SHNE: Representation Learning for Semantic-Associated Heterogeneous Networks

Chuxu Zhang
University of Notre Dame
Notre Dame, IN 46556, USA
czhang11@nd.edu

Ananthram Swami
US Army Research Laboratory
Adelphi, MD 20783, USA
ananthram.swami.civ@mail.mil

Nitesh V. Chawla
University of Notre Dame
Notre Dame, IN 46556, USA
nchawla@nd.edu

ABSTRACT

Representation learning in heterogeneous networks faces challenges due to heterogeneous structural information of multiple types of nodes and relations, and also due to the unstructured attribute or content (*e.g.*, text) associated with some types of nodes. While many recent works have studied homogeneous, heterogeneous, and attributed networks embedding, there are few works that have collectively solved these challenges in heterogeneous networks. In this paper, we address them by developing a Semantic-aware Heterogeneous Network Embedding model (SHNE). SHNE performs joint optimization of heterogeneous SkipGram and deep semantic encoding for capturing both heterogeneous structural closeness and unstructured semantic relations among all nodes, as function of node content, that exist in the network. Extensive experiments demonstrate that SHNE outperforms state-of-the-art baselines in various heterogeneous network mining tasks, such as link prediction, document retrieval, node recommendation, relevance search, and class visualization.

KEYWORDS

Semantic-Associated Heterogeneous Networks, Representation Learning, Network Embedding, Deep Learning

ACM Reference Format:
Chuxu Zhang, Ananthram Swami, and Nitesh V. Chawla. 2019. SHNE: Representation Learning for Semantic-Associated Heterogeneous Networks. In *The Twelfth ACM International Conference on Web Search and Data Mining (WSDM ’19), February 11–15, 2019, Melbourne, VIC, Australia.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3289600.3291001

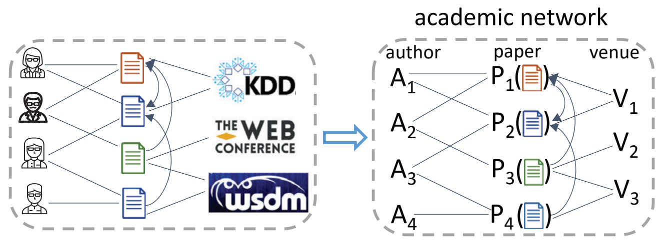


Figure 1: An Illustrative example of challenges in heterogeneous network representation learning.

1 INTRODUCTION

Heterogeneous information networks (HetNets) [27, 28], *e.g.*, academic networks, encode rich information through multi-typed nodes, relationships, and attribute/content associated with nodes. For example, academic networks can represent human-human relationship (authors), human-object relationship (author-paper or author-venue or author-organization), and object-object relationship (paper-paper, paper-venue, paper-organization). The nodes in this case (human and object) can carry attribute or semantic content (such as paper abstract). Given the multi-typed nodes, relationships, and content at the nodes, feature engineering is faced with a unique challenge for network mining tasks such as relation mining [26, 35], relevance search [10, 27], personalized recommendation [14, 22, 42]. The typical feature engineering activity responds to the requirements of the network mining task for an application, requiring both a domain understanding and large exploratory search space for possible features. Not only is this expensive, but it also may not result in optimal performance.

To that end, we ask the question: *Can we generalize the feature engineering activity through representation learning that addresses the complexity of multi-typed data in HetNets?* With the advent of deep learning, significant effort has been devoted to network representation learning in the last few years, starting with a focus on homogeneous networks [7, 19, 31] and more recently on HetNets [1, 6, 16, 23, 24, 37] and attributed networks [9, 12, 13, 33, 39]. The underlying theme of the models developed in these works is to automate the discovery of useful node latent features that can be further utilized in various network mining problems such as link prediction and node recommendation. However, these methods do not fully address the challenges of HetNets:

- (**C1**) HetNets include multiple types of nodes and relations. For example, in Figure 1, academic network involves three types of nodes, *i.e.*, *author*, *paper* and *venue*, which are connected by three types of relations, *i.e.*, *author-write-paper*, *paper-cite-paper* and *paper-publish-venue*. Most of the previous models (*e.g.*, Deepwalk [19] and node2vec [7]) employ homogeneous language models which make application to HetNets difficult. Thus challenge 1 is: how to extend homogeneous language model to heterogeneous network representation learning for maintaining structural closeness among multiple types of nodes and relations?
- (**C2**) HetNets include both structural content (*e.g.*, node type and relation connection) and unstructured semantic content (*e.g.*, text). For example, in Figure 1, paper in academic network connects to author & venue and contains semantic text. The

WSDM ’19, February 11–15, 2019, Melbourne, VIC, Australia

ACM ISBN 978-1-4503-5940-5/19/02.
https://doi.org/10.1145/3289600.3291001

current models (*e.g.*, metapath2vec [6]) purely depend on structural content but cannot leverage unstructured content to infer semantic relations that are far away in network. To be more specific (as we will show in Section 4.5), given query author "Jure Leskovec", conventional techniques (*e.g.*, node2vec and metapath2vec) tend to return authors who collaborated with "Jure" due to structural relations bridged by paper or return authors who are different from "Jure" in specific research interests due to structural relations bridged by venue. Thus challenge 2 is: how to capture both structural closeness and unstructured semantic relations among all nodes in HetNets?

- (**C3**) Most HetNets are partially semantic-associated. That is, only some types of nodes in network are associated with semantic content. For example, in Figure 1, only paper is associated with semantic content (*i.e.*, abstract). Thus, the recent attribute-aware network embedding models [12, 13, 25, 33, 39] cannot be directly applied to HetNets since attribute/semantic content of each node is required. Thus challenge 3 is: how to effectively incorporate unstructured content of some types of nodes into a representation learning framework for partially semantic-associated HetNets?

Our proposed method SHNE, a **S**emantic-aware **H**eterogeneous **N**etwork **E**mbedding model, addresses these challenges. Specifically, first, we develop a heterogeneous SkipGram model to maintain structural closeness among multiple types of nodes and relations. Next, we design two effective ways based on deep semantic encoding to incorporate unstructured content (*i.e.*, text) of some types of nodes into heterogeneous SkipGram for capturing semantic relations. Finally, a negative sampling technique and a walk sampling based strategy are utilized to optimize and train the proposed models. To summarize, the main contributions of our work are:

- We formalize the problem of semantic-aware representation learning in HetNets and develop a model, *i.e.*, SHNE, to solve the problem. SHNE performs joint optimization of heterogeneous SkipGram and deep semantic encoding.
- We design the corresponding optimization strategy and training algorithm to effectively learn node embeddings. The output embeddings are further utilized in various HetNet mining tasks, such as link prediction, document retrieval, node recommendation, relevance search, and class visualization, which demonstrate the superior performance of SHNE over state-of-the-art baselines.

2 PROBLEM DEFINITION

In this section, we first introduce the concepts of HetNets and random & meta-path walks, then formally define the problem of semantic-aware representation learning in HetNets.

Definition 2.1. (**Heterogeneous Networks**) A heterogeneous network [28] is defined as a network $G = (V, E, O_V, R_E)$ with multiple types of nodes V and links E. O_V and R_E represent the sets of object types and relation types. Each node $v \in V$ and each link $e \in E$ is associated with a node type mapping function $\psi_v : V \rightarrow O_V$ and a link type mapping function $\psi_e : E \rightarrow R_E$.

For example, in Figure 2(b), the academic network can be seen as a HetNet. The set of node types O_V includes *author* (A), *paper* (P) and *venue* (V). The set of link types R_E includes *author-write-paper*, *paper-cite-paper* and *paper-publish-venue*.

