A Hybrid Machine Learning Method to Author Name Disambiguation

Natan S. Rodrigues[1,2], **Celia G. Ralha**[2]

[1]Instituto Acadêmico de Ciências Tecnológicas
Universidade Estadual de Goiás (UEG) – Goiás – GO – Brasil

[2]Departamento de Ciência da Computação – Instituto de Ciências Exatas
Universidade de Brasília (UnB) – Brasília, DF – Brasil

natan.rodrigues@ueg.br, ghedini@unb.br

Abstract. *Digital bibliographic repositories, including publications, authors, and research fields are essential for sharing scientific information. Nevertheless, the information retrieval, extraction, and classification efficiency in such archives is threatened by author name ambiguity. This paper addresses the Author Name Disambiguation (AND) problem by proposing a hybrid machine learning method integrating Bidirectional Encoder Representations from Transformers (BERT), Graph Convolutional Network (GCN), and Graph Enhanced Hierarchical Agglomerative Clustering (GHAC) approaches. The BERT model extracts textual data from scientific documents, the GCN structures global data from academic graphs, and GHAC considers heterogeneous networks' global context to identify scientific collaboration patterns. We compare the hybrid method with AND state-of-the-art work using a publicly accessible data set consisting of 7,886 documents, 137 unique authors, and 14 groups of ambiguous authors, along with recognized validation metrics. The results achieved a high precision score of 93.8%, recall of 96.3%, F1-measure of 95%, Average Cluster Purity (ACP) of 96.5%, Average Author Purity (AAP) of 97.4% and K-Metric of 96.9%. Compared to the AND baseline approach, the hybrid method presents better results indicating a promising approach.*

1. Introduction

Digital bibliographic repositories are vast reservoirs of bibliographic citation information (DBLP [DBLP 2024], ArnetMiner [AMiner 2024b], CiteSeerX [CiteSeerX 2019]). They offer functionalities that allow the identification of works by scientists, authors, and their respective academic social networks. The DBLP currently lists around 7 million works in Computer Science, including journals and conference articles. In January 2024, DBLP gathered information on approximately 3.5 million authors, with 227 thousand names of researchers and publications manually verified by the DBLP team, corresponding to a curation of 34% of all publications in the database.[1] ArnetMiner stores information on approximately 2 million of scientific works, 1.7 million of authors, and 8 million of bibliographic citations [AMiner 2024a].[2]

By storing millions of information from bibliographic records, digital repositories become an essential source of information for the global academic and scientific

[1]https://dblp.org/
[2]https://www.aminer.org/

community, allowing retrieval, extraction and classification of relevant publications in a centralized manner [Ferreira et al. 2020]. In addition to these bibliographic features, such digital libraries provide helpful analysis useful for better decision-making by scientific funding agencies and academic institutions [Hussain and Asghar 2017].

However, a common problem in digital bibliographic repositories is automatic Author Name Disambiguation (AND). The AND problem occurs when different authors have the same name record or when an author has multiple name records in the same data set. Such a problem can significantly affect the document and information retrieval performance through *Web* search engines and obstruct entity integrity for integrated databases. Even though the author's name ambiguity problem has been studied for decades, it remains without a canonical solution. Thus, research efforts to solve the AND problem are essential, especially considering that digital bibliographic repositories are becoming more person-centric than document-centric [Shin et al. 2014].

This work addresses the AND problem with a novel hybrid method combining advanced machine learning techniques, such as the Bidirectional Encoder Representations from Transformers (BERT) [Devlin et al. 2019], Graph Convolutional Network (GCN) [Zhang et al. 2019], and Graph Enhanced Hierarchical Agglomerative Clustering (GHAC) [Qiao et al. 2019]. As proposed by [Kipf and Welling 2017], GCN is a powerful machine learning model that extends the Convolutional Neural Network (CNN) to handle data structured as graphs, capturing local and global dependencies within a network. Our method aims to enhance the AND accuracy in digital bibliographic repositories considering information retrieval, extraction, and classification by applying document content semantic treatment related to a graph representation of relationships between documents, authors, and other scientific attributes.

As presented in [Ferreira et al. 2020], there are several approaches to solving AND problem applying various techniques, but no works combining transfer learning with GCN and GHAC techniques. Also, according to a recent AND literature review, using the theory of the consolidated meta-analytic approach with quantitative techniques and bibliometric aspects, the hybrid method proposed is considered a novel solution to AND [Rodrigues et al. 2024].

