$ indicates quite inequivalent structural roles of two types of nodes connected via the relation r (one-centered-by-another), while a small value of $D(r)$ means compatible structural roles (peer-to-peer). In other words, relations with a large value of $D(r)$ show much stronger affiliation relationships. Nodes connected via such relations share much more similar properties (Faust 1997). While relations with a small value of $D(r)$ implicate much stronger interaction relationships. Therefore, we call the two categories of relations as *Affiliation Relations* (ARs) and *Interaction Relations* (IRs), respectively.

In order to better understand the structural difference between various relations, we take the DBLP network as an

[1] https://dblp.uni-trier.de

[2] https://www.yelp.com/dataset/

[3] https://www.aminer.cn/citation

example. As shown in Table 1, for the relation *PC* with $D(PC) = 718.8$, the average degree of nodes with type P is 1.0 while that of nodes with type C is 718.8. It shows that papers and conferences are structurally inequivalent. Papers are centered by conferences. While $D(AP) = 1.1$ indicates that authors and papers are compatible and peer-to-peer in structure. This is consistent with our common sense. Semantically, the relation *PC* means that *'papers are published in conferences'*, indicating an affiliation relationship. Differently, *AP* means that *'authors write papers'*, which explicitly describes an interaction relationship.

In fact, we can also define some other measures to capture the structural difference. For example, we compare the relations in terms of sparsity, which can be defined as:

$$S(r) = \frac{N_r}{N_{t_u} \times N_{t_v}}, \tag{2}$$

where N_r represents the number of relation instances following r. N_{t_u} and N_{t_v} mean the number of nodes with type t_u and t_v, respectively. The measure can also consistently distinguish the relations into two categories: ARs and IRs. The detailed statistics of all the relations in the three HINs are shown in Table 1.

Evidently, Affiliation Relations and Interaction Relations exhibit rather distinct characteristics: (1) ARs indicate one-centered-by-another structures, where the average degrees of the types of end nodes are extremely different. They imply an affiliation relationship between nodes. (2) IRs describe peer-to-peer structures, where the average degrees of the types of end nodes are compatible. They suggest an interaction relationship between nodes.

5 Relation Structure-Aware HIN Embedding

In this section, we present a novel Relation structure-aware HIN Embedding model (RHINE), which individually handles two categories of relations (ARs and IRs) with different models in order to preserve their distinct structural characteristics, as illustrated in Figure 1(c).

5.1 Basic Idea

Through our exploration with thorough mathematical analysis, we find that the heterogeneous relations can be typically divided into ARs and IRs with different structural characteristics. In order to respect their distinct characteristics, we need to specifically design different while appropriate models for the different categories of relations.

For ARs, we propose to take Euclidean distance as a metric to measure the proximity of the connected nodes in the low-dimensional space. There are two motivations behind this: (1) First of all, ARs show affiliation structures between nodes, which indicate that nodes connected via such relations share similar properties. (Faust 1997; Yang and Leskovec 2012). Hence, nodes connected via ARs could be directly close to each other in the vector space, which is also consistent with the optimization of Euclidean distance (Danielsson 1980). (2) Additionally, one goal of HIN embedding is to preserve the high-order proximity. Euclidean distance can ensure that both first-order and second-order proximities are preserved as it meets the condition of the triangle inequality (Hsieh et al. 2017).

Different from ARs, IRs indicate strong interaction relationships between compatible nodes, which themselves contain important structural information of two nodes. Thus, we propose to explicitly model an IR as a translation between nodes in the low-dimensional vector space. Additionally, the translation based distance is consistent with the Euclidean distance in the mathematical form (Bordes et al. 2013). Therefore, they can be smoothly combined in a unified and elegant manner.

5.2 Different Models for ARs and IRs

In this subsection, we introduce two different models exploited in RHINE for ARs and IRs, respectively.

Euclidean Distance for Affiliation Relations Nodes connected via ARs share similar properties (Faust 1997), therefore nodes could be directly close to each other in the vector space. We take the Euclidean distance as the proximity measure of two nodes connected by an AR.

