A Knowledge Graph Embeddings based Approach for Author Name Disambiguation using Literals

Cristian Santini[1,2,3*], Genet Asefa Gesese[1,3*], Silvio Peroni[2], Aldo Gangemi[2], Harald Sack[1,3] and Mehwish Alam[1,3*]

[1]FIZ Karlsruhe – Leibniz Institute for Information Infrastructure, Karlsruhe, Germany.
[2]University of Bologna, Bologna, Italy.
[3]Karlsruhe Institute of Technology, Institute AIFB, Karlsruhe, Germany.

*Corresponding author(s). E-mail(s): cristian.santini@fiz-karlsruhe.de; genet-asefa.gesese@fiz-karlsruhe.de; mehwish.alam@kit.edu;
Contributing authors: silvio.peroni@unibo.it; aldo.gangemi@unibo.it; harald.sack@kit.edu;

Abstract

Scholarly data is growing continuously containing information about the articles from plethora of venues including conferences, journals, etc. Many initiatives have been taken to make scholarly data available in the for of Knowledge Graphs (KGs). These efforts to standardize these data and make them accessible have also lead to many challenges such as exploration of scholarly articles, ambiguous authors, etc. This study more specifically targets the problem of Author Name Disambiguation (AND) on Scholarly KGs and presents a novel framework, Literally Author Name Disambiguation (LAND), which utilizes Knowledge Graph Embeddings (KGEs) using multimodal literal information generated from these KGs. This framework is based on three components: 1) Multimodal KGEs, 2) A blocking procedure, and finally, 3) Hierarchical Agglomerative Clustering. Extensive experiments have been conducted on two newly created KGs: (i) KG containing information from Scientometrics Journal from 1978 onwards (OC-782K), and (ii) a KG extracted from a well-known benchmark for AND provided by AMiner (AMiner-534K). The results show that our proposed architecture outperforms our baselines of 8-14% in terms of F_1 score and shows competitive performances on a challenging benchmark such as AMiner. The code and the datasets are publicly available through Github (https://github.com/sntcristian/and-kge) and Zenodo (https://zenodo.org/record/5675787#.YcCJzL3MJTY) respectively.

Keywords: Author Name Disambiguation, Bibliographic Data, Citation Data, Clustering, Knowledge Graph Embeddings, Open Citations

1 Introduction

Data available in scholarly knowledge graphs (SKGs) – i.e., "a graph of data intended to accumulate and convey knowledge of the real world, whose nodes represent entities of interest and whose edges represent potentially different relations between these entities" [1] – is growing continuously every day, leading to a plethora of challenges concerning, for instance, article exploration and visualization [2], article recommendation [3], citation recommendation [4], and Author Name Disambiguation (AND) [5], which is relevant for the purposes of the present article. In particular, AND refers to a specific task of entity resolution which aims at resolving author mentions in bibliographic references to real-world people.

Author persistent identifiers, such as ORCIDs and VIAFs, simplify the AND activity since such identifiers can be used for reconciling entities defined as different objects and representing the same real-world person. However, the availability of such persistent identifiers in SKGs – such as OpenCitations (OC) [6], AMiner [7] and Microsoft Academic Knowledge Graph (MAKG) [8] – is characterized by very low coverage and, as such, additional and computationally-oriented techniques must be adopted to identify different authors as the same person.

In the past, many automatic approaches have been developed to automatically address AND by using publications metadata (e.g., title, abstract, keywords, venue, affiliation, etc.) to extract some features which can be used in the disambiguation task. These methods vary widely from supervised learning methods to unsupervised learning including recently developed deep neural network-based architectures [9]. However, the existing SKGs do not provide all the relevant contextual information necessary to reuse effectively and efficiently such approaches, that often rely on pure textual data.

In contrast with the approaches mentioned above, this study focuses on performing AND for scholarly data represented as linked data or included in SKGs by considering the multi-modal information available in such collections, i.e., the structural information consisting of entities and relations between them as well as text or numeric values associated with the authors and publications defined in the form of literals (family name, given name, publication title, venue title, year of publication, etc.). The proposed framework to address this task is named *Literally Author Name Disambiguation* (LAND), which focuses on tackling the following research questions:

- Can Knowledge Graph Embeddings (KGEs) – i.e. a technique that enables the creation of a "dense representation of the graph in a continuous, low-dimensional vector space that can then be used for machine learning tasks" [10] – be used effectively for the downstream task of clustering, more specifically for author name disambiguation?
- Does the information present in attributive triples (i.e. titles, publication dates, etc.) in existing SKGs enhance the aforementioned representations for AND?

The goal of this article is to provide a representation learning method for extracting entity features from SKGs which do not require any labeled training data. To this end, LAND uses semantic matching models which incorporate literal information, namely **LiteralE** [11], to extract author-related features which can adapt to the sparsity of metadata in SKGs. LAND further integrates KGEs along with Hierarchical Agglomerative Clustering (HAC) [12] and Blocking [13] where LAND architecture is particularly suited for data modeling with the topology of an SKG.

The rest of the article is organized as follows. Section 2 discusses the related studies in the field. Section 3 introduces the SKGs created for conducting our experiments. Section 4 details the proposed framework, while Section 5 documents the conducted experiments and the achieved results. Finally, Section 6 provides a summary of the work and gives some future perspectives.

2 Related Work

This section describes the studies related to author name disambiguation which are further divided into rule-based approaches, machine learning based approaches, and more specifically neural network based approaches. It also details the studies using KGEs for scholarly data.

2.1 Author Name Disambiguation

In [14] the authors classify existing AND approach into two categories, i.e., *author assignment* and *author grouping*. The author assignment approach directly assigns a label to every item corresponding to the real-world author. This approach is often difficult to implement since it requires the actual list of authors to be known a priori. The second method, author grouping, consists in clustering the entries corresponding to authors via a similarity function which should produce output groups associated with real-world entities. Author grouping may not require the number of authors to be known *a priori* and is consequently easier to implement in most cases.

Moreover, in the aforementioned survey, the authors classify the evidence used according to three categories: 1) *web information* (e.g., information extracted from web pages), 2) *citation information* (i.e. metadata associated with publications), and 3) *implicit evidence*, such as topic modeling or graph embeddings [15]. Additionally, a common strategy in AND is to initially group author entries into subsets by looking at name compatibility, e.g., authors carrying the same last name are grouped and disambiguation is performed within each group. This activity is carried out to reduce the number of pairwise comparisons required by the task and is termed as *blocking* (for details see [13]). One of the simplest and most common approaches is to group authors by looking at the full last name and the first initial (hereafter, *LN-FI*) of the given name, therefore called *LN-FI blocking*.

