Camel: Content-Aware and Meta-path Augmented Metric Learning for Author Identification

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

Chao Huang
University of Notre Dame
Notre Dame, IN 46556, USA
chuang7@nd.edu

Lu Yu
King Abdullah University of Science and Technology
Thuwal, 23955, SA
lu.yu@kaust.edu.sa

Xiangliang Zhang
King Abdullah University of Science and Technology
Thuwal, 23955, SA
xiangliang.zhang@kaust.edu.sa

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

ABSTRACT

In this paper, we study the problem of author identification in big scholarly data, which is to effectively rank potential authors for each anonymous paper by using historical data. Most of the existing de-anonymization approaches predict relevance score of paper-author pair via feature engineering, which is not only time and storage consuming, but also introduces irrelevant and redundant features or miss important attributes. Representation learning can automate the feature generation process by learning node embeddings in academic network to infer the correlation of paper-author pair. However, the learned embeddings are often for general purpose (independent of the specific task), or based on network structure only (without considering the node content). To address these issues and make a further progress in solving the author identification problem, we propose Camel, a content-aware and meta-path augmented metric learning model. Specifically, first, the directly correlated paper-author pairs are modeled based on distance metric learning by introducing a push loss function. Next, the paper content embedding encoded by the gated recurrent neural network is integrated into the distance loss. Moreover, the historical bibliographic data of papers is utilized to construct an academic heterogeneous network, wherein a meta-path guided walk integrative learning module based on the task-dependent and content-aware Skipgram model is designed to formulate the correlations between each paper and its indirect author neighbors, and further augments the model. Extensive experiments demonstrate that Camel outperforms the state-of-the-art baselines. It achieves an average improvement of 6.3% over the best baseline method.

KEYWORDS

Author Identification; Heterogeneous Networks; Representation Learning; Metric Learning; Deep Learning

WWW 2018, April 23–27, 2018, Lyon, France

ACM ISBN 978-1-4503-5639-8/18/04.
https://doi.org/10.1145/3178876.3186152

ACM Reference Format:
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 2018: The 2018 Web Conference, April 23–27, 2018, Lyon, France.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3178876.3186152

1 INTRODUCTION

With the fast growth of academic data collections by various online services such as Google Scholar, Microsoft Academic and AMiner, big scholarly data mining problems have gained a lot of attention in the past decade. Typical examples include scientific impact modeling and prediction [4, 23, 24, 31], academic heterogeneous network analysis [12, 25, 26], personalized recommendation [8, 17, 21].

In this paper, we consider the problem of author identification for each anonymous paper in big scholarly data, which was proposed and briefly investigated in [9], and has been further studied in recent works [1, 19]. Specifically, as illustration in Figure 1, given an anonymous paper with content/attributes (e.g., abstract), we would like to design a machine learning model to predict the potential authors of this paper by using the historical data. Solutions of the problem bring broad implications to the academic community. Let's take the double-blind review process in many conferences (e.g., WWW 2018) as an example. Although the authors of the paper under double-blind review process are invisible to the reviewers, they sometimes can still be unveiled by the paper content. Thus our work can serve as a study for helping existing review systems to answer the question that whether or not double-blind review process is really effective [1, 29]. In addition, the proposed model can infer for each query paper the potential authors, which can be useful for general information retrieval or recommender system design such as reviewer recommendation [16, 30].

To solve the author identification problem, supervised leaning models have been applied to predict the correlation between paper and author, such as the ones used in the top solutions [5, 15, 35] of 2013 KDD Cup author-paper pair identification challenge and the multimodal approach in [19]. However, these methods heavily rely on time consuming and storage intensive feature engineering, which may extract irrelevant and redundant features or miss important features. In the past few years, a number of network

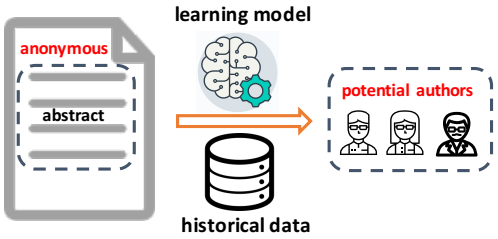


Figure 1: Illustration of author identification problem.

embedding models [3, 6, 20, 27] have been proposed to automatically learn node representations that can be further utilized in various academic mining tasks such as paper-author correlation inference and similar authors/venues search. Although the proximity among nodes is preserved by dense vectors, these methods learn general purpose embeddings that are independent of task and not suitable for the specific problem. To address this drawback, Chen et al. proposed HetNetE [1], a task-guided heterogeneous network embedding model, which outperforms the existing baselines. However, HetNetE mainly uses network structure and ignores semantic content of paper. In addition, it searches correlations among all kind of nodes (such as paper, reference and venue) for optimization.

To address above issues and make a further progress in solving the author identification problem, we develop Camel, a **c**ontent-**a**ware and **me**ta-path augmented **me**tric **l**earning model. First, we model the historical data of direct paper-author relations via distance metric learning according to the specific task. Next, we introduce the gated recurrent units to encode paper content and integrate the semantic embedding into the metric learning model. Moreover, we use the historical bibliographic data of papers to construct academic heterogeneous network, wherein we further design a learning module to augment the model. The augmented module employs meta-path walks to capture correlations between each paper and its indirect author neighbors and further formulate them via a task-dependent and content-aware Skipgram model. Finally, a sampling based mini-batch gradient descent algorithm is designed to infer model parameters.

To summarize, the main contributions of our work are:

- We develop a model, i.e., Camel, to solve the author identification problem. Camel performs joint optimization of content encoder based distance metric learning and Skipgram model based meta-path walk integrative learning.
- We design the corresponding optimization strategy and training algorithm for Camel. The learned model only needs partial content (i.e., abstract) of the target paper as the input and effectively predict the authors for each new paper in big scholarly data.
- We conduct extensive evaluations and analytical experiments to show the effectiveness of Camel on the well known AMiner dataset. The results demonstrate that our method outperforms a number of baseline methods and achieves a 6.3% average improvement over the best baseline.

2 PROBLEM DEFINITION

In this section, we first introduce the concepts of heterogeneous networks and meta-path, then formally define the author identification problem in big scholarly data.