Formally, given an affiliation node-relation triple $\langle p, s, q \rangle \in P_{AR}$ where $s \in R_{AR}$ is the relation between p and q with weight w_{pq}, the distance between p and q in the latent vector space is calculated as follows:

$$f(p, q) = w_{pq} ||\mathbf{X}_p - \mathbf{X}_q||_2^2, \tag{3}$$

in which $\mathbf{X}_p \in \mathbb{R}^d$ and $\mathbf{X}_q \in \mathbb{R}^d$ are the embedding vectors of p and q, respectively. As $f(p, q)$ quantifies the distance between p and q in the low-dimensional vector space, we aim to minimize $f(p, q)$ to ensure that nodes connected by an AR should be close to each other. Hence, we define the margin-based loss (Bordes et al. 2013) function as follows:

$$\begin{aligned} L_{EuAR} = & \sum_{s \in R_{AR}} \sum_{\langle p,s,q \rangle \in P_{AR}} \\ & \sum_{\langle p',s,q' \rangle \in P'_{AR}} \max[0, \gamma + f(p, q) - f(p', q')], \end{aligned} \tag{4}$$

where $\gamma > 0$ is a margin hyperparameter. P_{AR} is the set of positive affiliation node-relation triples, while P'_{AR} is the set of negative affiliation node-relation triples.

Translation-based Distance for Interaction Relations Interaction Relations demonstrate strong interactions between nodes with compatible structural roles. Thus, different from ARs, we explicitly model IRs as translations between nodes.

Formally, given an interaction node-relation triple $\langle u, r, v \rangle$ where $r \in R_{IR}$ with weight w_{uv}, we define the score function as:

$$g(u, v) = w_{uv} ||\mathbf{X}_u + \mathbf{Y}_r - \mathbf{X}_v||, \tag{5}$$

where $\mathbf{X}_u$ and $\mathbf{X}_v$ are the node embeddings of u and v respectively, and $\mathbf{Y}_r$ is the embedding of the relation r. Intuitively, this score function penalizes deviation of $(\mathbf{X}_u + \mathbf{Y}_r)$ from the vector $\mathbf{X}_v$.

For each interaction node-relation triple $\langle u, r, v\rangle \in P_{IR}$, we define the margin-based loss function as follows:

$$
\begin{aligned}
L_{TrIR} = & \sum_{r\in R_{IR}} \sum_{\langle u,r,v\rangle \in P_{IR}} \\
& \sum_{\langle u',r,v'\rangle \in P'_{IR}} \max[0, \gamma + g(u,v) - g(u',v')]
\end{aligned} \tag{6}
$$

where P_{IR} is the set of positive interaction node-relation triples, while P'_{IR} is the set of negative interaction node-relation triples.

5.3 A Unified Model for HIN Embedding

Finally, we smoothly combine the two models for different categories of relations by minimizing the following loss function:

$$
\begin{aligned}
& L = L_{EuAR} + L_{TrIR} \qquad (7) \\
& = \sum_{s\in R_{AR}} \sum_{\langle p,s,q\rangle \in P_{AR}} \sum_{\langle p',s,q'\rangle \in P'_{AR}} \max[0, \gamma + f(p,q) - f(p',q')] \\
& + \sum_{r\in R_{IR}} \sum_{\langle u,r,v\rangle \in P_{IR}} \sum_{\langle u',r,v'\rangle \in P'_{IR}} \max[0, \gamma + g(u,v) - g(u',v')]
\end{aligned}
$$

Sampling Strategy As shown in Table 1, the distributions of ARs and IRs are quite unbalanced. What's more, the proportion of relations are unbalanced within ARs and IRs. Traditional edge sampling may suffer from under-sampling for relations with a small amount or over-sampling for relations with a large amount. To address the problems, we draw positive samples according to their probability distributions. As for negative samples, we follow previous work (Bordes et al. 2013) to construct a set of negative node-relation triples $P'_{(u,r,v)} = \{(u', r, v)|u' \in V\} \cup \{(u, r, v')|v' \in V\}$ for the positive node-relation triple (u, r, v), where either the head or tail is replaced by a random node, but not both at the same time.

