Design of an Anomaly Detection Model Utilizing Semantic and Relational Features of Papers Assigned to Authors

Kenichirou Miyaki
NTT DOCOMO, INC.
Tokyo, Japan
kenichirou.miyaki.dk@nttdocomo.com

Sho Maeoki
NTT DOCOMO, INC.
Tokyo, Japan
syou.maeoki.rz@nttdocomo.com

Meisaku Suzuki
NTT DOCOMO, INC.
Tokyo, Japan
meisaku.suzuki.fw@nttdocomo.com

Abstract

The primary goal of academic data mining is to deepen the understanding of scientific development, nature, and trends, thereby maximizing scientific, technological, and educational value. Currently, many entity-centric applications, such as paper search, expert discovery, and venue recommendation, exist. However, the lack of appropriate public benchmarks significantly limits the progress of academic graph mining. To address this issue, the Knowledge Engineering Group (KEG) at Tsinghua University and Zhipu AI organized the OAG-Challenge in KDD Cup 2024. This paper introduces the solution presented by our DOCOMOLABS team, which achieved 6th place in the paper assignment error detection task (IND) as part of the OAG-Challenge. Our approach captures the semantic and relational features among papers assigned to each author using various methods and constructs a high-precision paper assignment error detection model by ensembling multiple binary classifier models. Our solution's source code is available on GitHub.

CCS Concepts

• **Information systems → Digital libraries and archives; Data mining**.

Keywords

KDD Cup, OAG, academic knowledge graph, academic graph mining

ACM Reference Format:
Kenichirou Miyaki, Sho Maeoki, and Meisaku Suzuki. 2024. Design of an Anomaly Detection Model Utilizing Semantic and Relational Features of Papers Assigned to Authors. 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

Despite the increasing number of papers being published via online academic systems, name disambiguation in online academic systems has long been a complex and persistent challenge due to the ambiguity of author names. This issue affects the overall process of data mining related to academic papers, potentially invalidating author rankings, leading to inappropriate awards, and significantly impacting the reliability of academic data. Accurately identifying authors with identical names or those using different name variations is particularly difficult. Therefore, precise author identification is essential for numerous data mining tasks, including citation relationships, co-author networks, and the identification of research fields.

To address this challenge, the WhoIsWho project [2] was launched. The WhoIsWho project aims to enhance the accuracy and reliability of name disambiguation by constructing a large-scale, high-quality dataset comprising over 1,000 names, 70,000 authors, and 1,000,000 papers. The project defines three primary tasks to tackle various name disambiguation challenges: From-scratch Name Disambiguation (SND), Real-time Name Disambiguation (RND) for assigning new papers to existing authors, and Incorrect Assignment Detection (IND) to correct erroneous assignments.

In the KDD Cup 2024 OAG-Challenge, a competition focusing on the IND task was held, where we achieved a sixth-place finish. Our approach involves extracting features from each paper and representing relationships between authors' papers using various methods to formulate an anomaly detection task. This paper presents our solution.

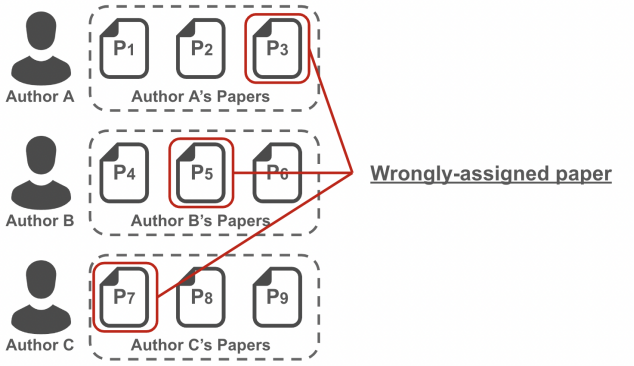

Figure 1: Overview of the IND Task. Given each author's profile, including author name and published papers, participants are asked to detect incorrect paper assignments among all one's papers.

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

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

2 Related Work

2.1 Challenges in Name Disambiguation

Name disambiguation in academic databases is a critical challenge arising when multiple authors share the same name or when the same author uses different name variations. This issue directly affects the accuracy of literature searches and citation analyses, leading to errors and inaccuracies in academic evaluation and recognition.

