 doi:10.20944/preprints201801.0147.v1

Article

TransPath: Representation Learning for Heterogeneous Information Networks via Translation Mechanism

Yang Fang *, Xiang Zhao and Zhen Tan

College of System Engineering, National University of Defense Technology, Changsha, China; fangyang12@nudt.edu.cn

* Correspondence: fangyang12@nudt.edu.cn; Tel.: +86-18858473547

Abstract: In this paper, we propose a novel network representation learning model TransPath to encode heterogeneous information networks (HINs). Traditional network representation learning models aim to learn the embeddings of a homogeneous network. TransPath is able to capture the rich semantic and structure information of a HIN via meta-paths. We take advantage of the concept of translation mechanism in knowledge graph which regards a meta-path, instead of an edge, as a translating operation from the first node to the last node. Moreover, we propose a user-guided meta-path sampling strategy which takes users' preference as a guidance, which could explore the semantics of a path more precisely, and meanwhile improve model efficiency via the avoidance of other noisy and meaningless meta-paths. We evaluate our model on two large-scale real-world datasets DBLP and YELP, and two benchmark tasks similarity search and node classification. We observe that TransPath outperforms other state-of-the-art baselines consistently and significantly.

Keywords: heterogeneous information network; representation learning

0. Introduction

Large network data become larger and ubiquitous nowadays, such as social networks, research publication networks and World Wide Web, which stimulate the research interest of mining and analyzing the information behind them [1]. Common network analysis tasks often involve node classification [2] or clustering [3], link prediction [4] and similarity search [5] [6]. A traditional approach to accomplish those tasks needs many artificial works such as designing domain-specific representation of data manually by domain experts, which is very time-consuming and inefficient. To address such issue, network representation learning (NRL) is proposed which embeds the structure and semantic information of the network into a low-dimensional space [7]. Heterogeneous information networks (HINs) are networks with multi-typed nodes and edges, which are more common in real life than homogeneous networks. Obviously in a HIN, relationships between nodes are more complex than those in a homogeneous network, thus more difficult to be represented. Classical network embedding methods like DeepWalk [8] and node2vec [9] leveraged random walks to explore the structural information of the network and utilized SkipGram [10] to project the node into a low-dimensional vector. LINE [11] represented the first-order proximity and second-order proximity of the network so as to capture the local and neighborhood network structures. All these models only aim to learn the representation for homogeneous networks, and perform poorly when applied to a HIN. The main reason is explained below. In a HIN, for example, in DBLP [12], a bibliographic dataset of computer science, nodes have various types, e.g., authors (A), papers (P), venues (V). It is common for us to explore the co-authors of a paper, or papers published in one venue. However, random walks are blind to those multi-typed nodes, and traditional SkipGram as a word2vec embedding method [10] is homogeneous network-oriented.

 doi:10.20944/preprints201801.0147.v1

Recently some approaches propose a concept 'meta-path' to help effectively represent a HIN. A meta-path is a sequence of vertice types connected by edge types, which represents a specific relationship. For example, consider a meta-path in DBLP: $A \rightarrow P \rightarrow V \rightarrow P \rightarrow A$, which denotes that two authors are connected by their publications in the same venue. Similarly, meta-path $A \rightarrow P \rightarrow A$ represents co-author relationships. From this we could observe that different meta-paths represent different semantic information, meanwhile they intuitively encode the structure information. Therefore, by fully leveraging such information behind meta-paths could be very helpful to learn the representation of a whole network.

In this paper, we propose a network representation learning model named TransPath, which provides a new perspective to encode the rich information behind meta-paths. We first introduce some concepts in knowledge graph embedding. Knowledge graphs are large-scale multi-relational structures such as Freebase [13], Wordnet [14] and YAGO [15], which are composed of larger amount of facts. A fact (knowledge) in a knowledge graph is described as a triplet (head_entity, relation, tail_entity), usually denoted as (h, r, t). In order to compress the knowledge graph into a low-dimensional space, a classical embedding model TransE [16] regards a relation as a translating operation between head entity and tail entity. The mathematical mechanism is represented as: $\mathbf{h} + \mathbf{r} = \mathbf{t}$, where $\mathbf{h}$, $\mathbf{r}$ and $\mathbf{t}$ denote the embedding vectors for head entities h, relations r, and tail entities t, respectively. Inspired by the translation mechanism in TransE, TransPath regards a meta-path as a translating operation between the first and the last node of the path. In a knowledge graph, a relation actually plays the same role as an entity when training since it has a specific label, however, in a network, an edge between two vertices does not have a label, so we extend the 'relation' to a meta-path since a meta-path could be labeled as, e.g., 'A-P-A' or 'A-P-V-P-A'.

Before applying the representation learning mechanism introduced above, we should conduct a data preprocess operation to extract meta-paths from the network. Traditional meta-path extraction method is based on random walks with a limited length which is much shorter than a common random walker. Here we propose a user-guided meta-path extraction to better capture the semantics of the network. Firstly, we confine the meta-paths to be symmetric, that is, the first node type is the same with the last one. Take similarity search task as an example, if we want to find two similar objects, they should not only be strongly connected, but the connection should also share the same structure, that is, the relation between peers should be symmetric. In addition, we limit the meta-paths to be merely several schemas guided by the user. For example, in oder to find the similar authors in DBLP, the meta-path schemas needed are only 'A-P-A' and 'A-P-V-P-A', which indicates that similar authors maybe co-authors of a paper or publish in the same venue. Adopting too many other meta-paths schemas may be noisy and inefficient. Another advantage of choosing specific schema of meta-paths is that semantic relationships between different types of nodes can be properly encoded. For instance, under the schema 'A-P-V-P-A', the path is biased towards paper nodes (P) given its previous author node(A), exactly following the semantics of this path.