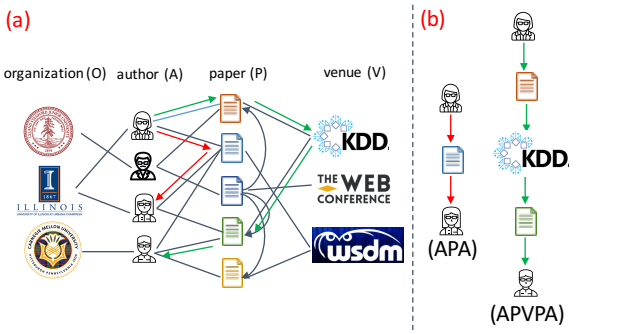


Figure 2: Illustrations of (a) academic heterogeneous network and (b) meta-path schemes.

Definition 2.1. (**Heterogeneous Networks**) A heterogeneous network (HetNet) [26] 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 objects and relation types. Each node $v \in V$ and each link $e \in E$ are associated with a node type mapping function $\psi_v : V \to O_V$ and a link type mapping function $\psi_e : E \to R_E$.

The academic network in big scholarly data can be seen as a HetNet, as shown in Figure 2(a). The set of node types O_V in the network includes *organization* (O), *author* (A), *paper* (P) and *venue* (V), and the set of link types R_E includes *author-write-paper*, *author-affiliate-organization*, *paper-cite-paper*, *paper-publish-venue*.

Definition 2.2. (**Meta-path**) A meta-path [25] in $G = (V, E, O_V, R_E)$ is defined in the form of $o_1 \xrightarrow{r_1} o_2 \xrightarrow{r_2} \cdots \xrightarrow{r_{m-1}} o_m$, 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-1}.

For example, in Figure 2(b), a meta-path "APA" extracted from HetNet denotes the coauthor relationship on a paper between two authors, and "APVPA" represents two authors publish papers in the same venue.

Definition 2.3. (**Author Identification Problem**) Given a set of previous papers $I_{<T}$ published before timestamp T, accompanying with bibliographic information (i.e., authors, abstract content, references and venue), the task is to rank all potential authors $u \in U$ (U: set of all authors) for each new anonymous paper $v \in I_{\geq T}$ ($I_{\geq T}$: set of papers published in or after T), such that its top ranked authors are true authors of v.

3 PROPOSED MODEL

We present the content-aware metric learning model for solving the problem and use historical bibliographic data to construct HetNet for modeling multiple indirect paper-author relations captured by meta-path walks, which benefits and augments the model.

3.1 Metric Learning with Gated Recurrent Neural Network

We denote each paper $v \in I_{<T}$ as embedding $\mathbf{E}_v \in \mathbb{R}^d$ (d: dimension of embedding) via a content encoder f: $\mathbf{E}_v = f(\mathbf{p}_v)$, where $\mathbf{p}_v$ denotes the word sequence of the paper abstract. Besides, feature vector $\mathbf{q}_u \in \mathbb{R}^d$ is used to represent each author $u \in U$. Considering distance metric [32] satisfies better triangle inequality and

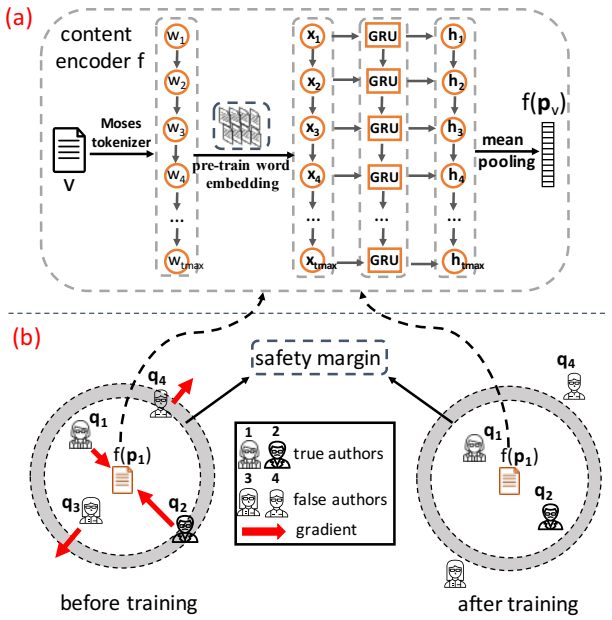


Figure 3: Illustrations of (a) paper content encoder based on gated recurrent neural network and (b) metric learning process for author identification.

transition property than inner-product, as demonstrated by CML [11], we introduce the following push loss function to formulate triple relations (v, u, u'):

$$\mathcal{L}_{Metric} = \sum_{v \in I_{<T}} \sum_{u \in l_v} \sum_{u' \notin l_v} \left[\xi + dist(v,u)^2 - dist(v,u')^2 \right]_+ \quad (1)$$

where l_v denotes the set of true authors of paper v, $\{x\}_+ = max(x, 0)$ is a standard hinge loss and ξ is a safety margin size. The distance metric $dist(v,u)$ between paper v and author u is defined as euclidean distance of feature representation:

$$dist(v,u) = ||\mathbf{E}_v - \mathbf{q}_u|| = ||f(\mathbf{p}_v) - \mathbf{q}_u|| \quad (2)$$

Hence, minimizing $\mathcal{L}_{Metric}$ obeys paper v's relative distances to different (true/false) authors.

To encode paper abstract content to fixed length embeddings $\mathbf{E} \in \mathbb{R}^{|I| \times d}$ (I: set of all papers), we introduce the gated recurrent units (GRU), a specific type of recurrent neural network, which has been widely adopted for many applications such as machine translation [2]. Figure 3(a) gives the illustration of paper content encoder. To be more specific, a paper 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 [18], where t_{max} is the maximum length of paper abstract. 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:

$$\begin{aligned} \mathbf{z}_t &= \sigma(\mathbf{A}_z \mathbf{x}_t + \mathbf{B}_z \mathbf{h}_{t-1}) \\ \mathbf{r}_t &= \sigma(\mathbf{A}_r \mathbf{x}_t + \mathbf{B}_r \mathbf{h}_{t-1}) \\ \hat{\mathbf{h}}_t &= \tanh[\mathbf{A}_h \mathbf{x}_t + \mathbf{B}_h (\mathbf{r}_t \circ \mathbf{h}_{t-1})] \\ \mathbf{h}_t &= \mathbf{z}_t \circ \mathbf{h}_{t-1} + (1 - \mathbf{z}_t) \circ \hat{\mathbf{h}}_t \end{aligned} \quad (3)$$

where σ is the sigmoid function, $\mathbf{A}$ and $\mathbf{B}$ are parameter matrices of GRU network, operator $\circ$ denotes element-wise multiplication, $\mathbf{z}_t$ and $\mathbf{r}_t$ are update gate vector and reset gate vector, respectively. The GRU network encodes word embeddings to deep semantic embeddings $\mathbf{h} \in \mathbb{R}^{t_{max} \times d}$, which is concatenated by a mean pooling layer to obtain the general semantic embedding of each paper. All of these steps construct the paper content encoder f. We have also explored other encoding architectures such as LSTM and achieved the similar result, as the discussion in Section 4.3.2.