6 Experiments

In this section, we conduct extensive experiments to demonstrate the effectiveness of our model RHINE.

6.1 Datasets

As described in Subsection 4.1, we conduct experiments on three datasets, including DBLP, Yelp and AMiner. The statistics of them are summarized in Table 1.

6.2 Baseline Methods

We compare our proposed model RHINE with six state-of-the-art network embedding methods.

- **DeepWalk** (Perozzi, Al-Rfou, and Skiena 2014) performs a random walk on networks and then learns low-dimensional node vectors via the skip-gram model.
- **LINE** (Tang et al. 2015) considers first-order and second-order proximities in networks. We denote the model that only uses first-order or second-order proximity as LINE-1st or LINE-2nd, respectively.

Table 2: Performance Evaluation of Node Clustering.

Methods	DBLP	Yelp	AMiner
DeepWalk	0.3884	0.3043	0.5427
LINE-1st	0.2775	0.3103	0.3736
LINE-2nd	0.4675	0.3593	0.3862
PTE	0.3101	0.3527	0.4089
ESim	0.3449	0.2214	0.3409
HIN2Vec	0.4256	0.3657	0.3948
metapath2vec	0.6065	0.3507	0.5586
RHINE	**0.7204**	**0.3882**	**0.6024**

- **PTE** (Tang, Qu, and Mei 2015) decomposes an HIN to a set of bipartite networks and then learns the low-dimensional representation of the network.
- **ESim** (Shang et al. 2016) takes a given set of meta-paths as input to learn a low-dimensional vector space. For a fair comparison, we use the same meta-paths with equal weights in Esim and our model RHINE.
- **HIN2Vec** (Fu, Lee, and Lei 2017) learns the latent vectors of nodes and meta-paths in an HIN by conducting multiple prediction training tasks jointly.
- **Metapath2vec** (Dong, Chawla, and Swami 2017) leverages meta-path based random walks and skip-gram model to perform node embedding. We leverage the meta-paths APCPA, UBSBU and APCPA in DBLP, Yelp and AMiner respectively, which perform best in the evaluations.

Parameter Settings For a fair comparison, we set the embedding dimension $d = 100$ and the size of negative samples $k = 3$ for all models. For DeepWalk, HIN2Vec and metapath2vec, we set the number of walks per node $w = 10$, the walk length $l = 100$ and the window size $\tau = 5$. For our model RHINE, the margin γ is set to 1.

6.3 Node Clustering

Experimental Settings We leverage K-means to cluster the nodes and evaluate the results in terms of normalized mutual information (NMI) (Shi et al. 2014).

Results As shown in Table 2, our model RHINE significantly outperforms all the compared methods. (1) Compared with the best competitors, the clustering performance of our model RHINE improves by 18.79%, 6.15% and 7.84% on DBLP, Yelp and AMiner, respectively. It demonstrates the effectiveness of our model RHINE by distinguishing the various relations with different structural characteristics in an HIN. In addition, it also validates that we utilize appropriate models for different categories of relations. (2) In all baseline methods, homogeneous network embedding models achieve the lowest performance, because they ignore the heterogeneity of relations and nodes. (3) RHINE significantly outperforms existing HIN embedding models (i.e., ESim, HIN2Vec and metapath2vec) on all datasets. We believe the reason is that our proposed RHINE with appropriate models for different categories of relations can better capture the structural and semantic information of HINs.

Table 3: Performance Evaluation of Link Prediction.