We conduct the experiments on two real-life network datasets DBLP and Yelp. The details of these two datasets will be introduced later. Experiment results show that TransPath outperforms classical state-of-the-art network embedding models significantly and consistently. It demonstrates our proposed model's power and efficiency on modeling relationships between nodes and edges, thus representing the whole network effectively.

The major contribution of the paper can be summarized into three ingredients:

- We propose a novel network embedding model TransPath, which encodes the network semantic and structure information by meta-paths, and regards the meta-path as a translation from the first node to the last node in the path;
- We propose a user-guided meta-path extraction method, which limits the meta-path schemas so as to better leverage the semantic information of the path, and improve the efficiency by avoiding other noisy and meaningless meta-paths; and

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 January 2018 doi:10.20944/preprints201801.0147.v1

- The new model is evaluated and compared with existing models on real-life benchmark datasets and tasks, and the experiment results further verify that TransPath outperforms state-of-the-art alternatives consistently and significantly.

The rest of the paper is structured as follows. We introduce the related work in Section 1, and then introduce the concept of HINs, meta-paths and problem definition in Section 2. We justify the intuitions of our method with its theoretical analysis in Section 3. Next, we conduct the empirical experiment studies using real-wold and present the results in Section 4. Finally, we conclude our findings in Section 5.

1. Related Work

In order to capture the embedded information of a network, traditional approaches depend much on human labor and knowledge, which is very insufficient. To address such issue, network representation learning (NRL) has been proposed to automatically learn the latent feature vectors to embed the whole network. NRL could be traced back to those factorization models which factorize the network into a matrix or tensor, e.g., the adjacency matrix [17]. However such models tend to suffer from expensive computational cost and source data sparsity, thus making these models both inefficient and ineffective.

With the development of machine learning techniques, many recent approaches are equipped with those techniques, e.g., word embedding model. DeepWalk [8] extracted random walks from the network and performed SkipGram, an efficient word2vec embedding model, on those walkers to learn the network representation. node2vec [9] adopted a biased random walks strategy to better explore the structure of the network. LINE [11] captured the first-order proximity and second-order proximity simultaneously to embed both the local and neighborhood structure information. Text-associated DeepWalk (TADW) [18] incorporated text information with the matrix factorization based DeepWalk. Max-margin DeepWalk (MMDW) [19] used vertices labeling information to learn discriminative network embedding. GrapRep [20] further explored high-order proximities in DeepWalk to capture global structure of the network. CANE [21] utilized mutual-attention mechanism to learn context-aware network embedding. SDNE [22] captured the non-linear structural information via a semi-supervised deep model equipped with autoencoder neural network.

However, approaches mentioned above only focus on learning representation for homogeneous network, and perform poorly when dealing with the complex heterogeneous network. Therefore, several models are proposed to learn the embeddings of a HIN. PTE [23] defined the conditional probability of vertices of one type generated by vertices of another type, and made the conditional distribution be close to its empirical distribution. HINE [24] proposed meta path based proximity, and preserved such proximity through minimizing the distance between nodes' joint probability defined by Sigmoid function and their empirical probability. Metapath2vec [25] proposed a heterogeneous SkipGram with its context window confined to one specific type. Esim [26] introduced the meta-path guided sampling strategy, however, some parts of its objective function like the multiplication of meta-paths are illy defined. HIN2Vec [27] introduced the Hadamard multiplication of nodes and meta-path to capture the semantic and structure information of a HIN, but it samples all meta-paths shorter than a certain length which is very inefficient.

In order to model the relationships between nodes and meta-paths, we borrow some concepts in knowledge graph. Knowledge graphs such as Freebase [13], Wordnet [14] and YAGO [15] contain a large amount of facts, which are described as triplets (head_entity, relation, tail_entity), denoted as (h, r, t). A classical knowledge embedding model TransE [16] regarded a relation as a translation operation between head entity and tail entity in the representation space, that is, $\mathbf{h} + \mathbf{r} = \mathbf{t}$, where $\mathbf{h}$, $\mathbf{r}$ and $\mathbf{t}$ denote the embedding vectors for head entities h, relations r, and tail entities t, respectively. Inspired by this translation mechanism, we argue that such mechanism also exists in a HIN between nodes and metapaths, and propose a translation based model, TransPath.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 January 2018 doi:10.20944/preprints201801.0147.v1

2. Preliminaries

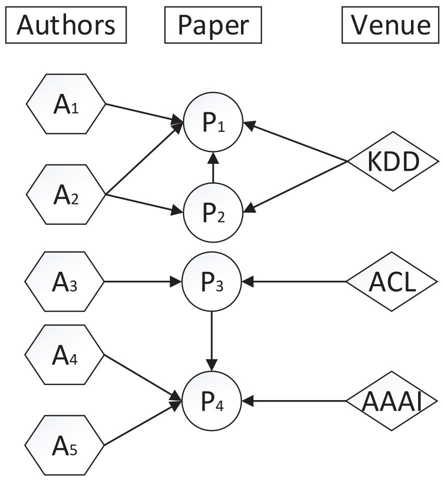


Figure 1. A Mini HIN in DBLP

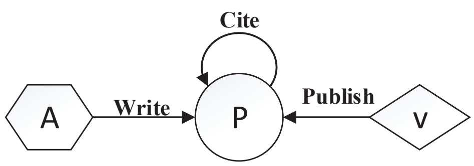


Figure 2. A Schema of a HIN

In this section, we formalize the problem of representation learning for heterogeneous information networks (HINs). First in the following, we introduce the definition of a HIN:

