A Hybrid Approach for Paper Source Tracing using Cross Encoder and Hand-Crafted Features

Masato Hashimoto
NTT DOCOMO, INC.
Tokyo, Japan
masato.hashimoto.px@nttdocomo.com

Yukiko Yoshikawa
NTT DOCOMO, INC.
Tokyo, Japan
yukiko.yoshikawa.ra@nttdocomo.com

Keiichi Ochiai
NTT DOCOMO, INC.
Tokyo, Japan
ochiaike@nttdocomo.com

ABSTRACT

Academic graph mining, specifically paper citation analysis, is crucial for identifying promising technologies and efficient citation-based paper retrieval. The influence of cited papers varies, necessitating the quantification of their impact using large citation datasets and ground truth data. Although traditional methods used hand-crafted features for a limited dataset, advances in large-scale language models (LLMs) suggest potential improvements. To this end, the organizers of KDD Cup 2024 launched a competition focused on academic graph mining, called OAG-Challenge, accompanied by a large scale dataset referred to as OAG-Bench dataset. In this paper, we, DOCOMOLABZ, present our solution that achieved an 8th place ranking on the public leaderboard for the Paper Source Tracing (PST) task within the OAG-Challenge. Our solution is based on two hypotheses: (1) Highly influential cited papers show high similarity between their titles and the context in which they are cited, and (2) Hand-crafted features, such as citation frequency, are effective indicators of influence. The source code of our solution is available at https://github.com/NTT-DOCOMO-RD/kddcup2024-oag-challenge-pst-9th-solution-nttdocomolabz

CCS CONCEPTS

• **Computing methodologies → Artificial intelligence**.

KEYWORDS

KDD Cup, Cross-Encoder, Ensemble

ACM Reference Format:
Masato Hashimoto, Yukiko Yoshikawa, and Keiichi Ochiai. 2024. A Hybrid Approach for Paper Source Tracing using Cross Encoder and Hand-Crafted Features. In *KDDCUP '24: ACM SIGKDD Conference on Knowledge Discovery and Data Mining,August 25–29, 2024, Barcelona, Spain.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Academic graph mining, especially paper citation analysis, is an important research topic with a variety of potential applications, including discovery of promising technologies and efficient citation-based paper retrieval. However, since cited papers do not equally influence the cited sources, it is useful to quantify the influence each paper has on the cited literature. In order to quantify the impact, a large dataset of paper citations and the ground truth about the impact are required. An existing study has used hand-crafted features to estimate the degree of importance for a small number of annotated data [16]. On the other hand, in the field of natural language processing, large-scale language models (LLMs) such as ChatGPT have made remarkable progress, and it is thought that the estimation of impact can be advanced by utilizing such technology. In addition to impact estimation, other important tasks in academic graph mining include detecting misassignment of authors and papers and answering technical questions. For this reason, the organizers of the KDD Cup 2024 constructed a larger dataset called OAG-Bench [24] and held OAG-Challenge as KDD Cup 2024. Especially for the impact estimation task, the KDD Cup 2024 constructed a larger dataset called PST-Bench [21] and held a task called Paper Source Tracing (PST).

In this paper, we introduce our solution that placed 8th on the test set leaderboard for PST task in the competition. We design the solution based on the following two hypotheses.
Hypothesis 1: A cited paper with a high degree of influence on a paper citing it has a high degree of similarity between their titles and between the title of the cited paper and the text around a sentence where the cited paper is introduced.
Hypothesis 2: Hand-crafted features such as *the more times a paper is cited, the more influential it is* are also effective.

Based on these hypotheses, we adopted a two-stage approach to address the competition task. In the first stage, we employed a cross-encoder model integrated with SciBERT to determine whether the source paper is among the most significant references. In the second stage, we utilized the features derived from both the target and source papers, along with the output from the first stage, to feed into several binary classifiers for prediction. Finally, we aggregated the results of these classifiers, using an ensemble method to enhance the overall accuracy and robustness of our solution.