Methods	DBLP (A-A)		DBLP (A-C)		Yelp (U-B)		AMiner (A-A)		AMiner (A-C)	
	AUC	F1	AUC	F1	AUC	F1	AUC	F1	AUC	F1
DeepWalk	0.9131	0.8246	0.7634	0.7047	0.8476	0.6397	0.9122	0.8471	0.7701	0.7112
LINE-1st	0.8264	0.7233	0.5335	0.6436	0.5084	0.4379	0.6665	0.6274	0.7574	0.6983
LINE-2nd	0.7448	0.6741	0.8340	0.7396	0.7509	0.6809	0.5808	0.4682	0.7899	0.7177
PTE	0.8853	0.8331	0.8843	0.7720	0.8061	0.7043	0.8119	0.7319	0.8442	0.7587
ESim	0.9077	0.8129	0.7736	0.6795	0.6160	0.4051	0.8970	0.8245	0.8089	0.7392
HIN2Vec	0.9160	0.8475	0.8966	0.7892	0.8653	0.7709	0.9141	0.8566	0.8099	0.7282
metapath2vec	0.9153	0.8431	0.8987	0.8012	0.7818	0.5391	0.9111	0.8530	0.8902	0.8125
RHINE	**0.9315**	**0.8664**	**0.9148**	**0.8478**	**0.8762**	**0.7912**	**0.9316**	**0.8664**	**0.9173**	**0.8262**

Table 4: Performance Evaluation of Multi-class Classification.

Methods	DBLP		Yelp		AMiner	
	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1
DeepWalk	0.7475	0.7500	0.6723	0.7012	0.9386	0.9512
LINE-1st	0.8091	0.8250	0.4872	0.6639	0.9494	0.9569
LINE-2nd	0.7559	0.7500	0.5304	0.7377	0.9468	0.9491
PTE	0.8852	0.8750	0.5389	0.7342	0.9791	0.9847
ESim	0.8867	0.8750	0.6836	0.7399	0.9910	0.9948
HIN2Vec	0.8631	0.8500	0.6075	0.7361	**0.9962**	**0.9965**
metapath2vec	0.8976	0.9000	0.5337	0.7208	0.9934	0.9936
RHINE	**0.9344**	**0.9250**	**0.7132**	**0.7572**	0.9884	0.9807

6.4 Link Prediction

Experimental Setting We model the link prediction problem as a binary classification problem that aims to predict whether a link exists. In this task, we conduct co-author (*A-A*) and author-conference (*A-C*) link prediction for DBLP and AMiner. For Yelp, we predict user-business (*U-B*) links which indicate whether a user reviews a business. We first randomly separate the original network into training network and testing network, where the training network contains 80% relations to be predicted (i.e., *A-A*, *A-C* and *U-B*) and the testing network contains the rest. Then, we train the embedding vectors on the training network and evaluate the prediction performance on the testing network.

Results The results of link prediction task are reported in Table 3 with respect to AUC and F1 score. It is clear that our model performs better than all baseline methods on three datasets. The reason behind the improvement is that our model based on Euclidean distance modeling relations can capture both the first-order and second-order proximities. In addition, our model RHINE distinguishes multiple types of relations into two categories in terms of their structural characteristics, and thus can learn better embeddings of nodes, which are beneficial for predicting complex relationships between two nodes.

6.5 Multi-Class Classification

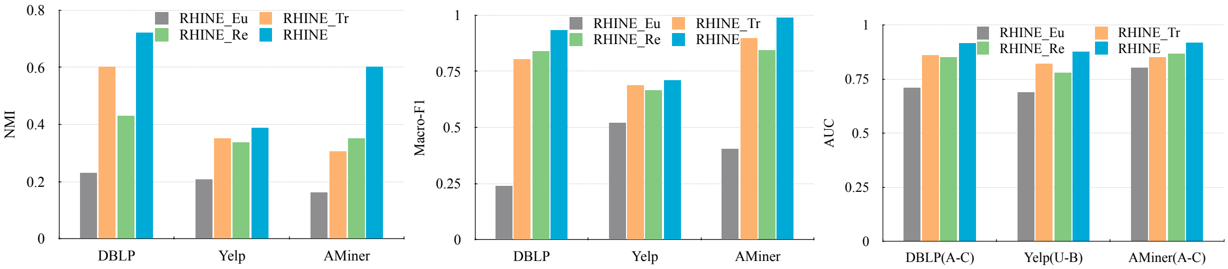
Experimental Setting In this task, we employ the same labeled data used in the node clustering task. After learning the node vectors, we train a logistic classifier with 80% of the labeled nodes and test with the remaining data. We use Micro-F1 and Macro-F1 score as the metrics for evaluation (Dong, Chawla, and Swami 2017).