Definition 1. ***Heterogeneous Information Network.*** *A heterogeneous information network (HIN) is defined as a directed graph $G = (V, E, T)$ where each vertice v is associated with an object type mapping function $\phi : V \rightarrow T_V$ and each edge is represented by a link type mapping function $\varphi : E \rightarrow T_E$. T_V and T_E denote the sets of vertice and edge types. If $|T_V| > 1$ or $|T_E| > 1$, the network should be a heterogeneous network, otherwise, it is a homogeneous network.*

Figure 1 illustrates a mini HIN in a bibliography network DBLP. From Figure 1 we can see that this HIN contains five authors, three papers, and three venues. In detail, author A_1 and A_2 are co-authors of paper P_1, author A_2 writes both paper P_1 and P_2, paper P_2 cites paper P_1, venue KDD publishes both paper P_1 and P_2, other edges could be inferred in the same way. For simplicity, we assume that each object only has one single corresponding type. In the following, we introduce the concept of HIN schema:

Definition 2. ***HIN schema.*** *Given a HIN $G = (V, E, T)$ with mapping functions $\phi : V \rightarrow T_V$ and $\varphi : E \rightarrow T_E$, the network schema T_G of G is defined as a directed graph on node types T_V and edge types T_E, denoted as $T_G = (T_V, T_E)$.*

The HIN schema defines all allowable edge types between node types. Figure 2 shows the schema of HIN in Figure 1, in which nodes A, P and V represent the type author, paper, and venue, respectively. Different edge types are also illustrated, such as 'write', 'publish' and 'cite'. The definition of meta-path is introduced as follows:

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 January 2018 doi:10.20944/preprints201801.0147.v1

Definition 3. ***Meta-path****.* *Given a HIN $G = (V, E, T)$, a meta-path P is a sequence of compatible node types and edge types defined on the graph of network schema $T_G = (T_V, T_E)$, and is denoted in the form of $T_{V_1} \xrightarrow{T_{E_1}} T_{V_2} \xrightarrow{T_{E_2}} \ldots \xrightarrow{T_{E_l}} T_{V_{l+1}}$. This meta-path defines a composite relation $T_E = T_{E_1} \circ T_{E_2} \circ \ldots T_{E_l}$ between node types T_{V_1} and $T_{V_{l+1}}$, where $\circ$ denotes the composition operation on relations.*

This work aims to learn the representation of a HIN. We formally define the problem as follows:

Definition 4. ***Heterogeneous Network Representation Learning****.* *Given a HIN $G = (V, E, T)$, representation learning is to learn a mapping function $f : V \rightarrow \mathbb{R}^d$ which projects each node $v \in V$ to a d-dimensional latent space $\mathbb{R}^d$, where $d \ll |V|$.*

In this work, we present a translation based model to realize this idea. Through embedding the relationships, i.e., meta-paths between nodes, our model is able to capture the semantic and structure information in a HIN and learn the node vectors and meta-path vectors simultaneously. The learned node representations are useful for many network mining tasks. For example, the embedding vector of nodes can be applied to traditional machine learning model like SVM as input features, so as to realize node classification or clustering tasks.

3. Proposed Model

In this section, we propose a novel heterogeneous network representation model TransPath. We separate the framework of TransPath into two phases: training data preprocess and representation learning. In specific, we first present the details in representation learning phase, which is based on translation mechanism. Next, we introduce how to sample the training data, which leverages the user-guided meta-paths to remarkably improve model efficiency and effectiveness.

3.1. Translation-Based Representation Learning

Inspired by the translation mechanism in knowledge graph embedding, we assume that meta-paths connecting two nodes can also be described as translating operations in representation space.

In specific, we take a triplet (h, p, t) as the input of TransPath, where h and t denote the first and last node of a meta-path, respectively, and p denotes the meta-path itself. $\mathbf{h}$, $\mathbf{p}$ and $\mathbf{t}$ are one-hot vectors representing h, p and t, where the length of $\mathbf{h}$ and $\mathbf{t}$ are $|V|$, and the length of $\mathbf{p}$ is $|P|$. In the latent space, $\mathbf{h}$ and $\mathbf{t}$ are projected into latent vectors $W'_{\mathbf{h}}\mathbf{h}$ and $W'_{\mathbf{t}}\mathbf{t}$, in which $W_{\mathbf{h}}$ and $W_{\mathbf{t}}$ are two $|V| \times d$ matrices describing the projection, $W'_{\mathbf{h}}$ and $W'_{\mathbf{t}}$ are their transpose matrices, and d is the latent space dimensionality. Similarly, we project $\mathbf{p}$ to the latent space via projection matrix $W_{\mathbf{p}} \in \mathbb{R}^{|P| \times d}$, the corresponding latent vector denoted as $W'_{\mathbf{p}}\mathbf{p}$.