2.1.1 Rule-based Approaches for AND

Rule-based methods adopt a predefined set of rules for considering if two publications belong to the same author or not. In [16], the authors propose a rule-based classifier which takes as input several attributes associated to a pair of publications (e.g., title, coauthors lists, referenced works, etc.) related to an ambiguous name and assigns a similarity score for each one of these attributes based on the overlapping information between two publications. Despite its simplicity, this method does not scale well, and its performance is often difficult to generalize on different domains. *GHOST* [15], is another rule-based method which adopts a graph-based approach. It constructs a co-authorship graph for each instance s related to a queried author name by collapsing all the co-authors with same name into one single node. The resulting graph contains all authors which are co-authors with s and all authors which have co-authored a paper with the co-authors of s. Then, the similarity between two instances of s is computed based on the number of valid paths and affinity propagation is used to group nodes into clusters. However, this method does not work for single-author papers and information contained in other metadata (e.g. publication titles, abstracts, or keywords) is not considered.

2.1.2 Machine Learning Based Approaches for AND

These approaches take into consideration several fields describing scholarly resources, such as title words, keywords, coauthors, venues, etc., and a classifier is trained in a supervised learning fashion to estimate the relevance of each of these features for author disambiguation. One of the seminal works in ML-based AND was *Author-ity* [17]. *Author-ity* makes use of LN-FI blocking to preliminarily split publications related to ambiguous author names into blocks; then, given a pair of publications p_1, p_2 corresponding to two author name instances s_1, s_2 respectively in a block, it constructs a multidimensional similarity profile $x(p_1, p_2)$, based on title, journal name, co-author names, MeSH, language, affiliation, email, and other name attributes. The similarity profile is the input feature of a classifier trained with Bayesian learning to estimate the probability of $x(p_1, p_2)$, given that p_1, p_2 are written by the same author or not. In the end, a maximum-likelihood based agglomerative clustering is used in order to group publications.

Another approach that makes use of supervised learning is *BEARD* [18]. This method adopts a phonetic-based blocking strategy to preliminarily group authors into blocks by looking at the phonetic representation of the normalized surname (e.g., “van der Waals, J. D.” → “Waals, J. D.”). Moreover, it associates a set of similarity features to each pair of author instances that are designed to be sensitive towards the ethnic group of the authors. Then, a classifier is trained on annotated data to learn a pairwise distance function using tree-based methods (i.e. Random Forest and Gradient Boosted Trees). Finally, author references are grouped using hierarchical agglomerative clustering. The novelty of this method is to introduce for the first time ethnicity-sensitive features to make author name disambiguation sensitive to the actual origin of authors. However, the impact of the phonetic-based blocking strategy is not adequately addressed.

2.1.3 Neural Networks based Approaches for AND

In [19], the authors propose an AND approach that works on anonymized graphs by using relational information learned via network embeddings. This method constructs three local graphs for a candidate set of documents: a person–person graph representing a collaboration between authors, a person–document graph representing the association between authors and bibliographic records, and a document–document similarity graph based on co-authorship relations. A representation learning model is proposed to embed the nodes of these graphs into a shared low-dimensional space by optimizing a joint objective function based on the pairwise ranking of similarity. The final results are generated by agglomerative hierarchical clustering. This method proposes a new representation learning framework that is particularly suited for downstream clustering tasks. However, since this approach is designed for anonymized graphs, it does not take into consideration many attributes of nodes rather than co-author sharing for computing document similarity.

In [9], the authors propose an AND method based on three components: a representation learning module which create embedding representations for each document by leveraging global information, a local-linkage learning framework which exploits shared information to refine the embeddings related to an ambiguous name a, and a recurrent neural network which estimates the number of clusters for each ambiguous name a. This model is by far the most complex among those surveyed and it outperformed previous models. However, this method requires labeled samples for the global learning framework and complex feature engineering.

2.2 Knowledge Graph Embeddings and Scholarly Data

Few studies have been made recently on the use of KGEs with an application on scholarly linked data. In [20], an entity retrieval system for the scholarly domain was proposed, combining information coming from textual embeddings and structural embeddings trained from the KG IOS Press LD Connect[1]. In this paper, the authors evaluate the quality of low-dimensional representations of papers and entities (i.e. authors, organizations, etc.) by extracting two benchmark datasets: 1) a benchmark dataset collected from Semantic Scholar in order to evaluate the semantic similarity of papers, and 2) a second benchmark dataset extracted from DBLP used in order to evaluate co-authorship recommendations based on KGEs. The authors extract paragraph vectors for representing papers' content by using doc2vec [21] and train TransE [22] for extracting embeddings of entities in the SKG of IOS LD Connect. In order to build the entity retrieval model, a logistic regression model which takes as input features both paragraph vectors and structural embeddings. It is trained on a dataset of similar papers automatically collected from Semantic Scholar. Reported results show that KGEs do not have a significant impact on paper similarity classification, whether textual embeddings alone achieve robust results. As a second step, a co-author inference evaluation is carried by using a benchmark dataset extracted from DBLP to demonstrate the ability of TransE for predicting coauthorship links based on the observed triples.

In [23], embeddings have been used to generate coauthorship recommendations on SKGs. One of the aims of this work is to propose a novel approach for training KGE models on SKGs where 1-to-N, N-to-1, and N-to-N relations are frequent (i.e. authorship relations or citation links). In order to address this issue, the authors present a reimplementation of TransE [22] and RotatE [24] by using a newly proposed loss function optimized for many-to-many relations, i.e. Soft Margin (SM) Loss. The results of their study show how the models equipped with SM loss outperform the original models. The novelty of this study is to propose a loss function that mitigates the adverse effects of false-negative sampling and to investigate the use of KGEs for co-authorship suggestions.

3 Creation of the Scholarly KGs

This section introduces the benchmark datasets **OC-782K** and **AMiner-534K** which are created for evaluating the LAND framework. OC-782K is a subset of the *Scientometrics* KG [25] which is built in compliance with the *OpenCitations Data Model (OCDM)* [26]. On the other hand, AMiner-534K is a KG generated from a well-established benchmark dataset[2] for AND made available by AMiner in [9].