The rest of the article includes in Section 2 related work focusing on approaches to the AND problem. Section 3 details the AND hybrid method. Section 4 includes the conducted experiments with the evaluation metrics. In Section 5, we present the results with discussion. Finally, the conclusion and future work are in Section 6.

2. Related Work

As presented in [Rodrigues et al. 2024], largely used AND solving approaches are author grouping associated with similarity functions and clustering methods, and some works with author assignment allied to classification methods. Also, approaches based on graphs, word embedding with supervised learning, and heuristics with probabilistic applications are common. The literature review highlights author clustering techniques' prevalence and effectiveness, especially when addressing issues associated with large bibliographic databases. In this section, we present works most related to our hybrid method.

The authors in [Kim and Owen-Smith 2020] explore supervised techniques using transfer learning on AND tasks where no labelled data was available for training. The results show that by training source data that well represent the main characteristics of the target datasets, the developed disambiguation models through transfer learning can produce results comparable to those achieved by traditional machine learning approaches, which train algorithms on specifically labelled subsets of the target data.

In [Waqas and Qadir 2021], the authors propose a method to perform AND based on heuristic clusters in several layers. They used global characteristics and those related to the structure of publications to group them. One of the differences pointed out by the authors is that instead of relying only on keyword information, the approach also considers the contextual structure of publications for grouping. The authors use an incremental classification method to reduce errors after creating clusters. A dataset called *CustAND* was presented for testing and executing the AND method.

The approach of [Pooja et al. 2022] uses GCN in conjunction with attention mechanisms for learning representations in a heterogeneous graph of documents. The work highlights the importance of using attention at different levels, both about the types of neighbors and relationships, to incorporate relevant context into learning node representations. The emphasis on attention allows a detailed analysis of the impact of this mechanism on capturing semantic and contextual information from documents in a graph. The authors used two ArnetMiner variants as data sets, the first with 110 and the second with 100 ambiguous name references.

3. The Hybrid Method

The hybrid method has four main steps as presented in Figure 1 and described in the sequence.

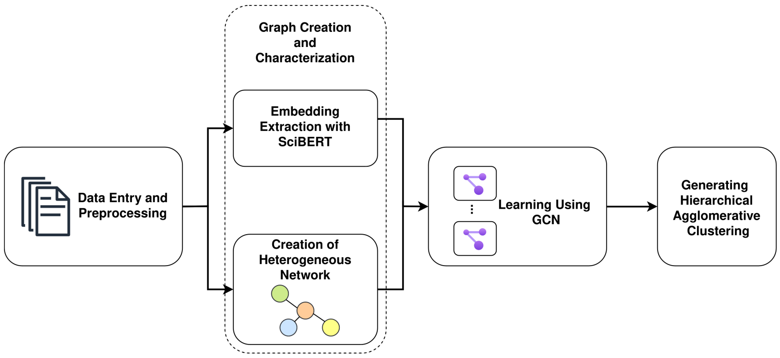

Figure 1. The hybrid method workflow.

3.1. Data Entry and Preprocessing

The hybrid method's first step deals with the input and preprocessing of document data (publications), when data is adjusted and formatted to ensure input suitability for the subsequent steps.

3.2. Graph Creation and Characterization

This step plays a fundamental role as it creates the structure of the heterogeneous network from the information received from the previous step and provides contextual information for the AND task.

