Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18)

Task-Guided and Semantic-Aware Ranking for Academic Author-Paper Correlation Inference

Chuxu Zhang[+], Lu Yu[*], Xiangliang Zhang[*] and Nitesh V. Chawla[+]
[+]University of Notre Dame, IN 46556, USA
[*]King Abdullah University of Science and Technology, Thuwal, 23955, SA
{czhang11,nchawla}@nd.edu, {lu.yu,xiangliang.zhang}@kaust.edu.sa

Abstract

We study the problem of author-paper correlation inference in big scholarly data, which is to effectively infer potential correlated works for researchers using historical records. Unlike supervised learning algorithms that predict relevance score of author-paper pair via time and memory consuming feature engineering, network embedding methods automatically learn nodes' representations that can be further used to infer author-paper correlation. However, most current models suffer from two limitations: (1) they produce general purpose embeddings that are independent of the specific task; (2) they are usually based on network structure but out of content semantic awareness. To address these drawbacks, we propose a task-guided and semantic-aware ranking model. First, the historical interactions among all correlated author-paper pairs are formulated as a pairwise ranking loss. Next, the paper's semantic embedding encoded by gated recurrent neural network, together with the author's latent feature is used to score each author-paper pair in ranking loss. Finally, a heterogeneous relations integrative learning module is designed to further augment the model. The evaluation results of extensive experiments on the well known AMiner dataset demonstrate that the proposed model reaches significant better performance, comparing to a number of baselines.

1 Introduction

Due to the growing evolution of scientific research and increasing data collections by various online services such as Google Scholar, Microsoft Academic or AMiner, the problems of mining big scholarly data have gained a lot of attention in the past decade. Examples include scientific impact modeling and prediction [Wang *et al.*, 2013; Shen *et al.*, 2014; Dong *et al.*, 2015], heterogeneous bibliographic network analysis [Sun *et al.*, 2012; Huang *et al.*, 2016; Chen and Sun, 2017], etc.

In this work, we consider the problem of author-paper correlation inference in big scholarly data. Specifically, given an author's previous correlated papers (e.g., publications or references), we would like to effectively infer the potential relevant works for him/her, such that the author will interact (e.g., cite) with those papers in the future. Solutions of the problem bring important implications to researchers with different knowledgeable levels. For example, an effective algorithm provides suitable academic paper reading lists for new PhD students with little historical feedback in database, or helps active scientists track the related or following works of their previous publications. In addition, it can be a good reference for recommender system design in digital libraries such as Elsevier or Springer.

As one of the representative solutions, supervised leaning algorithms can be applied to predict the correlation score between author and paper, as they were used in 2013 KDD cup author-paper pair identification challenge [Efimov *et al.*, 2013; Li *et al.*, 2015]. However, such methods heavily rely on time and memory consuming feature engineering, and the extracted features may be too simple to capture complicated relations. In recent years, unlike traditional supervised leaning, several network embedding models [Perozzi *et al.*, 2014; Grover and Leskovec, 2016; Chen and Sun, 2017; Dong *et al.*, 2017] have been proposed to automatically learn nodes' representations that can be further used for various applications in scholarly data such as author-paper correlation inference. Although the proximity among nodes is preserved by dense vectors, most of the existing embedding models suffer from following two drawbacks:

- They produce general purpose embeddings that are independent of task, no matter what kind (homogeneous or heterogeneous) of networked data is used. However, when it comes to the author-paper correlation inference problem, nodes should be embedded under the guidance of target for generating task-specific representations.
- Even though they take account of the task for embedding generation, e.g., TaskE [Chen and Sun, 2017], they are purely based on network structure and out of content awareness. Actually, content (e.g., abstract of paper) contains useful semantic information that should be used for encoding nodes into a better feature space.

To address the above issues and solve the given problem, we propose a task-guided and semantic-aware ranking model. First, we model the historical interactions among correlated author-paper pairs via pairwise ranking according to the spe-

cific task. Next, we introduce the gated recurrent neural network to encode paper's content, and combine the obtained semantic embedding with author's latent feature to score each author-paper pair in ranking loss. Moreover, we design a heterogeneous relations integrative learning module to formulate the indirect correlations among authors and papers, and further augment the model. Finally, a relations sampling based mini-batch gradient descent algorithm is designed for model training. The main contributions of this paper are summarized as follows:

- We study the author-paper correlation inference problem in big scholarly data, which brings important implications to academic community.
- To solve the problem, we propose a model by jointly content semantic encoding and heterogeneous relations augmented ranking, and design the corresponding learning algorithm.
- We conduct extensive experiments to evaluate the performance of the proposed model on the well known AMiner dataset. The results show that our model significantly outperforms a number of baselines.