For each triplet (h, p, t) satisfying the translation mechanism, the representation of node h plus the representation of meta-path p is expected to be close to the representation of node t, meanwhile far away from other entities. Using the notification above, we present the mathematical formula of such mechanism as follows:

$$W'_{\mathbf{h}}\mathbf{h} + W'_{\mathbf{p}}\mathbf{p} \approx W'_{\mathbf{t}}\mathbf{t} \tag{1}$$

A distance function should be applied to evaluate the degree that (h, p, t) matches the Eq. 1, and in practice, L_1 or L_2 norms is the most common choice. Consequently, the score function of TransPath can be presented as follows:

$$f_{\mathbf{p}}(\mathbf{h}, \mathbf{t}) = \|W'_{\mathbf{h}}\mathbf{h} + W'_{\mathbf{p}}\mathbf{p} - W'_{\mathbf{t}}\mathbf{t}\|^2_{L_1/L_2}. \tag{2}$$

To learn the representation above, for each triplet (h, p, t), we adopt the negative sampling strategy to create the corrupted sample (h', p', t'), which will be introduced in detail in next subsection. For training purpose, we minimize the margin-based ranking loss function over the training set as follows:

$$\mathcal{L} = \max(f(\mathbf{h}, \mathbf{r}, \mathbf{t}) + \gamma - f(\mathbf{h}', \mathbf{r}', \mathbf{t}'), 0), \tag{3}$$

where $\gamma > 0$ is a margin hyperparameter and the loss function encourages the discrimination between positive triplets and negative triplets.

3.2. Training Data Preprocess

To preprocess the train data, we adopt a user-guided meta-path algorithm to sample the triplets introduced above from the original HIN, in the form of (h, p, t). Notice that we have to balance the data quality, i.e., the training data should fully cover the semantic and structure information of the HIN, and the calculation efficiency, i.e, to randomly sample the training data rather than enumeration. Therefore, it is crucial to develop an efficient algorithm to extract samples for training data preprocess.

3.2.1. User-Guided Meta-Path Extraction

To address such issue above, traditional method is to employ random walks on a HIN to generate paths between different types of nodes. The transition probability at step i is denoted as $p(v^{i+1}|v^i)$, which is a normalized probability distributed on the neighbor nodes regardless of their types. However, heterogeneous random walks are biased to nodes with a dominant number of paths and concentrated nodes, thus failing to fully capture the structure information of a HIN. In light of this problem, we propose meta-path based random walks to generate the paths as input data.

As discussed above, a meta path P is defined in the form of $T_{V_1} \xrightarrow{T_{E_1}} T_{V_2} \xrightarrow{T_{E_2}} \dots \xrightarrow{T_{E_l}} T_{V_{l+1}}$. For example, a meta path 'A-P-A' denotes that two authors(A) co-write a paper(P), and 'A-P-V-P-A' represents that papers (P) of two authors (A) are published in the same venue (V).

Here we present the mechanism of meta-path guided random walkers. Given a HIN $G = (V, E, T)$ and a meta path schema $P : T_{V_1} \xrightarrow{T_{E_1}} T_{V_2} \xrightarrow{T_{E_2}} \dots \xrightarrow{T_{E_l}} T_{V_{l+1}}$, the transition probability at step i is adapted as follows:

$$p(v^{i+1}|v^i) = \begin{cases} \dfrac{1}{|N_{t+1}(v_t^i)|} & (v^{i=1}, v_t^i) \in E, \phi(v^{i+1}) = t+1, \\ 0 & (v^{i=1}, v_t^i) \in E, \phi(v^{i+1}) \neq t+1, \\ 0 & (v^{i=1}, v_t^i) \notin E, \end{cases} \tag{4}$$

where $v_t^i \in V_t$ and $N_{t+1}(v_t^i)$ denote the V_{t+1} type of the neighbor nodes of node v_i^t. From this, the flow of the path is conditioned by the pre-defined meta-path P, i.e., $v^{i+1} \in V_{t+1}$. Furthermore, we confine the meta-paths to be symmetric. For one thing, it could facilitate its recursive guidance for random walkers [28] [6] [29], presented as follows:

$$p(v^{i+1}|v_t^i) = p(v^{i+1}|v_1^i), if\ t = l. \tag{5}$$

More importantly, a symmetric meta-path could better capture the semantics of a HIN especially for some practical applications such as similarity search. Two similar peer objects should not only be strongly connected, but their connection should also share the same structure. In short, the relation of peer should be symmetric, thus restricting the meta-paths to be symmetric as well.

As for user guided meta-paths, limiting the meta-paths to be only several schemas could not only improve the efficiency of our model, but also help capture the specific semantics for a task needed. For example, under the meta-path schema 'A-P-V-P-A', the path is biased towards paper nodes (P) given its previous node of type author (A), instead of other types of nodes like author (A) itself, thus following the semantics of this path. In addition, in order to find the most similar authors, only schemas 'A-P-A' and 'A-P-V-P-A' are needed, since they demonstrate that similar authors are

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 January 2018 doi:10.20944/preprints201801.0147.v1

probably co-authors or publish papers in the same venue. Other meta-path schemas should be noisy thus influencing the effectiveness and efficiency of the representation model.

3.2.2. Weighted Combination of Meta-Path Schemas

In order to cover the subtle semantic meanings of a HIN, sometimes a single meta-path is not enough, the weighted combination of n meta-paths could improve the performance of our model. Suppose $w_1, w_2, \ldots, w_n$ are the weights ($\forall i, w_i > 0$ and $\sum_{i=1}^{n} w_i = 1$), then the unified loss function is the weighted sum over the individual meta-path loss function. Mathematically, $\mathcal{L}_{all} = \sum_{i=1}^{n} w_i \mathcal{L}_{M_i}$.

The weights for each meta-path schema could either be explicitly assigned by users or learned during the training. Because weight learning is beyond the scope of this paper, we simply adopt the grid searches to obtain the best wighted combination.

3.2.3. Negative Sampling

We adopt the idea of negative sampling [30] for training. Before applying such mechanism, we should filter the training data as some paths extracted may be 'cycles', for example, a training data entry could be $\langle A_1, A-P-A, A_1 \rangle$. Obviously, it hurts the quality of the training data, because a path instance with cycles is actually meaningless, e.g., an author has no relationship of co-authors with himself. Therefore, we drop the inputing data with any cycle by checking duplicate nodes.