Definition 2.2. (**Random Walk**) A random walk [7] is defined as a node sequence $S_{v_0} = \{v_0, v_1, v_2, ..., v_{L-1}\}$ wherein the i-th node v_{i-1} in the walk is randomly selected from the neighbors of its predecessor v_{i-2}.

Definition 2.3. (**Meta-path Walk**) A meta-path walk [6] in HetNet is defined as a random walk guided by a specific meta-path scheme with the form of $\mathcal{P} \equiv o_1 \xrightarrow{r_1} o_2 \xrightarrow{r_2} \cdots \xrightarrow{r_{m-1}} o_m$, where $o_i \in O_V$, $r_i \in R_E$ and $r = r_1 * r_2 \cdots * r_{m-1}$ represents a compositional relation between relation types r_1 and r_m. Each meta-path walk recursively samples a specific $\mathcal{P}$ until it meets the given length.

Figure 2(b) shows examples of random walk and "APVPA" meta-path walk in the academic network.

Definition 2.4. (**Semantic-aware Representation Learning for Heterogeneous Networks**) Given a HetNet with structural information among all nodes and unstructured semantic content of some types of nodes, the task is to design a model to learn a d-dimensional embeddings $\theta \in \mathbb{R}^{|V| \times d} (d \ll |V|)$, which can encode both structural closeness and unstructured semantic relations.

For example, in the network of Figure 1, author and venue nodes contain structural content, *i.e.*, node id, node type as well as link relations with others, and paper node contains both structural content and unstructured semantic content, *e.g.*, abstract text. The output θ denotes embeddings of all nodes via the same latent space, which can be further utilized in various HetNet mining tasks.

3 SHNE FRAMEWORK

In this section, we present the framework of semantic-aware heterogeneous network embedding model which will address the three challenges described in Section 1.

3.1 Heterogeneous Network Embedding (C1)

Deepwalk [19] and node2vec [7] leverage SkipGram and random walks to learn node embeddings. However, those approaches focus on homogeneous networks. Inspired by metapath2vec [6], we formulate the heterogeneous network representation learning as heterogeneous SkipGram (HSG) to address challenge **C1**. Specifically, given a HetNet $G = (V, E, O_V, R_E)$, the objective is to maximize the likelihood of each type of context node given the input node v:

$$o_1 = \arg\max_{\theta} \prod_{v \in V} \prod_{t \in O_V} \prod_{v_c \in N_t(v)} p(v_c | v; \theta) \tag{1}$$

where θ contains the embeddings of all nodes and $N_t(v)$ is the set of t-type context node of v which can be collected in different ways such as one-hop neighbors in random walks. For example, in Figure 2(b), A_3 is structurally close to other authors (*e.g.*, A_1 & A_4), papers (*e.g.*, P_2 & P_4) and venues (*e.g.*, V_1 & V_3). Thus objective o_1 is able to maintain structural closeness among multiple types of nodes and relations in G.

3.2 Incorporating Semantic Encoder (C2, C3)

Objective o_1 formulates structural closeness but ignores unstructured semantic relations. To address challenges **C2** and **C3**, we

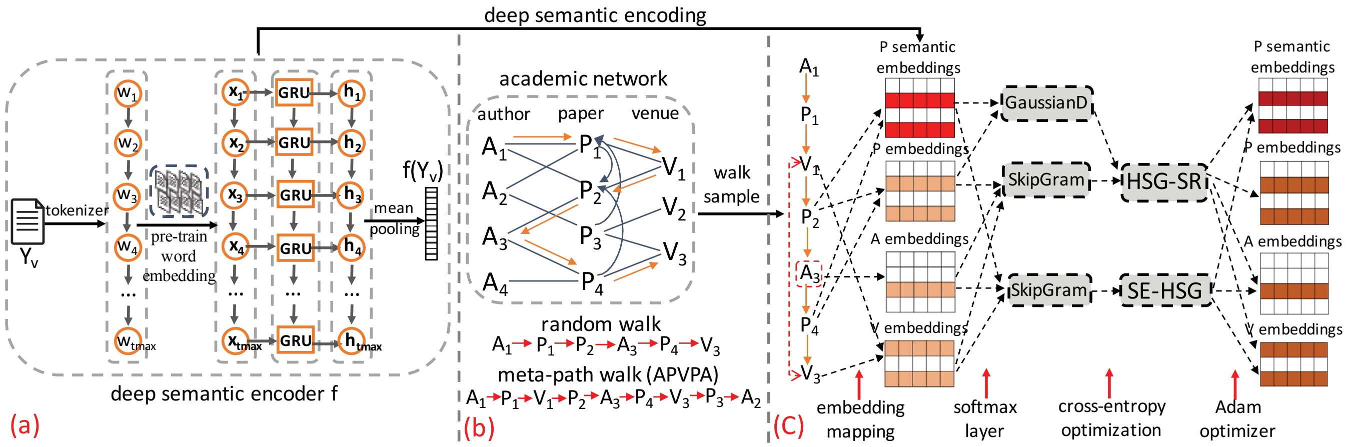


Figure 2: An illustrative example of SHNE in the academic network: (a) paper semantic encoder based on gated recurrent neural network; (b) academic network and random & meta-path walks; (c) framework of the proposed models.

design two ways to incorporate unstructured content of some types of nodes into HSG.

3.2.1 HSG with Unstructured Semantic Regularization (HSG-SR). One way is to tightly join HSG with the conditional probability of semantic constraint, leading to unstructured semantic regularization onto objective o_1. The objective is defined as:

$$o_2 = \arg\max_{\theta,\Phi} \prod_{v \in V} \prod_{t \in O_V} \prod_{v_c \in N_t(v)} p(v_c|v;\theta) \prod_{v \in V_S} p(\theta_v|Y_v;\Phi) \quad (2)$$

where V_S is the set of nodes with unstructured semantic content, Y_v represents unstructured content of node v and Φ are parameters of a deep semantic encoder that will be described later. The conditional probability $p(v_c|v;\theta)$ is defined as the heterogeneous softmax function: $p(v_c|v;\theta) = \frac{e^{\theta_{v_c}\cdot\theta_v}}{\sum_{v_k \in V_t} e^{\theta_{v_k}\cdot\theta_v}}$, where V_t is the set of t-type nodes. Besides, we model the conditional probability $p(\theta_v|Y_v;\Phi)$ as Gaussian distribution: $p(\theta_v|Y_v;\Phi) = \mathcal{N}(\theta_v|E_v, \sigma^2 I)$, where E_v denotes v's semantic representation encoded by deep learning architecture f: $E_v = f(Y_v)$. For example, in the network of Figure 2, V_S in o_2 is the set of papers and the formulation involves four kinds of embeddings, *i.e.*, author embedding, venue embedding, paper embedding and paper deep semantic embedding. Note that there are two kinds of paper embeddings and we will use paper deep semantic embedding for evaluation in Section 4.