The WhoIsWho project provides a high-quality and large-scale dataset to improve the accuracy and reliability of name disambiguation. This project includes over 1,000 names, more than 70,000 authors, and over 1,000,000 papers, offering benchmarks, contest leaderboards, and toolkits for name disambiguation. This dataset is manually labeled by expert annotators, associating papers with authors under ambiguous names through a detailed manual labeling process.

2.2 Existing Approaches

2.2.1 From-Scratch Name Disambiguation. The From-Scratch Name Disambiguation (SND) task aims to group papers by authors with the same name when constructing a new dataset. Specific solutions involve first extracting features using the metadata of the papers (titles, keywords, abstracts) and the collaborative relationships of the authors. Next, clustering algorithms, such as k-means clustering or DBSCAN, are applied to these features to group the clusters so that each corresponds to a single author. When using machine learning models, supervised learning is performed using the extracted features to enable the model to accurately identify authors.

2.2.2 Real-time Name Disambiguation. The Real-time Name Disambiguation (RND) task aims to assign newly added papers in an existing database to existing authors in real-time. This task requires the model to be updated each time a new paper is added. Common solutions include using incremental learning algorithms, such as online logistic regression or online SVM, to process data sequentially and update the model. Additionally, streaming data processing platforms (e.g., Apache Kafka or Apache Flink) are utilized to efficiently handle large volumes of data in real-time.

2.3 Significance of the IND Approach

The main objective of the IND (Incorrect Assignment Detection) task is to detect and correct errors made by the SND (From-Scratch Name Disambiguation) and RND (Real-time Name Disambiguation) systems. This enhances the reliability and accuracy of academic databases, maintaining data integrity. Correcting incorrect assignments improves the precision of academic evaluation and citation analysis. This paper proposes solutions to achieve these goals.

3 Methodology

3.1 Task Overview

The IND task aims to detect assignment errors for each author by using the given paper assignments and metadata of each paper. The paper metadata consists of id, title, abstract, author name, author org, venue, and year.

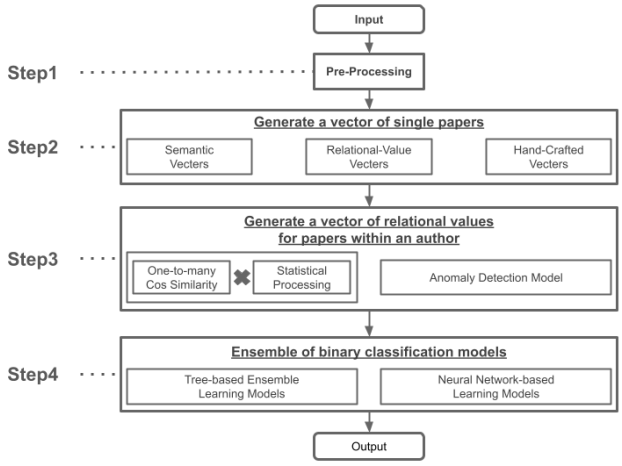

Figure 2: A four-step model for detecting paper assignment errors for authors: STEP 1: Data cleansing, STEP 2: Generating paper embeddings, STEP 3: Generating relationship values between papers for each author, STEP 4: Detecting paper assignment errors for each author using binary classification models.

3.2 Method Overview

In this section, we describe our solution. Our approach consists of four distinct steps. Figure 1 shows an overview of the solution. First, we perform data cleansing on the input data (STEP 1), then generate embedded representations for each paper (STEP 2). After that, we vectorize the relational values between papers within each author (STEP 3), and finally, we use an ensemble of multiple binary classification models to detect paper assignment errors for each author (STEP 4).

3.3 Implementation Details

Please refer to our code[1] for more details. Here we describe a brief explanation of our implementation points.

3.3.1 STEP 1: Data cleansing. In the OAG-IND task, two types of data were provided: paper assignment data for each author and paper metadata. The paper metadata contains a significant amount of noise, which can lead to a decline in accuracy if the learning process proceeds without addressing this noise. Therefore, we conducted comprehensive data cleansing on three types of data in the paper metadata: natural language data (such as paper titles and abstracts), author name information linked to the papers, and numerical information regarding the publication year of the papers.

