A Structural Representation Learning for Multi-relational Networks

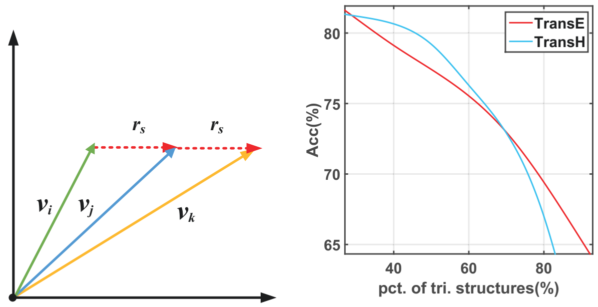
Lin Liu[1], **Xin Li**[1*],**William K. Cheung**[2] and **Chengcheng Xu**[1]
[1] BJ ER Center of HVLIP&CC, School of Comp. Sci., Beijing Institute of Technology, Beijing, China
[2] Department of Computer Science, Hong Kong Baptist University, Hong Kong, China

Abstract

Most of the existing multi-relational network embedding methods, e.g., TransE, are formulated to preserve pair-wise connectivity structures in the networks. With the observations that significant triangular connectivity structures and parallelogram connectivity structures found in many real multi-relational networks are often ignored and that a hard-constraint commonly adopted by most of the network embedding methods is inaccurate by design, we propose a novel representation learning model for multi-relational networks which can alleviate both fundamental limitations. Scalable learning algorithms are derived using the stochastic gradient descent algorithm and negative sampling. Extensive experiments on real multi-relational network datasets of WordNet and Freebase demonstrate the efficacy of the proposed model when compared with the state-of-the-art embedding methods.

1 Introduction

Representation learning has become an important research track in the area of machine learning, with the aim of providing more informative numerical representations of the observed data for applications like image classification, speech recognition and text mining, etc. More specifically, network embedding, which is to learn the distributed representations of information networks, has attracted much attention due to the promising empirical results obtained. In the literature, a number of network embedding methods have been proposed, including LINE [Tang *et al.*, 2015], IONE [Liu *et al.*, 2016], SDNE [Wang *et al.*, 2016], and DeepWalk [Perozzi *et al.*, 2014]. These methods learn only the representations of the nodes in a network, and the edges are assumed to be single-relational, that is, they are of the same type. For instance, edges represent only “friendship” in a social network , and only “collaboration” in the DBLP collaboration network.



(a) (b)

Figure 1: Trans-family vs. Triangular structures

A multi-relational network is represented by a directed graph with the edges of various relation types typically indicated by associating each edge from a source node to a target node with a discrete label, denoted as ***(source, label, target)*** or ***(h, r, t)***. Such multi-relational networks, e.g., Google Knowledge Graph, semantic networks and multi-relational social networks, have become important resources to support more advanced information retrieval, question-answering systems, etc. To learn the embedding of such a network, it is common for both the node and edge representations to be learned at the same time.

Following the success of TransE [Bordes *et al.*, 2013], a series of translation-based methods have been proposed for knowledge graph (KG) embedding to project the nodes (also called entities) and the edges (also called relations) of the KG onto a continuous vector space, e.g., TransH [Wang *et al.*, 2014b], TransR [Lin *et al.*, 2015b], pTransE[Wang *et al.*, 2014a] and TransG [Xiao *et al.*, 2015] (referred to as “trans-family” hereafter), so that the local structural relationship of the nodes and edges can be retained in their corresponding embeddings. These approaches differ from each other in the way of (1) whether the entities and relations are projected onto the same subspace (e.g., TransH and TransR project a KG onto different subspaces to reflect the relations' semantics); (2) how the embedding objective function is defined (e.g., TransE minimizes the so-called energy function of $f_r(h,t) = ||h + r - t||$, while pTransE maximizes the conditional probability of (h, r, t) with the constraint $h + r = t$).

[*] Corresponding Author: Xin Li (xinli@bit.edu.cn). This work has been partially supported by NSFC under Grant No. 61300178, National Program on Key Basic Research Project under Grant No. 2013CB329605.