3.2.2 Unstructured Semantic Enhanced HSG (SE-HSG). The other way is to replace the embeddings of semantic-associated nodes by deep semantic encoding in HSG, leading to unstructured semantic enhancement onto objective o_1. The objective is defined as:

$$o_3 = \arg\max_{\theta,\Phi} \prod_{v \in V} \prod_{t \in O_V} \prod_{v_c \in N_t(v)} p(v_c|v;\theta;Y;\Phi) \quad (3)$$

where Y is the set of all unstructured semantic content of V_S. The conditional probability $p(v_c|v;\theta;Y;\Phi)$ is defined as the semantic enhanced heterogeneous softmax function: $p(v_c|v;\theta;Y;\Phi) = \frac{e^{\Theta_{v_c}\cdot\Theta_v}}{\sum_{v_k \in V_t} e^{\Theta_{v_k}\cdot\Theta_v}}$, where Θ denotes the enhanced embeddings. That is, $\Theta_v = E_v = f(Y_v)$ for $v \in V_S$ otherwise $\Theta_v = \theta_v$. For example, in the network of Figure 2, Y in o_3 is text content of all papers and the formulation involves three kinds of embeddings, *i.e.*, author embedding, venue embedding and paper deep semantic embedding. Notice that, θ in o_3 only denotes embeddings of nodes without unstructured content (*e.g.*, author and venue in academic network), which is a bit different from o_2.

3.2.3 Unstructured Semantic Content Encoder. Both objectives o_2 and o_3 involve deep semantic encoding architecture. To encode unstructured content of some types of nodes into fixed length representations $E \in \mathbb{R}^{|V_S|\times d}$, we introduce gated recurrent units (GRU), a specific type of recurrent neural network, which has been widely adopted for many applications such as machine translation [4]. Figure 2(a) gives an illustrative example of this encoder for papers in the academic network. To be more specific, each paper's abstract is represented as a sequence of words: $\{w_1, w_2, \cdots, w_{t_{max}}\}$, followed by the word embeddings sequence: $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_{t_{max}}\}$ trained by word2vec [17], where t_{max} is the maximum length of text. For each step t with the input word embedding $\mathbf{x}_t$ and previous hidden state vector $\mathbf{h}_{t-1}$, the current hidden state vector $\mathbf{h}_t$ is updated by $\mathbf{h}_t = \mathbf{GRU}(\mathbf{x}_t, \mathbf{h}_{t-1})$, where the GRU module is defined as:

$$\begin{aligned} \mathbf{z}_t &= \sigma(\mathbf{A}_z\mathbf{x}_t + \mathbf{B}_z\mathbf{h}_{t-1}) \\ \mathbf{r}_t &= \sigma(\mathbf{A}_r\mathbf{x}_t + \mathbf{B}_r\mathbf{h}_{t-1}) \\ \hat{\mathbf{h}}_t &= \tanh[\mathbf{A}_h\mathbf{x}_t + \mathbf{B}_h(\mathbf{r}_t \circ \mathbf{h}_{t-1})] \\ \mathbf{h}_t &= \mathbf{z}_t \circ \mathbf{h}_{t-1} + (1 - \mathbf{z}_t) \circ \hat{\mathbf{h}}_t \end{aligned} \quad (4)$$

where σ is the sigmoid function, $\mathbf{A}$ and $\mathbf{B}$ are parameter matrices of the GRU (*i.e.*, Φ in objectives o_2 and o_3 includes $\mathbf{A}$ and $\mathbf{B}$), operator $\circ$ denotes element-wise multiplication, $\mathbf{z}_t$ and $\mathbf{r}_t$ are update gate vector and reset gate vector, respectively. The GRU encodes word embeddings into deep semantic embeddings $\mathbf{h} \in \mathbb{R}^{t_{max}\times d}$, which is concatenated with a mean pooling layer to obtain the general embedding of paper. All of these steps construct the deep semantic encoder f. We have also explored other encoding architectures such as LSTM and attention-based GRU, and obtain similar results. We choose GRU since it has a concise structure and reduces training time.

3.3 Model Optimization and Training

We leverage the negative sampling technique [17] to optimize our model and introduce a walk sampling based strategy for model training.

3.3.1 Optimization of HSG-SR. By applying negative sampling to the construction of softmax function, we can approximate the logarithm of $p(v_c|v;\theta)$ in objective o_2 as:

$$\log\sigma(\theta_{v_c}\cdot\theta_v)+\sum_{m=1}^{M}\mathbb{E}_{v_{c'}\sim P_t(v_{c'})}\log\sigma(-\theta_{v_{c'}}\cdot\theta_v) \tag{5}$$

where M is the negative sample size and $P_t(v_{c'})$ is the pre-defined sampling distribution $w.r.t.$ the t-type node. In our case, M makes little impact on the performance of the proposed models. Thus we set $M=1$ and obtain the cross entropy loss for optimization:

$$\log p(v_c|v;\theta)=\log\sigma(\theta_{v_c}\cdot\theta_v)+\log\sigma(-\theta_{v_{c'}}\cdot\theta_v) \tag{6}$$

That is, for each context node v_c of v, we sample a negative node $v_{c'}$ according to $P_t(v_{c'})$. Further, as $p(\theta_v|Y_v;\Phi)=\mathcal{N}(\theta_v|E_v,\sigma^2 I)$, the logarithm of $p(\theta_v|Y_v;\Phi)$ in objective o_2 is equivalent to:

$$\log p(\theta_v|Y_v;\Phi)=-\left[\theta_v-E_v\right]^T\left[\theta_v-E_v\right] \tag{7}$$

where $E_v=f(Y_v)$ and f is the deep semantic encoder. Therefore we rewrite objective o_2 as:

$$\begin{aligned} o_2=\sum_{\langle v,v_c,v_{c'}\rangle\in T_{walk}}&\Big\{\log\sigma(\theta_{v_c}\cdot\theta_v)+\log\sigma(-\theta_{v_{c'}}\cdot\theta_v)\\ &-\gamma\sum_{v_*\in T^S_{tri}}\left[\theta_{v_*}-f(Y_{v_*})\right]^T\left[\theta_{v_*}-f(Y_{v_*})\right]\Big\}\end{aligned} \tag{8}$$

where γ is a trade-off factor and T_{walk} denotes the set of triplets $\langle v,v_c,v_{c'}\rangle$ collected by walk sampling on HetNet, which will be described later. Here, T^S_{tri} is the set of nodes with unstructured content in each triplet of T_{walk} for semantic regularization. That is, $max\{|T^S_{tri}|\}=3$.

3.3.2 Optimization of SE-HSG. Similar to HSG-SR, the logarithm of $p(v_c|v;\theta;Y;\Phi)$ in objective o_3 is approximated by:

$$\log\ p(v_c|v;\theta;Y;\Phi)=\log\sigma(\Theta_{v_c}\cdot\Theta_v)+\log\sigma(-\Theta_{v_{c'}}\cdot\Theta_v) \tag{9}$$

where $\Theta_v=E_v=f(Y_v)$ for nodes with unstructured content otherwise $\Theta_v=\theta_v$. Therefore we rewrite objective o_3 as:

$$o_3=\sum_{\langle v,v_c,v_{c'}\rangle\in T_{walk}}\log\sigma(\Theta_{v_c}\cdot\Theta_v)+\log\sigma(-\Theta_{v_{c'}}\cdot\Theta_v) \tag{10}$$

As in HSG-SR, T_{walk} is the set of triplets $\langle v,v_c,v_{c'}\rangle$ collected from walk sequences on HetNet.

3.3.3 Model Training. Both optimized objectives o_2 and o_3 are accumulated on set T_{walk}. Similar to Deepwalk, node2vec and metapath2vec, we design a walk sampling strategy to generate T_{walk}. Specifically, first, we uniformly generate a set of random walks or meta-path walks S in HetNet. Then, for each node v in $S_i\in S$, we collect context node v_c which satisfies: $dist(v,v_c)\le\tau$, $i.e.$, neighbor v_c of v is within within distance τ in S_i. For example, in Figure 2(c), the context nodes of A_3 ($\tau=2$) in the sample walk are V_1, P_2, P_4 and V_3. Finally, for each v_c, we sample a negative node $v_{c'}$ with the same node type of v_c according to $P_t(v_{c'})\propto dg^{3/4}_{v_{c'}}$, where $dg_{v_{c'}}$ is the frequency of $v_{c'}$ in S. Furthermore, we design a mini-batch based Adam Optimizer [11] to train the model. Specifically, at each iteration, we sample a mini-batch of triplets in T_{walk} and accumulate the objective according to equation (8) or (10), then update the parameters via Adam. We repeat the training iterations until the change between two consecutive iterations is sufficiently small. Figure 2(c) shows an illustration of the framework of HSG-SR and SE-HSG on the academic network. The output representations θ and E can be utilized in various HetNet mining tasks, as we will show in Section 4.