Results We summarize the results of classification in Table 4. As we can observe, (1) RHINE achieves better performance than all baseline methods on all datasets except Aminer. It improves the performance of node classification by about 4% on both DBLP and Yelp averagely. In terms of AMiner, the RHINE performs slightly worse than ESim, HIN2vec and metapath2vec. This may be caused by over-capturing the information of relations *PR* and *APR* (*R* represents references). Since an author may write a paper referring to various fields, these relations may introduce some noise. (2) Although ESim and HIN2Vec can model multiple types of relations in HINs, they fail to perform well in most cases. Our model RHINE achieves good performance due to the respect of distinct characteristics of various relations.

6.6 Comparison of Variant Models

In order to verify the effectiveness of distinguishing the structural characteristics of relations, we design three variant models based on RHINE as follows:

- **RHINE**$_{Eu}$ leverages Euclidean distance to embed HINs without distinguishing the relations.
- **RHINE**$_{Tr}$ models all nodes and relations in HINs with translation mechanism, which is just like TransE (Bordes et al. 2013).

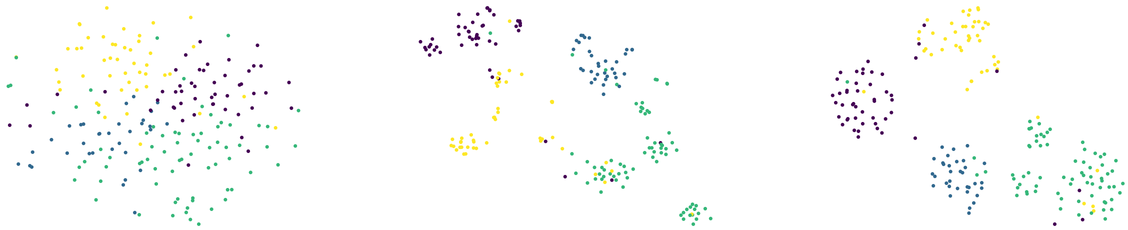


(a) Node Clustering on Three Datasets

(b) Classification on Three Datasets

(c) Link Prediction on Three Datasets

Figure 2: Performance Evaluation of Variant Models.



(a) DeepWalk

(b) metapath2vec

(c) RHINE

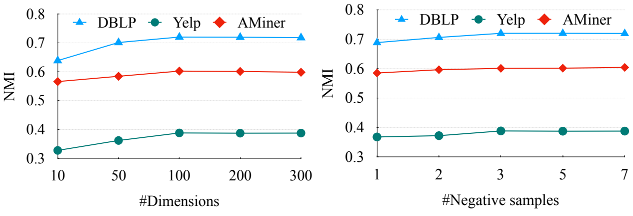
Figure 3: Visualization of Node Embeddings.

- **RHINE**$_{Re}$ leverages Euclidean distance to model IRs while translation mechanism for ARs, reversely.

We set the parameters of variant models as the same as those of our proposed model RHINE. The results of the three tasks are shown in Figure 2. It is evident that our model outperforms RHINE$_{Eu}$ and RHINE$_{Tr}$, indicating that it is beneficial for learning the representations of nodes by distinguishing the heterogeneous relations. Besides, we find that RHINE$_{Tr}$ achieves better performance than RHINE$_{Eu}$. This is due to the fact that there are generally more peer-to-peer relationships (i.e., IRs) in the networks. Directly making all nodes close to each other leads to much loss of information. Compared with the reverse model RHINE$_{Re}$, RHINE also achieves better performance on all tasks, which implies that two models for ARs and IRs are well designed to capture their distinctive characteristics.