As for the negative sampling, we generate negative inputing data with those positive samples simultaneously. For each extracted positive triplet as input data, $\langle h, p, t \rangle$, we generate negative triplets by randomly replacing either head nodes h' or tail nodes t'. We further filter the training data again as some triplets with replaced nodes are still positive, so we need to make sure that those triplets are truly negative. The reason we do not replace p is that for user-guided meta-path sampling, the choice of meta-path schema is rather limited, thus triplets after replacing p are very likely to be still positive.

4. Experiments

In this section, we first introduce two real-world large-scale datasets **DBLP** [12] and **YELP**[1], along with the state-of-the-art baseline modes. Then we evaluate our proposed model TransPath on two specific tasks,i.e., similarity search and node classification, and compare its performance with baselines.

4.1. Datasets and Baselines

DBLP is a bibliographic dataset mainly containing computer science researches. To form a HIN, we extract papers published between 1994 and 2014 from 20 conferences in 4 research fields. This HIN contains three types of nodes, i.e., papers(P), authors(A) and venues (V), and edges like authorships (P-A), authors' publishing venues (A-V) and paper's publishing venues (P-V). The interested meta-paths, i.e., user-guided meta-paths are the co-authorship meta-path A-P-A and the shared venue meta-path A-P-V-P-A.

YELP is released in Yelp Dataset Challenge, which is a social media dataset. To form a HIN, similarly, we extract data of the top 10 cities with the most businesses. This HIN contains four types of nodes, businesses(B), reviews of businesses (R), words in business names (N) and words in reviews (W), and edges like name word of a business (B-N), review of a business (B-R), and words of a review (R-W). Similarly the interested meta-paths we choose for training are the shared name word meta-path B-N-B and the shared review word meta-path B-R-W-R-B.

We compare the proposed model TransPath against six state-of-the-art network representation models, among which DeepWalk, node2vec and LINE are originally designed for homogeneous

[1] https://www.yelp.com/dataset_challenge.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 January 2018 doi:10.20944/preprints201801.0147.v1

networks. They are applied to networks which treat all nodes and edges as homogeneous ones. The other three namely, PTE, Esim and HIN2Vec are designed to address the learning issue on HIN representation.

DeepWalk [8] adopted uniform random walks to capture the structure information of the HIN, and conducted homogeneous SkipGram to learn the representation of the network.

node2vec [9] was derived from DeepWalk which used biased random walks to better explore the structure of the network and also applied SkipGram to learn the network embedding.

LINE [11] encoded the first-order proximity and second-order proximity simultaneously which could explore both the structure and the semantic information of a network, and it concatenated the learned vectors of first-order and second-order together to form the final representation.

PTE [23] decomposed a HIN to a set of bipartite networks by edge types, and defined the conditional probability of nodes of one type given other nodes of other types, and made such conditional distribution close to its empirical distribution.

Esim [26] applied the meta-path guided sampling strategy and could also learn the representation of meta-path, but its objective function is illy defined.

HIN2Vec [27] learned the embedding through the Hadamard multiplication of nodes and meta-path but it sampled all meta-paths shorter than a certain length which led to its inefficiency.

4.2. Similarity Search

In this section, we evaluate the models on similarity search, which aims to find the similar objects of a specific node. We first introduce the details of the experimental set up and then present the results generated by default parameters. We observe that our model TransPath outperforms those baselines significantly and consistently. At last, we analyze the parameter sensitivity.

4.2.1. Experimental Setup

We first introduce two groups which are labeled by human experts in DBLP. The first is research area, where 3750 authors are divided into four areas, that is, 'data mining', 'machine learning', 'database', and 'information retrieval'. The second is research group and it is more fine-grained compared to the research area, where 103 authors are separated into four groups, that is, 'JiaWei Han', 'Christos Faloutsos', 'Dan Roth' and 'Michael I. Jordan'. For YELP, similarly, we also set two groups labeled by human experts. One is business type, where 881 businesses are separated into four types, namely, 'restaurant', 'shopping', 'hotel' and 'health and medical'. Business with multiple types are discarded so as to avoid ambiguity and these four types are the most popular ones. The other is restaurant type, as the majority of the businesses belong to restaurants, and we further divide 270 restaurants into 5 cuisine types, namely, 'Chinese', 'Indian', 'Japanese', 'Greek' and 'Mexican'.

To evaluate the results of the experiment, we introduce the cosine similarity score and AUC evaluation metric. For any pair of nodes u and v, their similarity score is calculated via the cosine similarity between their learned latent vectors $\mathbf{u}$, $\mathbf{v}$, mathematically, $\mathrm{sim}(\mathrm{u}, \mathrm{v}) = \frac{\mathbf{u}^T\mathbf{v}}{\|\mathbf{u}\|\cdot\|\mathbf{v}\|}$. Furthermore, we could see that cosine similarity is equivalent to Euclidean distance if $\|\mathbf{u}\| = \|\mathbf{v}\| = 1$, which makes the top-k similar nodes of the given node u efficiently explored via approximate nearest neighbors [31] after normalizations.