2 Problem

We introduce few notations that will be used throughout this paper. Specifically, we denote the sets of authors and papers as U and I, respectively. Let $l^u_{<T}$ be the set of both author u's publications and references before a given timestamp T. Similarly, $l^u_{\geq T}$ represents u's papers and references after T. In this work, the correlated papers of each author are assumed as both publications and references in dataset. The problem is formalized as:

Academic Author-Paper Correlation Inference. *Given $l^u_{<T}$ of each author $u \in U$ and the content (i.e., abstract) of each paper $v \in I$, the goal is to learn a model to rank all potential papers $v' \in I \backslash l^u_{<T}$ for u, such that its top return papers are in $l^u_{\geq T}$.*

Note that each author can cite previous works or write new papers, thus the return papers can be published both before and after T. In addition, the overlapping between $l^u_{<T}$ and $l^u_{\geq T}$ are removed from $l^u_{\geq T}$, so that papers in $l^u_{\geq T}$ are never cited by u before T.

3 Proposed Model

In this section, we present how to design task-guided and semantic-aware ranking model for solving the problem, and use historical academic data to construct a heterogeneous network for capturing indirect author-paper relations which benefit and augment the model.

3.1 Pairwise Ranking with Gated Recurrent Neural Network

We model historical interactions among correlated author-paper pairs via pairwise ranking optimization [Rendle *et al.*, 2009]. Specifically, for a given author u, the correlated paper $v \in l^u_{<T}$ should be ranked higher than the uncorrelated paper $v' \in I \backslash l^u_{<T}$. In other words, the relevance score $s_{u,v}$ of $\langle u, v \rangle$ pair should be larger than that of $\langle u, v' \rangle$ pair as much as possible, leading to an author-paper pairwise ranking loss as follows:

$$\mathcal{L}_{rank} = \sum_{u \in U} \sum_{v \in l^u_{<T}} \sum_{v' \notin l^u_{<T}} \Big[-\log \, \sigma(s_{u,v} - s_{u,v'}) \Big] \quad (1)$$

where σ is the sigmoid function. For each author-paper pair $\langle u, v \rangle$, we introduce author u's latent feature $\mathbf{q}_u \in \mathbb{R}^d$ (d: the embedding dimension) and represent paper v as semantic embedding $\mathbf{E}_{\mathbf{p}_v} \in \mathbb{R}^d$ via content encoder f: $\mathbf{E}_{\mathbf{p}_v} = f(\mathbf{p}_v)$, where $\mathbf{p}_v$ represents paper v's content. The inner product of $\mathbf{q}_u$ and $f(\mathbf{p}_v)$ is used to measure the relevance score, i.e., $s_{u,v} = \mathbf{q}_u{}^T f(\mathbf{p}_v)$.

To encode papers' contents to fixed size embeddings $\mathbf{E}_{\mathbf{p}} \in \mathbb{R}^{|I| \times d}$, we use the gated recurrent units (GRU), a specific type of recurrent neural network (RNN), which has been widely adopted for many applications such as machine translation [Cho *et al.*, 2014]. Specifically, a paper is represented by a sequence of word embeddings: $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_{t_{max}}\}$, where $\mathbf{x}_t$ denotes the t-th word embedding pre-trained by word2vec [Mikolov *et al.*, 2013] and t_{max} is the maximum length of paper's abstract. For each step t with the input of word embedding $\mathbf{x}_t$ and hidden state vector $\mathbf{h}_{t-1}$, GRU computes the updated hidden state vector via $\mathbf{h}_t = \mathbf{GRU}(\mathbf{x}_t, \mathbf{h}_{t-1})$, where GRU module is defined as:

$$\begin{aligned} \mathbf{z}_t &= \sigma(\mathbf{N}_z \mathbf{x}_t + \mathbf{M}_z \mathbf{h}_{t-1}) \\ \mathbf{r}_t &= \sigma(\mathbf{N}_r \mathbf{x}_t + \mathbf{M}_r \mathbf{h}_{t-1}) \\ \hat{\mathbf{h}}_t &= \tanh[\mathbf{N}_h \mathbf{x}_t + \mathbf{M}_h(\mathbf{r}_t \circ \mathbf{h}_{t-1})] \\ \mathbf{h}_t &= \mathbf{z}_t \circ \mathbf{h}_{t-1} + (1 - \mathbf{z}_t) \circ \hat{\mathbf{h}}_t \end{aligned} \quad (2)$$

where σ is the sigmoid function, the operator $\circ$ denotes element-wise multiplication, $\mathbf{N}$ and $\mathbf{M}$ are parameter matrices of GRU network, $\mathbf{z}_t$ and $\mathbf{r}_t$ are updated gate vector and reset gate vector, respectively. We apply the above GRU network to encode words' contextual embeddings $\mathbf{h} \in \mathbb{R}^{t_{max} \times d}$ and use a mean pooling layer to obtain the general semantic embedding of each paper. All of these steps construct the paper's content encoder f, as illustrated by Figure 1(a). Note that, we also explore other encoding architectures such as LSTM or attention-based GRU but obtain similar result. Thus we choose GRU since it has a concise structure for reducing training time.