- Creation of a Heterogeneous Network - includes different types of nodes and edges. The graph is formally defined as $G_{\text{heterogeneous}} = (N_{\text{nodes}}, E_{\text{edges}})$, where N_{nodes} are nodes representing publications, authors, and words. The edges E_{edges} represent the different connections between nodes, such as $contains$ (between publications and words), $written\_by$ (between publications and authors), $co-authored$ (between authors who collaborated on the same publication), and $shared-word$ (between publications that share keywords).
- Embedding Extraction with BERT - BERT uses transfer learning pretraining its parameters on large sets of unlabeled texts with only minor modifications to perform tasks in a given domain. BERT converts every word in a text into a vector representation that captures the word's meaning given the context in which it appears. This representation can be combined to obtain a representation of entire sentences. In this work, we use the SciBERT variant of BERT pre-trained on scientific texts, which is particularly effective at capturing the contextual and semantic information of academic documents [Beltagy et al. 2019]. SciBERT calculates the embeddings of publications based on titles and abstracts. These embeddings are incorporated as features of the nodes that represent the publications in the graph. The algorithm is described in the sequence.
  Given a set of N documents with titles and abstracts, where each document i has a title T_i and an abstract R_i, the embedding extraction process with SciBERT can be detailed as follows:
  1. Tokenization of Titles and Abstracts: each title T_i and abstract R_i are tokenized into sequences of separate tokens, represented by $\{t_{i,1}, t_{i,2}, \ldots, t_{i,L_i}\}$ and $\{r_{i,1}, r_{i,2}, \ldots, r_{i,M_i}\}$, respectively.
  2. SciBERT Embedding Generation: the token sequences of the titles T_i and abstracts R_i are processed by BERT, which produces a vector of embeddings for each document. These embeddings capture the semantics of the texts, reflecting the main topics and contextual relations.
  3. Graph Embedding: the embeddings resulting from SciBERT are used as nodes features that represent the publications in the heterogeneous graph.

  In our algorithm, SciBERT performs the embedding extraction on the titles and abstracts of the documents. These embeddings are used as node features in the heterogeneous graph, allowing the subsequent GCN to use these representations to analyze the interactions between publications, authors, and words, including co-authorship relationships. An edge index represents sparsely the connections between nodes. This format allows the GCN to process large heterogeneous networks while maintaining essential connectivity information among entities.

3.3. Learning Using GCN

After extracting embeddings with SciBERT and constructing the heterogeneous graph, the titles and abstract embeddings are used as features of the nodes in the network. The propagation operations in the GCN layers use these embeddings to compute representations of neighboring nodes. The need to apply a GCN model to a heterogeneous network, instead of other traditional deep learning techniques arises from the particularities of networks, where relationships between different types of nodes and graph structures must be captured effectively.

In the GCN step, this proposed hybrid method initially processes the textual data to create a vocabulary with a feature matrix, where each row corresponds to the embedding of a node, such as documents, authors, or keywords. The edge index represents the connections between nodes, preserving the essential relationships in the heterogeneous network.

To capture local and global dependencies within the heterogeneous network data, each layer of the GCN updates the node representations based on the connections and features defined by the edge index. The proposed GCN model uses activation functions to introduce nonlinearities in the model. In our work, we use the ReLU function ($\sigma(x) = \max(0, x)$) at different stages of GCN (widely used to mitigate the vanishing gradient problem). GCN training is performed by minimizing the MSE loss function, defined as $\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{Z}_i - \mathbf{X}_i)^2$, where N is the number of nodes in the network, Z_i the final GCN output for node i, and X_i the original feature vector of node i. The Adam optimizer [Kingma 2014] was used to adjusts the model's weights, adjusting an initial learning rate as needed.

Finally, GCN produces embeddings of the nodes in the network, represented as low-dimensional vectors that capture both the nodes' initial features and the network structure. These embeddings are used for subsequent tasks, such as agglomerative hierarchical clustering, which will be performed in the next step.

3.4. Generating Hierarchical Agglomerative Clustering

The disambiguated authors' clustering results are generated based on their representations in the heterogeneous network. The goal is to group documents with similar characteristics the interactions between publications, authors, and keywords using the GHAC method [Qiao et al. 2019]. The GHAC is an agglomerative hierarchical clustering algorithm that integrates network structural information considering the average similarity of the embeddings between the connected nodes. The algorithm is suitable for complex heterogeneous networks as the one built from the embeddings generated by GCN.

Initially, each document is an individual cluster. The iterative algorithm proceeds to merge the clusters with the highest average similarity between their components until reaching the desired number of clusters. The similarity between the two clusters is defined based on the normalized inner products of the node embeddings, allowing GHAC to capture the semantic and contextual data relationships.

Documents are grouped to maximize the internal cohesion of the clusters while preserving the semantic and structural interaction characteristics between the different types of entities in the network. This method not only groups documents based on local similarities but also considers the global context of the heterogeneous network, making it particularly effective in organizing complex academic networks and identifying underlying co-authorship and scientific collaboration patterns.

