Text-Driven Neural Collaborative Filtering Model for Paper Source Tracing

Aobo Xu*†
upc20xab@s.upc.edu.cn
China University of Petroleum
Qingdao, Shandong, China

Bingyu Chang*
S23080035@s.upc.edu.cn
China University of Petroleum
Qingdao, Shandong, China

Qingpeng Liu*
S23080022@s.upc.edu.cn
China University of Petroleum
Qingdao, Shandong, China

Ling Jian‡
bebetter@upc.edu.cn
China University of Petroleum
Qingdao, Shandong, China

Abstract

Identifying significant references within the complex interrelations of a citation knowledge graph is challenging, which encompasses connections through citations, authorship, keywords, and other relational attributes. The Paper Source Tracing (PST) task seeks to automate the identification of pivotal references for given scholarly articles utilizing advanced data mining techniques. In the KDD CUP OAG-Challenge PST track, we design a recommendation-based framework tailored for the PST task. This framework employs the Neural Collaborative Filtering (NCF) model to generate final predictions. To process the textual attributes of the papers and extract input features for the model, we utilize SciBERT, a pre-trained language model. According to the experimental results, our method achieved a score of 0.37814 on the Mean Average Precision (MAP) metric, outperforming baseline models and ranking 11th among all participating teams. The source code is publicly available at https://github.com/MyLove-XAB/KDDCupFinal.

CCS Concepts

• **Information systems** → **Language models**; **Recommender systems**.

Keywords

Paper Source Tracing, Recommender Systems, Language Models

ACM Reference Format:
Aobo Xu, Bingyu Chang, Qingpeng Liu, and Ling Jian. 2024. Text-Driven Neural Collaborative Filtering Model for Paper Source Tracing. In *Proceedings of KDD CUP 2024 OAG-Challenges, Paper Source Tracing, Technical Report of Team AoboSama @ KDD CUP 2024. (KDD CUP 2024 OAG-Challenges).* ACM, New York, NY, USA, 4 pages. https://doi.org/XXXXXXX.XXXXXXX

*Team AoboSama
†Leader of Team AoboSama
‡Corresponding author

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
KDD CUP 2024 OAG-Challenges, August 25–29, 2024, Barcelona, Spain
© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-XXXX-X/18/06
https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Paper source tracing (PST) is vital for effective knowledge management. By linking common attributes between papers as relational edges, a comprehensive citation knowledge graph is constructed. Successfully addressing the PST task enables the clear visualization of whether paper A primarily inspires or contributes to paper B, as illustrated in Figure 1. Utilizing a citation knowledge graph facilitates the identification of critical references, elucidates the evolution of disciplines and techniques, and foster academic connections and collaborations. To this end, we propose a model based on Neural Collaborative Filtering (NCF) for the PST task. The model processes the textual attributes of papers obtained from DBLP-citation-network [10] (abbreviated as DBLP), with the assistance of SciBERT [3], a pre-trained language model. The proposed model demonstrates suitability for solving the PST task and achieves promising results on the Mean Average Precision (MAP) metric. The source code of our approach is publicly available at https://github.com/MyLove-XAB/KDDCupFinal. Our contributions are as follows:

- To the best of our knowledge, this is the first recommendation model applied to the paper source tracing task.
- We retrieve the text attributes from knowledge graph and process them with a language model. The experimental results demonstrate promising outcomes.

2 Related works

Two primary research fields are pertinent to the PST task from the perspective of the corresponding methodologies: machine learning-based methods and language model-based methods.

2.1 Machine Learning Based Methods for PST

The identification of significant references is generally framed as a classification or regression problem. This process involves the manual engineering of features, such as metadata-based parameters [8], citation counts[11] [7] , author overlap [2], textual similarity [5] [7], among others. These features are then utilized for training and inference with various machine learning methods, such as support