Using the above GRU encoder for papers' embeddings and the authors' latent features, we further minimize the pairwise ranking loss (i.e., Eq. (1)) via gradient descent approach, as illustrated by Part-1 of Figure 1(b). The process leads to task-guided and semantic-aware ranking and we name it TSR.

3.2 Heterogeneous Relations based Integrative Learning Augmentation

TSR trains model by only using direct correlations, i.e., correlated author-paper pairs in $l^u_{<T}$ for each author u. However, there are multiple indirect author-paper relations, which can be inferred from direct correlations and useful for improving TSR. Inspired by DeepWalk [Perozzi *et al.*, 2014], we apply random walk to collect those indirect correlations. Figure 1(b) Part-2 gives the illustration. First, we use the correlated papers set $l^u_{<T}$ (contains both author u's publications and references before T) of each author u to create a heterogeneous

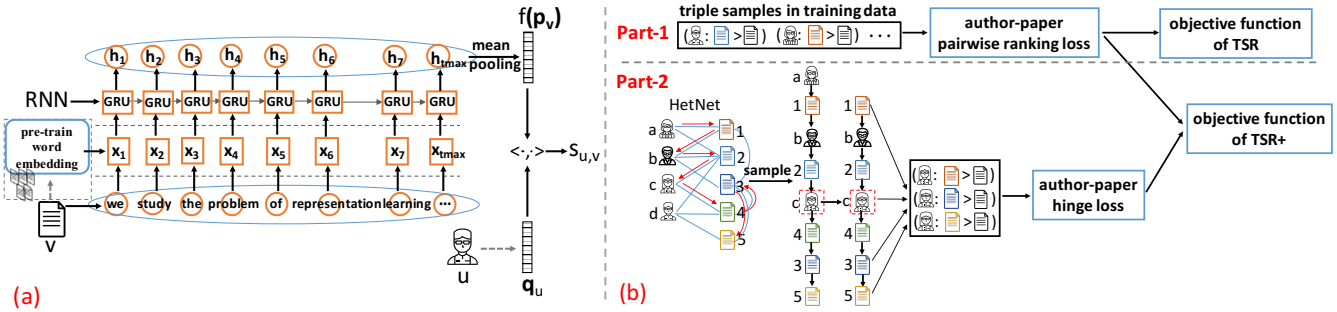


Figure 1: Illustration of (a) GRU network for encoding paper's semantic embedding and (b) heterogeneous relations integrative learning module for augmenting TSR.

network (HetNet) with two kinds of nodes (author and paper) and two kinds of undirected edges (author $\underset{\text{write}^{-1}}{\overset{\text{write}}{\rightleftharpoons}}$ paper and paper $\underset{\text{cite}^{-1}}{\overset{\text{cite}}{\rightleftharpoons}}$ paper). Then we perform random walk sampling over HetNet to generate a set of node sequences. For example, in Figure (b) Part-2, a heterogeneous walk: $\boldsymbol{w}_a \equiv \{a \rightarrow 1 \rightarrow b \rightarrow 2 \rightarrow c \rightarrow 4 \rightarrow 3 \rightarrow 5\}$ is generated and the surrounding context of each node in $\boldsymbol{w}_a$ implies different relations among authors and papers within $\boldsymbol{w}_a$. We use sub-sequence $\hat{\boldsymbol{w}}_1 \equiv \{1 \rightarrow b \rightarrow 2 \rightarrow c \rightarrow 4 \rightarrow 3 \rightarrow 5\}$ (window size equals to 3) centered at author c as an illustration. Besides the direct connections, e.g., author c writes paper 2, $\hat{\boldsymbol{w}}_1$ also captures multiple indirect relations. For example, author c has indirect citation relation with paper 5 since s/he has citation relation with paper 3. Therefore, each heterogeneous walk contains both direct correlations and indirectly transitive relations among authors and papers.

To formulate indirect relations within each walk, we design a heterogeneous relations integrative learning module (HRIL) to augment TSR based on a reasonable assumption that the relevance scores of indirectly correlated author-paper pairs should be larger than those of uncorrelated pairs. Specifically, we introduce a hinge loss to formulate the difference of author u's correlations to indirectly correlated paper v and uncorrelated paper v':

$$H(u, v, v') = \Big[\xi + s_{uv'} - s_{uv}\Big]_+ \tag{3}$$

where $\{x\}_+ = max(x, 0)$ and ξ is a positive margin value. A loss penalty will incur if the score of $\langle u, v\rangle$ is not at least ξ larger than that of $\langle u, v'\rangle$. Such formulation has been widely adopted in recent works [Chen and Sun, 2017; Zhang *et al.*, 2017; 2018] for modeling preference differences. Thus the overall loss in each walk $\boldsymbol{w}$ is formulated as:

$$\mathcal{L}_{hinge}(\boldsymbol{w}) = \sum_{u \in \boldsymbol{w}} \sum_{\substack{v \in \boldsymbol{w}[I_u - \tau : I_u + \tau] \\ v \notin l^u_{<T}}} \sum_{v' \notin \boldsymbol{l}^u_{<T}} \Big[\xi + s_{uv'} - s_{uv}\Big]_+ \tag{4}$$

where τ is the window size of surrounding context used for relations extraction, and I_u indicates the position of u in $\boldsymbol{w}$. Therefore constraining $\mathcal{L}_{hinge}$ obeys our assumption.