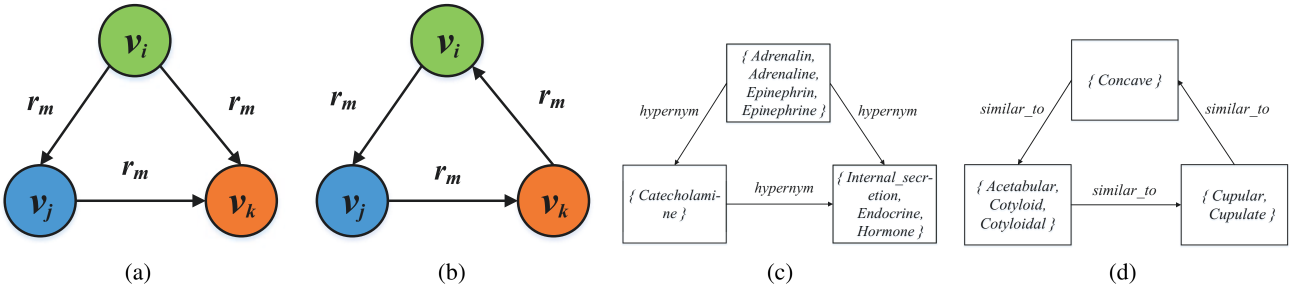


Figure 2: Triangular structure examples

In this paper, we focus on the representation learning of the multi-relational networks, and propose our approach based on the following two observations:

Observation 1: Methods in the trans-family are all constrained by $h + r = t$ which cannot capture the structures shown in Fig.2. For the directed graph with three nodes connecting to each other via a specific edge, there are two non-isomorphic modes. In this paper, this structure is referred to as the triangular structure which often appears in many multi-relational networks. For example, Fig.2(c) illustrates a fact in WordNet which accords with the mode in Fig.2(a), where {internal_secretion, endocrine, hormone} is the hypernym of {adrenalin, adrenaline, epinephrin, epinephrine} and {catecholamine}, {catecholamine} is the hypernym of {adrenalin, adrenaline, epinephrin, epinephrine}, and the relation edge is labeled "hypernym". Note that WordNet is organized by the concept of synonym sets (so-called synsets), where each node represents a set of words that are roughly synonymous in a given context. Fig.2(d) illustrates another fact in WordNet which accords with the mode in Fig.2(b), where the relation edge is labeled as "similar to". In the trans-family, the scoring function $f_r(h,t) = ||h + r - t||$ is used to ensure the plausibility of triple (h, r, t). Accordingly, the closeness of similar nodes can be guaranteed in the low-dimensional Euclidean space. However, Euclidean geometry breaks when encountering triangular structures. For example, TransE requires the forms of $v_i + r_s \approx v_j$, $v_j + r_s \approx v_k$ and $v_i + r_s \approx v_k$ to hold at the same time. However, as illustrated in Fig.1(a), for the former two equations to hold, we have $v_i + 2r_s \approx v_k$. The forcible updating rule in trans-family will compromise the accuracy. Fig.1(b) shows that the accuracy of the link prediction obtained by TransE and TransH decreases as the number of the triangular structures increases [1].

Observation 2: Network embedding methods like LINE [Tang *et al.*, 2015] have been proposed to capture network structures by exploring the first-order and second-order proximities. The former corresponds to the edge strength between two connected nodes, while the latter corresponds to the overlapping neighbors of the two nodes. Note that embedding methods like LINE are deliberately designed for single-relational networks in which these two properties are commonly seen. However, in multi-relational networks, the strengths of the edges do not vary as much as in single-relational networks[2]. For KGs like WordNet, most of nodes are linked with each other by an edge of a specific relation type only once. Besides, it is difficult to define the scale of the strength when the relations have different semantic meanings. In addition, the second-order proximity focuses on how many neighbors of two nodes are exactly the same, whereas in our framework we propose to relax such proximity definition by considering the proximity among the neighbors via *parallelogram structures*. We have found that parallelogram structures exist more often in multi-relational networks. Fig.3 illustrates the examples of parallelogram structures, where $\{v_1, v_2, v_5, v_6\}$ and $\{v_1, v_2, v_3, v_7\}$ are the two instances of the parallelogram structure with the parallel sides of the same relation type. As shown in Fig.3, the two nodes v_1 and v_2 are linked to v_3 and v_7 via the same relation r_1 respectively. When v_3 and v_7 are linked by a relation r_5, it is highly likely v_1 and v_2 can be linked together via the same relation of their neighbors, that is r_5. Intuitively, given any three sides of the parallelogram, we could infer the relation of the fourth one.

In this paper, we propose a multi-relational network embedding method. The objective function is designed to consider deliberately the triangular and parallelogram structures to define node proximity, and thus to infer the representations. In order to improve the efficiency, we adopt the stochastic gradient descent algorithm and negative sampling to optimize the objective function to reduce the training cost. We conduct extensive experiments over the tasks of triplet classification and link prediction on the real datasets like WordNet and Freebase. Experimental results demonstrate the effectiveness of our model over several state-of-the-art methods.

2 Related Work

There are two lines of research related to our work, namely *network embedding* and *knowledge graph embedding*.