The source code of our solution is available at https://github.com/NTT-DOCOMO-RD/kddcup2024-oag-challenge-pst-9th-solution-nttdocomolabz

2 RELATED WORK

Influence estimation task of citing papers is one of the important types of research in Academic graph mining. The dataset paper [21] states that three approaches are effective for the influence estimation task: statistical methods, graph-based methods, and pre-trained language model (PLM) based methods.

KDDCUP '24, August 25–29, 2024, Barcelona, Spain

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Statistical Methods:
The statistical method defines manually created features and employs a classifier to indicate the importance of the reference. Valenzuela et al. [16] defined features including citing count, citing position, author overlap, text similarity. They employed Random Forest (RF) [7] to classify, the importance of the reference with an accuracy of 0.65. Pride et al. [10] added abstract features such as the similarity of the paper abstracts, and showed that the accuracy was improved. Hassan et al. [6] showed the importance of using certain features for accuracy, including clue word-based features such as "used" and "according to" that scholars often use when referring to previous work and context-based features such as the similarity of the text around the sentence in which the cited paper is introduced and the abstract of the cited paper.

Graph-based Methods:
Graph-based models have achieved significant success in various domains, such as molecular structure analysis [5], community detection [2], and forecasting retweet count [17]. In the analysis of paper citation graphs, research has been conducted on the task of predicting the number of citations of a paper. Wahid et al. [18]evaluated the task of predicting the number of citations of a paper by constructing a graph of the citation and back-citation relationships between each paper. In the dataset paper [21], they focused on structural similarity between the target paper and the reference paper, that is, the number of references shared between the target paper and the reference paper. They evaluated the impact estimation task by embedding the paper citation graph and measuring the similarity between the graphs.

PLM-based Methods:
PLM-based Methods are trained on large-scale text data, therefore they can be used for general purposes. In fact, they have achieved significant success in various domains, such as sentiment analysis [12], news article recommendation [19], and criminal verdict prediction [20]. Although there is no existing research that considers the task of predicting the importance of cited papers, it is possible also for this task to use a pre-trained language model, focusing on the fact that the importance of cited papers appears in the context. Even in the dataset paper [21], the influence estimation task can be evaluated by encoding the surrounding text where the cited paper is cited using a pre-trained model and performing binary prediction in the classifier layer.

3 APPROACH

Task definition. The task of the PST competition is to predict the most relevant reference paper for a given target paper. The provided dataset consists of a target paper and a set of reference papers. The target paper is a paper that cites the reference papers, and the reference papers are the papers that are cited by the target paper. And this competition provides XML files that contains full text of the paper. Competitors are required to predict the most significant reference paper for each target paper by using these provided dataset.

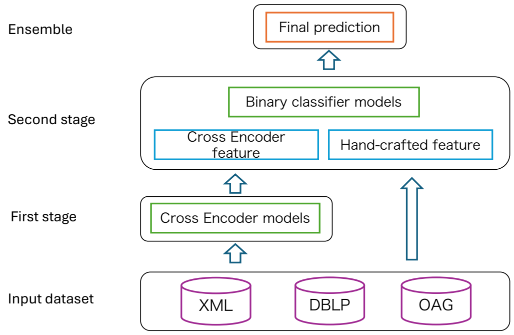

Figure 1: Overview of our solution pipeline

Solution overview. we used a two-stage approach to address the PST task. In the first stage, we used a cross-encoder model to detect which reference paper is the most relevant to the target paper. In the second stage, we used the output of the first stage and a variety of hand-crafted features to train several binary classifiers to predict the most relevant reference paper. Finally, we aggregated the results of these classifiers, using an ensemble method with a simple average method. The Figure 1 shows the overview of our solution.

3.1 Pre processing

To obtain a given paper's information such as title, authors, abstract, keywords, we tried to parse the paper's full text on XML files provided by the competition. However, the text in the XML file was incomplete, and we needed to fill in the missing information. For solving this problem, we used two kinds of extra datasets that are DBLP [14] and OAG-BERT [22],[23],[15],[13] in addition to the PST-Bench dataset provided by the competition. Both datasets represent large-scale paper citation network data, which contains meta data of papers such as authors, titles, abstracts, and references. We used them to fill in the missing information in the provided dataset for training and testing. For a paper that has an abstract but no title even after the information filling process, we used OAG-BERT [9],[24],[3] to generate the title of the paper from the abstract and fill in the missing title information. The OAG-BERT is a pre-trained language model that is trained on the OAG dataset. We used the `oagbert-v2` model to generate the title from the abstract.