According to $\mathcal{L}_{Metric}$, the target neighbors of each paper are its true authors and the model incorporates semantic information of papers via GRU based content encoder. To infer model's parameters, we can minimize $\mathcal{L}_{Metric}$ via gradient descent approach. For true authors of a given paper, gradients of loss function pull them inward to create a smaller radius. As for the false authors, gradients push them outward until they are out of the perimeter by a safety margin. Illustration of such process is shown in Figure 3(b). Thereafter the learned encoder f for inferring the semantic embedding of each future paper $v \in I_{\geq T}$ and the optimized author latent features can be utilized to rank all potential authors for v according to the relevance score (e.g., the inner product of embedding) between paper and author. The learned model only needs abstract content of the target paper as the input for prediction since f and author latent features are optimized by using historical training data.

3.2 Model Augmentation via Meta-path Walk Integrative Learning

In Section 3.1, $\mathcal{L}_{Metric}$ essentially models direct triple relations, i.e., (v, u, u') - (paper-true author-false author), for each paper $v \in I_{<T}$. However, there are multiple indirect relations between paper and author, which can be inferred from the HetNet of previous papers' bibliographic data and beneficial to the model. Hence, we aim to further augment the content encoder based metric learning by enforcing the smoothness of representation among indirectly correlated paper-author neighbors on the academic HetNet.

3.2.1 Meta-path Walks. Although we can naturally take random walk on HetNet to capture indirect paper-author relations, as did in Deepwalk [20] and node2vec [6], such random walks are biased on highly visible types of nodes and concentrated nodes, as demonstrated by metapath2vec [3]. Thus we apply meta-path walks to capture indirect correlations between paper and author. Specifically, given a meta-path $\mathcal{P} \equiv \left\{ o_1 \xrightarrow{r_1} o_2 \xrightarrow{r_2} \cdots o_i \xrightarrow{r_i} o_{i+1} \cdots \xrightarrow{r_{m-1}} o_m \right\}$ on the academic HetNet $G = (V, E, O_V, R_E)$, the transition probability of walk at step t is defined as:

$$p(v^{t+1} | v_i^t, \mathcal{P}) = \begin{cases} \frac{1}{|N_{i+1}(v_i^t)|} & (v^{t+1}, v_i^t) \in E,\ \psi(v^{t+1}) = i+1 \\ 0 & (v^{t+1}, v_i^t) \in E,\ \psi(v^{t+1}) \neq i+1 \\ 0 & (v^{t+1}, v_i^t) \notin E \end{cases} \quad (4)$$

where $v_i^t \in o_i$ and $N_{i+1}(v_i^t)$ denotes the set of the o_{i+1} type of neighborhood of node v_i^t, which guarantees that $v^{t+1} \in o_{i+1}$ and the flow of walk is conditioned on $\mathcal{P}$. In addition, we use symmetric meta-path whose first node type o_1 is the same as the last one o_m. Each random walk guided by $\mathcal{P}$ recursively samples nodes sequence until it meets the fixed length, leading to its ability in capturing

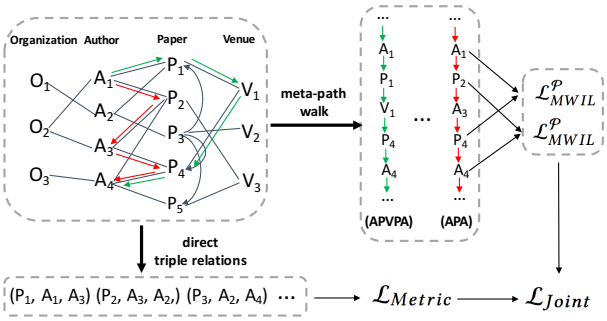


Figure 4: Illustration of the joint representation learning model with metric learning for formulating direct paper-author relations and meta-path walk integrative learning for modeling indirect paper-author correlations.

both direct correlations and indirectly transitive relations between paper and author within the walk of setting $\mathcal{P}$, as illustrated by Figure 4. In this figure, we take walk $\boldsymbol{w}_o \equiv \{\cdots \rightarrow A_1 \rightarrow P_2 \rightarrow A_3 \rightarrow P_4 \rightarrow A_4 \rightarrow \cdots\}$ guided by $\mathcal{P} \equiv \{A \overset{write}{\rightarrow} P \overset{write^{-1}}{\rightarrow} A\}$ as an example. Besides the direct paper-author connections, e.g., A_1 writes P_2 or A_4 writes P_4, $\boldsymbol{w}_o$ also captures indirect relations. For example, A_1 may pay attention to P_4 since s/he collaborates with A_3 on P_2. Therefore, multiple useful indirect relations between paper and author will be inferred if we generate plenty of walks guided by different meta-path schemes and collect the surrounding author context of each paper node within each walk.

3.2.2 Smoothness Constraint as Task-dependent and Content-aware Skipgram Model. To formulate indirect paper-author relations within each walk and force the corresponding smoothness of representation, we design a **m**eta-path guided **w**alk **i**ntegrative **l**earning module (MWIL) based on the Skipgram model [18], which has been widely adopted in recent works [3, 6, 20, 34] for representation learning on networks. Specifically, given a set of collected walks $W_{\mathcal{P}}$ under the guidance of meta-path $\mathcal{P}$, the loss for predicting indirectly correlated author u of paper v is defined as:

$$\mathcal{L}^{\mathcal{P}}_{MWIL} = -\sum_{\boldsymbol{w}\in W_{\mathcal{P}}}\sum_{v\in\boldsymbol{w}}\sum_{\substack{u\in\boldsymbol{w}[I_v-\tau:I_v+\tau]\\ u\notin l_v}} \log p(u|v,\mathcal{P}) \tag{5}$$

where τ is the window size of surrounding context and I_v indicates the position of v in walk $\boldsymbol{w}$. The likelihood probability $p(u|v,\mathcal{P})$ is defined as content-aware Softmax function:

$$p(u|v,\mathcal{P}) = \frac{exp\Big[f(\mathbf{p}_v)\mathbf{q}_u\Big]}{\sum_{u'\in C_{\mathcal{P}}} exp\Big[f(\mathbf{p}_v)\mathbf{q}_{u'}\Big]} \tag{6}$$