Firstly, we explain the data cleansing techniques applied to the natural language data. This includes elements such as titles, abstracts, author affiliations, keywords, and venues in the provided paper metadata. These data contain numerous unnecessary symbols and spaces, making them noisy. To maintain accuracy in the subsequent natural language embedding generation, we conducted the

[1]https://github.com/NTT-DOCOMO-RD/kddcup2024-oag-challenge-ind-7th-aqa-7th-solution-nttdocomolabs/tree/main/IND)

following processes: removal of stop words, symbols, and spaces, conversion of numbers to 0, and stemming.

Secondly, we explain the data cleansing techniques applied to the author name data. The provided paper metadata includes author names, which serve as crucial keys when aggregating papers associated with each author. Thus, it is necessary to standardize all author names into a unified format. For the IND task, we decided to Romanize all author names and carried out the following procedures. Specifically, we handled abbreviations in English names, standardized the order of given names and family names, and also converted Japanese names from Kanji to Roman letters, Chinese names from Kanji to Pinyin.

Finally, we explain the data cleansing techniques applied to the numerical data. The provided paper metadata includes the publication year of the papers, which is crucial for understanding the author's publication timeline. For this data, we performed processes to impute outliers with the mean value and to impute missing values with the mean value.

3.3.2 STEP 2: Generating Paper Embeddings. The quality of embeddings is crucial for comparing the provided papers. Therefore, we aimed to generate diverse embeddings by applying multiple methods to a single paper. Specifically, we combined the following three types of vectors to obtain paper embeddings: hand-crated vectors, semantic vectors, and relational-value vectors.

Hand-Crafted Vectors: This method aims to extract basic information from each paper. Specifically, we calculated the number of characters and languages in the natural language data, the number of authors and organizations assigned to each paper, and the publication year. These features were used to create the embeddings for each paper.

Semantic Vectors: This method aims to extract semantic features from the textual content of each paper. To obtain semantic information, we employed two primary methods. The first method involved using word2vec [7] to generate embeddings trained on the natural language data within the papers. The second method utilized pre-trained BERT [4] models to generate embeddings. By combining these two methods, we captured comprehensive semantic information for each paper.

Relational-Value Vectors: This method aims to extract features from the relationships between papers. Specifically, it involves a three-stage process. In the first stage, we create graph data to represent the relationships between papers. Each paper is assigned to a node, and edges are created only when there are co-authors or shared organizations between papers. This constructs a graph that represents the relationships between papers. In the second stage, we perform random walks on the graph data to generate meta-paths that express the relationships between papers. Specifically, each paper is taken as a starting point, and nodes are selected randomly by traversing co-author edges or shared organization edges. This generates meta-paths that express the relationships between papers. Finally, these meta-paths are treated as vocabulary and used as training data for word2vec. During the training of word2vec, we apply the skip-gram algorithm to generate embedding representations for each paper.

By repeating this three-stage process multiple times and averaging the outputs, we were able to obtain stable embedding representations for each paper.

3.3.3 STEP 3: Generating Relationship Values Between Papers for Each Author. To detect paper assignment errors for each author, we calculated the relationship values between papers associated with each author. The quality of these relationship values directly impacts the accuracy of subsequent processing steps, so we employed a multifaceted approach to enhance precision. Specifically, we combined the three methods as follows.

Calculating Cosine Similarity Using Embedding Representations: Utilizing the embedding representations of papers associated with each author, we calculated the distances between papers using cosine similarity. Papers that are significantly distant from others in terms of cosine similarity are identified as potential errors.

Performing Clustering: We conducted clustering on the papers associated with each author. Specifically, we employed DBSCAN, KMeans, and GaussianMixture to group the papers into clusters, identifying patterns and outliers.

Ensembling Anomaly Detection Models: We ensembled multiple anomaly detection models on the papers associated with each author. Specifically, we utilized the One-Class Support Vector Machine [9], the Elliptic Envelope, the Isolation Forest [6], and the Local Outlier Factor, combining their outputs to form the final relationship value vectors.

3.3.4 STEP 4: Detecting Paper Assignment Errors for Each Author Using Binary Classification Models. Finally, we created training data by combining the embeddings obtained in STEP 3 with the provided paper assignment data. We then built binary classification models using multiple algorithms to predict whether each paper is a paper assignment error. Specifically, we ensembled primarily decision-tree based models and neural network based models.

Tree-based Ensemble Learning Models: This approach utilized tree-structured ensemble learning models such as lightGBM [5], CatBoost [8], Xgboost [3], and RandomForest.