2.1 Network Embedding

One of the recent attempts to address network embedding is graph factorization (GF) [Ahmed *et al.*, 2013] which utilizes

[1] The experiments are conducted on WN18 dataset. The nodes and edges which do not belong to a triangular structure are gradually added to simulate the deceasing number of the triangular structures.

[2] For example, the number of mentions(@) of a user by another user could be considered as the strength of these two users (nodes) in single relational social networks.

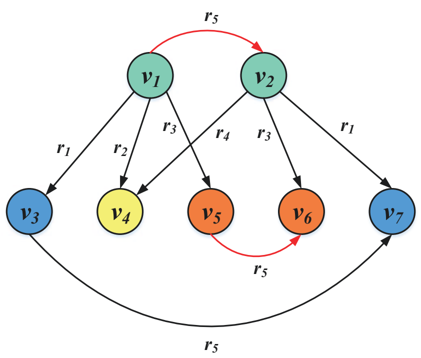


Figure 3: Parallelogram structure examples.

matrix factorization over undirected graph's affinity matrix to infer the low-dimensional embedding. Only first-order proximity is preserved and nodes with close interaction are represented closely in the projected vector space. LINE [Tang *et al.*, 2015] is another recently proposed method to handle large-scale network embedding for both directed and undirected graphs, where both first-order and second-order proximity measures have been considered. DeepWalk [Perozzi *et al.*, 2014] utilizes the distribution of node degree to model network community structure via random walk and skip-gram together to infer the network embedding. However, the studies show that DeepWalk tends to preserve the second-order proximity only. SDNE [Wang *et al.*, 2016] offers a semi-supervised deep learning framework to address the problem of learning representations of networks, in which the first-order proximity and the second-order proximity are jointly preserved. All the aforementioned approaches try to learn the representations of nodes in single-relational networks. The semantics of multiple relations are not addressed. Besides, as explained in Sec. I, the first-order and second-order proximity measures may not be the representative local structures in multi-relational networks.

2.2 Knowledge Graph Embedding

Recent advance of relational learning for knowledge graph embedding has attracted much attention from industry and academia. Among them, TransE [Bordes *et al.*, 2013] is the most well-known pioneer work which embeds both nodes and edges of different relation types onto a low-dimensional vector space. The basic idea is to represent the edge (relation) of two nodes (entities) as a translation operation in the embedding space. Given the triplet (h, r, t), we expect the representation vector of the node t to be as close as possible to the representation vector of the node h plus the relation r. The objective function is $||h + r - t||$. TransE is an efficient algorithm for the embedding. However, it does not do well in dealing with some mapping properties of relations, such as reflexive, one-to-many, many-to-one, and many-to-many. To alleviate the limitations, Wang *et al.* proposed TransH [Wang *et al.*, 2014b] to project the nodes in a relation-specific subspace (a hyperplane w_r) to obtain h' and t' respectively for each triplet (h, r, t). The translation is performed in the relation subspace and constrained by the function of $h' + r = t'$. Lin *et al.* extended the idea of TransH and proposed TransR [Lin *et al.*, 2015b] to project the entities and relations onto different vector spaces respectively to further increase the degrees of freedom for the representations. In [Lin *et al.*, 2015a], the authors argued that multiple-step relation paths also contain rich inference patterns between entities, and proposed a path-based representation learning model by considering relation paths as translations between entities. Wang *et al.* proposed a probabilistic TransE to encode the knowledge graph by maximizing the conditional probability of (h, r, t), in which the conventional scoring function of $||h + r - t||$ is still being utilized.

These translation-based approaches inherit the efficiency from TransE but also the underlying flaws when using the scoring function in one way or another. As illustrated in Section I, the use of the constraint of $h + r = t$ cannot handle the triangular structures of multi-relational networks. In this paper, we propose a novel multi-relational network embedding approach to overcome the flaws of the trans-family where the observed local structures are incorporated into the objective function to infer a more robust network representation.

3 Model Framework

Let $G = (V, E, R)$ be the graph representation of a directed multi-relational network where $V = \{v_1, v_2, \ldots, v_{|V|}\}$ corresponds to the set of nodes, $R = \{r_1, r_2, \ldots, r_{|R|}\}$ corresponds to the set of relation labels, and E corresponds to the set of typed edges. Each typed edge in E is denoted as a triplet (v_i, r_m, v_j) with v_i being the source node, r_m being the associated relation label, and v_j being the target node.