However, it is hard to obtain the explicit rankings among all pairs of nodes, in which case we can evaluate the similarity according to the labels of the node groupings $l(\cdot)$, thus improving model efficiency. Intuitively, to rank other nodes given an individual node based on the similarity score, the nodes from the same group (similar ones) tend to be ranked top in the list while the dissimilar ones ranked correspondingly at the bottom. More specifically, we adopt the AUC score, and if a

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 January 2018 doi:10.20944/preprints201801.0147.v1

model explores the similarity better, its AUC score should be higher. Mathematically, the AUC score is defined as follows:

$$\text{AUC} = \frac{1}{|V|}\sum_{v\in V}\frac{\sum_{v,v'\in V\wedge l(u)=l(v)\wedge l(u)\neq l(v')} 1_{\text{sim}(u,v)>\text{sim(u,v}')}}{\sum_{v,v'\in V\wedge l(u)=l(v)\wedge l(u)\neq l(v')} 1}. \tag{6}$$

Notice that the models that conduct the similarity search over the whole dataset, while the AUC metric is adopted merely on the subset of nodes where it has the group labels. The subset is usually small as pairwise similarities among the subset are needed when calculating the AUC score.

As for the default parameters, the dimension of the learned node and meta-path is set to 128, the negative sampling ratio is set to 5 and the sampling times is set to 2 million. The initial learning rate in stochastic gradient descent is set to 0.01. The combination weights for different meta-paths are learned by the grid searches among different groupings. For other baselines, we set the best configuration introduced in their original paper so as to achieve good performances. We do not conduct this task on HIN2Vec, as its source code is not available, and we directly borrow its results on next task node classification.

4.2.2. Evaluation

Table 1. AUC Evaluation on DBLP dataset

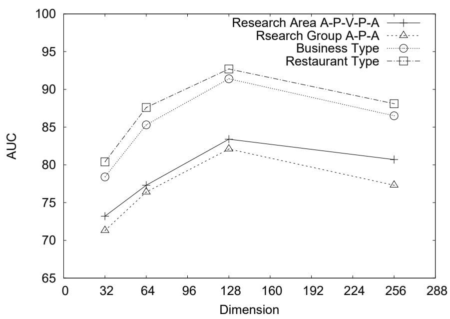
Model	Research Area	Research Gruop
DeepWalk	50.12%	50.31%
LINE	51.83%	52.89%
node2vec	52,47%	53.42%
PTE	74.40%	77.07%
Esim	83.58%	81.46%
TransPath(A-P-A)	55.25%	83.64%
TransPath(A-P-V-P-A)	85.35%	56.33%
TransPath(Combination)	**86.23**%	**85.35**%

Table 2. AUC Evaluation on YELP dataset

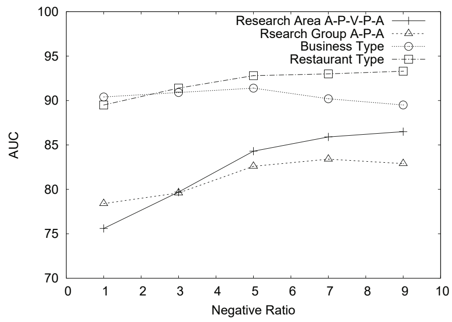
Model	Business Type	Restaurant Type
DeepWalk	77.25%	76.45 %
LINE	78.89%	82.39%
node2vec	78.37%	81.36%
PTE	85.67%	83.77%
Esim	89.22%	88.73%
TransPath(B-N-B)	70.36%	72.48%
TransPath(B-R-W-R-B)	**92.35**%	**93.04** %

From Table 1 and 2, we have the following observations. First of all, models like DeepWalk, LINE, node2vec which treat every nodes and edges in a network homogeneously all suffer from poor performance on similarity search, which further proves the necessity of the heterogeneous embedding methods. TransPath and Esim outperform the model PTE significantly, which proves the effectiveness of the usage of meta-paths, since PTE merely adopts the heterogeneous SkipGram whereas extracting samples using the traditional random walks. In addition, TransPath also outperforms Esim consistently and we attribute this to the fact that our translation mechanism could better represent the semantic and structure information than that of Esim.

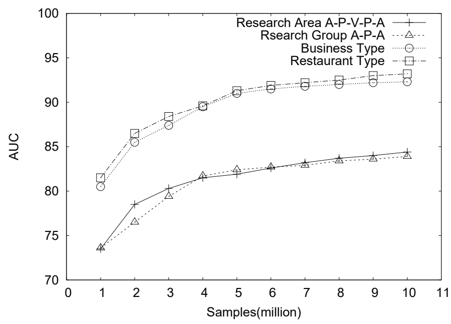
To be more specific, we could see that the choice of user-selected meta-path can make a real difference to the embedding performance. For example, in DBLP, A-P-V-P-A works much better than A-P-A on research area because the shared venue is more related to the shared research interests.



(a) Dimension



(b) Negative Ratio



(c) Total Samples

Figure 3. Parameter Sensitivity

On the contrary, working collaborations are more important in research groups, thus A-P-A is more straightforward and effective. In YELP, words in review can better describe a business or restaurant, despite that business names may contain some semantics. As a result, B-R-W-R-B performs better on both business and restaurant type. Such phenomena demonstrate that user-guided meta-path is essential when searching the similar nodes, i.e., a more suitable meta-path defined by the user leads to a better performance.

Regarding the combination of meta-paths, in DBLP, we can see that the combination of A-P-A and A-P-V-P-A is able to obtain the highest AUC scores. However, in YELP, B-R-W-R-B always works better so the combination is not necessary, which reminds us that we could compute the AUC scores for each meta-path separately and then consider the combination method. In addition, the improvement of combined meta-paths is marginal which further illustrates the importance of choosing meta-paths guided by the user more suitably.