4. Experiments

To validate the hybrid method, we conducted experiments comparing to the multi-layer approach with clustering techniques of [Waqas and Qadir 2021] as a baseline, using the public data set *CustAND*,[3] which is composed of 14 ambiguous name groups with 137

[3] `https://github.com/humaira699/CustAND_Full.git`

distinct authors and 7,886 documents [Waqas and Qadir 2022]. This dataset is valuable for AND studies with various attributes and complex data relationships. The execution pipeline and the code for implementing this method are available in the repository.[4]

4.1. Experimental Setup

We used the document titles and abstracts to extract embeddings with SciBERT. We then concatenated these features to form the input text tokenized using the BERT tokenizer limited to 512 tokens. The output was a 768-dimensional embedding representing each document. The empirically defined GCN configuration includes three layers with an embedding size of 768, ReLU activation function, Mean Squared Error (MSE) loss function, and the optimization performed with a 0.001 learning rate for the Adam algorithm. We executed the training for 200 epochs with a batch size of 128. Python language was used to execute the experiments in a Google Colab L4 environment with the hardware accelerator L4, GPU with 22.5 GB of RAM, CPU with 53 GB of RAM, 201.2 GB disk, and the runtime type configured for Python 3.

4.2. Evaluation Metrics

The precision, recall, F1-measure, and specific metrics for clustering, such as Average Cluster Purity (ACP), Average Author Purity (AAP), and K-Metric metrics commonly presented in the AND literature are used to evaluate the experimental results.[5]

Precision measures the proportion of correctly classified documents relative to the total number of author documents, assessing the algorithm's ability to assign documents to authors correctly as $\text{Precision} = \frac{\text{Documents Correctly Classified}}{\text{Total Documents Classified}}$. Recall evaluates the ability of the algorithm to retrieve all documents from a real author, measuring the retrieval capacity of the algorithm about real authors as $\text{Recall} = \frac{\text{Documents Correctly Retrieved}}{\text{Total Documents from Real Author}}$. F1-measure is the harmonic mean of precision and recall, providing a balanced metric between these metrics (general performance metric) as $\text{F1-measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$.

ACP evaluates the average purity of the clusters generated by the algorithm about the theoretical clusters. ACP measures how well the documents were grouped into clusters that represent real authors as $\text{ACP} = \frac{1}{N} \sum_{i=1}^{q} \sum_{j=1}^{R} \frac{n_{ij}^2}{n_i}$, where N is the total size of the publication/paper records in the test set, q is the number of hybrid method/predicted clusters, R is the number of manually generated reference/real clusters, n_{ij} is the number of elements in common between the hybrid method-predicted clusters i and the reference clusters j, and n_j is the number of elements in the reference cluster j. The purer the clusters, the higher the ACP value. AAP measures how fragmented or cohesive the clusters predicted by the algorithm are relative to the reference clusters. A higher AAP indicates that the clusters are less fragmented, as $\text{AAP} = \frac{1}{N} \sum_{j=1}^{R} \sum_{i=1}^{q} \frac{n_{ij}^2}{n_j}$.

K-metric determines the trade-off between the average purity of clusters (ACP) and the average purity of authors (AAP). It is a metric that provides a single measure that considers both the quality of clusters and the quality of document attribution to real authors, as $\text{K-metric} = \sqrt{\text{ACP} \times \text{AAP}}$. K-metric helps evaluate the overall performance

[4] `https://github.com/natansr/adan_hybrid_method.git`

[5] Cluster purity measures how well the items in a cluster belong to the same real class. For AND it reflects the authorship records belonging to a single author within a cluster. A higher purity indicates a more homogeneous cluster where one is the ideal value [Ferreira et al. 2020].

of the disambiguation algorithm by balancing the quality of clusters and the quality of document attribution.

