Leveraging Knowledge Graph Embeddings to Disambiguate Author Names in Scientific Data

Laura Rettig
eXascale Infolab
University of Fribourg
Fribourg, Switzerland
laura.rettig@unifr.ch

Kurt Baumann
SWITCH
Zurich, Switzerland
kurt.baumann@switch.ch

Sebastian Sigloch
SWITCH
Zurich, Switzerland
sebastian.sigloch@switch.ch

Philippe Cudré-Mauroux
eXascale Infolab
University of Fribourg
Fribourg, Switzerland
philippe.cudre-mauroux@unifr.ch

***Abstract*—Access to scientific data is dependent on the proper indexing of such data for findability (alongside other FAIR standards) on portals, aggregators and more generally-speaking on the Web. Due to a lack of uptake in terms of standards (e.g., on unique identifiers, ORCID records, etc.), author name disambiguation continues to represent a major issue in organizing such research data. In this work, we present a novel approach to resolving name ambiguity for scientific authors as they appear in data about publications, grants or scientific datasets. Specifically, we leverage metadata present in a document in order to cluster similar authors: In addition to commonly-used information such as co-authorship, we include named entity similarities obtained from knowledge graphs as an additional source of information to further improve document representation and, subsequently, cluster the documents by authors. Due to the computational complexity of graph algorithms, we leverage knowledge graph embeddings to approximate the structure of large graphs. We evaluate our approach against an existing solution on a gold standard dataset and show that our approach provides notable improvement, especially when other information is sparse. In addition, we provide a novel, manually-annotated dataset for this task, consisting of scientific publications and project data.**

***Index Terms*—author name disambiguation, data set, knowledge graphs, scientific data, word embeddings**

I. Introduction

Researchers spend considerable time and energy publishing their results in the best possible outlets. Publication data is in that context often key for the advancement of their careers, as it is routinely taken into account by hiring, promotion, or tenure committees. Beyond those standard use-cases, bibliographic data is actually used for an increasingly diverse set of applications. Publications are for example used when allocating competitive funding, when identifying experts on a given topic, or when analyzing citation or peer-review graphs for detecting abnormal behavior or scientific fraud.

The quality of bibliographic data is hence of utmost importance. Beyond punctual quality issues relating to inconsistent sources or errors that are entered manually [1], [2], a major issue in that context is author name disambiguation. Despite recent efforts—such as ORCID[1] records to identify researchers through unique identifiers—most researchers are still mentioned using their names in project reports, grants, and in most publications to date. However, as we explain in more detail below (Section III), such names can often be ambiguous, in the sense that different researchers can share the same name, while the name of a given researcher can change over time or be serialized differently depending on the platform and context. As such, it can be challenging to identify researchers and discover their work on previous publications and research projects.

At a larger scale, the issue of name disambiguation addresses data quality. A scientific data platform will typically aggregate data from various providers and sources. With ORCID only being adapted by some platforms, each provider uses their own standard in terms of author identifiers and data quality varies greatly as a function of the intents of the data providers and the motivation to maintain high quality data. A research data platform that organizes information on publications and research projects thus follows *FAIR* principles, that is to say, organizing data in such a way that it is findable, accessible, interoperable, and reusable [3]. In a first step, aggregating different data sources scattered across the web into a single platform improves the findability of such data. A platform with the aim of helping researchers find each other's work, and reuse existing results, may have downstream applications that rely on high quality data in order to deliver the best results. Having several profiles per person affects the findability of the researcher's work, as not all work will be associated with one profile. On the other hand, multiple distinct researchers sharing one profile leads to poor results e.g. of recommender systems, as these may consider work related when written by presumably the same author.

In this paper, we tackle this problem by introducing a new method for author name disambiguation in scientific data. Prior work tackling this issue have suggested various approaches, leveraging clustering [4], [5] or crowdsourcing [6], [7]. In this paper, we set out to demonstrate how the inclusion of more extensive structured data and semantic links, in the form of knowledge graphs, can improve the results on this task.