3.3 Model Training

We generate a plenty of walks rooted at each author node to collect indirect correlations among authors and papers, and let

Algorithm 1: *Learning Framework of TSR+*

input : C_{rank} in training data and C_{hinge} collected by random walk sampling on HetNet
output: authors' latent features $\boldsymbol{q}$, GRU encoder parameter matrices $\boldsymbol{N}$ and $\boldsymbol{M}$ (for papers' embeddings $f(\boldsymbol{p})$)

while *not converged* **do**
  sample a mini-batch of (u, v, v') triples in C_{rank};
  sample a mini-batch of (u, v, v') triples in C_{hinge};
  accumulate the loss by Equation (6) ;
  update the parameters by Adam Optimizer
end

$\boldsymbol{W}$ be the set of all walks. The objective of augmented TSR (TSR+) is defined as the combination of TSR and HRIL:

$$\mathcal{L} = \mathcal{L}_{rank} + \sum_{\boldsymbol{w} \in \boldsymbol{W}} \mathcal{L}_{hinge}(\boldsymbol{w}) + \mathcal{L}_{reg} \tag{5}$$

where $\mathcal{L}_{reg}$ is the regularization term for avoiding overfitting. We denote all model parameters including GRU network coefficients (for generating papers' embeddings) and authors' latent features as $\boldsymbol{\Theta}$, and let C_{rank} and C_{hinge} be the sets of (u, v, v') triples in $\mathcal{L}_{rank}$ and (u, v, v') triples in $\mathcal{L}_{hinge}$, respectively. Thereafter we can rewrite the objective of TSR+ as:

$$\begin{aligned}\mathcal{L} = &\sum_{(u,v,v') \in C_{rank}} \Big\{ -\log\, \sigma\Big[\mathbf{q}_u{}^T f(\mathbf{p}_v) - \mathbf{q}_u{}^T f(\mathbf{p}_{v'})\Big]\Big\} \\ &+ \sum_{(u,v,v') \in C_{hinge}} \Big\{\xi + \mathbf{q}_u{}^T f(\mathbf{p}_{v'}) - \mathbf{q}_u{}^T f(\mathbf{p}_v)\Big\}_+ + \lambda \left\|\boldsymbol{\Theta}\right\|^2\end{aligned} \tag{6}$$

where parameter λ controls regularization penalty. To minimize the above objective function, we design a relations sampling based mini-batch Adam Optimizer [Kingma and Ba, 2014], as illustrated by the pseudocode in Algorithm 1.

4 Experiments

In this section, we conduct extensive experiments to evaluate the proposed model and various baselines. Case studies are also provided.

Statistics	AMiner-T	AMiner-F
# authors	28,646	571,563
# papers	21,044	483,319
# venues	18	492
# correlated papers of authors	271,777	4,646,671

Table 1: Statistics of datasets.

4.1 Experimental Design

Dataset.

We use the public available dataset[1] AMiner [Tang *et al.*, 2008] between 2005 and 2015, and remove the papers published in venues (e.g., workshop) with limited number of publications and the instances without content (i.e., abstract). Besides, considering most of researchers pay attention to papers published in top venues and each research area has its own community, we extract a subset data of six areas according to Google Scholar Metrics[2], namely Artificial Intelligence (AI), Computer Vision (CV), Data Mining (DM), Databases (DB), Computational Linguistics (CL) and Information System (IS). For each area, we choose three top venues[3] that are considered to have influential publications. The main statistics of two datasets (AMiner-T and AMiner-F) are summarized in Table 1. Note that, there are some missing citations in the dataset. It makes the inference task more difficult due to potential noise in model training.

Baselines.

We compare TSR and TSR+ with eight baseline methods that span three categories: (1) feature-based supervised learning, (2) content-based ranking and (3) network embedding.

- **Feature-based supervised learning.** It first extracts author-paper paired features and then applies supervised learning algorithms to predict the relevance score of each author-paper pair. We extract 14 kinds of features (as shown in Table 2) and choose Bayesian Regressor (BayesR), Neural Network (NeuNet) and Random Forest (RandomF) as learning algorithms. For each correlated author-paper pair, we randomly sample 5 negative pairs to train model.
- **Content-based ranking.** It first encodes each paper's content (i.e., abstract) via language modeling and then applies pairwise ranking BPR [Rendle *et al.*, 2009] to learn each author's latent feature. We employ two popular models word2vec [Mikolov *et al.*, 2013] and paragraph vector (doc2vec) [Le and Mikolov, 2014] to learn papers' embeddings. Note that word2vec encodes embedding of each word in content, we connect the output of word2vec with a mean pooling layer to obtain general embedding of each paper.
- **Network embedding.** It learns embeddings of both authors and papers based on the structure of author-paper heterogeneous network (same as the HetNet in TSR+). We use both homogeneous model Deepwalk [Perozzi *et al.*, 2014] and heterogeneous model metapath2vec [Dong *et al.*, 2017]. In addition, a task-guided network embedding model (TaskE) [Chen and Sun, 2017] for author identification is introduced for comparison.