4.2.3. Parameter Analysis

To conduct the analysis of parameter sensitivity, we select the dimension of learned latent vectors, the negative ratio of sampling, and the amount of total samples. We use A-P-V-P-A for research area groupings, A-P-A for research group groupings and B-R-W-R-B for both business type and restaurant type, since these meta-paths achieve relatively better performance.

Fig 3 illustrates the results of AUC score when varying the corresponding parameters. Specifically, from the curves in Fig 3(a), we observe that when dimension is set to 128, it obtains the best performance, this is due to that too small dimension is insufficient to capture the features, while being too large will cause overfitting problem. Fig 3(b) demonstrates that the AUC scores are relatively not sensitive to the negative sampling ratio, and setting it to 5 is reasonable to balance the efficiency and effectiveness. From Fig 3(c), we find that if more samples are trained

 doi:10.20944/preprints201801.0147.v1

during optimization, the AUC scores keep an increasing trend and finally get converged. Therefore, extracting 2 million samples is also rational considering the efficiency problem.

4.3. Node Classification

In this section, we apply models on another common task node classification. We introduce the experimental setup in the first place and analyze the results compared with other baselines in detail. We observe that TransPath is able to outperform other baseline models dramatically which further proves our model's power on representing the HIN.

4.3.1. Experimental Setup

After learning the node vectors, we further apply these representations to a linear SVM classifier as input feature vectors. We use $micro - f_1$ and $macro - f_1$ score as evaluation metrics.

Regarding the training datasets, we directly apply the research area datasets in DBLP and restaurant type datasets in YELP, in which research area could be classified as 'data mining', 'machine learning', 'database', and 'information retrieval' and restaurant type could be classified as 'Chinese', 'Indian', 'Japanese', 'Greek' and 'Mexican'.

As for the default parameters, the dimensionality of node vectors is set to 128, the ratio of negative sampling is set to 5 and the sampling times is set to 1 million. The learning rate in stochastic gradient descent is initiated as 0.025. For training, we choose meta-path A-P-V-P-A for DBLP and B-R-W-R-B for YELP. For other baseline models, we directly adopt the best configuration published in their own papers.

4.3.2. Evaluation

Table 3. Node Classification on DBLP

Model	$micro - f_1$	$macro - f_1$
DeepWalk	0.483	0.467
LINE	0.453	0.433
node2vec	0.497	0.477
PTE	0.414	0.396
Esim	0.517	0.502
HIN2Vec	0.605	0.594
TransPath	**0.645**	**0.638**

Table 4. Node Classification on YELP

Model	$micro - f_1$	$macro - f_1$
DeepWalk	0.278	0.167
LINE	0.273	0.163
node2vec	0.279	0.168
PTE	0.224	0.133
Esim	0.231	0.132
HIN2Vec	0.302	0.192
TransPath	0.356	0.254

Notice that the source code of HIN2Vec is unavailable, so we directly borrow the results published in its original paper. Table 3 and table 4 illustrate the performance of node classification in detail, from which we have the following observations. First of all, Esim, HIN2Vec, and TransPath which take advantage of meta-paths outperform DeepWalk, LINE, node2vec and PTE which only adopt the traditional random walks. Such phenomenon illustrates that meta-paths could better explore the semantics and structure of a network, compared to much longer random walkers without any schema. In addition, we find that HIN2Vec and TransPath are still better than Esim, which we attribute to that through learning the representation for meta-paths, the relationships between nodes are captured more precisely, thus facilitating the classification. Moreover, TransPath obtains the best results on both datasets, compared with all the state-of-the-art baseline models including HIN2Vec, which further proves the advantage of user-guided meta-path sampling strategy and translation mechanism.

5. Conclusion

In this paper, we propose a new model TransPath to address the representation learning issues of a heterogeneous information network (HIN). Transpath borrows the concept of translation mechanism in knowledge graph so as to capture the semantic and structure information of a HIN. Such mechanism regards a meta-path as a translation operation between head node and tail node, thus learning the embeddings of meta-paths and nodes simultaneously. In addition, TransPath also adopts a user-guided meta-path sampling strategy, so as to better explore the semantics of a path, and meanwhile improve model efficiency. We compare our proposed model TransPath with other state-of-the-art baseline models on two real-world datasets. We observe that TransPath outperforms these baselines significantly and consistently on both tasks, i.e., similarity search and node classification.

As future work, we plan to further explore at least the following two directions:

- An automatic mechanism should be developed to learn and extract the interesting meta-paths or their weighted combination from user-guided preferences and rankings; and
- We intend to further improve the translation mechanism so as to capture the semantics and structure of a network more precisely. As illustrated in knowledge graph works, translation mechanism still has many flaws, e.g., it is not able to represent complex relationships effectively.