Knowledge graphs (KGs) have become a powerful tool in representing semi-structured information and concepts. They are defined through their nodes—entities or concepts—and edges—labeled relationships between nodes. While compa-

[1] The Open Researcher and Contributor ID, see https://orcid.org

978-1-6654-8045-1/22/$31.00 ©2022 IEEE

nies are building their own KGs internally to power their applications, a number of open-source KGs are open and available to all. Wikidata[2], for instance, is a prominent KG that is free and open and that can be accessed by both humans and machines. Wikidata contains most entities (e.g., persons, companies, events, or concepts) that are of common interest or that are available on other Wikimedia portals, along with series of property-value pairs and relationships to further entities. With nearly 100 million entities alone in Wikidata[3], not to mention relationships between entities, Wikidata captures a vast array of human knowledge on various general and highly specific topics. Taken as a whole, this represents invaluable information that can be leveraged in many text analysis tasks. As an example, Figure 1 illustrates some of the relationships between the *human* entity and related concepts, visualized via the Wikidata Graph Builder[4].

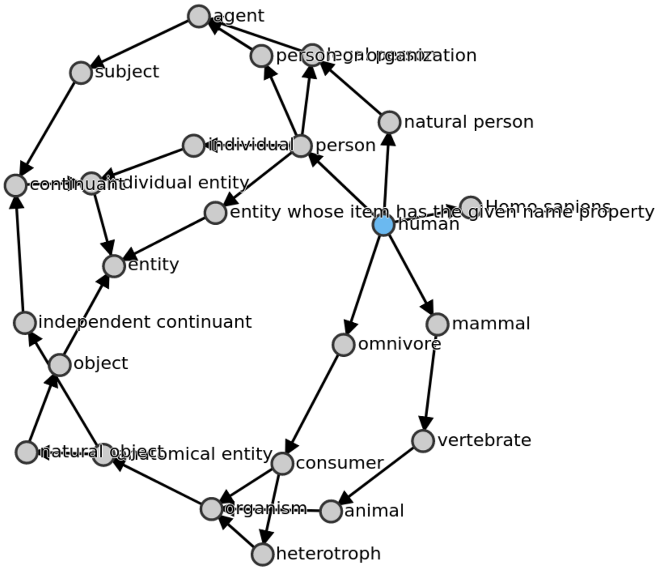


Fig. 1. Wikidata represents knowledge in a structured manner and facilitates knowledge discovery and management through graph algorithms. This example shows the *subclassOf* relationships between the *human* entity and related concepts.

KGs are particularly powerful thanks to their graph structure and the relationships they define between entities, allowing for the utilization of processes based on various graph algorithms. However, due to the immense size of the graphs, such algorithms are costly in practice [8]. Recent advancements on knowledge graph embedding [9]–[13] offer a promising solution to this problem: Graph structures are preserved, while the graph representation is compressed into vectors of fixed size, allowing for efficient computation. For example, distance computation between two nodes in a graph requires the use of algorithms that traverse the graph, a costly endeavor, whereas the distance can be approximated to a degree that is sufficient to most applications [14], [15] by a simple cosine similarity operation between their respective vector representations [16]. In this work, we show how one can use KGs, and more specifically, KG embeddings to help disambiguate names appearing on scientific documents, i.e., to correctly attribute each work to its corresponding author(s).

A. Contribution

In summary, the contributions of this work are as follows:

1) We propose a new framework mixing graph and text embeddings to disambiguate author names appearing in scientific documents. In addition to commonly used information such as co-authorship, our method uses named entity similarities obtained from knowledge graphs as outside sources of information to further improve document representation and, subsequently, the clustering of documents by unique authors.
2) In addition, we propose a novel multilingual and manually annotated dataset for the evaluation of name disambiguation including both publication and research project information.
3) Finally, we empirically show the merits of our name disambiguation framework compared to a strong baseline on both standard data as well as on our new corpus.