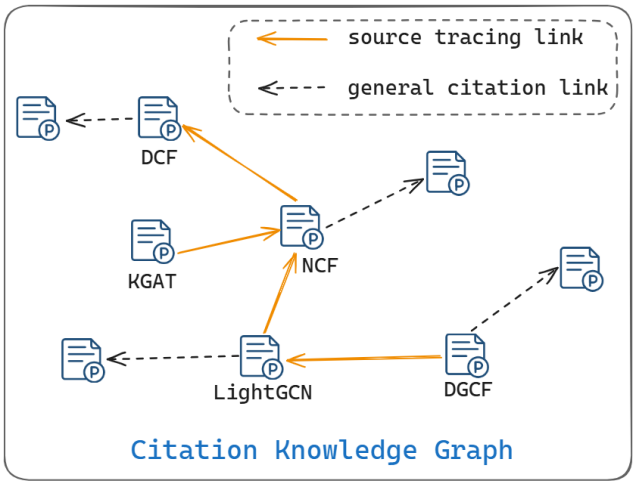

Figure 1: Paper Source Tracing with the Citation Knowledge Graph

vector machine, random forest, decision tree, kernel logistic regression, among others [5] [1] [8]. Machine learning-based methods offer the advantages of automating the processing of extensive data and efficiently extracting valuable insights for literature analysis. However, these methods may suffer from limitations in accuracy and interpretability, which can be attributed to factors such as data quality and the inherent complexity of the models.

2.2 Language Model Based Methods for PST

Recent advancements in language models have significantly enhanced text processing capabilities. Fine-tuning these models with annotated data has demonstrated promising results for PST task. Previous research has explored the integration of text embedding techniques such as Word2Vec [6] and GloVe [9]. By framing the PST as a binary classification problem for citation significance, the application of models such as SciBERT, GLM, and Galactica in conjunction with a Multi-Layer Perceptrons (MLPs) classifier has yielded promising results [12]. Building on these advancements, the PST-Bench dataset has been introduced, utilizing pre-trained models like BERT and SciBERT, which further demonstrate the effectiveness of these models in identifying significant references [13]. Language model-based methods are widely used due to their ability in deeply understanding contextual semantics. Nevertheless, the full potential of these models remains largely untapped.

3 Methodology

In this section, we first frame the PST task within the context of recommender systems. We then propose a NCF-based model that integrates both the interactions between papers and their references, as well as the textual attributes associated with the papers.

3.1 Problem Formulation

3.1.1 Citation Knowledge Graph. We define a citation knowledge graph as $\mathcal{G} = (\mathcal{E}, \mathcal{R})$, where $\mathcal{E}$ represents the set of entities, including papers, authors, and other relevant elements, while $\mathcal{R}$ donates the set of relations that describe the interactions between different types of nodes. Each paper node is associated with various text attributes, such as the paper title, abstract and body.

3.1.2 Recommendation Modeling. A conventional approach to address the PST task involves modeling it as a matching problem, where similarity between the query paper and potential references is calculated. Similarly, recommender systems operate by filtering items based on user preferences through similarity measures. Given the conceptual overlap between the PST task and recommendation systems, we frame the PST task as a recommendation problem. From the perspective of recommender systems, the query papers can be conceptualized as "users" and references as "items". In this framework, "citation" relations represent interactions between "users" and "items", with "critical citations" interpreted as positive interactions. By reformulating the problem as a recommendation task, we can leverage the family of algorithms used in recommender systems. Specifically, we focus on predicting the probability of a "critical citation" for each paper-reference pair (p, r), $p, r \in \mathcal{E}$. To this end, we employ collaborative filtering algorithms, which, despite their simplicity, have proven to be effective.

3.2 NCF-based PST Model with SciBERT

We employ the Neural Collaborative Filtering (NCF) model, which processes the features of papers and references separately in two channels, subsequently computing their mutual similarity as the prediction output. Drawing inspiration from the potential of language models in text-attributed graphs [4], we utilize a pre-trained language model, specifically SciBERT, to process the textual attributes of the papers and references. The SciBERT module comprises 12 BERT layers, each implementing attention and feed-forward network (FFN) mechanisms. Two separate SciBERT modules encode the query paper and reference inputs independently, after which the [CLS] token representations from the outputs are merged and fed into Multi-Layer Perceptrons (MLPs) to calculate the prediction values. The model architecture is depicted as Figure 2. Denoting the inputs of paper and reference as p_{in} and r_{in} respectively, the complete process of the model is as follows:

$$p_h = \text{SciBERT}_\text{p}(p_{in}), \tag{1}$$

$$r_h = \text{SciBERT}_\text{r}(r_{in}), \tag{2}$$

$$h = \text{concat}(p_h[CLS], r_h[CLS]), \tag{3}$$

$$x = \text{MLPs}(h), \tag{4}$$

$$o = \text{softmax}(\text{F}(x)), \tag{5}$$

where F represents the final prediction layer and the softmax operation converts the output into probability predictions. By training a sophisticated recommender model, the model is competent for solving the PST task.