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

[2]https://scholar.google.com/citations?view_op=metrics_intro&hl=en

[3]AI: ICML, AAAI, IJCAI. CV: CVPR, ICCV, ECCV. DM: KDD, WSDM, ICDM. DB: SIGMOD, VLDB, ICDE. CL: ACL, EMNLP, NAACL. IS: WWW, SIGIR, CIKM.

No.	Feature description
1	# of the paper's references being cited by the author before
2	ratio of the paper's references being cited by the author before
3	ratio of the author's citations in the paper's references
4	# of paper's references in the author's previous publications
5	ratio of the paper's references in the author's previous publications
6	ratio of the author's publications in the paper's references
7	# of share keywords between author and paper
8	ratio of the author's keywords in share keywords
9	ratio of the paper's keywords in share keywords
10	whether the author attend the paper's venue before
11	# of times the author attend the paper's venue before
12	ratio of times the author attend the paper's venue before
13	# of papers the author published in 3 years before the paper's time
14	ratio of papers the author published in 3 years before the paper's time

Table 2: Features selection of supervised learning baselines. Keywords are extracted from title of each paper.

Evaluation Metrics.

As described in problem definition, for each author $u \in U$, papers in $l^u_{<T}$ are treated as training data and papers in $l^u_{\geq T}$ are left for evaluation. The overlapping between $l^u_{<T}$ and $l^u_{\geq T}$ are removed from $l^u_{\geq T}$. We use three popular metrics, i.e., Recall@k, Precision@k and AUC, to evaluate the performance of each method. The Recall@k shows the ratio of true correlated papers returned in the top-k list, which is defined as: $Rec@k = \frac{1}{|U|}\sum_{u\in U}\frac{|\hat{l}^u_{\geq T}\bigcap l^u_{\geq T}|}{|l^u_{\geq T}|}$, where $\hat{l}^u_{\geq T}$ denotes the set of top-k papers for author u. The Precision@k reflects the accuracy of top-k papers by a method and it can be computed according to: $Pre@k = \frac{1}{|U|}\sum_{u\in U}\frac{|\hat{l}^u_{\geq T}\bigcap l^u_{\geq T}|}{k}$. The AUC measures the accuracy of pairwise orders between correlated and uncorrelated papers of each author, which is formulated as: $AUC = \frac{1}{|U|}\sum_{u\in U}\frac{1}{|E(u)|}\sum_{(v,v')\in E(u)}\delta(s_{uv} > s_{uv'})$, where $E(u) \equiv \{(v,v')|v \in l^u_{\geq T}, v' \notin (l^u_{<T} \cup l^u_{\geq T})\}$ and δ is indicator function which equals 1 when the condition holds otherwise 0. The k is set to 10 and a larger Recall@k, Precision@k or AUC value means a better performance.

Experimental Settings.

All of information used for model training such as triple/pair samples in our models or the selected features in supervised learning baselines, are extracted from training data. We design two different training/test splits by setting $T = 2012$ and 2013. Besides, two key issues of the experiments are set as:

- **Reproducibility.** For the fair of comparison, we use the same feature dimension $d = 128$ for all content-based ranking and network embedding models. In the proposed models, the regularization parameter λ equals to 0.001. For HRIL module of TSR+, we set the number of walks (start from each author node) as 10 and the walk length as 20. Besides, we fix window size $\tau = 5$ and hinge loss margin $\xi = 0.1$. In addition, we employ TensorFlow to implement the proposed model and further conduct it via NVIDIA TITAN X GPU.