6.7 Visualization

To understand the representations of the networks intuitively, we visualize the vectors of nodes (i.e., papers) in DBLP learned with DeepWalk, metapath2vec and RHINE in Figure 3. As we can see, our model clearly clusters the paper nodes into four groups. It demonstrates that our model learns superior node embeddings by distinguishing the heterogeneous relations in HINs. In contrast, DeepWalk barely splits papers into different groups. Metapath2vec performs better than DeepWalk, but the boundary is blurry.



(a) NMI w.r.t #dimensions

(b) NMI w.r.t #negative samples

Figure 4: Parameter Analysis.

6.8 Parameter Analysis

In order to investigate the influences of different parameters in our model, we evaluate the RHINE in node clustering task. Specifically, we explore the sensitivity of two parameters, including the number of embedding dimensions and the number of negative samples. As shown in Figure 4(a), the performance of our model improves with the increase in the number of dimensions, and then tends to be stable once the dimension of the representation reaches around 100. Similarly, Figure 4(b) shows that as the number of negative examples increases, the performance of our model first grows and then becomes stable when the number reaches 3.

7 Conclusion

In this paper, we make the first attempt to explore and distinguish the structural characteristics of relations for HIN embedding. We present two structure-related measures which can consistently distinguish heterogeneous relations into two categories: affiliation relations and interaction relations. To respect the distinctive structures of relations, we propose a novel relation structure-aware HIN embedding model (RHINE), which individually handles these two categories of relations. Experimental results demonstrate that RHINE outperforms state-of-the-art baselines in various tasks. In the future, we will explore other possible measures to differentiate relations so that we can better capture the structural information of HINs. In addition, we will exploit deep neural network based models for different relations.

8 Acknowledgments

This work is supported by the National Key Research and Development Program of China (2017YFB0803304), the National Natural Science Foundation of China (No. 61772082, 61806020, 61702296, 61375058), the Beijing Municipal Natural Science Foundation (4182043), and the CCF-Tencent Open Fund.

References

Bordes, A.; Usunier, N.; Garcia-Duran, A.; Weston, J.; and Yakhnenko, O. 2013. Translating embeddings for modeling multi-relational data. In *Proceedings of NIPS*, 2787–2795.

Cai, H.; Zheng, V. W.; and Chang, K. 2018. A comprehensive survey of graph embedding: problems, techniques and applications. *IEEE Transactions on Knowledge and Data Engineering*.

Cao, S.; Lu, W.; and Xu, Q. 2016. Deep neural networks for learning graph representations. In *Proceedings of AAAI*, 1145–1152.

Chang, S.; Han, W.; Tang, J.; Qi, G.-J.; Aggarwal, C. C.; and Huang, T. S. 2015. Heterogeneous network embedding via deep architectures. In *Proceedings of SIGKDD*, 119–128. ACM.

Cui, P.; Wang, X.; Pei, J.; and Zhu, W. 2018. A survey on network embedding. *IEEE Transactions on Knowledge and Data Engineering*.

Danielsson, P.-E. 1980. Euclidean distance mapping. *Computer Graphics and image processing* 14(3):227–248.

Dong, Y.; Chawla, N. V.; and Swami, A. 2017. metapath2vec: Scalable representation learning for heterogeneous networks. In *Proceedings of SIGKDD*, 135–144. ACM.

Faust, K. 1997. Centrality in affiliation networks. *Social networks* 19(2):157–191.

Fu, T.-y.; Lee, W.-C.; and Lei, Z. 2017. Hin2vec: Explore metapaths in heterogeneous information networks for representation learning. In *Proceedings of CIKM*, 1797–1806. ACM.

Grover, A., and Leskovec, J. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of SIGKDD*, 855–864. ACM.