Neural Network-based Learning Models: This approach employed a 7-layer neural network model and TabNet [1], allowing us to capture more complex patterns and relationships.

In the end, we ensembled the outputs of all models and averaged their predictions to produce the final output.

4 Evaluation

4.1 Experiment Settings

4.1.1 Dataset. The dataset used in the IND task [10] consists of correct paper IDs associated with each author and paper IDs incorrectly assigned to authors. The task schedule is composed of two stages; we refer to a dataset offered in the first two-month period as training/validation split, a dataset offered in the last one-week as test split. The training dataset includes 779 authors, while the test dataset includes 515 authors.

4.1.2 Evaluation Metrics. The evaluation metric used is the AUC (Area Under the ROC Curve), which is widely adopted in anomaly detection. AUC is a measure of the overall performance of a model,

Table 1: Results of methods in leaderboard (LB)

Method	Feature	LB score
LightGBM	Relational-Value Vectors	0.7427
LightGBM	Hand-Crafted Vectors	0.7635
LightGBM	Semantic Vectors	0.7756
XgBoost	All	0.7792
LightGBM	All	0.7874
7-layer Neural Network	All	0.7892
TabNet	All	0.7921
CatBoost	All	0.8045
Proposed Method + Xgboost	All	0.8045
Proposed Method	All	0.8048

representing the area under the ROC curve. Additionally, the number of errors incorrectly assigned to authors is weighted, allowing for an evaluation that reflects the importance of the errors.

4.1.3 Environment. The experiments were conducted on an Amazon EC2 instance (known as g5.2xlarge) equipped with an NVIDIA A10G Tensor Core GPU and 32GB of memory.

4.2 Results and Discussion

Table 1 shows the public leaderboard scores for each method of our solution. These results were evaluated online using weighted AUC on the public leaderboard. Below, we explain the experimental results of our methods as shown in Table 1.

We constructed our baseline by ensembling Tree-based Ensemble Learning Models. Initially, to correct for imbalanced data, we added a model trained with LightGBM after undersampling the training data. We also adopted XGBoost as part of the ensemble, but the accuracy did not meet our expectations. Ultimately, by excluding similar algorithm models and focusing on a more diverse set of models for ensembling, we achieved a leaderboard score of 0.80487, ranking 6th.

5 Conclusion

In this paper, we presented our solution that ranked 6th on the test set leaderboard of the KDD Cup 2024 IND task. Our approach consists of the following four main steps.

Data Cleansing: Initially, we performed data preprocessing to correct or remove incomplete or inaccurate data. This improved data quality and enhanced the accuracy of subsequent processes.

Generating Embedding Representations for Each Paper: Next, we vectorized metadata such as the titles and abstracts of each paper to generate semantic embedding representations. This involved using natural language processing techniques like BERT and word2vec.

Vectorizing Relationship Values Between Papers for Each Author: Furthermore, based on information such as co-author relationships and shared publications, we represented the relationships between authors as numerical vectors. This allowed for a quantitative understanding of the relationships between authors.

Inference Using an Ensemble of Binary Classification Models: Finally, we built an ensemble model by combining multiple binary classification models to detect paper assignment errors for each author. This achieved higher accuracy than a single model.

As a result, our proposed solution achieved 6th place in the KDD Cup 2024 IND task.

References

[1] Sercan Ö Arik and Tomas Pfister. 2019. TabNet: Attentive Interpretable Tabular Learning. In *AAAI*.

[2] Bo Chen, Jing Zhang, Fanjin Zhang, Tianyi Han, Yuqing Cheng, Xiaoyan Li, Yuxiao Dong, and Jie Tang. 2023. Web-Scale Academic Name Disambiguation: the WhoIsWho Benchmark, Leaderboard, and Toolkit. *CoRR* (2023).

[3] Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. In *KDD*.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL*.

[5] 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 *NeurIPS*.

[6] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. 2008. Isolation Forest. In *ICDM*.

[7] Tomas Mikolov, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. In *ICLR*.

[8] Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. 2018. CatBoost: unbiased boosting with categorical features. In *NeurIPS*.

[9] Bernhard Schölkopf, John C. Platt, John Shawe-Taylor, Alex J. Smola, and Robert C. Williamson. 2000. Support Vector Method for Novelty Detection. In *NuerIPS*.

[10] 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 *KDD*.