where f is the content encoder defined in Section 3.1, $C_{\mathcal{P}}$ denotes the set of all authors in corpus $W_{\mathcal{P}}$. To train the Skipgram model, we apply the popular negative sampling approach [18] to approximate the intractable normalization:

$$\log p(u|v,\mathcal{P}) \approx \log\sigma\Big[f(\mathbf{p}_v)\mathbf{q}_u\Big] + \sum_{i=1}^{k}\mathbb{E}_{u'\sim P_{\mathcal{P}}(u')}\Big\{\log\sigma\Big[-f(\mathbf{p}_v)\mathbf{q}_{u'}\Big]\Big\} \tag{7}$$

where σ is the sigmoid function, u' is the negative author node sampled from a pre-defined noise distribution $P_{\mathcal{P}}(u')$ [18] in $C_{\mathcal{P}}$, k is the number of negative samples. In our case, k makes little impact on the performance of propose model. Thus we choose $k = 1$ and $\log p(u|v,\mathcal{P})$ is degenerated to the cross entropy loss of classifying pair (u,u') for v:

$$-\log p(u|v,\mathcal{P}) = -\log\sigma\Big[f(\mathbf{p}_v)\mathbf{q}_u\Big] - \log\sigma\Big[-f(\mathbf{p}_v)\mathbf{q}_{u'}\Big] \tag{8}$$

That is, for each positive author u of paper v within walk $\boldsymbol{w}$, we sample a negative author u' from $C_{\mathcal{P}}$ according to $P_{\mathcal{P}}(u')$.

Comparing to the objective function of metapath2vec [3], $\mathcal{L}^{\mathcal{P}}_{MWIL}$ has three main differences:

- It forces task-dependent smoothness constraint between paper and its indirectly correlated author neighbors but not among all kind of neighbor pairs for general purpose.
- The likelihood probability for predicting surrounding context is degenerated to cross entropy loss of classifying the positive/negative authors for each paper.
- More importantly, the paper representations are encoded by GRU content encoder f for integrating paper semantic information into model.

3.3 Joint Model Inference

The objective function of joint model is defined as the combination of $\mathcal{L}_{Metric}$ and $\mathcal{L}^{\mathcal{P}}_{MWIL}$:

$$\mathcal{L}_{Joint} = \mathcal{L}_{Metric} + \gamma\sum_{\mathcal{P}\in S(\mathcal{P})}\mathcal{L}^{\mathcal{P}}_{MWIL} + \lambda\mathcal{L}_{reg} \tag{9}$$

where $S(\mathcal{P})$ denotes all meta-path schemes, $\mathcal{L}_{reg}$ is the regularization term for avoiding over-fitting, parameter λ controls penalty of regularization, γ is a trade-off factor between $\mathcal{L}_{Metric}$ and $\mathcal{L}^{\mathcal{P}}_{MWIL}$. We denote all model parameters including the GRU network coefficients of paper content encoder and the author latent features as $\boldsymbol{\Theta}$. Let T_{Metric} and $T^{\mathcal{P}}_{MWIL}$ be the sets of (v,u,u') triples in $\mathcal{L}_{Metric}$ and (v,u,u') triples in $\mathcal{L}^{\mathcal{P}}_{MWIL}$, respectively. Thereafter we can rewrite $\mathcal{L}_{Joint}$ as:

$$\begin{aligned}\mathcal{L}_{Joint} = &\sum_{(v,u,u')\in T_{Metric}}\Big[\xi + ||f(\mathbf{p}_v) - \mathbf{q}_u||^2 - ||f(\mathbf{p}_v) - \mathbf{q}_{u'}||^2\Big]_+\\ &+\gamma\sum_{\mathcal{P}\in S(\mathcal{P})}\sum_{(v,u,u')\in T^{\mathcal{P}}_{MWIL}} -\Big\{\log\sigma\Big[f(\mathbf{p}_v)\mathbf{q}_u\Big] + \log\sigma\Big[-f(\mathbf{p}_v)\mathbf{q}_{u'}\Big]\Big\}\\ &+\lambda\,\|\boldsymbol{\Theta}\|^2\end{aligned} \tag{10}$$

To minimize $\mathcal{L}_{Joint}$, we design a sampling based mini-batch Adam optimizer [13]. The pseudocode of learning algorithm is summarized in Algorithm 1. The proposed model performs joint optimization of content encoder based metric learning and meta-path walk integrative learning thus we name it **c**ontent-**a**ware and **me**ta-path augmented **me**tric **l**earning (Camel).

4 EXPERIMENTS

In this section, we conduct extensive evaluations and analytical experiments to compare Camel with various baselines. Case studies are also provided to show performance differences of different methods.

Algorithm 1: *Learning Framework of Camel*

input : T_{Metric} in training data, $T^{\mathcal{P}}_{MWIL}$ extracted by meta-path walks on the academic HetNet
output: author latent features $\boldsymbol{q}$, GRU encoder matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ (for generating paper embeddings $f(\boldsymbol{p})$)

```
while not converged do
    sample a batch of (v, u, u') in T_Metric;
    for P ∈ S(P) do
        sample a batch of (v, u, u') in T^P_MWIL;
    end
    accumulate the loss by Equation (10);
    update the parameters by Adam;
end
```

Table 1: Statistics of datasets used in this paper.

Statistics	AMiner-Top	AMiner-Full
# authors	28,646	571,563
# papers	21,044	483,319
# venues	18	492
# citations	245,420	3,154,421
ave. # authors per paper	3.294	3.087

4.1 Experimental Design

4.1.1 Dataset. AMiner [28] is a well known platform for academic search and mining, which contains millions of author and paper information from major computer science venues for more than 50 years. We utilize the AMiner dataset[1] of 10 years from 2006 to 2015, and remove the papers published in venues (e.g., workshop) with limited publications and the instances without semantic content (i.e., abstract). In addition, considering most of researchers pay attention to papers published in top venues and each research area has its own community, we extract one more subset data of six domains according to Google Scholar Metrics, namely Artificial Intelligence (AI), Data Mining (DM), Databases (DB), Information System (IS), Computer Vision (CV) and Computational Linguistics (CL). For each domain, we choose three top venues[2] that are considered to have influential papers. The main statistics of two datasets (AMiner-Top and AMiner-Full) are summarized in Table 1.

4.1.2 Baseline Methods. We consider nine baseline methods that span four types: (1) citation-based matching, (2) feature engineering based supervised learning, (3) pairwise ranking with content embedding and (4) heterogeneous network embedding.