Dataset	T	Metric	BayesR	NeuNet	RandomF	word2vec +BPR	doc2vec +BPR	Deepwalk	metapath2vec	TaskE	TSR	TSR+
AMiner-T	2012	Rec@10	0.211	0.220	0.288	0.267	0.224	0.230	0.305*	0.298	0.312	**0.330**
		Pre@10	0.248	0.263	0.341	0.311	0.274	0.312	0.364*	0.355	0.350	**0.374**
		AUC	0.674	0.663	0.705	0.801*	0.739	0.675	0.723	0.738	0.831	**0.846**
	2013	Rec@10	0.224	0.255	0.301	0.272	0.228	0.189	0.302	0.303*	0.329	**0.343**
		Pre@10	0.221	0.246	0.304	0.276	0.244	0.237	0.325*	0.321	0.320	**0.341**
		AUC	0.694	0.693	0.730	0.785*	0.735	0.647	0.715	0.723	0.835	**0.842**
AMiner-F	2012	Rec@10	0.333	0.361	0.378*	0.372	0.341	0.245	0.365	0.351	0.444	**0.461**
		Pre@10	0.344	0.375	0.379	0.377	0.357	0.327	0.419*	0.397	0.433	**0.450**
		AUC	0.709	0.726	0.708	0.844*	0.808	0.698	0.740	0.721	0.869	**0.878**
	2013	Rec@10	0.353	0.406	0.404	0.399	0.362	0.286	0.412*	0.398	0.475	**0.496**
		Pre@10	0.331	0.383	0.368	0.366	0.344	0.331	0.423*	0.405	0.427	**0.445**
		AUC	0.721	0.742	0.720	0.850*	0.817	0.726	0.774	0.739	0.880	**0.883**
	Impv. over Baseline		37.7%	28.3%	18.1%	16.9%	28.9%	43.8%	13.0%	16.2%	–	–

Table 3: Performance comparisons of different models. The last row reports the average improvements (%) of TSR+ over baselines. The best baseline of each case is indicated by star notation. TSR+ achieves the best results (highlighted in bold) in all cases.

- **Evaluation candidates.** It is time and memory consuming to extract and store features for all author-paper pairs (which amounts to over 2.7×10^{11} pairs in AMiner-F). Thus the supervised learning baselines cannot scale up to such large amount of data. To deal with this issue and reduce evaluation time, we follow the setting [Chen and Sun, 2017] that randomly samples a set of negative (uncorrelated) papers and combines it with the set of correlated papers to form a candidate set of total 200 papers for each author. In addition, we eliminate the authors who have few correlated papers (less than 3) in test set to avoid noise. After that, the average sizes of authors' correlated papers sets respectively equal to 14.7 and 12.5 in AMiner-F test data for $T = 2012$ and 2013, and the corresponding values in AMiner-T test data are 15.7 and 12.7. Those relative small values make the 200 candidates large enough for a convincing evaluation. The reported results are averaged over 10 experiments of such setting.

4.2 Result Comparison

The performances of all methods are reported in Table 3, where the best results are highlighted in bold and the best baselines are indicated by star notation. The last row reports the average improvements (%) of TSR+ over different baselines. Note that, the embeddings of out-of-matrix papers (published after T) are missing in Deepwalk and metapath2vec. Accordingly we use the average of their in-matrix references' (published before T) embeddings to represent them. The main takeaways from this table are summarized as follows:

- The best content-based ranking method (word2vec+BPR) and the best network embedding models (TaskE, metapath2vec) have better average performances than the best supervised learning baseline (RandomF), which suggests that the vectorized representations generated by network embedding or content embedding are better for capturing the complicated correlations among author-paper pairs than the simple features extracted from data.
- TSR achieves better results than all baselines in most cases, showing that the joint model of deep semantic embedding and task-guided ranking is better than supervised learning, content-based ranking and network embedding for the given task.
- TSR+ performs best in all cases for both datasets. The average improvements of TSR+ over different baselines range from 13.0% to 43.8%. In addition, TSR+ outperforms TSR, which indicates that the heterogeneous relations integrative learning module further improves TSR.

4.3 Analysis and Discussion

Parameters Sensitivity.

The hyper-parameters play important roles in TSR+, as they determine how the model will be trained. We conduct experiments to analyze the impacts of two key parameters, i.e., window size τ for model augmentation module and embedding dimension d. We investigate a specific parameter by changing its value and fixing the others. The performances of TSR+ (in terms of $Rec@10$ and $Pre@10$ on AMiner-T test data with $T = 2013$) on various settings of τ and d are reported in Figure 2. According to this figure:

- With the increment of τ, $Rec@10$ and $Pre@10$ increase at first since a larger window means more useful indirect correlations among authors and papers. But when τ goes beyond a certain value, the results decrease with the further increment of τ due to the possible involvement of uncorrelated noise. The best τ is around 5.
- Similar to τ, an appropriate value should be set for d such that the best representations of authors and papers are learned. The optimal value of d is around 128.

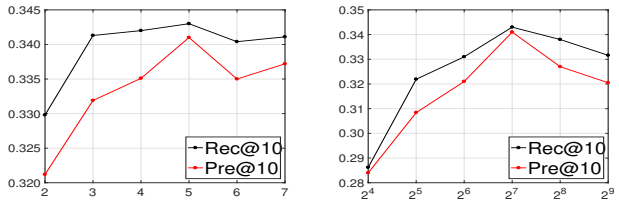
Besides d and τ, we also investigate the impacts of other hyper-parameters such as regularization parameter λ, and reveal the similar point. Therefore the certain settings of the hyper-parameters lead to the best performance of TSR+.

Performances on Different Author Groups.