4 Experiments

In this section, we perform comparison experiments to assess the suitability of the NCF-SciBERT model for the PST tasks.

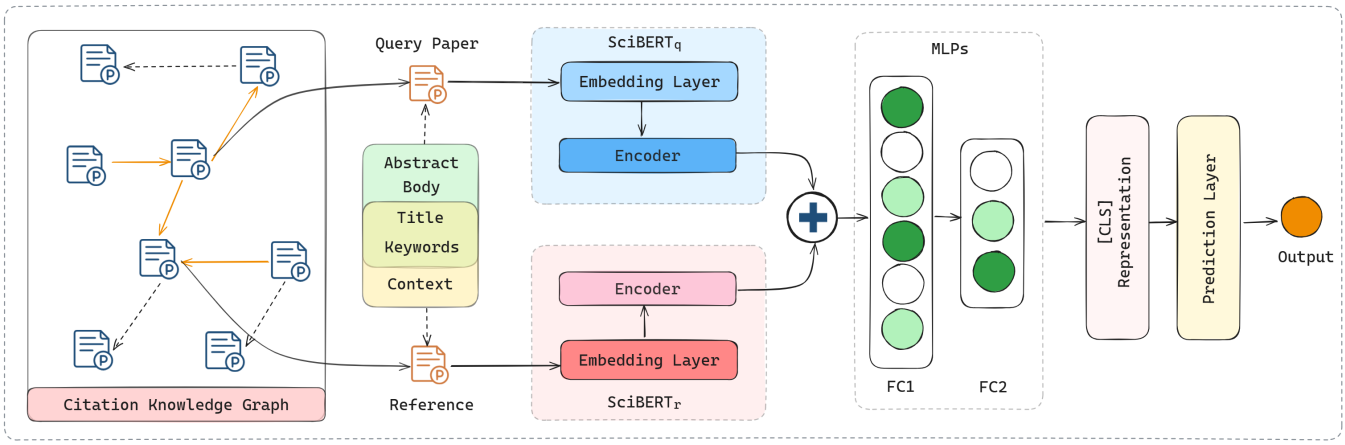

Figure 2: NCF-SciBERT Model for PST Task

4.1 Dataset

We conduct experiments on the PST dataset provided by the organizer. The text input for the query papers includes the title, abstract, keywords, and the first 500 characters of the body. For the references, the inputs comprise the title, keywords, and the contextual text in which each reference appears within the full text. Negative sampling is employed with a positive-to-negative sample ratio of 1:10. From the total of 788 training papers, we construct 8905 samples. The dataset is divided into training and validation subsets with a ratio of 8:2.

4.2 Experimental Results

The model is trained on a NVIDIA A100-PCIE-40GB GPU. The total number of model parameters is approximately 2.3×10^9. For hyperparameters, the batch size is set to 16, and the maximum length for both paper and reference inputs is 512 tokens. The model's memory usage does not exceed 36 GB. We employ the ADAM optimizer with an initial learning rate of 1×10^{-5}.

We evaluate the final performance of our model using the test data provided by the organizer. As baseline comparisons, we select the Random Forest, ProNE, and SciBERT methods proposed in [12]. Table 1 presents the results of various methods evaluated using the Mean Average Precision (MAP) metric. Our proposed model (NCF-SciBERT) achieves 0.37814 in MAP score, representing a significant improvement over the baselines, with a 28.23% enhancement relative to the best-performing baseline.

To demonstrate the significance of textual attributes in the task, we design an ablation experiment by introducing an NCF-id method, which excludes textual attributes and uses only the paper ID as input. The ablation results shows the text attribute greatly improve the performance, as shown in Table 1. Furthermore, the improvement of the NCF-SciBERT model over the SciBERT model alone suggests that the NCF architecture has a positive impact on the PST task.