- **Citation-based matching.** The approach was proposed in [9] and represents each paper and author by citation-based vector, and further matches potential authors for each query paper according to the vector similarity (VecS).
- **Feature engineering based supervised learning.** Such approaches have been utilized in top solutions [5, 15, 35] for 2013 KDD Cup challenge. It first extracts both author features and paper-author paired features, and then utilizes supervised learning algorithms to predict the correlation score of each paper-author pair. Similar to HetNetE [1], we extract 16 kinds of features (as reported in Table 2) based on AMiner data and select Bayes Regression (BayesR), Random Forest (RandF) and Neural Network (NeuN) as learning algorithms. In addition, an ensemble approach (MultiM) of three algorithms is introduced for comparison.
- **Pairwise ranking with content embedding.** Another possibility to consider content information is to first encode each paper content embedding via language modeling and then apply pairwise ranking [22] (BPR, which utilizes the inner product to measure paper-author correlation) to learn author latent features. We apply two popular models Word2V [18] and Par2V [14] to generate paper embeddings. In addition, the joint learning model (GRUBPR) of GRU [2] based content encoder and BPR is also introduced for comparison. As Word2Vec generates embedding of each word in content, we concatenate the output with a mean pooling layer to obtain general embedding of each paper. The learned feature representations of paper and author are further utilized to predict the authors of each paper.
- **Heterogeneous network embedding.** We also compare Camel with a recent model HetNetE in [1], which optimizes feature representations of author and paper via task-guided heterogeneous network embedding, and further applies them to identify authors of each paper.

Table 2: Selected features of supervised learning baselines.

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

4.1.3 Evaluation Metrics. As illustrated in problem definition, papers published before a given timestamp T are treated as training data and papers published in or after T (denoted as set $I_{\geq T}$) are left for evaluation. We use four popular metrics, i.e., Recall@k, Precision@k, F1 score and AUC, to evaluate the performance of each method.

- **Recall@k.** It shows the ratio of true authors being retrieved in the top-k return list, which can be computed according to:

$$Rec@k = \frac{1}{|I_{\geq T}|} \sum_{v \in I_{\geq T}} \frac{|\hat{l}_v \bigcap l_v|}{|l_v|} \tag{11}$$

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

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

where l_v and $\hat{l}_v$ denote the sets of true authors of paper v and top-k ranked authors by a specific method, respectively.

- **Precision@k.** It reflects the accuracy of top-k ranked authors by a specific method and is defined as:

$$Pre@k = \frac{1}{|I_{\geq T}|} \sum_{v \in I_{\geq T}} \frac{|\hat{l}_v \bigcap l_v|}{k} \tag{12}$$

- **F1 score.** It balances the trade-off between precision and recall, and is defined as the harmonic average of precision and recall:

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{13}$$

- **AUC.** It measures the accuracy of pairwise orders between correlated and uncorrelated papers of each author, which is formulated as:

$$AUC = \frac{1}{|I_{\geq T}|} \sum_{v \in I_{\geq T}} \frac{1}{|E(v)|} \sum_{(u,u') \in E(v)} \delta(s_{vu} > s_{vu'}) \tag{14}$$

where $E(v) \equiv \{(u,u') | u \in l_v, u' \notin l_v\}$.

For all evaluations, we set k = 10. A larger Recall@k, Precision@k, F1 or AUC value means a better performance.

4.1.4 Experimental Settings. All information utilized for model training such as triple samples in Camel 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, there are three key settings of experiments:

- **Parameters.** The embedding dimension d is set to 128 and the regularization parameter λ equals 0.001. We fix hinge loss margin $\xi = 0.1$ for metric learning and window size $w = 6$ for meta-path walk augmentation. In addition, the trade-off factor γ of the joint model equals 0.1.
- **Meta-path selections.** We empirically investigate the performance of our model by greedily selecting and combining different meta-path walks and find that "APA", "APPA" and "APVPA" are the most effective meta-path schemes. Notice that, "APA" denotes collaboration relationship, "APPA" represents citation link and "APVPA" indicates correlation in the same publication venue. The set of node sequences capture multiple correlations between paper and author under these meta-path settings.
- **Evaluation candidates.** It is time consuming and memory intensive to extract and store features of all paper-author pairs (which amounts to over 2.7×10^{11} pairs in AMiner-Full). The supervised learning algorithms cannot scale up to such large amount of data. Hence, we adopt the setting in HetNetE [1] that randomly samples a set of negative authors and combines it with the set of true authors to form a candidate set of total 100 authors for each paper. The reported results are averaged over 10 experiments of such setting. For completeness, we also conduct evaluation of different representation learning models on the whole authors set.

4.2 Performance 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 of table reports the average

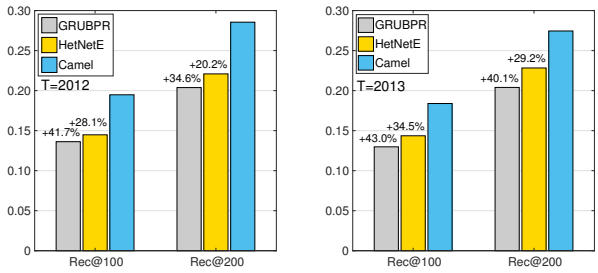


Figure 5: Result comparisons on the whole authors set. Camel significantly outperforms the others.

improvements (%) of Camel over different baselines. The main takeaways from this table are summarized as follows:

- The pairwise ranking models with content embedding have better average performances than the supervised learning baselines with one algorithm (i.e., BayesR, RandF and NeuN), which suggests that the feature representations generated by content embedding are better for capturing the complicated paper-author relations than the simple features extracted directly from data. In addition, VecS achieves poor performance since there are some missing citation information in AMiner data.
- HetNetE achieves better results than the supervised learning methods and the pairwise ranking models with content embedding, showing that the task-guided heterogeneous network embedding model generates task-specific feature representations and performs better than the other two for the author identification problem.
- Camel performs best in all experimental settings. The average improvements of Camel over different baselines range from 6.3% to 158.7%, demonstrating the effectiveness of our proposed model.

To make thorough evaluation, we also conduct comparison experiment of Camel and two selected baselines (GRUBPR and HetNetE) on the whole author candidate set of AMiner-Top dataset. The results (in terms of *Rec*@100 and *Rec*@200) are shown in Figure 5. It can be seen that Camel significantly outperforms the other two methods (with 39.8% and 28.0% average improvements, respectively), which further shows the effectiveness of Camel.