3.2 Stage 1: Cross-encoder model with SciBERT

As a first stage, we used a cross-encoder model to determine whether the source paper is among the most significant references. The cross-encoder model is a model that takes a pair of two sentences as input and outputs a binary classification result. The SciBERT [1] model is used as the base model for the cross-encoder, because the SciBERT is a pre-trained language model that is trained on scientific papers. In our solution, the pairs are constructed from the title of the source paper and the title of the reference paper that concatenated the text around a sentence where the cited paper is introduced. To train the cross-encoder model, we used a binary cross-entropy with logit loss function and the AdamW optimizer. As a cross-validation strategy, grouped k-fold cross-validation is used (k was set to 5). Here the paper id was used as grouped key. And to prevent data leakage, we used 20% of the training dataset as a validation dataset after the grouped k-fold split. The validation dataset was used to find best model parameters while training. `max_seq_length` was set to 512, and epoch was set to 1.

3.3 Stage 2: Binary classifiers

In the second stage, we constructed hand-crafted features by processing the data and trained several binary classifiers. The features used in the stage are as follows:

- Cross-encoder based feature: The output of the cross-encoder model trained in the first stage is used as a feature.
- DBLP based feature: Features are created based on the DBLP dataset such as the number of citations, the number of co-authors, and published years.
- XML based feature: The feature as to which section of the paper the reference paper is introduced in the target paper
- SciBERT-encode based feature: The paper meta data such as title, keywords, and abstract of the target and reference papers are input to SciBERT, and the output is compressed to 20 dimensions using Truncated SVD and used as a feature. And the cosine similarity of the output of SciBERT for the target and reference papers is used as a feature.
- OAGBERT-encode based feature: The target paper and its reference paper are encoded using `oagbert-v2` and `oagbert-v2-sim`, and the cosine similarity of each pair is calculated.
- Text-based feature: The count of the number of common words and Levenshtein distance in all combination of the paper's meta data of target and reference papers like title, abstract, keywords, venue and organizers.

As a feature engineering, a few of features are created by the above features:

- Normalized by the maximum value of each target paper for some features such as counting features, the output of the first stage, and the number of citations of the reference paper.
- A multiplication of the cosine similarity of the SciBERT output of the target and reference papers. e.g. $cos\_sim_{title} \times cos\_sim_{abstract} \times cos\_sim_{keyword} \cdots$.
- Cosine similarities of the SciBERT output embedding of the text around which the reference paper is introduced in the target paper and the following texts that are inspired by the task definition of the competition.
  - "Main idea of this paper is inspired by the reference",
  - "The core method of this paper is derived from the reference",
  - "The reference is essential for this paper without the work of this reference, this paper cannot be completed"

We created several binary classifiers using the features described above. The classifiers used in the second stage are CatBoost [11], LightGBM [8], Support Vector Machine (SVM) [4], and Random Forest [7]. Here two types of LightGBM and CatBoost models are trained, one with the post-processed of cosine-similarity feature and the other without it. Note that the post-processed cosine-similarity means that the feature was set to 0 if the both of the elements of target and reference papers are missing. So that we create the 6 binary classifiers in total. For the training of the SVM, all the features of 20-dimension embedding obtained by SciBERT and Truncated SVD are removed because of the SVM model's training time, and the missing values of the features are filled with the mean value or 0 for each paper. As a cross-validation strategy, we used grouped k-fold cross-validation with k equals 5 as same as the first stage. And for avoiding data leakage, 25% of the training dataset was used as a validation dataset after the k-fold split. The hyperparameters of the classifiers are tuned to prevent overfitting.