3.1 Model Description

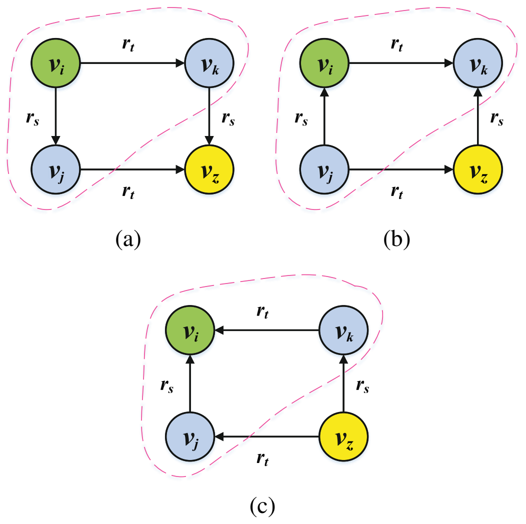
We propose a novel probabilistic embedding model for representing multi-relational networks. Similar to most of existing representation learning methods, we represent each node $v_i \in V$ as a d-dimensional vector in an embedded space via a projection function f : $V \rightarrow \Re^d$. For directed networks, since each node can take the role of either a source node or a target node in a relation-specific edge, we represent each node v_i using two vector representations: a source vector $\vec{u}_i \in \Re^d$, a target vector $\vec{u}'_i \in \Re^d$. Also, we introduce $\vec{u}_{r_i}$ as the vector representation of relation r_i.

Given a node v_i, we first define the probability that the node links to v_j via a relation r_s, when compared with how v_i is related to other nodes via its outgoing edges, denoted as

$$p(v_j, r_s|v_i) = \frac{\exp(\vec{u}_j'^{\,T}(\vec{u}_i + \vec{u}_{r_s}))}{\sum\limits_{(v_i, r_p, v_x) \in E'} \exp(\vec{u}_x'^{\,T}(\vec{u}_i + \vec{u}_{r_p}))} \quad (1)$$

$$E' = \{(v_i, r_x, v_p) | v_p \in V, r_x \in R\} \quad (2)$$

where the source vector $\vec{u}_i$, the target vector $\vec{u}_j'$ and the relation vector $\vec{u}_{r_s}$ for the directed edge (v_i, r_s, v_j) are related by adding $\vec{u}_{r_s}$ to the source vector $\vec{u}_i$. Note that such addition adopted in Eq.(1) is merely to include the relation to obtain the probability compared with LINE, instead of enforcing the hard constraint as in trans-family.



(a)

(b)

(c)

Figure 4: Local connectivity structures of parallelogram

Furthermore, to characterize the parallelogram structures, we take into account different possible directions of the relation edges so that three distinct non-isomorphic local connectivity structures are considered for each node in a parallelogram, as shown in Fig.4. For the three cases, we define the corresponding probability distributions as follow:

Case 1 (Fig.4(a)): As the out-degree of v_i is 2 and the in-degree of v_i is 0, p_1 is defined as the probability that v_i will "contribute" to such a situation, given as

$$p_1(v_j^{r_s}, v_k^{r_t}|v_i)= \frac{\exp(\vec{u}_j'^{\,T}(\vec{u}_i+\vec{u}_{r_s}) + \vec{u}_k'^{\,T}(\vec{u}_i+\vec{u}_{r_t}))}{\sum\limits_{\substack{(v_i,r_p,v_x)\in E' \\ \wedge(v_i,r_q,v_y)\in E'}} \exp(\vec{u}_x'^{\,T}(\vec{u}_i+\vec{u}_{r_p}) + \vec{u}_y'^{\,T}(\vec{u}_i+\vec{u}_{r_q}))} \quad (3)$$

We utilize $v_i^{r_s}$ as a neater representation of the pair of (v_i, r_s) in the sequel.

Case 2 (Fig.4(b)): As the out-degree of v_i is 1 and the in-degree of v_i is 1, p_2 is defined as:

$$p_2(v_j^{r_s}, v_k^{r_t}|v_i)= \frac{\exp(\vec{u}_i'^{\,T}(\vec{u}_j+\vec{u}_{r_s}) + \vec{u}_k'^{\,T}(\vec{u}_i+\vec{u}_{r_t}))}{\sum\limits_{\substack{(v_x,r_p,v_i)\in E' \\ \wedge(v_i,r_q,v_y)\in E'}} \exp(\vec{u}_i'^{\,T}(\vec{u}_x+\vec{u}_{r_p}) + \vec{u}_y'^{\,T}(\vec{u}_i+\vec{u}_{r_q}))} \quad (4)$$