3.1 The OC-782K Knowledge Graph

In this paper, the Scientometrics KG from [25] is referred to as *Scientometrics-OC*. This publicly available KG contains bibliographic information about the

[1] http://ld.iospress.nl/
[2] https://static.aminer.cn/misc/na-data-kdd18.zip

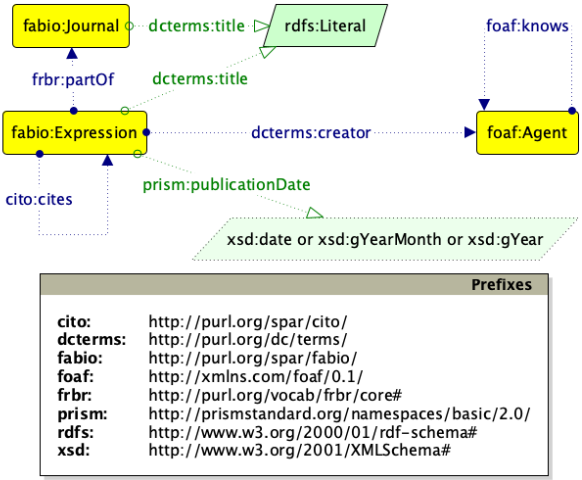


Fig. 1 A Graffoo diagram [27] describing the data Model used for OC-782K.

articles published by the journal *Scientometrics*[3] from 1978 to the present, along with bibliographic information of all the cited academic works. The dataset named *OC-782K* is created from *Scientometrics-OC* by modeling entities related to authors, publications, and venues with different data models suited for the task of AND.

This data model contains three types of entities: `fabio:Expression`, which represents articles, books, conference papers, and other academic works, `fabio:Journal` for representing journal venues (if the related `fabio:Expression` is a journal article), and authors which are described as `foaf:Agent`. The data model is an abstraction of the *OCDM* [26] and is created for two reasons: i) for collecting triples only related to the entities of interest (e.g. bibliographic resources, venues, and authors), ii) create an abstract representation of *Scientometrics-OC* in order to perform representation learning more efficiently. The data model of OC-782K is represented in *Figure 1*.

OC-782K is extracted from *Scientometrics-OC* by first collecting information about the bibliographic resources with at least a title and an author. Then, the publication dates and journal venues of these works (if available) were collected. A `foaf:knows` relation is added between two authors who have co-authored the same work, and the relation between two bibliographic resources, a citing expression and a cited one, is represented with the `cito:cites` relation.

The dataset consists of 781,917 triples, with 620,321 structural triples (i.e. triples with object relations). In the original *Scientometrics-OC*, while duplicate bibliographic resources and journals were merged by using the DOIs associated with each article, authors are not disambiguated (i.e., there is

[3] https://www.springer.com/journal/11192

Table 1 Number of entities and triples in OC-782K.

Object triples	Textual triples	Numeric triples	Entities
620,321	104,621	56,975	293,186

Table 2 Number of entities and relations counted by type in OC-782K.

Entities			Relations			
Publications	Venues	Authors	dc:creator	foaf:knows	cito:cites	frbr:partOf
57,266	47,355	188,565	188,565	253,942	128,738	49,076

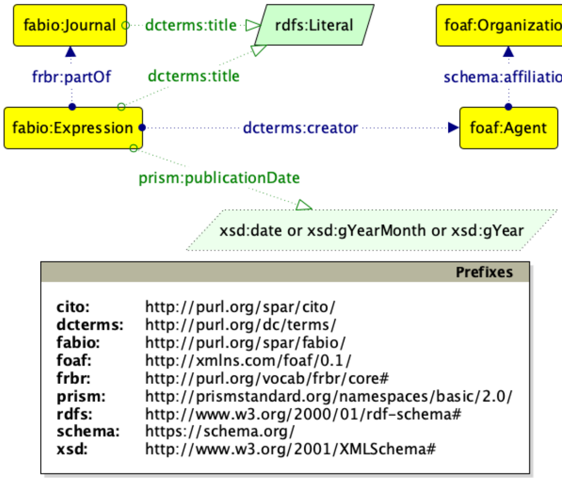


Fig. 2 A Graffoo diagram [27] describing the data model used for *AMiner-534K*.

one author for each `dcterms:creator` relation.) Statistics of the dataset are reported in *Table 1* and *Table 2*.

3.2 The AMiner-534K Knowledge Graph

In order to evaluate the generalizability of the proposed approach on a different dataset, a second scholarly KG named *AMiner-534K* is extracted from the AMiner AND benchmark dataset introduced in [9]. The AMiner benchmark for AND contains a sub-set of publications from AMiner and sampled from 100 ambiguous Asian names. This dataset contains the following information for each scholarly article: title, publication date, venue, keywords, abstract, authors, and affiliations. The AMiner-534K KG is created by extending the data model of OC-782K with the additional author affiliation information (using the property `schema:affiliation`). A representation of the data model is available in *Figure 2*. However, for AMiner-534K the `cito:cites` and the `foaf:knows` properties are absent since these properties are not present in the original benchmark.

Table 3 Number of entities and triples in AMiner-534K.

Object triples	Textual triples	Numeric triples	Entities
428,473	70,046	35,021	179,377

Table 4 Number of entities and relations counted by type in AMiner-534K.

Entities			
Publications	Venues	Authors	Organizations
35,023	5,889	110,837	27,628
Relations			
dc:creator	schema:affiliation		frbr:partOf
197,249	196,201		35,023

Statistics of the dataset are reported in *Table 3* and *Table 4*. As for the previous dataset, the extracted files are available on Zenodo at https://doi.org/10.5281/zenodo.5675801 [28] in order to allow the reproducibility of the studies herein described.

4 Literally Author Name Disambiguation (LAND)

In this section, the different components of the proposed framework, Literally Author Name Disambiguation (LAND), are described in detail. Figure 4 shows the overall architecture of the approach which is based on three main components:

- **Multimodal KG Embeddings.** This strategy is aimed at learning representative features of entities and relations in a KG by taking into account the structure of the graph itself along with the semantics contained in the literals about these entities (e.g., titles of academic works or publication dates).
- **Blocking.** This strategy is used to reduce the number of pairwise comparisons required by the AND task by initially grouping authors into blocks characterized by name similarity, so that disambiguation is carried within each block. LAND uses a rather simple but effective blocking strategy called LN-FI blocking.
- **Clustering.** Hierarchical Agglomerative Clustering (HAC) is used to group the embeddings associated with each author to be disambiguated into k-clusters by using vector-based similarity measure along with a distance threshold.

The output of these components is then used for refining the original KG. In the following, each of these components is discussed in detail.