Table 1: Offline evaluation results

Model	CV
SciBERT (First stage only)	0.44997
CatBoost without output of the first stage	0.45430
CatBoost	0.46922
LightGBM	0.46805
SVM	0.39906
Random Forest	0.45751
CatBoost + post-processed cosine similarity	0.47165
LightGBM + post-processed cosine similarity	0.47330

3.4 Ensemble method

As a final step, we aggregated the results of the 6 binary classifiers using an ensemble method. We used a simple average of the output of the classifiers as an ensemble method. The ensemble method is used to enhance the overall accuracy and robustness of our solution.

4 EVALUATION

4.1 Evaluation setting

Dataset. The dataset used in the competition is the PST-Bench dataset. The dataset consists of a target paper and a set of reference papers. The training dataset contains 788 target papers. The test dataset contains 394 target papers.

Metric. The evaluation metrics of this competition are the mean average precision (MAP). MAP is a metric that evaluates the ranking of the prediction results in which a significant reference paper must be ranked higher than the other reference papers.

Environment. The experiments were conducted on an Amazon EC2 instance with 1 NVIDIA T4 Tensor Core GPU and 16GB of memory that is known as `g4dn.xlarge`.

4.2 Evaluation of Solution

We used the 5-fold grouped k fold strategy for offline evaluation in which we call cross-validation (CV) scores. The MAP score of each fold is averaged to obtain the final offline MAP score. The results of the offline evaluation are shown in Table 1. Here, the Catboost, LightGBM, SVM, and Random Forest models have higher CV scores than the SciBERT model. This result indicates that the hand-crafted features using 2-stage method are effective in the PST task. In this result we can see that the SVM model has a lower accuracy than the other models. Nevertheless, we adopted it in the ensemble method to increase the diversity of the models.

The leaderboard score (LB) of the validation dataset using our approach (excluding SVM, Random Forest classifier and OAG-BERT feature) was 0.44049. Moreover, we show the online evaluation

Table 2: Online evaluation results

Method	LB (Test set)
SciBERT (First stage only)	0.35423
CatBoost	0.40394
Ensemble	**0.41668**

results in Table 2, which are the LBs of the test dataset. The score of CatBoost of the second stage is higher than the first-stage-only model. This means our two-stage method is effective in the PST task as is the case with the offline evaluation. In addition, the ensemble method with 6 classification models is more effective than the single model and has the highest score in our trials. The MAP score of the ensemble method is 0.41668, which is the 8th place on the leaderboard.

5 CONCLUSIONS

In this paper, we have introduced our solution that placed 8th on the test set leaderboard for PST task in the competition. Our solution method consists of a two-stage process that uses a cross encoder to compute the similarity of the paper titles and a variety of hand-crafted features. Our solution was awarded 8th place in PST task in the KDD Cup 2024 competition.

REFERENCES

[1] Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: Pretrained Language Model for Scientific Text. In *EMNLP*. arXiv:arXiv:1903.10676

[2] Joan Bruna and X Li. 2017. Community detection with graph neural networks. *stat* 1050 (2017), 27.

[3] Bo Chen, Jing Zhang, Jie Tang, Lingfan Cai, Zhaoyu Wang, Shu Zhao, Hong Chen, and Cuiping Li. 2020. CONNA: Addressing Name Disambiguation on The Fly. *IEEE Transactions on Knowledge and Data Engineering* (2020).

[4] Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning* 20, 3 (1995), 273–297.

[5] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. 2017. Neural message passing for quantum chemistry. In *International conference on machine learning*. PMLR, 1263–1272.

[6] 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.

[7] Tin Kam Ho. 1995. Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, Vol. 1. IEEE, 278–282.

[8] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. LightGBM: a highly efficient gradient boosting decision tree. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (Long Beach, California, USA) *(NIPS'17)*. Curran Associates Inc., Red Hook, NY, USA, 3149–3157.

[9] Xiao Liu, Da Yin, Xingjian Zhang, Kai Su, Kan Wu, Hongxia Yang, and Jie Tang. 2021. OAG-BERT: Pre-train Heterogeneous Entity-augmented Academic Language Model. *arXiv preprint arXiv:2103.02410* (2021).