Case 3 (Fig.4(c)): As the out-degree of v_i is 0 and the in-degree of v_i is 2, p_3 is defined as:

$$p_3(v_j^{r_s}, v_k^{r_t}|v_i)= \frac{\exp(\vec{u}_i'^{\,T}(\vec{u}_j+\vec{u}_{r_s}) + \vec{u}_i'^{\,T}(\vec{u}_k+\vec{u}_{r_t}))}{\sum\limits_{\substack{(v_x,r_p,v_i)\in E' \\ \wedge(v_y,r_q,v_i)\in E'}} \exp(\vec{u}_i'^{\,T}(\vec{u}_x+\vec{u}_{r_p}) + \vec{u}_i'^{\,T}(\vec{u}_y+\vec{u}_{r_q}))} \quad (5)$$

To preserve the three parallelogram structures, we minimize the KL-divergence of p_1, p_2, p_3 and their empirical distributions over all the nodes. The empirical distributions $\hat{p}_1$, $\hat{p}_2$ and $\hat{p}_3$ are defined as $\omega_{ij} * \omega_{ik}/(d_{out}^i * d_{out}^i)$, $\omega_{ji} * \omega_{ik}/(d_{in}^i * d_{out}^i)$ and $\omega_{ji} * \omega_{ki}/(d_{in}^i * d_{in}^i)$ respectively, where ω_{ij} denotes the weight[3] of edge (v_i, v_j), $d_{out}^i = \sum_{k\in N_{out}(v_i)} w_{ik}$ and $d_{in}^i = \sum_{k\in N_{in}(v_i)} w_{ki}$, $N_{out}(v_i)$ and $N_{in}(v_i)$ are the sets of out-neighbors and in-neighbors of v_i respectively. As the importance of the nodes in the network may be different, we introduce λ_i to represent the importance of v_i in the network. In this paper, we set λ_i according to its degree. Therefore, the objective function is defined as:

$$O = \sum_{i\in V} \lambda_i KL\left(\hat{p}\left(\cdot|v_i\right)||p\left(\cdot|v_i\right)\right) \quad (6)$$

Then we set λ_i to be $d_{out}^i * d_{out}^i$, $d_{in}^i * d_{out}^i$ and $d_{in}^i * d_{in}^i$ respectively, the corresponding objective function becomes:

$$O_1 = -\sum_{\substack{(v_i,r_s,v_j)\in E \\ \wedge(v_i,r_t,v_k)\in E}} \omega_{ij} * \omega_{ik} * \log p_1(v_j^{r_s}, v_k^{r_t}|v_i) \quad (7)$$

$$O_2 = -\sum_{\substack{(v_j,r_s,v_i)\in E \\ \wedge(v_i,r_t,v_k)\in E}} \omega_{ji} * \omega_{ik} * \log p_2(v_j^{r_s}, v_k^{r_t}|v_i) \quad (8)$$

$$O_3 = -\sum_{\substack{(v_j,r_s,v_i)\in E \\ \wedge(v_k,r_t,v_i)\in E}} \omega_{ji} * \omega_{ki} * \log p_3(v_j^{r_s}, v_k^{r_t}|v_i) \quad (9)$$

Then, the source and target vector representations for each node, i.e., $\{\vec{u}_i\}_{i=1\ldots|V|}$, $\{\vec{u}_i'\}_{i=1\ldots|V|}$ and the relation representation for each relation type, i.e., $\{\vec{u}_{r_i}\}_{i=1\ldots|R|}$ can be obtained by minimizing the combined objective function $O = O_1 + O_2 + O_3$ where O_1, O_2 and O_3 collaboratively help retain parallelogram structures as much as possible. In fact, the triangular structures are also implicitly preserved at the same time under such design.

3.2 Model Inference

The stochastic gradient descent is adopted to learn the vector representations of the multi-relational network. For example, to update the source vector of node v_i, the gradient w.r.t. $\vec{u}_i$ is computed as:

$$\frac{\partial O}{\partial \vec{u}_i} = \omega_{ij} * \omega_{ik} * \frac{\partial \log p_1(v_j^{r_s}, v_k^{r_t}|v_i)}{\partial \vec{u}_i} + \omega_{ji}(\omega_{ik} * \frac{\partial \log p_2(v_j^{r_s}, v_k^{r_t}|v_i)}{\partial \vec{u}_i} + \omega_{ki} * \frac{\partial \log p_3(v_j^{r_s}, v_k^{r_t}|v_i)}{\partial \vec{u}_i}) \quad (10)$$

To reduce the computational cost of calculating the summation over the entire set of nodes when addressing the conditional probability p_1, p_2 and p_3, we utilize the negative sampling approach [Mikolov *et al.*, 2013] which has been widely

[3] The weight indicates the strength of a labeled edge. In multi-relational social networks, the weight of a friendship relation between two users can be defined using the retweet frequency.