4.1 Multimodal Knowledge Graph Embeddings

The first step of the LAND framework is to learn the latent representation of the KGs described in Section 3 including the representations of the authors.

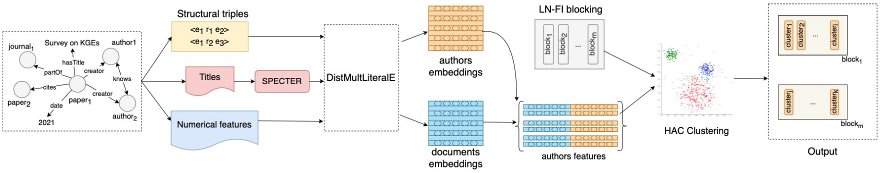


Fig. 3 The overview of the LAND architecture

To this end, the Multimodal KGEs component of LAND is designed to learn embeddings of entities and relationships in a KG by combining the structural information and literals associated with the entities such as a string or a date value. LAND adopts LiteralE [11] embedding model in this component to learn the KGEs. It incorporates literal information into entity representations by using a learnable mapping function where the literals can either be numeric or text. More specifically, LiteralE is a multimodal extension of *semantic matching models* for learning KGEs, such as DistMult [29]. DistMult scores each triple in the KG with a simple bilinear transformation $f(h, r, t) = \mathbf{h}^T diag(\mathbf{r})\mathbf{t}$. Meanwhile, LiteralE aims to modify the scoring function f by enhancing the entity embeddings with the information coming from literal values. At the core of this method is the mapping function $g : R^h \times R^d \rightarrow R^h$ which takes as input an entity embedding $\mathbf{e} \in R^h$ and a literal vector $\mathbf{l} \in R^d$ and maps them to a new embedding of the same dimension as the entity embedding.

LAND makes use of SPECTER [30], a pre-trained BERT language model for scientific documents in order to encode the textual attributes of the entities (e.g., publication titles) in the vector space R^d before incorporating them into entity vectors with the g function. Each title in our scholarly KG is mapped to a 768-dimensional sentence embedding by utilizing this model. Meanwhile, the numeric datatypes such as *xsd:gYear* are converted to a literal vector as described in LiteralE.

In this study, the following two varieties of the DistMultLiteralE model are used and compared against their corresponding base (unimodal) model *DistMult*.

- **DistMultLiteralE-**g_{lin}. This architecture incorporates textual embeddings extracted from the titles of the entities (scholarly articles) into their representations by means of a linear transformation defined as follows:

$$g_{lin}(\mathbf{e}, \mathbf{l}) = \mathbf{W}[\mathbf{e}, \mathbf{l}],$$

  where $\mathbf{e} \in R^h$ is the vector associated to the ith entity in a KG, $\mathbf{l} \in R^d$ is the title embedding, $\mathbf{W} \in R^{(h,d+h)}$ is a linear transformation matrix and $[\mathbf{e}, \mathbf{l}] \in R^{(h+d)}$ is the concatenation vector of the entity embedding $\mathbf{e}$ and the literal embedding $\mathbf{l}$.

- **DistMultLiteralE-**g_{gru}. The goal here is to leverage both text (titles) and numeric literals (publication dates). This architecture combines the information coming from numeric and textual literals into the entity representations by means of a Gated Recurrent Unit (GRU), defined as follows:

$$g_{gru}(\mathbf{e}, \mathbf{l}, \mathbf{n}) = \mathrm{z} \circ \mathrm{h} + (1 - \mathrm{z}) \circ \mathbf{e}$$
$$\mathrm{z} = \sigma(\mathbf{W}_{ze}\mathbf{e} + \mathbf{W}_{zl}\mathbf{l} + \mathbf{W}_{zn}\mathbf{n} + \ \mathbf{b})$$
$$\mathrm{h} = h(\mathbf{W}_h[\mathbf{e}, \mathbf{l}, \mathbf{n}]),$$

  where $\circ$ is the element-wise multiplication, $\sigma(\cdot)$ is the sigmoid function, $\mathbf{W}_{ze} \in R^{(h,h)}, \mathbf{W}_{zl} \in R^{(h,h+d)}, \mathbf{W}_{zn} \in R^h$ and $\mathbf{W}_h \in R^{(h,h+d+1)}$ are linear transformation matrices, $\mathbf{b}$ is a bias vector, $h(\cdot)$ is a component-wise nonlinearity (e.g. the hyperbolic tangent) and $[\mathbf{e}, \mathbf{l}, \mathbf{n}]$ is the concatenation of the entity vector, the textual vector and the numeric literal value.

Finally, after having each model trained on a given KG, every author A's embedding $\mathbf{E}$ is modified by concatenating it with the embedding $\mathbf{D}$ of the document D (i.e. scholarly article) associated to the author A, in order to obtain feature $\mathbf{F}$ where $\mathbf{F} = \mathbf{E} + \mathbf{D}$. This is carried out to reflect both the structural information of the two entities (the author and the document) and the literal information present in the document attributes (i.e. title and publication date) in the embedding of the author.

4.2 Blocking

Blocking is a strategy that is widely used in AND systems. A comparative analysis of these approaches is already discussed in [13]. The idea is to split the set of features F related to authors into separate groups, also called *blocks*, $F_{b_1}, F_{b_2}, \ldots, F_{b_n}$, each one associated with an ambiguous name, so that AND is carried out independently within these blocks. This leads to the reduction in the computational complexity of the disambiguation algorithm typically from a pairwise comparison among all the author features in F to a pairwise comparison among the features in each block. Mathematically, the complexity gets reduced from $O(\|F\|^2)$ to $O(\sum_{i=1}^{n} \|F_{b_i}\|^2)$.

LAND uses a common blocking technique **LN-FI** (Last Name First Initial). LN-FI blocking divides the set of author features into blocks by looking at the full last name and the first initial of the given name of each author. This blocking technique is chosen since it's computationally less expensive than other blocking approaches which are based on distance measures or string normalization and it's also compliant with the way publishers often mention author names in publications' metadata. Moreover, LN-FI creates high recall blocks and thus allows for a higher number of pairwise comparisons among author features if compared with other methods [13].

In order to implement the blocking procedure, first, the list of authors is extracted and sorted according to the family name and given name. Then, LN-FI blocking is applied to group the authors in this list into multiple sub-sets, each one containing authors with the same last name and first name initial.

Because of how LN-FI works, each block has a lower limit of 2 members to be disambiguated. Moreover, due to the size of our dataset, no upper limit is given in the number of members belonging to each block.