As presented in introduction, an author-paper inference model should be effective for researchers with different knowledgeable levels. In order to validate the effectiveness of TSR+ on different author groups (from "cold-start" to active), we classify all authors into 6 groups (i.e., 1∼3, 4∼6, 7∼9, 10∼12, 13∼15 and >15) based on the number of observed correlated papers they have in training data, then evaluate the performance in each group. The performances (in

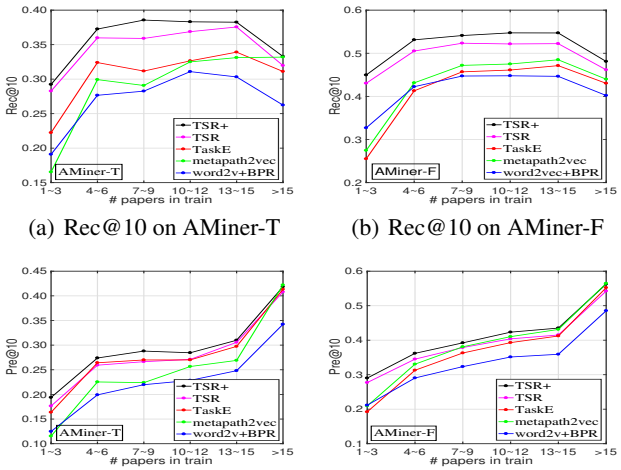
Author	Rank	Paper	Year	Venue	Relation
Jure Leskovec (precision: 4/5)	1	Overlapping community detection at scale: a nonnegative matrix factorization approach.	2013	WSDM	□
	2	Predicting emerging social conventions in online social networks.	2012	CIKM	△
	3	No country for old members: user lifecycle and linguistic change in online communities.	2013	WWW	□
	4	Fast mining and forecasting of complex time-stamped events.	2012	KDD	×
	5	Earthquake shakes Twitter users: real-time event detection by social sensors.	2010	WWW	△
Yuxiao Dong (precision: 4/5)	1	Social influence analysis in large-scale networks.	2009	KDD	△
	2	Confluence: conformity influence in large social networks.	2013	KDD	△
	3	Mining topic-level influence in heterogeneous networks.	2010	CIKM	△
	4	Yes, there is a correlation: from social networks to personal behavior on the web.	2008	WWW	×
	5	Inferring user demographics and social strategies in mobile social networks.	2014	KDD	□

Table 4: Two case studies of TSR+'s result. Notations □, △ and × represent writing, citing and uncorrelated relations between author and paper, respectively.



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

Figure 2: The impacts of window size and embedding dimension on TSR+. TSR+ performs best when τ is around 5 and d is around 128.



(a) Rec@10 on AMiner-T (b) Rec@10 on AMiner-F

(c) Pre@10 on AMiner-T (d) Pre@10 on AMiner-F

Figure 3: Performances of our models and three selected baselines in different author groups. TSR+ consistently reaches the best performance in all groups.

terms of $Rec@10$ and $Pre@10$ with $T = 2013$) of our models and three selected baselines (i.e., TaskE, metapath2vec and word2vec+BPR) are shown by Figure 3. We can find that TSR+ consistently outperforms TSR and the baselines in all groups. The results indicate that TSR+ achieves robust performance across different author groups and is convincible for different purposes.

Case Study.

We present two case studies to show the details of TSR+'s result. Table 4 lists the top 5 ranked papers for two data mining researchers of different groups in previous discussion, i.e., Jure Leskovec (in the last group) and Yuxiao Dong (in the second group) in AMiner-T data (with $T = 2013$). It is easy to know that all return papers for both researchers belong to data mining or information system area. In addition, both of them will interact (cite or write) with 4 papers (among 5) after T, which shows that TSR+ performs well for researchers with different knowledgeable levels. As for papers rank at 4 for both cases, their topics or methods are quite similar to those of the researchers' correlated papers, which results wrong predictions of TSR+. Moreover, there are some top ranked papers published after T (publication year $\geq$ 2013, highlighted in red), indicating that TSR+ can correctly return not only previous works but also new papers.

5 Related Work

In the past decade, some works have devoted to academic data mining problems, such as heterogeneous bibliographic network analysis [Sun *et al.*, 2012; Huang *et al.*, 2016], citation recommendation [He *et al.*, 2010; Ren *et al.*, 2014] or collaborator recommendation [Tang *et al.*, 2012; Li *et al.*, 2014]. In this paper, we study the problem of author-paper correlation inference in big scholarly data.

The network embedding has attracted lots of attention in recent years. Most of embedding models [Perozzi *et al.*, 2014; Grover and Leskovec, 2016; Dong *et al.*, 2017] preserve the proximities among nodes by learning vectorized representations. Some of the extended studies have been applied to various applications in big scholarly data, like correlation inference [Huang *et al.*, 2016; Chen and Sun, 2017] or node classification [Gui *et al.*, 2016; Dong *et al.*, 2017]. Unlike task-independent attribute or content unawareness of these models, our model TSR+ is task-specific and incorporates both semantic content and heterogeneous relations.