adopted, e.g., [Tang *et al.*, 2015], [Goldberg and Levy, 2014]. Negative sampling basically transforms the computationally expensive learning problem into a binary classification proxy problem that uses the same parameters but requires the statistics much easier to compute. The equivalent counterparts of the objective function Eq.(10) can then be derived, given as:

$$\log p_1(v_j^{r_s}, v_k^{r_t}|v_i) \propto \log\sigma(\vec{u}_j^{'T}(\vec{u}_i + \vec{u}_{r_s}) + \vec{u}_k^{'T}(\vec{u}_i + \vec{u}_{r_t})) + \sum_{m=1}^{K} E_{\substack{v_n \sim P_{n(v)} \\ r_l \sim P_{l(r)}}} \log\sigma(-\vec{u}_j^{'T}(\vec{u}_i + \vec{u}_{r_s}) - \vec{u}_n^{'T}(\vec{u}_i + \vec{u}_{r_l})) \quad (11)$$

$$\log p_2(v_j^{r_s}, v_k^{r_t}|v_i) \propto \log\sigma(\vec{u}_i^{'T}(\vec{u}_j + \vec{u}_{r_s}) + \vec{u}_k^{'T}(\vec{u}_i + \vec{u}_{r_t})) + \sum_{m=1}^{K} E_{\substack{v_n \sim P_{n(v)} \\ r_l \sim P_{l(r)}}} \log\sigma(-\vec{u}_i^{'T}(\vec{u}_j + \vec{u}_{r_s}) - \vec{u}_n^{'T}(\vec{u}_i + \vec{u}_{r_l})) \quad (12)$$

$$\log p_3(v_j^{r_s}, v_k^{r_t}|v_i) \propto \log\sigma(\vec{u}_i^{'T}(\vec{u}_j + \vec{u}_{r_s}) + \vec{u}_i^{'T}(\vec{u}_k + \vec{u}_{r_t})) + \sum_{m=1}^{K} E_{\substack{v_n \sim P_{n(v)} \\ r_l \sim P_{l(r)}}} \log\sigma(-\vec{u}_i^{'T}(\vec{u}_j + \vec{u}_{r_s}) - \vec{u}_i^{'T}(\vec{u}_n + \vec{u}_{r_l})) \quad (13)$$

Each of the first terms of Eqs.(11-13) models the observed local structures (positive samples), while each of the second terms models the way the negative samples drawn from the noise distribution (we adopt uniform distribution in this paper). $\sigma(x) = 1/(1 + exp(-x))$ denotes the sigmoid function. v_n and r_l denote the negative samples for nodes and relation edges drawn from a uniform distribution where v_i, r_l and v_n cannot constitute the fact triplet, and K is the number of the negative samples. Then the partial derivative of Eq.(11) w.r.t. $\vec{u}_i$ can be rewritten as:

$$\begin{aligned} \frac{\partial O}{\partial \vec{u}_i} &= \{[1 - \sigma(\vec{u}_j^{'T}(\vec{u}_i + \vec{u}_{r_s}) + \vec{u}_k^{'T}(\vec{u}_i + \vec{u}_{r_t}))](\vec{u}_j^{'} + \vec{u}_k^{'}) \\ &- \sigma(\vec{u}_j^{'T}(\vec{u}_i + \vec{u}_{r_s}) + \vec{u}_n^{'T}(\vec{u}_i + \vec{u}_{r_l}))(\vec{u}_j^{'} + \vec{u}_n^{'})\} * \omega_{ij} * \omega_{ik} \\ &+ \{[1 - \sigma(\vec{u}_i^{'T}(\vec{u}_j + \vec{u}_{r_s}) + \vec{u}_k^{'T}(\vec{u}_i + \vec{u}_{r_t}))]\vec{u}_k^{'} \\ &- \sigma(\vec{u}_i^{'T}(\vec{u}_j + \vec{u}_{r_s}) + \vec{u}_n^{'T}(\vec{u}_i + \vec{u}_{r_l}))\vec{u}_n^{'}\} * \omega_{ji} * \omega_{ik} \end{aligned} \quad (14)$$

With reference to Eq.(14), the updating rule for the embedding vector $\vec{u}_i$ can be obtained. The target vectors $\vec{u}_i^{'}$ and relation vectors $\vec{u}_{r_l}$ can be obtained similarly. They are not listed due to the page limit.