4 EXPERIMENTS

In this section, we conduct extensive experiments with the aim of answering the following research questions:

- (**RQ1**) How does SHNE perform vs. state-of-the-art network embedding models for different HetNet mining tasks, such as link prediction (**RQ1-1**), document retrieval (**RQ1-2**), and node recommendation (**RQ1-3**)? In addition, how do hyper-parameters impact SHNE's performance in each task?
- (**RQ2**) What is the performance difference between SHNE and baselines in relevance search $w.r.t.$ each task in **RQ1**?
- (**RQ3**) What is the quality of embedding generated by SHNE $w.r.t.$ class visualization?

In this work, we focus on the experiments on academic HetNet but our model can be applied to or modified for different partially semantic-associated HetNets.

4.1 Experimental Setup

4.1.1 Data. We use the AMiner computer science dataset [32], which is publicly available[1]. To avoid noise, we remove the papers published in venues ($e.g.$, workshop) with limited publications and the instances without abstract text. In addition, topic of each area changes over time. For example, according to our analysis of the data, the most popular topics in data mining change from web mining and clustering (1996~2005) to network mining and learning (2006~2015). To make a thorough evaluation of SHNE and verify its effectiveness for networks in different decades, we independently conduct experiments on two datasets, $i.e.$, AMiner-*I* (1996~2005) and AMiner-*II* (2006~2015). As a result, AMiner-*I* contains 160,713 authors, 111,409 papers and 150 venues, AMiner-*II* contains 571,693 authors, 483,449 papers and 492 venues. The structure of the academic network used in this work is shown in Figure 1.

4.1.2 Comparison Baselines. We compare SHNE with four state-of-the-art models, $i.e.$, Deepwalk [19], LINE [31], node2vec [7] and metapath2vec [6] $w.r.t.$ both homogeneous and heterogeneous networks. Note that some recent attribute/semantic-aware network embedding models [12, 13, 25, 33, 39] cannot be directly applied to partially semantic-associated HetNets because of the requirement of semantic content of each node. We use either random walk (rw) or meta-path walk (mw) to collect context node in HSG-SR and SE-HSG, resulting in four variants of SHNE: SHNE$^{rw}_{HSG-SR}$, SHNE$^{mw}_{HSG-SR}$, SHNE$^{rw}_{SE-HSG}$ and SHNE$^{mw}_{SE-HSG}$.

[1] https://aminer.org/citation

4.1.3 Reproducibility. For fairness of comparison, we use the same embedding dimension d = 128 for all models. The window size τ = 7, the number of walks per node N = 10 and the walk length L = 30 are used for Deepwalk, node2vec, metapath2vec and SHNE. The size of negative samples M is set to 5 for node2vec, LINE and metapath2vec. In addition, $\gamma = 1.0$ for SHNE$_{HSG-SR}$ and three meta-path schemes "APA", "APPA" and "APVPA" are jointly used to generate meta-path walks for SHNE^{mw}. We employ TensorFlow to implement all variants of SHNE and further conduct them on NVIDIA TITAN X GPU.

4.2 Link Prediction (RQ1-1)

Who will be your academic collaborators? As a response to **RQ1-1**, we design an experiment to evaluate SHNE's performance on the author collaboration link prediction task.

4.2.1 Experimental Setting. Unlike past work [7] that randomly samples a portion of links for training and uses the remaining for evaluation, we consider a more realistic setting that splits training and test data via a given time stamp T. Specifically, first, the network before T is utilized to learn node embeddings. Then, the collaboration links before T are used to train a binary logistic classifier. Finally, the collaboration relations after T with equal number of random non-collaboration links are used to evaluate the trained classifier. In addition, only new collaborations among current authors (who appear before T) are considered and duplicated collaborations are removed from evaluation. The link embedding is formed by element-wise multiplication of embeddings of the two edge nodes. We use **Accuracy** and **F1 score** as evaluation metrics for the binary classification task. Besides, T is set as 2003/2004 and 2013/2014 for AMiner-I and AMiner-II, respectively.

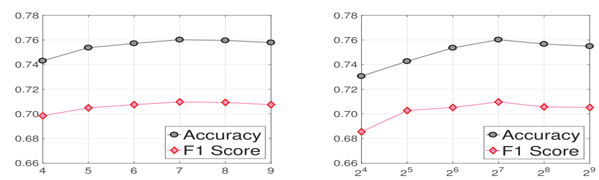
4.2.2 Results. The performances of different models are reported in Table 1. According to the table: (a) All variants of SHNE perform better than baselines, demonstrating the effectiveness of incorporating unstructured semantic content to learn author embeddings; (b) SHNE$^{mw}_{SE-HSG}$ achieves the best performances in all cases. The average improvement of SHNE$^{mw}_{SE-HSG}$ over different baselines ranges from 10.9% to 41.0% and 6.7% to 30.9% on AMiner-I and AMiner-II, respectively. (c) SHNE^{mw} outperforms SHNE^{rw}, showing that meta-path walk is better than random walk for collecting context node in SHNE. In addition, SHNE$_{SE-HSG}$ has better performance than SHNE$_{HSG-SR}$, indicating that enhancing heterogeneous SkipGram with text encoder is more effective than taking text encoding as semantic regularization. It may be because author embeddings are directly improved by paper semantic embedding in SHNE$_{SE-HSG}$ while indirectly influenced by paper embeddings which are regularized by paper semantic embeddings in SHNE$_{HSG-SR}$.

4.2.3 Parameter Sensitivity. We conduct experiment to analyze the impact of two key parameters, *i.e.*, the window size τ of walk sampling and the embedding dimension d. We investigate a specific parameter by changing its value and fixing the others. The prediction results of SHNE$^{mw}_{SE-HSG}$ as a function of τ and d on AMiner-II (T = 2013) are shown in Figure 3. We see that: (a) With increasing τ, accuracy and F1 score increase at first since a larger window

Table 1: Collaboration prediction results.