4.3 Clustering

The clustering algorithm in LAND helps in grouping together the author features in each block F_{b_i} into k-clusters $\{C_1, ...C_k\}$ where all the features in C_j, where $j = 1, ..., k$, ideally belong to the same real-world author. The **Hierarchical Agglomerative Clustering (HAC)** approach [12] is used which builds clusters of features in a bottom-up manner. The approach conceives each embedding in a block as a singleton cluster and works by iteratively merging the two most similar clusters until all features have been merged in one final cluster.

In our implementation of HAC, similarity among clusters is computed with a *single linkage* strategy which, at each step, merges the clusters whose two closest members have the smallest distance, based on cosine similarity. In order to get the final clusters, a threshold on the maximum distance is defined and clusters above this threshold are considered to be corresponding to different authors. The threshold is defined globally over all the blocks by testing different values over an evaluation dataset and by trying to maximize precision to minimize false positives.

5 Experimental Results

This section discusses the empirical evaluation of the LAND framework. It first shows how the ground truth is generated for the task of AND, then it presents the achieved experimental results of LAND on the newly generated dataset OC-782K and on a KG extracted from a widely used AND benchmark, i.e. AMiner-534K (refer to Section 3 for more details). In addition, an error analysis is carried out for the results on OC-782K.

5.1 Generation of the Ground Truth

In order to obtain the ground truth for testing LAND on *OC-782K*, a list of $(author, ORCIDiD)$ pairs is extracted. This is performed for the purpose of having an evaluation dataset of scholarly articles labeled with a unique identifier associated with their real-world authors. In order to handle the unbalance in the dataset, only those authors whose last name and first initial are associated with at least two different ORCID iDs are considered. The final evaluation dataset contains 630 bibliographic works organized into 184 blocks and 497 different ORCID iDs.

For measuring the generalizability of the proposed approach, another manually-labeled benchmark dataset is used, i.e., AMiner-534K. This evaluation dataset is larger than the one extracted for OC-782K, with 35,023 scholarly articles and 6,395 unique authors. As for the previous dataset, each

Table 5 Hyper-parameter ranges for the HPO studies.

Hyper-parameters	Ranges
Embedding dimension	128, 256, 512
Learning rate (log scale)	[0.0001, 0.01)
Number of negatives per triple (log scale)	[1, 50)
Batch size	128, 256, 512
Smoothing coefficient (log scale)	[0.001, 1.0)

ambiguous name is considered as a block and disambiguation is performed within each block.

5.2 Experimental Setup

The performance of LAND is evaluated based on three variants of the KGE models: **DistMult**, **DistMultLiteralE-**g_{lin}, and **DistMultLiteralE-**g_{gru}. The first variant DistMult only considers the structural information and is used in order to have a baseline to measure the impact of literals. The second variant DistMultLiteralE-g_{lin} incorporates titles of papers into the representation learning. The third variant DistMultLiteralE-g_{gru} uses numeric attributes of the nodes (e.g., publication dates) along with titles. The implementation of the multimodal KGE models is made compatible with `PyKEEN (v.1.4.0)` [31]. The source code of different variants as discussed previously is available on Github[4]. The KGE models are trained and evaluated using Colab Pro notebooks with ≈ 24GB of RAM and Nvidia Tesla T4/K80 GPUs.

Two major tasks are involved in these experiments, i.e., i) an evaluation of LAND against a candidate set of authors associated with an ORCID iD in OC-782K and ii) a generalizability analysis of LAND on the benchmark dataset provided by AMiner, where LAND is compared to the SOTA models surveyed in [9]. Inspired by [9], the evaluation metrics **pairwise Precision**, **Recall**, and $\mathbf{F}_1$ are used. For studying the generalizability of LAND, these metrics are macro-averaged across all 100 test names.

5.3 Model Selection

The models are trained using the Binary Cross Entropy Loss function **BCE**, the **Adam** optimizer, the Stochastic Local Closed World Assumption **SLCWA** training approach, and **label smoothing** as a regularization technique. Note that for training, each KG is split with a ratio of 64% training, 16% validation, and 20% testing. Random search has been used to perform the hyper-parameter optimizations over the range of values given in *Table 5*. Each model is trained for a maximum of 1000 epochs and early stopping is applied to speed up the optimization process and avoid overfitting.

Note that due to limitation of resources, we ran the optimization study only for the unimodal model (i.e., DistMult) on both datasets and chose the set of optimal hyperparameter values which gave the best results, and then decided

[4]https://github.com/sntcristian/and-kge

Table 6 Rules to compute the similarity of two pairs of publications for the baseline Score Pairs. This table is a subset of the rules originally introduced in [16]

Field	Criterion	Score
Shared words in titles	1 / 2 / > 2	3 / 5 / 8
Shared coauthors	1 / 2 / > 2	4 / 7 / 10
Journal	Exact match	6
Shared cited works	1 / 2 / 3 / 4 / > 4	2 / 3 / 6 / 8 / 10
Self-citation	one publication citing the other	10

to apply them also for training the multimodal models. The optimal hyperparameters are as follows: for OC-782K, embedding dimension: 512, learning rate: 0.0003, number of negatives: 12, batch size: 512, smoothing coefficient: 0.001, epochs: 120; for AMiner-534K, embedding dimension: 128, learning rate: 0.0001, number of negatives: 32, batch size: 512, smoothing coefficient: 0.1, epochs: 300.

For HAC, we define the distance threshold for the final clusters experimentally by trying to find a trade-off between Precision and Recall. However, since high recall systems tend to group different authors together and this negatively affects the performances for AND, we decided to favor high precision over recall. For *OC-782K*, the resulting best threshold is 0.6, while for *AMiner-534K* it is 0.26.

5.4 Baseline Methods

To better assess the performances of the LAND framework, two baseline methods are implemented: (i) a rule-based method originally proposed in [16], which assigns a pairwise score of similarity to two publications based on several rules; and (ii) a simple disambiguation algorithm based on blocking and clustering of sentence embeddings extracted from titles. The rule-based method is inspired by [16], hereby mentioned as *Score Pairs*, classifies if two publications belong to the same author or not by looking at several features (e.g., shared words in title, co-authors, citations, etc.) and computes an affinity score for each one of these features based on a list of criteria, i.e., exact string matching or number of co-occurrences. A list of the features compared, along with the respective comparison criteria and scores are reported in *Table 6*. Then, a threshold on the sum of the affinity scores is chosen in order to decide whether the publications, given the similarity of their attributes, belong to the same author or not. In our experiments, the value of the threshold is 10.