4 Experiment

To evaluate the performance of the proposed multi-relational network embedding (MNE), we employ two well-known benchmark datasets, namely, WN18 and FB15K which are extracted from the real-world multi-relational networks WordNet [Miller, 1995] and Freebase [Bollacker *et al.*, 2008] respectively. Table 1 tabulates their statistics where tri-nodes refers to the nodes conforming a triangular structure in networks. We compare our proposed MNE with several existing methods in trans-family, including TransE, TransH and TransR where the two settings "unif" and "bern" to sample negative instances are used for the embedding learning [Lin *et al.*, 2015b]. We also compare our proposed approach with the state-of-the-art approaches for network embedding, including DeepWalk and LINE. [4] For LINE, both first-order proximity and second-order proximity terms are investigated for comparison, denoted as LINE-1st-order and LINE-2nd-order respectively. The experiments are evaluated using 80/20 rule for the train-test split.

Table 1: Statistics of the datasets used for evaluation

dataset	#Entity	#Relation	#Triplet	#Tri-nodes
WN18	40943	18	151442	895 (2.19%)
FB15K	14951	1345	592213	6198 (41.46%)

4.1 Triplet Classification

The triplet classification task has been widely investigated for the performance evaluation of representation learning approaches, which is usually translated into a binary classification task to judge whether a given triplet is a fact or not in a given knowledge base.

Evaluation Protocol In this task, we perform binary classification as in [Grover and Leskovec, 2016]. The triplet facts (h, r, t) appeared in the dataset are taken as the positive samples. And we randomly sampled the same number of triplets that have not appeared in the dataset as the negative triplets. We concatenate the obtained low-dimensional vectors of the head entity, relation and tail entity as the input of a classifier. Both logistic regression (LR) and support vector machine (SVM) are adopted for the classifier with similar results achieved. We adopt LR for its efficiency in this paper. And we use the classification accuracy as the evaluation criterion.

Results Table 2 shows the performance comparison among the existing approaches for triplet classification. We observe that: (1) The proposed MNE and the trans-family perform consistently better than the network embedding methods (i.e. DeepWalk and LINE) which treat the relations semantically indistinguishable; (2) For both benchmark datasets, our proposed approach MNE outperforms all the baseline methods; (3) The trans-family does not work well on FB15K while our proposed MNE can still achieve high accuracy. As reported in Table 1, FB15K is a far more dense multi-relational network with more relation types than WN18. The relation-specific local structures are intuitively more complex. And in FB15K dataset, there are more nodes with the triangular structures compared to WN18. That accounts for the performance degradation of trans-family enforcing the constraints of $h + r = t$.

[4] As LINE and Deepwalk can only deal with single relational networks, we treat the linkages of various types between two nodes in multi-relational networks as a weighted single relation.

Table 2: Performance comparison on triplet classification

WN18	Methods	MNE	LINE-1st-order	LINE-2nd-order	DeepWalk	TransE(bern)
	Acc.	**86.74%**	50.47%	54.34%	53.28%	81.31%
	Methods	TransE(unif)	TransH(bern)	TransH(unif)	TransR(bern)	TransR(unif)
	Acc.	80.42%	81.44%	80.83%	80.43%	80.73%
FB15K	Methods	MNE	LINE-1st-order	LINE-2nd-order	DeepWalk	TransE(bern)
	Acc.	**90.08%**	58.67%	70.52%	69.31%	70.46%
	Methods	TransE(unif)	TransH(bern)	TransH(unif)	TransR(bern)	TransR(unif)
	Acc.	71.40%	71.72%	70.98%	70.49%	71.48%

Table 3: Performance comparison on link prediction

WN18	Methods	MNE	LINE-1st-order	LINE-2nd-order	DeepWalk	TransE(bern)
	Acc.	**85.04%**	50.94%	54.12%	54.54%	82.76%
	Methods	TransE(unif)	TransH(bern)	TransH(unif)	TransR(bern)	TransR(unif)
	Acc.	82.46%	83.48%	82.22%	82.36%	82.38%
FB15K	Methods	MNE	LINE-1st-order	LINE-2nd-order	DeepWalk	TransE(bern)
	Acc.	**91.81%**	59.27%	64.13%	69.55%	69.40%
	Methods	TransE(unif)	TransH(bern)	TransH(unif)	TransR(bern)	TransR(unif)
	Acc.	71.23%	69.77%	72.46%	71.35%	71.77%

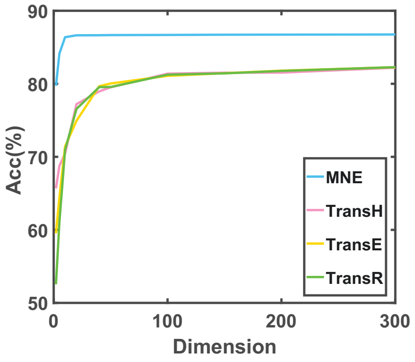


Figure 5: Acc. vs. Dimension

4.2 Link Prediction

Link prediction is to predict the missing *h* or *t* for a triplet fact (h, r, t) in a given KG. That is to obtain the best answer of *t* given (h, r) or to obtain the best answer of *h* given (r, t).