Figure 2 presents an illustrative example with geometric figures corresponding to an authorship record, where equal figures represent the same author. There are three theoretical clusters and four empirical ones, with one empirical cluster not pure and two authorship circle records fragmented across two clusters. The results of the metrics applied to this example, considering the ACP with the empirical clusters include in the first two clusters three author records ($\frac{3^2}{3}$), the third and fourth clusters two different authors ($\frac{1^2}{2}$), and the last cluster has a single record ($\frac{1^2}{1}$), the ACP is 0.888 ($\frac{1}{9} \times (\frac{3^2}{3} + \frac{3^2}{3} + \frac{1^2}{2} + \frac{1^2}{2} + \frac{1^2}{1})$). The AAP values numerators remain the same, but the denominators reflect the number of records in the theoretical clusters. For instance, $\frac{3^2}{4}$ represents three records from the same author in an empirical cluster out of four in the theoretical one. The final AAP value is 0.722 ($\frac{1}{9} \times (\frac{3^2}{4} + \frac{3^2}{3} + \frac{1^2}{4} + \frac{1^2}{2} + \frac{1^2}{2})$), and the K-metric is the geometric mean of ACP and AAP ($\sqrt{0.888 \times 0.722} = 0.8$). Precision is 0.857 considering the sum of three authorship record pairs from the same author in the first and second empirical clusters and none in the last three clusters. The denominator sums the total number of authorship record pairs from each empirical ($\frac{3+3+0+0+0}{3+3+1+0}$). Recall is 0.6 using the same Precision numerator with the denominator the sum of the authorship record pairs that refer to the same author in the theoretical clusters 6, 3, and 1 in the first, second, and third theoretical clusters, respectively ($\frac{3+3+0+0+0}{6+3+1}$). Finally, the F1-measure $= \frac{2\times(0.857\times0.6)}{0.857+0.6} = 0.7$.

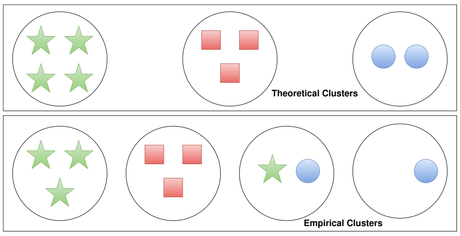

Figure 2. Theoretical and empirical clusters.

5. Results and Discussion

In this section, we present our hybrid method results with the evaluation metrics (Section 4.2) for the *CustAND* dataset with 14 groups of ambiguous names compared to the baseline work of [Waqas and Qadir 2021]. In the CustAND dataset, an example of an ambiguous name group for “A Choudhary” consists of 12 distinct authors that share the same name in document citation, namely “Ashish Choudhary”, “Amit Choudhary”, “Anil Choudhary”, “Arvind Choudhary”, “Anupam Choudhary”, “Ajay Choudhary”, “Abhishek Choudhary”, “Aniruddha Choudhary”, “Anjali Choudhary”, “Arjun Choudhary”, “Akshay Choudhary”, and “Arun Choudhary”. Table 1 summarizes the metrics for each ambiguous name group presenting average values for our method and the baseline.

Analysis of our method performance metrics for the 14 ambiguous name groups of the *CustAND* dataset reveals attractive results. Compared to the results reported by [Waqas and Qadir 2021], the average precision across the 14 groups is slightly lower (93.8% versus 94.6%), which may indicate a loss of precision when classifying documents for specific authors. However, the higher Recall (96.3% versus 92.5%) suggests that the method applied to ambiguous groups has a better recall capacity and is more efficient in identifying all documents of an author. The F1-measure of 95% across the 14 groups, compared to 93.5% for [Waqas and Qadir 2021], demonstrates that the method achieves a better balance between Precision and Recall.

The ACP and AAP metrics across the 14 groups also outperform the baseline with values of 96.5% and 97.4%, compared to 95.8% and 87%, respectively. These results suggest a higher average purity of the generated clusters and a lower fragmentation of the predicted clusters, reflecting a more cohesive and representative grouping of the real authors. Finally, the 96.9% K-metric in the 14 clusters of our method is significantly higher than the 91.24% reported by [Waqas and Qadir 2021], indicating that our method achieves a superior balance between the quality of the clusters and the correct attribution of documents to authors.

Table 1. Performance metrics by ambiguous name group.

Ambiguous Name Group	# Authors	Precision	Recall	F1-measure	ACP	AAP	K-metric
A Choudhary	12	1.000	1.000	1.000	1.000	1.000	1.000
J Martin	9	1.000	1.000	1.000	1.000	1.000	1.000
M A Qadir	15	1.000	1.000	1.000	1.000	1.000	1.000
J Mitchell	10	1.000	1.000	1.000	1.000	1.000	1.000
A Gupta	8	0.853	0.878	0.865	0.875	0.875	0.875
J Robinson	12	1.000	1.000	1.000	1.000	1.000	1.000
A Kumar	9	1.000	1.000	1.000	1.000	1.000	1.000
J Smith	12	0.938	0.988	0.964	0.972	0.972	0.972
Bin Li	8	0.592	0.671	0.632	0.763	0.889	0.826
S Kim	10	0.754	0.944	0.839	0.897	0.895	0.896
D Eppstein	3	1.000	1.000	1.000	1.000	1.000	1.000
Z Zhang	10	1.000	1.000	1.000	1.000	1.000	1.000
J Lee	8	1.000	1.000	1.000	1.000	1.000	1.000
K Tanaka	11	1.000	1.000	1.000	1.000	1.000	1.000
Baseline [Waqas and Qadir 2021]	137	**0.946**	0.925	0.935	0.958	0.870	0.912
Our Method	137	0.938	**0.963**	**0.950**	**0.965**	**0.974**	**0.969**