[10] David Pride and Petr Knoth. 2017. Incidental or influential?-challenges in automatically detecting citation importance using publication full texts. In *Research and Advanced Technology for Digital Libraries: 21st International Conference on Theory and Practice of Digital Libraries*. Springer, 572–578.

[11] Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. 2018. CatBoost: unbiased boosting with categorical features. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems* (Montréal, Canada) *(NIPS'18)*. Curran Associates Inc., Red Hook, NY, USA, 6639–6649.

[12] Chinmayee Sahoo, Mayur Wankhade, and Binod Kumar Singh. 2023. Sentiment analysis using deep learning techniques: a comprehensive review. *International Journal of Multimedia Information Retrieval* 12, 2 (2023), 41.

[13] Arnab Sinha, Zhihong Shen, Yang Song, Hao Ma, Darrin Eide, Bo-June (Paul) Hsu, and Kuansan Wang. 2015. An Overview of Microsoft Academic Service (MAS) and Applications. In *Proceedings of the 24th International Conference on World Wide Web* (Florence, Italy) *(WWW '15 Companion)*. Association for Computing Machinery, New York, NY, USA, 243–246. https://doi.org/10.1145/2740908.2742839

[14] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. 2008. ArnetMiner: Extraction and Mining of Academic Social Networks. In *KDD'08*. 990–998.

[15] 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* (Las Vegas, Nevada, USA) *(KDD '08)*. Association for Computing Machinery, New York, NY, USA, 990–998. https://doi.org/10.1145/1401890.1402008

[16] Marco Valenzuela, Vu Ha, and Oren Etzioni. 2015. Identifying meaningful citations. In *Proceedings of the "Scholarly Big Data: AI Perspectives, Challenges, and Ideas" Workshop at the Twenty-Ninth AAAI Conference on Artificial Intelligence.*

[17] Raghavendran Vijayan and George Mohler. 2018. Forecasting retweet count during elections using graph convolution neural networks. In *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, 256–262.

[18] Abdul Wahid, Rajesh Sharma, and Chandra Sekhara Rao Annavarapu. 2021. A Graph Convolutional Neural Network based Framework for Estimating Future Citations Count of Research Articles. *arXiv preprint arXiv:2104.04939* (2021).

[19] Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. 2021. Empowering news recommendation with pre-trained language models. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*. 1652–1656.

[20] Chaojun Xiao, Xueyu Hu, Zhiyuan Liu, Cunchao Tu, and Maosong Sun. 2021. Lawformer: A pre-trained language model for chinese legal long documents. *AI Open* 2 (2021), 79–84.

[21] 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).

[22] Fanjin Zhang, Xiao Liu, Jie Tang, Yuxiao Dong, Peiran Yao, Jie Zhang, Xiaotao Gu, Yan Wang, Evgeny Kharlamov, Bin Shao, Rui Li, and Kuansan Wang. 2023. OAG: Linking Entities Across Large-Scale Heterogeneous Knowledge Graphs. *IEEE Transactions on Knowledge and Data Engineering* 35, 9 (2023), 9225–9239. https://doi.org/10.1109/TKDE.2022.3222168

[23] Fanjin Zhang, Xiao Liu, Jie Tang, Yuxiao Dong, Peiran Yao, Jie Zhang, Xiaotao Gu, Yan Wang, Bin Shao, Rui Li, and Kuansan Wang. 2019. OAG: Toward Linking Large-scale Heterogeneous Entity Graphs. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (Anchorage, AK, USA) *(KDD '19)*. Association for Computing Machinery, New York, NY, USA, 2585–2595. https://doi.org/10.1145/3292500.3330785

[24] Fanjin Zhang, Shijie Shi, Yifan Zhu, Bo Chen, Yukuo Cen, Jifan Yu, Yelin Chen, Lulu Wang, Qingfei Zhao, Yuqing Cheng, et al. 2024. OAG-Bench: A Human-Curated Benchmark for Academic Graph Mining. *arXiv preprint arXiv:2402.15810* (2024).