Besides academic data mining and network embedding, this paper is also related to pairwise ranking optimization [Rendle *et al.*, 2009] in recommender systems, gated recurrent neural network [Cho *et al.*, 2014] in deep learning, word and document embedding [Mikolov *et al.*, 2013; Le and Mikolov, 2014] in natural language processing, etc.

6 Conclusion and Future Work

In this paper, we propose the author-paper correlation inference problem in big scholarly data, and design a model TSR+ to solve it. The model performs joint optimization of

GRU-based content encoding and task-guided ranking, and is further augmented by a heterogeneous relations integrative learning module. The extensive experiments on the well known AMiner data demonstrate that TSR+ achieves significant better performance, comparing to a number of baselines. Some potential future work includes: (1) TSR+ can be extended by using more context information like publication venue of paper; (2) the dynamics of authors' embeddings should be considered for the task since authors keep publishing new papers, and citing more papers.

Acknowledgments

We would like to thank Yuxiao Dong for suggestions. This work is supported by the Army Research Laboratory under Cooperative Agreement Number W911NF-09-2-0053 and the National Science Foundation (NSF) grant IIS-1447795. This work is partially supported by King Abdullah University of Science and Technology (KAUST).

References

[Chen and Sun, 2017] Ting Chen and Yizhou Sun. Task-guided and path-augmented heterogeneous network embedding for author identification. In *WSDM*, pages 295–304, 2017.

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

[Dong *et al.*, 2015] Yuxiao Dong, Reid A Johnson, and Nitesh V Chawla. Will this paper increase your h-index?: Scientific impact prediction. In *WSDM*, pages 149–158, 2015.

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

[Efimov *et al.*, 2013] Dmitry Efimov, Lucas Silva, and Benjamin Solecki. Kdd cup 2013-author-paper identification challenge: second place team. In *KDD Cup Workshop*, 2013.

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

[Gui *et al.*, 2016] Huan Gui, Jialu Liu, Fangbo Tao, Meng Jiang, Brandon Norick, and Jiawei Han. Large-scale embedding learning in heterogeneous event data. In *ICDM*, pages 907–912, 2016.

[He *et al.*, 2010] Qi He, Jian Pei, Daniel Kifer, Prasenjit Mitra, and Lee Giles. Context-aware citation recommendation. In *WWW*, pages 421–430, 2010.

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

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

[Le and Mikolov, 2014] Quoc Le and Tomas Mikolov. Distributed representations of sentences and documents. In *ICML*, pages 1188–1196, 2014.

[Li *et al.*, 2014] Jing Li, Feng Xia, Wei Wang, Zhen Chen, Nana Yaw Asabere, and Huizhen Jiang. Acrec: a co-authorship based random walk model for academic collaboration recommendation. In *WWW*, pages 1209–1214, 2014.

[Li *et al.*, 2015] Chun-Liang Li, Yu-Chuan Su, Ting-Wei Lin, Cheng-Hao Tsai, Wei-Cheng Chang, Kuan-Hao Huang, Tzu-Ming Kuo, Shan-Wei Lin, Young-San Lin, Yu-Chen Lu, et al. Combination of feature engineering and ranking models for paper-author identification in kdd cup 2013. *JMLR*, 16(1):2921–2947, 2015.

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

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

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

[Rendle *et al.*, 2009] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. BPR: Bayesian personalized ranking from implicit feedback. In *UAI*, pages 452–461, 2009.

[Shen *et al.*, 2014] Hua-Wei Shen, Dashun Wang, Chaoming Song, and Albert-László Barabási. Modeling and predicting popularity dynamics via reinforced poisson processes. In *AAAI*, pages 291–297, 2014.

[Sun *et al.*, 2012] Yizhou Sun, Brandon Norick, Jaiwei Han, Xifeng Yan, Philip Yu, and Xiao Yu. PathSelClus: Integrating meta-path selection with user-guided object clustering in heterogeneous information networks. In *KDD*, pages 1348–1356, 2012.

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

[Tang *et al.*, 2012] Jie Tang, Sen Wu, Jimeng Sun, and Hang Su. Cross-domain collaboration recommendation. In *KDD*, pages 1285–1293, 2012.

[Wang *et al.*, 2013] Dashun Wang, Chaoming Song, and Albert-László Barabási. Quantifying long-term scientific impact. *Science*, 342(6154):127–132, 2013.

[Zhang *et al.*, 2017] Chuxu Zhang, Lu Yu, Xiangliang Zhang, and Nitesh Chawla. ImWalkMF: Joint matrix factorization and implicit walk integrative learning for recommendation. In *IEEE Big Data*, pages 857–866, 2017.

[Zhang *et al.*, 2018] Chuxu Zhang, Chao Huang, Lu Yu, Xiangliang Zhang, and Nitesh V Chawla. Camel: Content-aware and meta-path augmented metric learning for author identification. In *WWW*, pages 709–718, 2018.