Author Contributions: Yang Fang and Xiang Zhao conceived and designed the experiments; Yang Fang performed the experiments; Xiang Zhao analyzed the data; Yang Fang and Zhen Tan wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Watts, D.J.; Strogatz, S.H. Collective dynamics of ‘[lsquo]’small-world’[rsquo]’ networks. *Nature* **1998**, *393*, 440–442.
2. Ji, M.; Han, J.; Danilevsky, M. Ranking-based classification of heterogeneous information networks. Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, CA, USA, August 21-24, 2011, 2011, pp. 1298–1306.
3. Sun, Y.; Norick, B.; Han, J.; Yan, X.; Yu, P.S.; Yu, X. Integrating meta-path selection with user-guided object clustering in heterogeneous information networks. The 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD ’12, Beijing, China, August 12-16, 2012, 2012, pp. 1348–1356.
4. Liben-Nowell, D.; Kleinberg, J.M. The link-prediction problem for social networks. *JASIST* **2007**, *58*, 1019–1031.
5. Sun, Y.; Han, J. Mining heterogeneous information networks: a structural analysis approach. *SIGKDD Explorations* **2012**, *14*, 20–28.
6. Sun, Y.; Han, J.; Yan, X.; Yu, P.S.; Wu, T. PathSim: Meta Path-Based Top-K Similarity Search in Heterogeneous Information Networks. *PVLDB* **2011**, *4*, 992–1003.
7. Bengio, Y.; Courville, A.C.; Vincent, P. Representation Learning: A Review and New Perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1798–1828.
8. Perozzi, B.; Al-Rfou, R.; Skiena, S. DeepWalk: online learning of social representations. KDD. ACM, 2014, pp. 701–710.
9. Grover, A.; Leskovec, J. node2vec: Scalable Feature Learning for Networks. KDD. ACM, 2016, pp. 855–864.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 January 2018 doi:10.20944/preprints201801.0147.v1

10. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient Estimation of Word Representations in Vector Space. *CoRR* **2013**, *abs/1301.3781*, [1301.3781].
11. Tang, J.; Qu, M.; Wang, M.; Zhang, M.; Yan, J.; Mei, Q. LINE: Large-scale Information Network Embedding. WWW. ACM, 2015, pp. 1067–1077.
12. Ley, M. The DBLP Computer Science Bibliography: Evolution, Research Issues, Perspectives. String Processing and Information Retrieval, 9th International Symposium, SPIRE 2002, Lisbon, Portugal, September 11-13, 2002, Proceedings, 2002, pp. 1–10.
13. Bollacker, K.D.; Evans, C.; Paritosh, P.; Sturge, T.; Taylor, J. Freebase: a collaboratively created graph database for structuring human knowledge. Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, Vancouver, BC, Canada, June 10-12, 2008, 2008, pp. 1247–1250.
14. Miller, G.A. WordNet: A Lexical Database for English. *Commun. ACM* **1995**, *38*, 39–41.
15. Suchanek, F.M.; Kasneci, G.; Weikum, G. Yago: a core of semantic knowledge. Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Banff, Alberta, Canada, May 8-12, 2007, 2007, pp. 697–706.
16. Bordes, A.; Usunier, N.; García-Durán, A.; Weston, J.; Yakhnenko, O. Translating Embeddings for Modeling Multi-relational Data. 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., 2013, pp. 2787–2795.
17. Koren, Y. Factorization meets the neighborhood: a multifaceted collaborative filtering model. Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA, August 24-27, 2008, 2008, pp. 426–434.
18. Yang, C.; Liu, Z.; Zhao, D.; Sun, M.; Chang, E.Y. Network Representation Learning with Rich Text Information. IJCAI. AAAI Press, 2015, pp. 2111–2117.
19. Tu, C.; Zhang, W.; Liu, Z.; Sun, M. Max-Margin DeepWalk: Discriminative Learning of Network Representation. IJCAI. IJCAI/AAAI Press, 2016, pp. 3889–3895.
20. Cao, S.; Lu, W.; Xu, Q. GraRep: Learning Graph Representations with Global Structural Information. Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015, Melbourne, VIC, Australia, October 19 - 23, 2015, 2015, pp. 891–900.
21. Tu, C.; Liu, H.; Liu, Z.; Sun, M. CANE: Context-Aware Network Embedding for Relation Modeling. ACL (1). Association for Computational Linguistics, 2017, pp. 1722–1731.
22. Wang, D.; Cui, P.; Zhu, W. Structural Deep Network Embedding. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016, 2016, pp. 1225–1234.
23. Tang, J.; Qu, M.; Mei, Q. PTE: Predictive Text Embedding through Large-scale Heterogeneous Text Networks. Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, August 10-13, 2015, 2015, pp. 1165–1174.
24. Huang, Z.; Mamoulis, N. Heterogeneous Information Network Embedding for Meta Path based Proximity. *CoRR* **2017**, *abs/1701.05291*, [1701.05291].
25. Dong, Y.; Chawla, N.V.; Swami, A. metapath2vec: Scalable Representation Learning for Heterogeneous Networks. Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017, 2017, pp. 135–144.
26. Shang, J.; Qu, M.; Liu, J.; Kaplan, L.M.; Han, J.; Peng, J. Meta-Path Guided Embedding for Similarity Search in Large-Scale Heterogeneous Information Networks. *CoRR* **2016**, *abs/1610.09769*, [1610.09769].
27. Fu, T.; Lee, W.; Lei, Z. HIN2Vec: Explore Meta-paths in Heterogeneous Information Networks for Representation Learning. Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, Singapore, November 06 - 10, 2017, 2017, pp. 1797–1806.
28. Sun, Y.; Han, J. *Mining Heterogeneous Information Networks: Principles and Methodologies*; Synthesis Lectures on Data Mining and Knowledge Discovery, Morgan & Claypool Publishers, 2012.
29. Sun, Y.; Yu, Y.; Han, J. Ranking-based clustering of heterogeneous information networks with star network schema. Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, June 28 - July 1, 2009, 2009, pp. 797–806.
30. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed Representations of Words and Phrases and their Compositionality. Advances in Neural Information Processing Systems 26: 27th Annual

 doi:10.20944/preprints201801.0147.v1

Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States., 2013, pp. 3111–3119.

31. Muja, M.; Lowe, D.G. Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration. VISAPP 2009 - Proceedings of the Fourth International Conference on Computer Vision Theory and Applications, Lisboa, Portugal, February 5-8, 2009 - Volume 1, 2009, pp. 331–340.