The second baseline *Title Similarity* is chosen to estimate the representativeness of textual embeddings for the task of AND. This baseline performs HAC on the title embeddings encoded by the SPECTER language model [30]. It is implemented as follows: single linkage as linkage method, cosine similarity as affinity measure, and a threshold of 0.18. As for the architecture using KGEs, the threshold for clustering is selected by maximizing the F_1 score while favoring Precision over Recall.

Table 7 Results of AND on OC-782K. Best results are highlighted in bold.

Model	Precision	Recall	F_1
Score Pairs	84.66	50.20	63.03
Title Similarity	71.56	66.64	69.02
DistMult+HAC	**91.71**	67.11	**77.50**
DistMultLiteralE-glin+HAC	89.63	66.98	76.67
DistMultLiteralE-ggru+HAC	82.76	**67.59**	74.40

Table 8 Confusion matrix of DistMult+HAC on OC-782K with high precision setup.

	Positive label	Negative label	Total
Positive Classification	996	90	1086
Negative Classification	488	1582	2030
Total	1484	1672	3110

5.5 Results

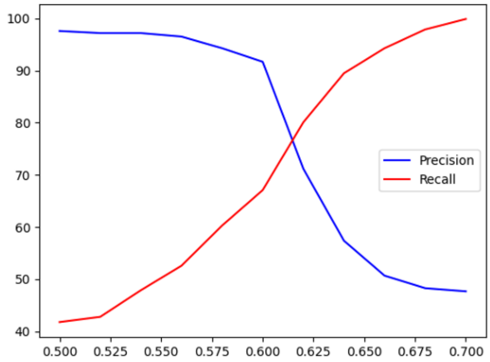
Evaluation on OC-782K

This section compares the results of different LAND variants, i.e., DistMult + HAC, DistMultLiteralE-g_{lin} + HAC and DistMultLiteralE-g_{gru} + HAC, on OC-782K with the two previously described baseline models.

Table 7 shows the results of the experiment. The embedding-based model outperforms the baseline methods except for the precision of DistMultLiteralE-g_{gru}+HAC. More precisely, there is an increment in the pairwise F_1 score of the best performing model DistMult+HAC, i.e., more than 14% and 8% as compared to the baselines *Score Pairs* and *Title Similarity* respectively. The best precision of **91.71** is obtained by DistMult+HAC. The best recall is **67.59** obtained by DistMultLiteralE-g_{gru}+HAC. However, the difference of the recall as compared to DistMult+HAC is marginal. Finally, the structural variant of LAND (DistMult+HAC) had the highest F_1 score of **77.50**. For the other multimodal models, improvements in the results are not significant, i.e., the precision of 89.63, recall of 66.98, and the F_1 score of 76.67 for the architecture which incorporates textual literals into the entity embeddings, and precision: 82.76, recall: 67.59 and F_1: 74.40 for the architecture which uses DistMult with textual and numerical embeddings. However, it's interesting to note that the model which uses textual and numerical information, i.e., DistMultLiteralE-g_{gru} has the highest recall; besides that, the low F_1 score achieved by this model suggests the negligible influence of multimodal information for this dataset.

As it is noticeable in *Table 8*, the performances of DistMult+HAC with respect to recall are far from being optimal, since our models ignored a relevant number of matching authors (> 30% avg.) in the evaluation dataset. However, we decided to avoid higher thresholds in order to reduce the number of false positives produced by our clustering algorithm and, as a consequence, to avoid attributing papers written by different persons to the same author. A plot of Precision and Recall curves for OC-782K is available in the *Figure 4*.

By applying DistMult+HAC to the whole set of authors in OC-782K with the high precision setup, we are able to reduce the author entities from 188,565



A p

Fig. 4 Plot of the precision and recall curves of our best AND system on different distance thresholds.

Table 9 Results of author name disambiguation for the AMiner benchmark [9]. Best results are reported in bold and the underlined results show the values in which our models showed competitive performances.

Model	Precision	Recall	F_1
GHOST [15]	**81.62**	40.43	50.23
BEARD [18]	57.09	77.22	63.10
Zhang and Al Hasan, 2017 [19]	70.63	59.53	62.81
Zhang et al., 2018 [9]	77.96	63.03	**67.79**
DistMult+HAC	<u>78.36</u>	59.68	63.36
DistMultLiteralE-g_{lin}+HAC	77.24	61.21	<u>64.18</u>
DistMultLiteralE-g_{gru}+HAC	77.62	59.91	63.07

to 135,325 (a reduction of more than 28%). This shows how relevant KGEs can be for AND on SKGs and how they can be effective in removing duplicates.

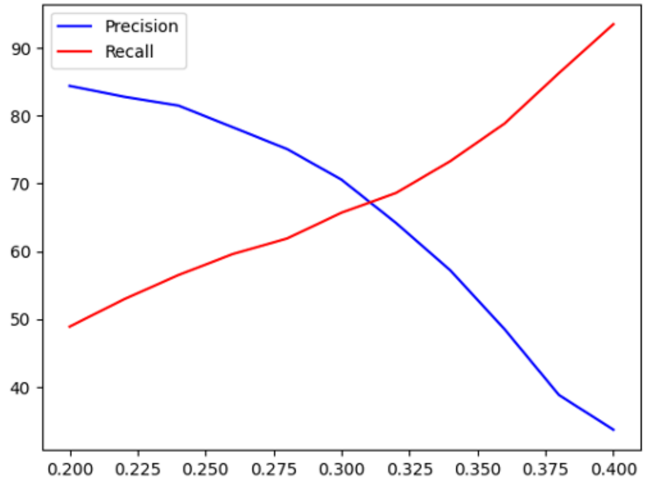
Evaluation on AMiner Dataset

We tested the generalizability of our approach on a newly collected KG extracted from the AMiner benchmark dataset for AND [9]. The results of LAND are compared to the performances of SOTA AND models reported in the benchmark study in [9] (A description of these models is provided in *Section 2.1*). However, we have to state that this comparison is not fair, since our LAND architectures are trained on a KG, i.e. AMiner-534K, which contains less information than the training dataset used in the original benchmark study.

Despite the unfairness in the comparison, two out of three of our LAND variants, i.e. DistMult+HAC and DistMult-LiteralE-g_{lin}, achieve the second and third best F_1 score. Moreover, our architecture DistMult+HAC achieves the second best precision score, only outperformed by GHOST [15], which in turn has a comparably lower recall than our models.