AMiner-I	T = 2004 Accuracy	T = 2004 F1	T = 2003 Accuracy	T = 2003 F1	Gain
Deepwalk	0.6341	0.4323	0.6244	0.4058	41.0%
LINE	0.6858	0.5605	0.6946	0.5729	14.5%
node2vec	0.6758	0.5291	0.6821	0.5409	18.9%
metapath2vec	0.7013*	0.5914*	0.7041*	0.5935*	10.9 %
SHNE$^{rw}_{HSG-SR}$	0.7302	0.6561	0.7378	0.6623	–
SHNE$^{mw}_{HSG-SR}$	0.7367	0.6618	0.7401	0.6635	–
SHNE$^{rw}_{SE-HSG}$	0.7388	0.6579	0.7419	0.6648	–
SHNE$^{mw}_{SE-HSG}$	**0.7482**	**0.6753**	**0.7525**	**0.6881**	–

AMiner-II	T = 2014 Accuracy	T = 2014 F1	T = 2013 Accuracy	T = 2013 F1	Gain
Deepwalk	0.6559	0.5024	0.6487	0.4833	30.9%
LINE	0.7034	0.6048	0.6956	0.5898	14.3%
node2vec	0.7136	0.6122	0.7066	0.5965	12.7%
metapath2vec	0.7299*	0.6628*	0.7254*	0.6512*	6.7%
SHNE$^{rw}_{HSG-SR}$	0.7498	0.7061	0.7495	0.6946	–
SHNE$^{mw}_{HSG-SR}$	0.7511	0.7084	0.7503	0.6962	–
SHNE$^{rw}_{SE-HSG}$	0.7562	0.7125	0.7546	0.6978	–
SHNE$^{mw}_{SE-HSG}$	**0.7627**	**0.7208**	**0.7602**	**0.7097**	–



(a) Window size τ (b) Embedding dimension d

Figure 3: Parameter sensitivity in collaboration prediction.

means more useful context information. But when τ exceeds a certain value, performances decrease slowly with τ possibly due to uncorrelated noise. The best value of τ is around 7; (b) Similar to τ, an appropriate value should be set for d such that the best node embeddings are learned. The optimal value of d is around 128.

4.3 Document Retrieval (RQ1-2)

Which relevant papers should be retrieved for your query? As a response to **RQ1-2**, we design an experiment to evaluate SHNE's performance on the paper retrieval task.

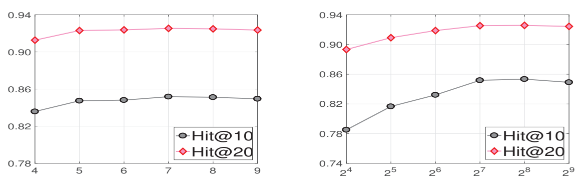
4.3.1 Experimental Setting. As in the previous task, the network before T is utilized to learn node embeddings. The ground truth of relevance is assumed as the co-cited relation (cited by the same paper) between two papers after T. The relevant score of two papers is defined as the cosine similarity between embeddings of two papers. We use **HitRatio@k** as the evaluation metric for retrieval task. Due to large number of candidate papers, we follow the sampling strategy in [3] to reduce evaluation time. Specifically, for each evaluated paper, we randomly generate 100 negative samples for comparison with the true relevant paper. The hit ratio equals 1 if the true relevant paper is ranked in the top-k list of relevant

Table 2: Paper retrieval results.

AMiner-*I*	T = 2004		T = 2003		Gain
	Hit@10	Hit@20	Hit@10	Hit@20	
Deepwalk	0.8120	0.8816	0.8217	0.8967	6.7%
LINE	0.7568	0.8139	0.7641	0.8139	15.7%
node2vec	0.8552*	0.9148*	0.8653*	0.9250*	2.3%
metapath2vec	0.8239	0.8910	0.8366	0.9081	5.3%
SHNE$^{rw}_{HSG-SR}$	0.8685	0.9351	0.8696	0.9375	–
SHNE$^{mw}_{HSG-SR}$	0.8673	0.9342	0.8722	0.9389	–
SHNE$^{rw}_{SE-HSG}$	0.8741	0.9412	**0.8788**	0.9455	–
SHNE$^{mw}_{SE-HSG}$	**0.8751**	**0.9425**	0.8783	**0.9470**	–
AMiner-*II*	T = 2014		T = 2013		Gain
	Hit@10	Hit@20	Hit@10	Hit@20	
Deepwalk	0.7460	0.8366	0.7392	0.8316	13.2%
LINE	0.7004	0.7785	0.6659	0.7453	23.6%
node2vec	0.8041*	0.8785*	0.7981*	0.8749*	6.3%
metapath2vec	0.7214	0.8136	0.7257	0.8201	15.9%
SHNE$^{rw}_{HSG-SR}$	0.8263	0.9162	0.8141	0.9101	–
SHNE$^{mw}_{HSG-SR}$	0.8287	0.9184	0.8153	0.9096	–
SHNE$^{rw}_{SE-HSG}$	0.8619	**0.9281**	0.8516	0.9243	–
SHNE$^{mw}_{SE-HSG}$	**0.8625**	0.9278	**0.8518**	**0.9254**	–

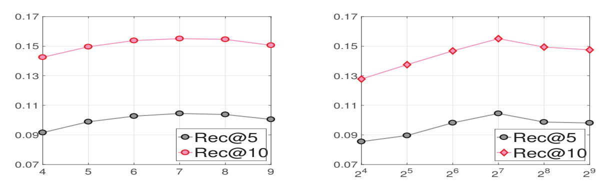
Table 3: Venue recommendation results.

AMiner-*I*	T = 2004		T = 2003		Gain
	Rec@5	Rec@10	Rec@5	Rec@10	
Deepwalk	0.1051*	0.1628*	0.0864*	0.1403*	18.9%
LINE	0.0716	0.1179	0.0727	0.1183	55.1%
node2vec	0.0945	0.1570	0.0774	0.1386	27.2%
metapath2vec	0.0878	0.1527	0.0714	0.1395	33.3%
SHNE$^{rw}_{HSG-SR}$	0.1156	0.1772	0.0988	0.1593	–
SHNE$^{mw}_{HSG-SR}$	0.1208	0.1813	0.1050	0.1631	–
SHNE$^{rw}_{SE-HSG}$	0.1225	0.1831	0.1054	0.1648	–
SHNE$^{mw}_{SE-HSG}$	**0.1250**	**0.1852**	**0.1073**	**0.1667**	–
AMiner-*II*	T = 2014		T = 2013		Gain
	Rec@5	Rec@10	Rec@5	Rec@10	
Deepwalk	0.1039*	0.1549*	0.0925*	0.1384*	11.7%
LINE	0.0771	0.1151	0.0641	0.0983	54.8%
node2vec	0.0766	0.1182	0.0691	0.1074	47.8%
metapath2vec	0.0654	0.1077	0.0608	0.1001	66.1%
SHNE$^{rw}_{HSG-SR}$	0.1092	0.1643	0.0977	0.1466	–
SHNE$^{mw}_{HSG-SR}$	0.1116	0.1713	0.1032	0.1526	–
SHNE$^{rw}_{SE-HSG}$	0.1107	0.1725	0.1023	0.1534	–
SHNE$^{mw}_{SE-HSG}$	**0.1136**	**0.1742**	**0.1045**	**0.1551**	–



(a) Window size τ (b) Embedding dimension d

Figure 4: Parameter sensitivity in paper retrieval.



(a) Window size τ (b) Embedding dimension d

Figure 5: Parameter sensitivity in venue recommendation.

score, otherwise 0. The overall result is the average value of HitRatio@k among all evaluated papers. Duplicated co-cited relations are removed from evaluation and k is set to 10 or 20. In addition, T is set as 2003/2004 and 2013/2014 for AMiner-*I* and AMiner-*II*, respectively.

4.3.2 Results. The results are reported in Table 2. From the table: (a) All variants of SHNE achieve better performance than baselines, demonstrating the benefit of incorporating unstructured semantic content to learn paper embeddings; (b) The average improvement of SHNE$^{mw}_{SE-HSG}$ over different baselines ranges from 2.3% to 15.7% and 6.3% to 23.6% on AMiner-*I* and AMiner-*II*, respectively; (c) SHNE$_{SE-HSG}$ outperforms SHNE$_{HSG-SR}$, showing that semantic enhanced SkipGram is better than SkipGram with semantic regularization. However, SHNE^{rw} has close performance to SHNE^{mw}, indicating that meta-path walk has little impact on this task. This is reasonable since the paper embeddings depend on the deep semantic encoder in our model.