6. Conclusion

The main objective of this work was accomplished by proposing and evaluating the resolution capacity of the AND problem using a hybrid method that involves transfer learning with SciBERT, GCN, and GHAC. When comparing the effectiveness of our hybrid method with the state-of-the-art work of [Waqas and Qadir 2021], using the *CustAND* dataset, we note that the proposed method outperformed the baseline regarding average accuracy, considering five of six commonly used metrics of precision, recall, F1-measure, ACP, AAP, and K-metric.

Future experiments include comparison to [Pooja et al. 2022] including the use of other machine learning methods, diverse textual extract information methods, and the adoption of graph neural networks approaches, such as Graph Attention Network (GAT) and GraphSAGE with larger datasets. Also, a manageable data entry implementation for the end user as a graphical user interface to make the solution more user-friendly.

References

AMiner (2005-2024b). Search and mining of academic social networks. `https://www.aminer.org/`. Tsinghua University, Beijing, 100084. China.

AMiner (2024a). Aminer dataset. Disponível em `https://www.aminer.cn/data/?nav=openData`.

Beltagy, I., Cohan, A., and Lo, K. (2019). Scibert: A pretrained language model for scientific text. *CoRR*, abs/1903.10676.

CiteSeerX (2007-2019). Scientific literature digital library and search engine. `https://citeseerx.ist.psu.edu/index`. Pennsylvania State University, University Park, PA 16802, USA.

DBLP (1993-2024). The digital bibliography & library project. `https://dblp.uni-trier.de/`. Schloss Dagstuhl, Leibniz-Zentrum für Informatik, LZI GmbH.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. page 4171–4186, Minneapolis, Minnesota, USA. Proceedings of NAACL-HLT 2019, Association for Computational Linguistics.

Ferreira, A. A., Gonçalves, M. A., and Laender, A. H. F. (2020). Automatic disambiguation of author names in bibliographic repositories. *Synthesis Lectures on Information Concepts, Retrieval, and Services*, 12(1):1–146.

Hussain, I. and Asghar, S. (2017). A survey of author name disambiguation techniques: 2010-2016. *Knowledge Eng. Review*, 32:e22.

Kim, J. and Owen-Smith, J. (2020). Model reuse in machine learning for author name disambiguation: An exploration of transfer learning. *IEEE Access*, 8:188378–188389.

Kingma, D. P. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kipf, T. N. and Welling, M. (2017). Semi-supervised classification with graph convolutional networks. *International Conference on Learning Representations (ICLR)*.

Pooja, K. M., Mondal, S., and Chandra, J. (2022). Exploiting higher order multi-dimensional relationships with self-attention for author name disambiguation. *ACM Transactions on Knowledge Discovery from Data*, 16(5).

Qiao, Z., Du, Y., Fu, Y., Wang, P., and Zhou, Y. (2019). Unsupervised author disambiguation using heterogeneous graph convolutional network embedding. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 910–919.

Rodrigues, N. S., Mariano, A. M., and Ralha, C. G. (2024). Author name disambiguation literature review with consolidated meta-analytic approach. *International Journal on Digital Libraries*, pages 1–21.

Shin, D., Kim, T., Choi, J., and Kim, J. (2014). Author name disambiguation using a graph model with node splitting and merging based on bibliographic information. *Scientometrics*, 100(1):15–50.

Waqas, H. and Qadir, A. (2022). Completing features for author name disambiguation (AND): An empirical analysis. *Scientometrics*, 127(2):1039–1063.

Waqas, H. and Qadir, M. A. (2021). Multilayer heuristics based clustering framework (MHCF) for author name disambiguation. *Scientometrics*, 126(9):7637–7678.

Zhang, S., Tong, H., Xu, J., and Maciejewski, R. (2019). Graph convolutional networks: a comprehensive review. *Computational Social Networks*, 6(11).