Han, X.; Shi, C.; Wang, S.; Philip, S. Y.; and Song, L. 2018. Aspect-level deep collaborative filtering via heterogeneous information networks. In *Proceedings of IJCAI*, 3393–3399.

Hsieh, C.-K.; Yang, L.; Cui, Y.; Lin, T.-Y.; Belongie, S.; and Estrin, D. 2017. Collaborative metric learning. In *Proceedings of WWW*, 193–201.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of NIPS*, 3111–3119.

Ou, M.; Cui, P.; Pei, J.; Zhang, Z.; and Zhu, W. 2016. Asymmetric transitivity preserving graph embedding. In *Proceedings of SIGKDD*, 1105–1114. ACM.

Perozzi, B.; Al-Rfou, R.; and Skiena, S. 2014. Deepwalk: Online learning of social representations. In *Proceedings of SIGKDD*, 701–710. ACM.

Ribeiro, L. F.; Saverese, P. H.; and Figueiredo, D. R. 2017. struc2vec: Learning node representations from structural identity. In *Proceedings of SIGKDD*, 385–394. ACM.

Shang, J.; Qu, M.; Liu, J.; Kaplan, L. M.; Han, J.; and Peng, J. 2016. Meta-path guided embedding for similarity search in large-scale heterogeneous information networks. *arXiv preprint arXiv:1610.09769*.

Shi, C.; Kong, X.; Huang, Y.; Philip, S. Y.; and Wu, B. 2014. Hetesim: A general framework for relevance measure in heterogeneous networks. *IEEE Transactions on Knowledge and Data Engineering* 26(10):2479–2492.

Shi, C.; Li, Y.; Zhang, J.; Sun, Y.; and Philip, S. Y. 2017. A survey of heterogeneous information network analysis. *IEEE Transactions on Knowledge and Data Engineering* 29(1):17–37.

Shi, C.; Hu, B.; Zhao, X.; and Yu, P. 2018. Heterogeneous information network embedding for recommendation. *IEEE Transactions on Knowledge and Data Engineering*.

Sun, Y.; Han, J.; Yan, X.; Yu, P. S.; and Wu, T. 2011. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. *Proceedings of VLDB* 4(11):992–1003.

Sun, Y.; Norick, B.; Han, J.; Yan, X.; Yu, P. S.; and Yu, X. 2013. Pathselclus: Integrating meta-path selection with user-guided object clustering in heterogeneous information networks. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 7(3):11.

Tang, J.; Zhang, J.; Yao, L.; Li, J.; Zhang, L.; and Su, Z. 2008. Arnetminer: extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 990–998. ACM.

Tang, J.; Qu, M.; Wang, M.; Zhang, M.; Yan, J.; and Mei, Q. 2015. Line: Large-scale information network embedding. In *Proceedings of WWW*, 1067–1077.

Tang, J.; Qu, M.; and Mei, Q. 2015. Pte: Predictive text embedding through large-scale heterogeneous text networks. In *Proceedings of SIGKDD*, 1165–1174. ACM.

Wang, H.; Zhang, F.; Hou, M.; Xie, X.; Guo, M.; and Liu, Q. 2018. Shine: signed heterogeneous information network embedding for sentiment link prediction. In *Proceedings of WSDM*, 592–600. ACM.

Wang, D.; Cui, P.; and Zhu, W. 2016. Structural deep network embedding. In *Proceedings of SIGKDD*, 1225–1234. ACM.

Wasserman, S., and Faust, K. 1994. *Social network analysis: Methods and applications*, volume 8. Cambridge university press.

Xu, L.; Wei, X.; Cao, J.; and Yu, P. S. 2017. Embedding of embedding (eoe): Joint embedding for coupled heterogeneous networks. In *Proceedings of WSDM*, 741–749. ACM.

Yang, J., and Leskovec, J. 2012. Community-affiliation graph model for overlapping network community detection. In *Proceedings of ICDM*, 1170–1175. IEEE.