Table 1: Results of Different Methods on MAP Metric

Method	Valid MAP	Test MAP	Description
Random Forest	0.21551	0.18020	Manual-feature based
ProNE Baseline	0.19104	0.16387	GNN based
SciBERT	0.29489	0.22237	LM based
NCF-id	0.18111	0.15569	No text attribute
NCF-SciBERT	**0.39065**	**0.37814**	Our method

5 Conclusion

In this paper, we formalize the paper source tracing task as a problem within the domain of recommender systems. We integrate the Neural Collaborative Filtering model with a language model, specifically SciBERT, by leveraging textual attributes retrieved from the citation knowledge graph. The experimental results demonstrate that the model achieves a MAP score of 0.37814 on the test dataset, which is a notable improvement over baseline methods and ranking 11th among all participating teams. Future work will explore the application of graph reasoning techniques to further address this task.

Acknowledgments

This work was supported by the National Key Research and Development Program of China under Grant Nos. 2021YFA1000100 and 2021YFA1000104, Laboratory Project of Higher Education Institutions in Shandong Province-Energy System Intelligent Management and Policy Simulation Laboratory at China University of Petroleum, and Youth Innovation Team of Higher Education Institutions in Shandong Province-Data Intelligence Innovation Team at China University of Petroleum.

References

[1] Naif Radi Aljohani, Ayman Fayoumi, and Saeed-Ul Hassan. 2021. An in-text citation classification predictive model for a scholarly search system. *Scientometrics* 126, 7 (2021), 5509–5529.

[2] Xin An, Xin Sun, and Shuo Xu. 2022. Important citations identification with semi-supervised classification model. *Scientometrics* 127, 11 (2022), 6533–6555.

[3] Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A Pretrained Language Model for Scientific Text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 3615–3620.

[4] Zhikai Chen, Haitao Mao, Hang Li, Wei Jin, Hongzhi Wen, Xiaochi Wei, Shuaiqiang Wang, Dawei Yin, Wenqi Fan, Hui Liu, et al. 2024. Exploring the potential of large language models (llms) in learning on graphs. *ACM SIGKDD Explorations Newsletter* 25, 2 (2024), 42–61.

[5] Saeed-Ul Hassan, Anam Akram, and Peter Haddawy. 2017. Identifying important citations using contextual information from full text. In *2017 ACM/IEEE joint conference on digital libraries (JCDL)*. IEEE, 1–8.

[6] Arshad Iqbal, Abdul Shahid, Muhammad Roman, Tanveer Afzal, and Muhammad Yahya. 2023. Exploiting Contextual Word Embedding for Identification of Important Citations: Incorporating Section-Wise Citation Counts and Metadata Features. *IEEE Access* (2023).

[7] Shahzad Nazir, Muhammad Asif, Shahbaz Ahmad, Hanan Aljuaid, Rimsha Iftikhar, Zubair Nawaz, and Yazeed Yasin Ghadi. 2022. Important citation identification by exploding the sentiment analysis and section-wise in-text citation weights. *IEEE Access* 10 (2022), 87990–88000.

[8] Faiza Qayyum and Muhammad Tanvir Afzal. 2019. Identification of important citations by exploiting research articles' metadata and cue-terms from content. *Scientometrics* 118 (2019), 21–43.

[9] Iqra Safder, Momin Ali, Naif Radi Aljohani, Raheel Nawaz, and Saeed-Ul Hassan. 2023. Neural machine translation for in-text citation classification. *Journal of the Association for Information Science and Technology* 74, 10 (2023), 1229–1240.

[10] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. 2008. Arnetminer: extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. 990–998.

[11] Marco Valenzuela, Vu Ha, and Oren Etzioni. 2015. Identifying meaningful citations. In *Workshops at the twenty-ninth AAAI conference on artificial intelligence*.

[12] Fanjin Zhang, Kun Cao, Yukuo Cen, Jifan Yu, Da Yin, and Jie Tang. 2024. PST-Bench: Tracing and Benchmarking the Source of Publications. *arXiv preprint arXiv:2402.16009* (2024).

[13] Fanjin Zhang, Shijie Shi, Yifan Zhu, Bo Chen, Yukuo Cen, Jifan Yu, Yelin Chen, Lulu Wang, Qingfei Zhao, Yuqing Cheng, Tianyi Han, Yuwei An, Dan Zhang, Weng Lam Tam, Kun Cao, Yunhe Pang, Xinyu Guan, Huihui Yuan, Jian Song, Xiaoyan Li, Yuxiao Dong, and Jie Tang. 2024. OAG-Bench: A Human-Curated Benchmark for Academic Graph Mining. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009