4.3.3 Parameter Sensitivity. Following the same setup as in link prediction, we investigate the impact of window size τ and embedding dimension d on SHNE$^{mw}_{SE-HSG}$'s performance on AMiner-*II* (T = 2013), as shown by Figure 4. It can be seen that: (a) The results are little sensitive to τ when $\tau \geq 7$. As we noted above, paper embeddings depend on the deep semantic encoder; (b) The dimension d plays significant role on generating paper embeddings. The best embeddings are learned when d is around 128 for the paper retrieval task.

4.4 Node Recommendation (RQ1-3)

Which venues are suitable for you? As a response to **RQ1-3**, we design an experiment to evaluate SHNE's performance on the venue recommendation task.

4.4.1 Experimental Setting. As in the previous two tasks, the network before T is utilized to learn node embeddings. The ground truth of recommendation is based on author's appearance (has papers published) in venue after T. The preference score is defined as the cosine similarity between embeddings of author and venue. We use **Recall@k** as the evaluation metric for recommendation task and k is set to 5 or 10. In addition, duplicated author-venue pairs are removed from evaluation. The reported score is the average value over all evaluated authors.

4.4.2 Results. The results are reported in Table 3. From the table: (a) All variants of SHNE achieve better performance than baselines, showing the benefit of incorporating unstructured semantic content for learning author and venue embeddings; (b) The average improvement of SHNE$^{mw}_{SE-HSG}$ over different baselines is significant, and ranges from 18.9% to 55.1% and 11.7% to 66.1% on AMiner-*I* and AMiner-*II*; (c) The results of different variants of SHNE are close due to relative small recall values. However, we find that SHNE$^{rw}_{HSG-SR}$ is the worst among the four, indicating both meta-path walk and semantic enhanced SkipGram help improve the performance of SHNE in the venue recommendation task.

4.4.3 Parameter Sensitivity. Figure 5 shows the impact of window size τ and embedding dimension d on the performance of SHNE$^{mw}_{SE-HSG}$ on AMiner-*II* (T = 2013). Note that SHNE$^{mw}_{SE-HSG}$ achieves the best results when τ is around 7 and d is around 128 for the venue recommendation task.

4.5 Relevance Search: Case Study (RQ2)

To answer **RQ2**, we present three case studies of relevance search on AMiner-*II* (T = 2013) to show the performance difference between SHNE$^{mw}_{SE-HSG}$ and baselines. The ranking of each search result is based on the cosine similarity of embeddings.

4.5.1 Relevant Author Search. Table 4 lists the top-5 returned authors for query author **"Jure Leskovec"** of SHNE$^{mw}_{SE-HSG}$, node2vec, and metapath2vec. According to this table: (a) most of the authors returned by node2vec have collaboration relations with "Jure" before T, indicating that node2vec highly depends on structural closeness and cannot find relevant authors who are far away from "Jure" in the network; (b) metapath2vec returns some authors (*e.g.*, P. Nguyen) who are different from "Jure" in their specific research interests, illustrating that "APVPA" meta-path walks (used by metapath2vec) may collect context nodes that are different from target node since it is common that authors bridged by the same venue study different research topics; (c) SHNE$^{mw}_{SE-HSG}$ not only returns structurally close authors who have collaboration relations with "Jure" before T but also finds farther authors (*e.g.*, D. Romero) who share similar research interest with "Jure", demonstrating that SHNE$^{mw}_{SE-HSG}$ captures both structural closeness and unstructured semantic relations for learning author embeddings.

4.5.2 Relevant Paper Search. Table 5 lists the top-5 returned papers for query paper **"When will it happen?"** of SHNE$^{mw}_{SE-HSG}$, node2vec, and metapath2vec. From this table: (a) all papers returned by node2vec are written by at least one author in query paper, showing that node2vec only returns structurally close papers but has difficulty finding farther semantically related papers; (b) all returned papers of metapath2vec belong to WSDM venue, *i.e.*, the same venue as the query paper, indicating "APVPA" meta-path walks (used by metapath2vec) only capture structural closeness inferred by meta-path walks; (c) SHNE$^{mw}_{SE-HSG}$ not only returns structurally close papers which have common authors with the query paper but also finds semantically related papers without authorship overlap, showing that SHNE$^{mw}_{SE-HSG}$ utilizes both structural content and unstructured semantic content for learning paper embeddings.

4.5.3 Relevant Author-Venue Search. Table 6 lists the top-10 returned venues for query author **"Chi Wang"** of SHNE$^{mw}_{SE-HSG}$, Deepwalk, node2vec, and metapath2vec. We find that: (a) Deepwalk, node2vec, and metapath2vec recommend both data mining

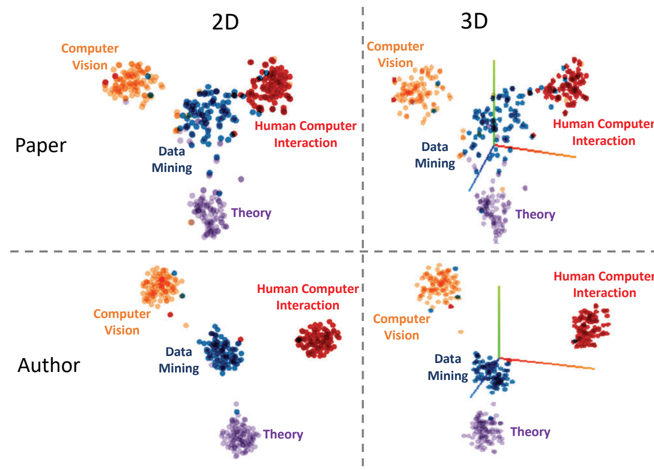


Figure 6: Embedding visualizations of paper and author in four selected research classes.

(*e.g.*, KDD/ICDM) and database (*e.g.*, SIGMOD/ PVLDB) venues to "Chi" since some of his works cite database papers and some of his co-authors focus on database research, which illustrates that both Deepwalk and node2vec return structurally close venues, some of which are not the most suitable ones; (b) most of the venues in SHNE$^{mw}_{SE-HSG}$'s recommendation list belong to data mining related areas, demonstrating that incorporating semantic content helps learn better embeddings of author and venue.

4.6 Class Visualization (RQ3)

The previous experiments and case studies demonstrate the effectiveness of SHNE in learning embeddings of different types of nodes. Moreover, to answer **RQ3** and provide a visual study of node embeddings, we employ the Tensorflow embedding projector to visualize paper and author embeddings learned by SHNE$^{mw}_{SE-HSG}$. Figure 6 shows the results of paper and author nodes of four selected research classes, *i.e.*, Data Mining (DM), Computer Vision (CV), Human Computer Interaction (HCI) and Theory, on AMiner-*II* (T = 2013). Specifically, we choose three top venues[2] for each area. Each paper is assigned according to venue's area and the class of each author is assigned to the area with the majority of his/her publications. We randomly sample 100 papers/authors of each class. According to Figure 6: (a) The embeddings of papers in the same class cluster closely and can be well discriminated from others for both 2D and 3D visualizations, indicating that the semantic encoder of SHNE achieves satisfactory performance in inferring deep semantic embeddings of papers. Note that papers belonging to DM have few intersections with the other three since semantic content of few DM papers are quite similar to CV, HCI and Theory papers *w.r.t.* model, application and theoretical basis, respectively. (b) The embeddings of authors in the same class are clearly discriminated from others for both 2D and 3D visualizations, which demonstrates the effectiveness of SHNE in learning author embeddings.