Another interesting fact is that, for this dataset, the multimodal model DistMultLiteralE-g_{lin}+HAC performes better than the unimodal model in the

Fig. 5 Plot of the precision and recall curves of DistMult+HAC on AMiner.



recall, while keeping slightly lower levels of precision. This shows that, for this KG, integrating textual literals did enhance the model performances by allowing to find more matching authors. However, it is to be noticed that as for OC-782K, DistMult-LiteralE-g_{gru}+HAC receives the lower scores among the KGE-based architectures, thus allowing us to infer the neglectable influence that numeric features bring for AND. For this dataset, we adopted as configuration for HAC single linkage, cosine similarity, and a distance threshold of 0.26. A plot of precision and recall curves is available in *Figure 5*.

By comparing our results with those of the other SOTA models on the AMiner benchmark dataset, we showed that LAND achieves competitive performances on large-scale author name disambiguation, only being outperformed by more complex models such as *Zhang et al.* [9].

5.6 Error Analysis

We randomly sampled a subset of 50 wrongly matched pairs (i.e. false positives) from the disambiguated OC-782K in order to analyze the most frequent errors produced by our AND system. We found out that most of the wrong matches are related to Asian authors with common surnames and first initials, like "Chen B", "Kim S", "Li Y", "Wang J", "Li J", "Hu Z" and "Chen J". This is probably due to the fact that LN-FI blocking tends to create huge blocks for very frequent surnames and this causes wrong authors to slide inside the final clusters, especially when they share some features (like references or publishing venue). However, we found out that it is possible to remove all these errors by using a post-blocking strategy which poses the condition that $fullname_i = fullname_j$ before merging two authors. Indeed, we found out that all the wrongly matched pairs in our sample which share the same full name are the same person and their entities are wrongly labeled due to the fact that they used multiple ORCIDs across different scholarly works.

6 Summary & Future Perspectives

This article has introduced a framework, named LAND, to perform Author Name Disambiguation (AND) for scholarly data represented as linked data or included in SKGs by developing KGE models based on relationships between entities and the related literal information associated to them. We have demonstrated that these models can be used in the downstream task of clustering for AND effectively. The proposed framework outperforms state-of-the-art methods on a newly created benchmark dataset defining a SKG (named OC-782K) compliant with the OpenCitations Data Model (OCDM) as well as another SKG (named AMiner-534K) created using an existing benchmark dataset, i.e., AMiner. Our method is able to maintain competitive levels of precision, recall and F_1 even when dealing with more complex models. Moreover, LAND is designed for dealing with data within knowledge graphs.

In future, we plan to extend our approach to include also author collaboration network information along with the topic of interest/area of expertise extracted by processing author's publications via deep learning approaches. Having such additional data will allow us to test if they can improve the results for the task of author name disambiguation.

Acknowledgments. This study was partially funded by the "Scholarship for research periods abroad aimed at the preparation of the master thesis" by the department of Classical Philology and Italian Studies, University of Bologna (https://ficlit.unibo.it/it).

References

[1] Hogan, A., Blomqvist, E., Cochez, M., d'Amato, C., de Melo, G., Gutiérrez, C., Kirrane, S., Gayo, J.E.L., Navigli, R., Neumaier, S., Ngomo, A.N., Polleres, A., Rashid, S.M., Rula, A., Schmelzeisen, L., Sequeda, J.F., Staab, S., Zimmermann, A.: Knowledge graphs. ACM Comput. Surv. **54**(4), 71–17137 (2021). https://doi.org/10.1145/3447772

[2] Liu, J., Tang, T., Wang, W., Xu, B., Kong, X., Xia, F.: A Survey of Scholarly Data Visualization. IEEE Access **6**, 19205–19221 (2018). https://doi.org/10.1109/ACCESS.2018.2815030. Conference Name: IEEE Access

[3] Beel, J., Gipp, B., Langer, S., Breitinger, C.: Research-paper recommender systems: a literature survey. International Journal on Digital Libraries **17**(4), 305–338 (2016). https://doi.org/10.1007/s00799-015-0156-0. Accessed 2021-12-03

[4] Färber, M., Jatowt, A.: Citation recommendation: approaches and datasets. International Journal on Digital Libraries **21**(4), 375–405 (2020). https://doi.org/10.1007/s00799-020-00288-2. Accessed 2021-12-03

[5] Sanyal, D.K., Bhowmick, P.K., Das, P.P.: A review of author name disambiguation techniques for the PubMed bibliographic database. Journal of Information Science **47**(2), 227–254 (2021). https://doi.org/10.1177/0165551519888605. Accessed 2021-07-30

[6] Peroni, S., Shotton, D.: OpenCitations, an infrastructure organization for open scholarship. Quantitative Science Studies **1**(1), 428–444 (2020). https://doi.org/10.1162/qss_a_00023. Accessed 2021-08-26

[7] Wan, H., Zhang, Y., Zhang, J., Tang, J.: AMiner: Search and Mining of Academic Social Networks. Data Intelligence **1**(1), 58–76 (2019). https://doi.org/10.1162/dint_a_00006. Accessed 2021-12-03

[8] Färber, M.: The Microsoft Academic Knowledge Graph: A Linked Data Source with 8 Billion Triples of Scholarly Data. In: Ghidini, C., Hartig, O., Maleshkova, M., Svátek, V., Cruz, I., Hogan, A., Song, J., Lefrançois, M., Gandon, F. (eds.) The Semantic Web – ISWC 2019. Lecture Notes in Computer Science, pp. 113–129. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-30796-7_8

[9] Zhang, Y., Zhang, F., Yao, P., Tang, J.: Name Disambiguation in AMiner: Clustering, Maintenance, and Human in the Loop. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1002–1011. ACM, London United Kingdom (2018). https://doi.org/10.1145/3219819.3219859. https://dl.acm.org/doi/10.1145/3219819.3219859 Accessed 2021-08-13

[10] Gesese, G.A., Biswas, R., Alam, M., Sack, H.: A survey on knowledge graph embeddings with literals: Which model links better literal-ly? Semantic Web **12**(4), 617–647 (2021). https://doi.org/10.3233/SW-200404. Accessed 2021-07-28

[11] Kristiadi, A., Khan, M.A., Lukovnikov, D., Lehmann, J., Fischer, A.: Incorporating Literals into Knowledge Graph Embeddings. arXiv:1802.00934 [cs, stat] (2019). arXiv: 1802.00934. Accessed 2021-07-28