4.3 Analysis and Discussion

The analytical experiments in this section are based on AMiner-Top data, results in the other dataset reveal similar conclusion but are omitted due to page limit.

4.3.1 Parameters Sensitivity. The hyper-parameters play important roles in Camel, as they determine how the model will be trained. We conduct experiments to analyze the impacts of two key parameters, i.e., the window size τ of meta-path augmentation module and the embedding (latent feature) dimension d of author and paper. We investigate a specific parameter by changing its value and fixing the others. The performances of Camel (in terms of *Rec*@10 and *Pre*@10) on various settings of τ and d are shown in Figure 6. According to this figure:

Table 3: Performance comparisons of different methods. The last row shows the average improvements (%) of Camel over different baselines. HetNetE is the best baseline (indicated by star notation) and Camel has the best performances (highlighted in bold) in all cases.

Dataset	T	Metric	citation	feature engineering + supervised learning				content embedding + pairwise ranking			network embedding	our proposed
			VecS	BayesR	RandF	NeuN	MultiM	Word2V + BPR	Par2V + BPR	GRUBPR	HetNetE	Camel
AMiner-Top	2012	Rec@10	0.2577	0.5790	0.6450	0.6388	0.6541	0.6671	0.6423	0.6580	0.6938*	**0.7543**
		Pre@10	0.0636	0.1302	0.1462	0.1442	0.1496	0.1501	0.1454	0.1485	0.1532*	**0.1682**
		F1	0.1020	0.2126	0.2384	0.2353	0.2435	0.2449	0.2371	0.2423	0.2510*	**0.2750**
		AUC	0.6254	0.7835	0.7651	0.8132	0.8267	0.8947	0.8801	0.8815	0.8993*	**0.9257**
	2013	Rec@10	0.3120	0.6070	0.6371	0.6506	0.6612	0.6478	0.6462	0.6612	0.6782*	**0.7476**
		Pre@10	0.0853	0.1497	0.1576	0.1612	0.1628	0.1597	0.1593	0.1620	0.1653*	**0.1838**
		F1	0.1339	0.2402	0.2528	0.2583	0.2613	0.2563	0.2556	0.2602	0.2658*	**0.2951**
		AUC	0.6519	0.8091	0.8339	0.8456	0.8524	0.8872	0.8768	0.8849	0.8938*	**0.9205**
AMiner-Full	2012	Rec@10	0.2217	0.6994	0.6922	0.7217	0.7302	0.7478	0.7034	0.6842	0.8166*	**0.8446**
		Pre@10	0.0518	0.1636	0.1634	0.1694	0.1735	0.1812	0.1663	0.1658	0.1904*	**0.2021**
		F1	0.0839	0.2651	0.2644	0.2744	0.2804	0.2917	0.2690	0.2670	0.3088*	**0.3249**
		AUC	0.6106	0.8497	0.8129	0.8697	0.8754	0.9204	0.9014	0.8809	0.9346*	**0.9526**
	2013	Rec@10	0.2895	0.7176	0.6977	0.7226	0.7354	0.7612	0.7105	0.6874	0.8127*	**0.8392**
		Pre@10	0.0736	0.1756	0.1719	0.1771	0.1803	0.1891	0.1751	0.1702	0.2065*	**0.2197**
		F1	0.1173	0.2821	0.2758	0.2845	0.2895	0.3029	0.2810	0.2728	0.3293*	**0.3482**
		AUC	0.6445	0.8592	0.8126	0.8628	0.8749	0.9256	0.8940	0.8752	0.9313*	**0.9501**
	Ours v.s. baseline		158.65%	21.55%	19.15%	15.90%	13.83%	10.55%	15.12%	15.70%	6.28%	–

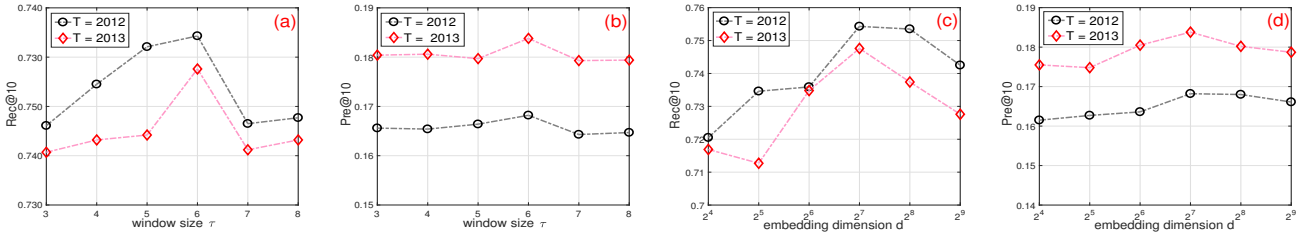


Figure 6: The impacts of window size τ and embedding dimension d on the performance of Camel. Camel achieves the best result when τ is around 6 and d is around 128.

- With the increment of τ, *Rec*@10 and *Pre*@10 increase at first since a larger window represents more useful indirect paper-author correlations. But when τ goes beyond a certain value, the performances decrease with the further increment of τ due to the possible involvement of uncorrelated noise. The best τ is around 6.
- Similar to τ, an appropriate value should be set for d such that the best representations of author and paper are learned. The optimal value of d is around 128.

Besides d and τ, we have also investigated the impacts of other hyper-parameters such as regularization parameter λ, and revealed the similar point. Therefore the certain settings of the hyper-parameters result in the best performance of Camel.

4.3.2 Performances of Variant Proposed Models. Camel is a joint representation learning model of content encoder based metric learning and meta-path walk integrative learning. Whether each learning component plays a role on the joint model? How meta-path schemes impact the model's performance? Whether the selection of recurrent unit or correlation measurement has influence on the model's performance? To answer these questions, we conduct experiments to evaluate the performances of variant proposed models w.r.t. different analytical categories:

- **Objective Function.** The joint objective function $\mathcal{L}_{Joint}$ contains two main components: $\mathcal{L}_{Metric}$ and $\mathcal{L}^{\mathcal{P}}_{MWIL}$. To shows the effectiveness of meta-path walk integrative learning module, we conduct evaluation for the model with only content encoder based metric learning, i.e., $\mathcal{L}_{Metric}$, and report its performance in Table 4 part (a). According to this result, we can find that Camel significantly outperforms $\mathcal{L}_{Metric}$, showing the large benefit of incorporating $\mathcal{L}^{\mathcal{P}}_{MWIL}$ into the joint model.
- **Random Walk.** We design a meta-path walk integrative learning module to augment the model. In order to show the larger benefit of meta-path walk over random walk, we design the joint learning model (Camel-RW) with random walk integrative learning module and compare it to Camel. As the result shown

Table 4: Performance comparisons of various proposed models w.r.t different analytical categories: (a) different components of the objective function; (b) different choices of random walk; (c) selection of meta-path scheme for random walk sampling and (d) selection of the recurrent unit for paper content encoder; (e) selection of paper-author correlation measurement.