[2] DM: KDD, WSDM, ICDM. CV: CVPR, ICCV, ECCV. HCI: CHI, CSCW, UIST. T: SODA, STOC, FOCS

Table 4: Case study of relevant author search. "Coauthor-b" denotes whether two authors have a collaboration relation before T and "Similar-I" represents whether two authors have similar research interests.

Query: Jure Leskovec (2009 SIGKDD Dissertation Award, Research Interest: Network Mining & Social Computing)									
Rank	node2vec			metapath2vec			$\text{SHNE}^{mw}_{SE-HSG}$		
	Author	Coauthor-b ?	Similar-I ?	Author	Coauthor-b ?	Similar-I ?	Author	Coauthor-b ?	Similar-I ?
1	S. Kairam	✓	✓	L. Backstrom	✓	✓	J. Kleinberg	✓	✓
2	M. Rodriguez	✓	✓	P. Nguyen	✗	✗	D. Romero	✗	✓
3	D. Wang	✓	✓	S. HanhijÃďrvi	✗	✗	A. Dasgupta	✓	✓
4	J. Yang	✓	✓	S. Myers	✓	✓	L. Backstrom	✓	✓
5	A. Jaimes	✗	✗	V. Lee	✗	✓	G. Kossinets	✗	✓

Table 5: Case study of relevant paper search.

Query: When will it happen?: relationship prediction in heterogeneous information networks (WSDM2012), A: Y. Sun, J. Han, C. Aggarwal, N. Chawla		
Model	Rank	Returned Paper
node2vec	1	Co-author relationship prediction in heterogeneous bibliographic networks (ASONAM2011), A: Y. Sun, R. Barber, M. Gupta, C. Aggarwal, J. Han
	2	A framework for classification and segmentation of massive audio data streams (KDD2007), A: C. Aggarwal
	3	Mining heterogeneous information networks: the next frontier (KDD2012), A: J. Han
	4	Ranking-based classification of heterogeneous information networks (KDD2011), A: M. Ji, J. Han, M. Danilevsky
	5	Evolutionary clustering and analysis of bibliographic networks (ASONAM2011), A: M. Gupta, C. Aggarwal, J. Han, Y. Sun
metapath2vec	1	Multi-relational matrix factorization using bayesian personalized ranking for social network data (WSDM2012), A: A. Grimberghe, et al.
	2	Collective extraction from heterogeneous web lists (WSDM2011), A: A. Machanavajjhala, A. Lyer, P. Bohannon, S. Merugu
	3	The life and death of online groups: predicting group growth and longevity (WSDM2012), A: S. Kairam, D. Wang, J. Leskovec
	4	Exploiting statistical and relational information on the web and in social media (WSDM2011), A: L. Getoor, L. Mihalkova
	5	Query reformulation using anchor text (WSDM2010), A: V. Dang, B. W. Croft
$\text{SHNE}^{mw}_{SE-HSG}$	1	Collective prediction of multiple types of links in heterogeneous information networks (ICDM2014), A: B. Cao, X. Kong, P. Yu
	2	Community detection in incomplete information networks (WWW2012), A: W. Lin, X. Kong, P. Yu, Q. Wu, Y. Jia, C. Li
	3	Meta path-based collective classification in heterogeneous information networks (CIKM2012), A: X. Kong, P. Yu, Y. Ding, D. Wild
	4	Ranking-based classification of heterogeneous information networks (KDD2011), A: M. Ji, J. Han, M. Danilevsky
	5	Fast computation of SimRank for static and dynamic information networks (EDBT2010), A: C. Li, J. Han, G. He, X. Jin, Y. Sun, Y. Yu, T. Wu

Table 6: Case study of relevant author-venue search.

Query: Chi Wang (2015 SIGKDD Dissertation Award) Research Interest: Unstructured Data/Text Mining				
Rank	Deepwalk	node2vec	metapath2vec	$\text{SHNE}^{mw}_{SE-HSG}$
1	KDD	KDD	KDD	KDD
2	CIKM	WSDM	ASONAM	CIKM
3	DASFAA	SIGMOD	EDBT	ICDM
4	SIGMOD	PVLDB	SIGMOD	PAKDD
5	ICDM	WWW	DEXA	WWW
6	WSDM	ICDM	DASFAA	WWWC
7	ICDE	EDBT	SIGIR	TKDE
8	PVLDB	PAKDD	PVLDB	KAIS
9	WWW	GRC	ICDE	DASFAA
10	EDBT	FPGA	GRC	WSDM

5 RELATED WORK

In the past decade, many works have been devoted to mining HetNets for different applications, such as relevance search [3, 10, 27, 43, 44], node clustering [28, 29], personalized recommendation [22, 38, 42].

Network representation learning [5] has gained a lot of attention in the last few years. Walk sampling based models [6, 7, 19] have been proposed to learn vectorized node embeddings that can be further utilized in various tasks. Specifically, inspired by word2vec [17] for learning distributed representations of words in text corpus, Perozzi *et al*. developed the innovative Deepwalk [19] which introduces node-context concept in network (analogy to word-context) and feeds a set of random walks over network (analogy to "sentences") to SkipGram for learning node embeddings. In order to deal with neighborhood diversity, Grover & Leskovec suggested taking biased random walks (a mixture of BFS and DFS) as the input of SkipGram. More recently, heterogeneous network representation learning approaches [1, 2, 6, 16, 21, 23, 24, 37] and attribute/semantic-aware network embedding methods [12, 13, 25, 33, 39] were proposed to tackle network heterogeneity and network attributed content. However, they either cannot capture unstructured semantic relations of HetNets or require attribute/semantic content of each node in HetNets. In addition, many other models have been proposed [8, 15, 18, 20, 30, 34, 36, 40, 41], such as PTE [30] for text data embedding, GraphSAGE [8] for inductive representation learning on network and NetMF [20] for unifying network representation models via matrix factorization.

Our work furthers the investigation of network representation learning by developing a semantic-aware heterogeneous network embedding model SHNE. Unlike previous models, SHNE leverages both structural closeness and unstructured semantic relations to learn node embeddings, and can be directly applied to partially semantic-associated HetNets.

6 CONCLUSION

In this paper, we formalized the problem of semantic-aware representation learning in HetNets and proposed a novel model SHNE to solve the problem. SHNE performs joint optimization of heterogeneous SkipGram and deep semantic encoding for capturing both structural closeness and unstructured semantic relations in a HetNet. Extensive experiments demonstrated that SHNE outperforms state-of-the-art baselines in various HetNet mining tasks, such as link prediction, document retrieval, node recommendation, and relevance search. The class visualization of node embeddings further showed the effectiveness of SHNE.

ACKNOWLEDGMENTS

This work is supported by the Army Research Laboratory under Cooperative Agreement Number W911NF-09-2-0053 (Network Science CTA) and the National Science Foundation (NSF) grant IIS-1447795.

REFERENCES

[1] Shiyu Chang, Wei Han, Jiliang Tang, Guo-Jun Qi, Charu C Aggarwal, and Thomas S Huang. 2015. Heterogeneous network embedding via deep architectures. In *KDD*. 119–128.

[2] Hongxu Chen, Hongzhi Yin, Weiqing Wang, Hao Wang, Quoc Viet Hung Nguyen, and Xue Li. 2018. PME: Projected Metric Embedding on Heterogeneous Networks for Link Prediction. In *KDD*. 1177–1186.

[3] Ting Chen and Yizhou Sun. 2017. Task-Guided and Path-Augmented Heterogeneous Network Embedding for Author Identification. In *WSDM*. 295–304.

[4] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv:1406.1078* (2014).

[5] Peng Cui, Xiao Wang, Jian Pei, and Wenwu Zhu. 2018. A survey on network embedding. *TKDE* (2018).