Evaluation Protocol Again, the link prediction problem can be posed as a binary classification problem by employing the low-dimensional vectors obtained from our proposed model. While the triplets in a KG can form the positive samples, the negative samples can be generated by corrupting each triplet of fact (h, r, t) with the head (h) or tail (t) replaced. Compared to triplet classification, the testing set will no longer included in the dataset for representation learning. Again, a LR classifier is trained by using the obtained low-dimensional vectors and tested on the corrupted edges. Again, we use the classification accuracy as the evaluation criterion.

Results The evaluation results are shown in Table 3. We made similar observations as those for triplet classification. In particular, the proposed MNE and the trans-family are performing obviously better than the network embedding methods on WN18. The trans-family methods do not perform well on FB15K. The phenomenon further confirms that the triangular structures in multi-relational networks will degrade the performance of the trans-family. MNE outperforms all the other methods on both WN18 and FB15K consistently.

Among the methods proposed for multi-relational networks, we also compare their performances on the triplet classification (WN18) under the settings using representations of different dimensions. The results are shown in Fig.5. We observe that: 1) There is a positive correlation between the classification accuracy and the dimension. After reaching a specific dimension, the classification accuracy converges; 2) MNE outperforms other state-of-the-art methods for all the dimensionality settings. In particular, MNE can work very well even at a very low dimension (2 to 5); 3) MNE converges when the dimension reaches 20, while the other methods reach the good performance when the dimension is around 100. We conclude that MNE could obtain a more compact representation compared with other approaches. Besides, similar to LINE, we adopt the negative sampling to substantially reduce the computational cost of learning, which allows MNE to scale up to the network of large size.

5 Conclusion

In this paper, we propose a novel multi-relational network embedding model. Many existing knowledge graph embedding methods share an intrinsic limitation of adopting a hard constraint on the inferred embedding. By defining an objective function which can implicitly preserve triangular and parallelogram structures, the proposed model can give more flexible embedding results. Negative sampling are used to reduce the computational cost for the learning process. The extensive experiments conducted on two real world datasets demonstrate that our proposed model outperforms a number of state-of-the-art embedding methods. This paper only explores the local structures to obtain embedding without considering other information carried in the network. We would like to explore the idea of incorporating semantic information in our framework for the future work.

References

[Ahmed *et al.*, 2013] Amr Ahmed, Nino Shervashidze, Shravan M. Narayanamurthy, Vanja Josifovski, and Alexander J. Smola. Distributed large-scale natural graph factorization. In *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013*, pages 37–48, 2013.

[Bollacker *et al.*, 2008] Kurt D. Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, Vancouver, BC, Canada, June 10-12, 2008*, pages 1247–1250, 2008.

[Bordes *et al.*, 2013] Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pages 2787–2795, 2013.

[Goldberg and Levy, 2014] Yoav Goldberg and Omer Levy. word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *CoRR*, abs/1402.3722, 2014.

[Grover and Leskovec, 2016] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pages 855–864, 2016.

[Lin *et al.*, 2015a] Yankai Lin, Zhiyuan Liu, Huan-Bo Luan, Maosong Sun, Siwei Rao, and Song Liu. Modeling relation paths for representation learning of knowledge bases. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 705–714, 2015.

[Lin *et al.*, 2015b] Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA.*, pages 2181–2187, 2015.

[Liu *et al.*, 2016] Li Liu, William K. Cheung, Xin Li, and Lejian Liao. Aligning users across social networks using network embedding. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 1774–1780, 2016.

[Mikolov *et al.*, 2013] Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pages 3111–3119, 2013.

[Miller, 1995] George A. Miller. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, 1995.

[Perozzi *et al.*, 2014] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: online learning of social representations. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 701–710, 2014.

[Tang *et al.*, 2015] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. LINE: large-scale information network embedding. In *Proceedings of the 24th International Conference on World Wide Web, WWW 2015, Florence, Italy, May 18-22, 2015*, pages 1067–1077, 2015.

[Wang *et al.*, 2014a] Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. Knowledge graph and text jointly embedding. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1591–1601, 2014.

[Wang *et al.*, 2014b] Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada.*, pages 1112–1119, 2014.

[Wang *et al.*, 2016] Daixin Wang, Peng Cui, and Wenwu Zhu. Structural deep network embedding. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pages 1225–1234, 2016.

[Xiao *et al.*, 2015] Han Xiao, Minlie Huang, Yu Hao, and Xiaoyan Zhu. Transg : A generative mixture model for knowledge graph embedding. *Computer Science*, 2015.