Analytical Category	Variant Proposed Models	T = 2012				T = 2013			
		Rec@10	Pre@10	F1	AUC	Rec@10	Pre@10	F1	AUC
(a) Objective Function	$\mathcal{L}_{Metric}$	0.6856	0.1539	0.2514	0.8901	0.6695	0.1651	0.2649	0.8758
(b) Random Walk	**Camel-RW**	0.7364	0.1656	0.2704	0.9186	0.7242	0.1787	0.2867	0.9132
(c) Meta-path Selection	**Camel-APA**	0.7122	0.1601	0.2613	0.8939	0.6841	0.1701	0.2725	0.8810
	Camel-APPA	0.7371	0.1643	0.2687	0.9194	0.7226	0.1772	0.2847	0.9109
	Camel-APVPA	0.7315	0.1622	0.2655	0.9209	0.7075	0.1727	0.2777	0.9099
(d) Recurrent Unit Selection	**Camel-LSTM**	0.7538	0.1680	0.2749	0.9252	0.7472	0.1836	0.2948	0.9203
(e) Correlation Measurement	**GRUBPR**	0.6580	0.1485	0.2423	0.8815	0.6612	0.1620	0.2602	0.8849
Camel		**0.7543**	**0.1682**	**0.2750**	**0.9257**	**0.7476**	**0.1838**	**0.2951**	**0.9205**

in Table 4 part (b), Camel has higher identification accuracy than Camel-RW. Thus the meta-path walk is better than the random walk for capturing indirect paper-author correlations on academic HetNet.

- **Meta-path Selection.** In meta-path augmentation module, we select three kinds of meta-path schemes: "APA", "APPA" and "APVPA". To study the impacts of different meta-path schemes on the model's performance, we design three joint learning models, i.e., Camel-APA, Camel-APPA and Camel-APVPA, which are augmented by "APA", "APPA" and "APVPA" walk integrative learning modules, respectively. The performances of three models are reported in Table 4 part (c). We can observe that Camel-APPA achieves relative better performance than the other two, indicating that an author tends to have stronger correlation/preference to his/her references than co-author's papers or papers published in the same venue. In addition, all of three models have worse performance than Camel, demonstrating that the combination of different meta-path schemes leads to better performance.
- **Recurrent Unit Selection.** We select the GRU as the basic recurrent unit for paper content encoder of Camel. Besides GRU, there are various deep architectures constructed by different recurrent units for sequence modeling, such as long short term memory networks (LSTM). In order to test the influence of the recurrent unit selection on model's performance, we conduct comparison experiment between Camel and the model with LSTM (Camel-LSTM). According to the results shown in Table 4 part (d), Camel-LSTM and Camel have close performance. In other words, the selection of GRU or LSTM has little impact on the performance. We choose GRU since it has a more concise structure than LSTM for reducing training time.
- **Correlation Measurement.** As illustrated in Section 3.1, we use distance metric rather than inner-product to measure the correlation between paper and author. To show the rationality of such a choice, we compare the performances of the model with $\mathcal{L}_{Metric}$ and the baseline method GRUBPR since GRUBPR is a joint learning model of GRU based content encoder and pairwise ranking, which employs inner-product to measure paper-author correlation. According to the result shown in Table 4 part (a) and part (e), $\mathcal{L}_{Metric}$ has better performance than GRUBPR in most cases, demonstrating distance metric is better than inner-product in measuring paper-author correlation for the author identification problem.

To summarize, according to the above discussion: (1) the meta-path walk integrative learning module brings large benefits to improve the proposed model; (2) meta-path walk is better than random walk for capturing indirect paper-author correlations on academic HetNet; (3) "APPA" is the best meta-path scheme among the three, while the combination of different meta-path schemes leads to the best performance of model; (4) the choice of different recurrent unit has little influence on model's performance; and (5) the distance metric is better than inner-product for measuring direct paper-author correlation.

4.4 Case Studies

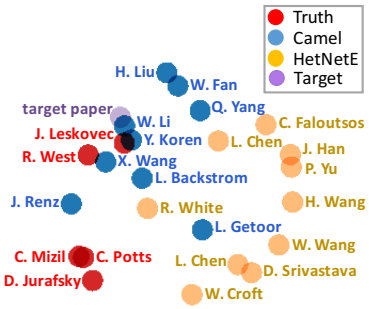
We present two case studies on AMiner-Top dataset to show the performance differences between Camel and two selected baselines, i.e., GRUBPR and HetNetE, which achieve relative better performances. Table 5 lists the top 10 ranked authors for two query papers published in WWW 2013 and WSDM 2013, respectively. For a better comparison, we also provide the embedding visualizations[3] of the target papers and top authors ranked by different methods. Comparing to the authors set of a given paper, the number of whole author set is much larger. Besides, there are many false authors whose feature representations are quite similar to the true authors of target paper. Thus many of the true authors may not be presented in the top list. However, according to Table 5, Camel achieves 2/5 and 2/4 w.r.t. *Rec*@100 in two cases, and predicts true authors more accurately than the other methods in top 10 lists, as shown by the authors (i.e., J. Leskovec and Q. Mei) highlighted in red color. In addition, the embeddings of top authors ranked by Camel cluster closer to the embeddings of the target paper and its true authors than those of HetNetE. We remove visualization result of GRUBPR due to its scattered behavior. Therefore, our model generates more accurate feature representations of paper and author, and achieves better performance than the other methods.

[3] http://projector.tensorflow.org/

Table 5: Top ranked authors for two query papers and the corresponding embedding visualizations.

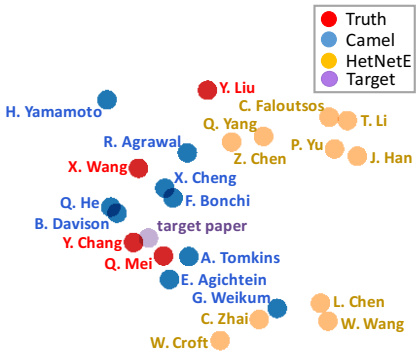
(1) (WWW 2013) No country for old members: User lifecycle and linguistic change in online communities.