[6] Yuxiao Dong, Nitesh V Chawla, and Ananthram Swami. 2017. metapath2vec: Scalable representation learning for heterogeneous networks. In *KDD*. 135–144.

[7] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *KDD*. 855–864.

[8] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *NIPS*. 1024–1034.

[9] Xiao Huang, Qingquan Song, Jundong Li, and Xia Hu. 2018. Exploring expert cognition for attributed network embedding. In *WSDM*. 270–278.

[10] Zhipeng Huang, Yudian Zheng, Reynold Cheng, Yizhou Sun, Nikos Mamoulis, and Xiang Li. 2016. Meta structure: Computing relevance in large heterogeneous information networks. In *KDD*. 1595–1604.

[11] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[12] Jundong Li, Harsh Dani, Xia Hu, Jiliang Tang, Yi Chang, and Huan Liu. 2017. Attributed network embedding for learning in a dynamic environment. In *CIKM*. 387–396.

[13] Jie Liu, Zhicheng He, Lai Wei, and Yalou Huang. 2018. Content to Node: Self-Translation Network Embedding. In *KDD*. 1794–1802.

[14] Xiaozhong Liu, Yingying Yu, Chun Guo, and Yizhou Sun. 2014. Meta-path-based ranking with pseudo relevance feedback on heterogeneous graph for citation recommendation. In *CIKM*. 121–130.

[15] Jianxin Ma, Peng Cui, Xiao Wang, and Wenwu Zhu. 2018. Hierarchical Taxonomy Aware Network Embedding. In *KDD*. 1920–1929.

[16] Yao Ma, Zhaochun Ren, Ziheng Jiang, Jiliang Tang, and Dawei Yin. 2018. Multi-Dimensional Network Embedding with Hierarchical Structure. In *WSDM*. 387–395.

[17] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*. 3111–3119.

[18] Jingchao Ni, Shiyu Chang, Xiao Liu, Wei Cheng, Haifeng Chen, Dongkuan Xu, and Xiang Zhang. 2018. Co-Regularized Deep Multi-Network Embedding. In *WWW*. 469–478.

[19] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online learning of social representations. In *KDD*. 701–710.

[20] Jiezhong Qiu, Yuxiao Dong, Hao Ma, Jian Li, Kuansan Wang, and Jie Tang. 2018. Network embedding as matrix factorization: Unifying deepwalk, line, pte, and node2vec. In *WSDM*. 459–467.

[21] Meng Qu, Jian Tang, and Jiawei Han. 2018. Curriculum Learning for Heterogeneous Star Network Embedding via Deep Reinforcement Learning. In *WSDM*. 468–476.

[22] Xiang Ren, Jialu Liu, Xiao Yu, Urvashi Khandelwal, Quanquan Gu, Lidan Wang, and Jiawei Han. 2014. Cluscite: Effective citation recommendation by information network-based clustering. In *KDD*. 821–830.

[23] Yu Shi, Huan Gui, Qi Zhu, Lance Kaplan, and Jiawei Han. 2018. AspEm: Embedding Learning by Aspects in Heterogeneous Information Networks. In *SDM*. 144–152.

[24] Yu Shi, Qi Zhu, Fang Guo, Chao Zhang, and Jiawei Han. 2018. Easing Embedding Learning by Comprehensive Transcription of Heterogeneous Information Networks. In *KDD*. 2190–2199.

[25] Xiaofei Sun, Jiang Guo, Xiao Ding, and Ting Liu. 2016. A General Framework for Content-enhanced Network Representation Learning. *arXiv preprint arXiv:1610.02906* (2016).

[26] Yizhou Sun, Jiawei Han, Charu C Aggarwal, and Nitesh V Chawla. 2012. When will it happen?: relationship prediction in heterogeneous information networks. In *WSDM*. 663–672.

[27] Yizhou Sun, Jiawei Han, Xifeng Yan, Philip S Yu, and Tianyi Wu. 2011. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. *VLDB* 4, 11 (2011), 992–1003.

[28] Yizhou Sun, Brandon Norick, Jaiwei Han, Xifeng Yan, Philip Yu, and Xiao Yu. 2012. PathSelClus: Integrating Meta-Path Selection with User-Guided Object Clustering in Heterogeneous Information Networks. In *KDD*. 1348–1356.

[29] Yizhou Sun, Yintao Yu, and Jiawei Han. 2009. Ranking-based clustering of heterogeneous information networks with star network schema. In *KDD*. 797–806.

[30] Jian Tang, Meng Qu, and Qiaozhu Mei. 2015. Pte: Predictive text embedding through large-scale heterogeneous text networks. In *KDD*. 1165–1174.

[31] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. 2015. Line: Large-scale information network embedding. In *WWW*. 1067–1077.

[32] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. 2008. Arnetminer: extraction and mining of academic social networks. In *KDD*. 990–998.

[33] Cunchao Tu, Han Liu, Zhiyuan Liu, and Maosong Sun. 2017. Cane: Context-aware network embedding for relation modeling. In *ACL*. 1722–1731.

[34] Ke Tu, Peng Cui, Xiao Wang, Philip S Yu, and Wenwu Zhu. 2018. Deep Recursive Network Embedding with Regular Equivalence. In *KDD*. 2357–2366.

[35] Chi Wang, Jiawei Han, Yuntao Jia, Jie Tang, Duo Zhang, Yintao Yu, and Jingyi Guo. 2010. Mining advisor-advisee relationships from research publication networks. In *KDD*. 203–212.

[36] Daixin Wang, Peng Cui, and Wenwu Zhu. 2016. Structural deep network embedding. In *KDD*. 1225–1234.

[37] Hongwei Wang, Fuzheng Zhang, Min Hou, Xing Xie, Minyi Guo, and Qi Liu. 2018. SHINE: signed heterogeneous information network embedding for sentiment link prediction. In *WSDM*. 592–600.

[38] Carl Yang, Lanxiao Bai, Chao Zhang, Quan Yuan, and Jiawei Han. 2017. Bridging Collaborative Filtering and Semi-Supervised Learning: A Neural Approach for POI Recommendation. In *KDD*. 1245–1254.

[39] Cheng Yang, Zhiyuan Liu, Deli Zhao, Maosong Sun, and Edward Y Chang. 2015. Network representation learning with rich text information. In *IJCAI*. 2111–2117.

[40] Zhilin Yang, William W Cohen, and Ruslan Salakhutdinov. 2016. Revisiting semi-supervised learning with graph embeddings. In *ICML*. 40–48.

[41] Wenchao Yu, Cheng Zheng, Wei Cheng, Charu C Aggarwal, Dongjin Song, Bo Zong, Haifeng Chen, and Wei Wang. 2018. Learning Deep Network Representations with Adversarially Regularized Autoencoders. In *KDD*. 2663–2671.

[42] Xiao Yu, Xiang Ren, Yizhou Sun, Quanquan Gu, Bradley Sturt, Urvashi Khandelwal, Brandon Norick, and Jiawei Han. 2014. Personalized entity recommendation: A heterogeneous information network approach. In *WSDM*. 283–292.

[43] Chuxu Zhang, Chao Huang, Lu Yu, Xiangliang Zhang, and Nitesh V Chawla. 2018. Camel: Content-Aware and Meta-path Augmented Metric Learning for Author Identification. In *WWW*. 709–718.

[44] Chuxu Zhang, Lu Yu, Xiangliang Zhang, and Nitesh V Chawla. 2018. Task-Guided and Semantic-Aware Ranking for Academic Author-Paper Correlation Inference.. In *IJCAI*. 3641–3647.