[12] Ward, J.H.: Hierarchical Grouping to Optimize an Objective Function. Journal of the American Statistical Association **58**(301), 236–244 (1963). https://doi.org/10.1080/01621459.1963.10500845. Publisher: Taylor & Francis _eprint: https://www.tandfonline.com/doi/pdf/10.1080/01621459.1963.10500845. Accessed 2021-09-11

[13] Backes, T.: The Impact of Name-Matching and Blocking on Author Disambiguation. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, pp. 803–812. ACM, Torino Italy (2018). https://doi.org/10.1145/3269206.3271699. https://dl.acm.org/doi/10.1145/3269206.3271699 Accessed 2021-07-28

[14] Ferreira, A.A., Gonçalves, M.A., Laender, A.H.F.: A brief survey of automatic methods for author name disambiguation. ACM SIGMOD Record **41**(2), 15–26 (2012). https://doi.org/10.1145/2350036.2350040. Accessed 2021-07-30

[15] Fan, X., Wang, J., Pu, X., Zhou, L., Lv, B.: On Graph-Based Name Disambiguation. Journal of Data and Information Quality **2**(2), 1–23 (2011). https://doi.org/10.1145/1891879.1891883. Accessed 2021-07-31

[16] Caron, E., van Eck, N.-J.: Large scale author name disambiguation using rule-based scoring and clustering: International conference on science and technology indicators. Proceedings of the Science and Technology Indicators Conference 2014, 79–86 (2014). Place: Leiden Publisher: Universiteit Leiden. Accessed 2021-09-17

[17] Torvik, V.I., Smalheiser, N.R.: Author Name Disambiguation in MEDLINE. ACM transactions on knowledge discovery from data **3**(3), 11 (2009). https://doi.org/10.1145/1552303.1552304

[18] Louppe, G., Al-Natsheh, H.T., Susik, M., Maguire, E.J.: Ethnicity Sensitive Author Disambiguation Using Semi-supervised Learning. In: Ngonga Ngomo, A.-C., Křemen, P. (eds.) Knowledge Engineering and Semantic Web vol. 649, pp. 272–287. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-45880-9_21. Series Title: Communications in Computer and Information Science. http://link.springer.com/10.1007/978-3-319-45880-9$_2 1 Accessed 2021-08-13$

[19] Zhang, B., Al Hasan, M.: Name Disambiguation in Anonymized Graphs using Network Embedding. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, pp. 1239–1248. ACM, Singapore Singapore (2017). https://doi.org/10.1145/3132847.3132873. https://dl.acm.org/doi/10.1145/3132847.3132873 Accessed 2021-08-13

[20] Mai, G., Janowicz, K., Yan, B.: Combining Text Embedding and Knowledge Graph Embedding Techniques for Academic Search Engines. (2018)

[21] Le, Q., Mikolov, T.: Distributed Representations of Sentences and Documents. In: Proceedings of the 31st International Conference on Machine Learning, pp. 1188–1196. PMLR, ??? (2014). ISSN: 1938-7228. https://proceedings.mlr.press/v32/le14.html Accessed 2021-09-28

[22] Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating Embeddings for Modeling Multi-relational Data. In: Advances in Neural Information Processing Systems, vol. 26. Curran Associates, Inc., ??? (2013). https://papers.nips.cc/paper/2013/hash/1cecc7a77928ca8133fa24680a88d2f9-Abstract.html Accessed 2021-09-15

[23] Nayyeri, M., Vahdati, S., Zhou, X., Shariat Yazdi, H., Lehmann, J.: Embedding-Based Recommendations on Scholarly Knowledge Graphs. In: Harth, A., Kirrane, S., Ngonga Ngomo, A.-C., Paulheim, H., Rula, A., Gentile, A.L., Haase, P., Cochez, M. (eds.) The Semantic Web. Lecture Notes in Computer Science, pp. 255–270. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-49461-2_15

[24] Sun, Z., Deng, Z.-H., Nie, J.-Y., Tang, J.: RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space. arXiv:1902.10197 [cs, stat] (2019). arXiv: 1902.10197. Accessed 2021-09-15

[25] Massari, A.: Bibliographic dataset based on Scientometrics, containing provenance information compliant with the OpenCitations Data Model and non disambigued authors. Zenodo. Type: dataset (2021). https://doi.org/10.5281/zenodo.5151264. https://zenodo.org/record/5151264 Accessed 2021-09-29

[26] Daquino, M., Peroni, S., Shotton, D., Colavizza, G., Ghavimi, B., Lauscher, A., Mayr, P., Romanello, M., Zumstein, P.: The OpenCitations Data Model. arXiv:2005.11981 [cs] (2020). arXiv: 2005.11981. Accessed 2021-07-28

[27] Falco, R., Gangemi, A., Peroni, S., Shotton, D., Vitali, F.: Modelling OWL Ontologies with Graffoo. In: Presutti, V., Blomqvist, E., Troncy, R., Sack, H., Papadakis, I., Tordai, A. (eds.) The Semantic Web: ESWC 2014 Satellite Events. Lecture Notes in Computer Science, vol. 8798, pp. 320–325. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-11955-7_42. https://doi.org/10.1007/978-3-319-11955-7$_4 2 Accessed 2019-04-02$

[28] Santini, C., Alam, M., Asefa, G.G., Peroni, S., Gangemi, A., Sack, H.: AMiner-534K: Knowledge Graph of AMiner benchmark for Author Name Disambiguation. Zenodo. Type: dataset (2021). https://doi.org/10.5281/zenodo.5675801. https://zenodo.org/record/5675801 Accessed 2021-11-24

[29] Yang, B., Yih, W.-t., He, X., Gao, J., Deng, L.: Embedding Entities and Relations for Learning and Inference in Knowledge Bases. arXiv:1412.6575 [cs] (2015). arXiv: 1412.6575. Accessed 2021-09-28

[30] Cohan, A., Feldman, S., Beltagy, I., Downey, D., Weld, D.S.: SPECTER:

Document-level Representation Learning using Citation-informed Transformers. arXiv:2004.07180 [cs] (2020). arXiv: 2004.07180. Accessed 2021-09-10

[31] Ali, M., Berrendorf, M., Hoyt, C.T., Vermue, L., Sharifzadeh, S., Tresp, V., Lehmann, J.: PyKEEN 1.0: A Python Library for Training and Evaluating Knowledge Graph Embeddings. Journal of Machine Learning Research **22**(82), 1–6 (2021). Accessed 2021-09-17