Ground-truth	GRUBPR	HetNetE	Camel
C. Mizil	P. Yu	J. Han	H. Liu
R. West	L. Liu	P. Yu	Y. Koren
D. Jurafsky	Q. Yang	C. Faloutsos	W. Li
J. Leskovec	J. Yen	L. Chen	**J. Leskovec**
C. Potts	G. Weikum	D. Srivastava	J. Renz
	L. Chen	Z. Chen	L. Backstrom
	W. Hsu	H. Wang	Q. Yang
	N. Ramakrishnan	R. White	W. Fan
	K. Visweswariah	W. Croft	L. Getoor
	L. Li	W. Wang	X. Wang
Rec@100	1/5	1/5	2/5



(2) (WSDM 2013) Towards Twitter context summarization with user influence models.

Ground-truth	GRUBPR	HetNetE	Camel
Y. Chang	I. King	J. Han	Q. He
X. Wang	D. P	P. Yu	A. Tomkins
Q. Mei	S. Ma	Z. Chen	**Q. Mei**
Y. Liu	C. Eickhoff	Q. Yang	H. Yamamoto
	D. Ramage	C. Zhai	R. Agrawal
	Y. Shen	C. Faloutsos	F. Bonchi
	T. Sakaki	W. Croft	B. Davison
	J. Tian	W. Wang	E. Agichtein
	T. Hope	T. Li	G. Weikum
	C. Yeung	L. Chen	X. Cheng
Rec@100	0/4	2/4	2/4



5 RELATED WORK

In the past few years, some works have devoted to paper-author pair identification problem in big scholarly data, such as studies in [9, 19] and various solutions in [5, 15, 35] for 2013 KDD Cup author-paper identification challenge. Most of these works focused on feature engineering and utilized supervised learning algorithms to infer the correlation between paper and author. However, feature engineering is time consuming and storage intensive, and the extracted features can be irrelevant, insufficient or redundant.

The representation learning models for networks have attracted a lot of attention in recent years. Most of these models [6, 20, 27] preserve the proximities among nodes by learning vectorized representations. Some extended studies have been applied to various applications in big scholarly data such as node classification [3, 7]. Unlike task-independent attribute of these methods, Chen et al. proposed HetNetE [1], a task-guided heterogeneous network embedding model. Comparing to the existing baselines, HetNetE achieves better performance for author identification problem. Despite the consideration of specific task for embedding generation, HetNetE ignores paper content (e.g., abstract) that contains useful semantic information and searches indirect correlations among all kind of nodes for augmentation.

Besides paper-author identification and network embedding, the loss function of our model for formulating direct paper-author correlations is based on distance metric learning [32, 33]. In addition, we introduce word embedding models [14, 18] and gated recurrent neural network [2, 10] to generate deep semantic embeddings of paper. Moreover, the design of meta-path walk integrative learning module is inspired by heterogeneous network analysis works [3, 25, 26] as well as the Skipgram model [18].

6 CONCLUSION

In this paper, we study the author identification problem in big scholarly data, and design a representation learning model, i.e., Camel, to solve it. The model performs joint optimization of GRU content encoder based metric learning and Skipgram model based meta-path walk integrative learning. The extensive experiments on the well known AMiner data demonstrate that Camel outperforms a number of baselines. Detail discussions are also provided to show the effectiveness of different components in Camel. Some potential future work includes: (1) the dynamics of author embeddings should be considered for the task since authors keep publishing new papers; (2) the attention-based RNN can be applied to generate better semantic embeddings of paper.

ACKNOWLEDGMENT

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. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on. This work is partially supported by King Abdullah University of Science and Technology (KAUST).

REFERENCES

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

[2] 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).

[3] Yuxiao Dong, Nitesh V Chawla, and Ananthram Swami. 2017. metapath2vec: Scalable Representation Learning for Heterogeneous Networks. In *KDD*. 135–144.

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

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

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

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

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

[9] Shawndra Hill and Foster Provost. 2003. The myth of the double-blind review?: author identification using only citations. *Acm Sigkdd Explorations Newsletter* 5, 2 (2003), 179–184.

[10] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[11] Cheng-Kang Hsieh, Longqi Yang, Yin Cui, Tsung-Yi Lin, Serge Belongie, and Deborah Estrin. 2017. Collaborative metric learning. In *WWW*. 193–201.

[12] 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.

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

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

[15] 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. 2015. Combination of feature engineering and ranking models for paper-author identification in KDD Cup 2013. *JMLR* 16, 1 (2015), 2921–2947.

[16] Xiang Liu, Torsten Suel, and Nasir Memon. 2014. A robust model for paper reviewer assignment. In *RecSys*. 25–32.

[17] 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.

[18] 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.

[19] Mathias Payer, Ling Huang, Neil Zhenqiang Gong, Kevin Borgolte, and Mario Frank. 2015. What you submit is who you are: A multimodal approach for deanonymizing scientific publications. *TIFS* 10, 1 (2015), 200–212.

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

[21] 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.

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

[23] Hua-Wei Shen, Dashun Wang, Chaoming Song, and Albert-László Barabási. 2014. Modeling and Predicting Popularity Dynamics via Reinforced Poisson Processes.. In *AAAI*, Vol. 14. 291–297.

[24] Roberta Sinatra, Dashun Wang, Pierre Deville, Chaoming Song, and Albert-László Barabási. 2016. Quantifying the evolution of individual scientific impact. *Science* 354, 6312 (2016), aaf5239.

[25] 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.

[26] 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.

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

[28] 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.

[29] Andrew Tomkins, Min Zhang, and William D Heavlin. 2017. Single versus Double Blind Reviewing at WSDM 2017. *arXiv preprint arXiv:1702.00502* (2017).

[30] Susan Van Rooyen, Fiona Godlee, Stephen Evans, Nick Black, and Richard Smith. 1999. Effect of open peer review on quality of reviews and on reviewers' recommendations: a randomised trial. *BMJ* 318, 7175 (1999), 23–27.

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

[32] Kilian Q Weinberger and Lawrence K Saul. 2009. Distance metric learning for large margin nearest neighbor classification. *JMLR* 10, 2 (2009), 207–244.

[33] Eric P Xing, Michael I Jordan, Stuart J Russell, and Andrew Y Ng. 2003. Distance metric learning with application to clustering with side-information. In *NIPS*. 521–528.

[34] 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.

[35] Xing Zhao. 2013. The scorecard solution to the author-paper identification challenge. In *KDD Cup Workshop*. 4.