The rest of this paper is structured as follows. First, we provide an overview of related work on author name disambiguation and named entities in Section II. We introduce our problem in Section III and our disambiguation method in Section IV. Finally, we describe our new dataset as well as the results of our empirical evaluation in Section V before concluding.

II. Related Work

In this work, we leverage entity extraction and graph embeddings to solve the problem of author name disambiguation on scientific documents. This section presents previous work in these two related areas.

A. Author Name Disambiguation

Several works use graphs to represent documents and their similarities. GHOST [17] uses co-authorship as a sole feature to generate a similarity matrix and cluster documents using a so-called *Affinity Propagation*. Similarly, Zhang et al. [18] create graph embeddings from a network topology based on collaboration networks, while Amancio et al. [19] also leverage co-author structures and graphs in order to regroup candidate ambiguous persons. Louppe et al. [20] use supervised learning to obtain a pairwise linkage function by placing particular emphasis on differing strategies for author names from different cultural backgrounds. Zhang et al. [4] leverage both global embeddings and refinement in local linkage structures using more document features to obtain similarity representations. Chen et al. [21] propose a bilingual (Chinese/English)

[2] see https://www.wikidata.org/
[3] https://www.wikidata.org/wiki/Wikidata:Statistics
[4] https://angryloki.github.io

dataset for author name disambiguation evaluation and utilizes Graph Convolutional Networks in order to handle both paper and author information. Wang et al. [22], use adverserial representation learning on heterogeneous graphs in order to eliminate the need for feature engineering altogether; however, the authors do not take into account external information as we do in the rest of this paper. Chen et al. [23], finally, propose a framework that leverages reinforcement learning in order to decide on the fly whether an incoming document shall be merged with an existing author profile, or whether to create a new profile for a previously unseen author.

As author name disambiguation relates to the extraction and improvement of textual metadata related to scientists, another common approach in this context is the identification of research topics from title and abstract of a paper using Latent Dirichlet Allocation (LDA) [24]–[26] or other forms of topic analysis [27]. By extracting entities from textual metadata, we are also enhancing the document representation with core concepts.

B. Named Entity Extraction in Author Name Disambiguation

Author name disambiguation can be considered a specific case of named entity linking, the process of matching an entity mention to its correct corresponding entity in a Knowledge Base (KB) [28]. However, little work has been conducted connecting author name disambiguation with entities in existing KBs and leveraging their structural features. Vu et al. [29] utilize external textual documents that the author refer to as a "Knowledge Base" to enrich the given and often sparse textual information and improve the representation of relevant terms in a tf-idf model. Song et al. [30] utilize Named Entity Recognition (NER) on author affiliation and country in order to eliminate noise from different variants of entering affiliation, by taking the KB representation to replace the user-entered affiliation. Farber et al. [31] link identified authors to their ORCID records based on a fixed set of rules and as such enrich their data platform with the structural information in ORCID. However, they do not leverage this information going forward.

Query resolution is a related issue to author name disambiguation, in that it also poses the problem of document similarity for information retrieval in the case of homonyms entered as a query. In order to improve an academic search engine, Xiong et al. [32] identify entities in publication metadata and create a Knowledge Graph consisting of entities in their dataset, i.e., documents, authors, and venues, in conjunction with entities from *Freebase*. They use this information to compute similarity metrics for information retrieval. Compared to the use of a KG embedding trained on the entire KG, limiting entities to strictly those contained within the dataset implies loosing part of the structural information (e.g., the similarity between two entities that are not directly adjacent but linked through other entities not contained in the dataset).

To the best of our knowledge, none of these previous works leverages entities extracted from external data structured as a knowledge graph as we suggest in the present work.

C. Applications of Knowledge Graph Embeddings

KG embeddings are used to address the task of *relation extraction*, that is, identifying the relationship between several entities mentioned in text [33]–[36]. In these cases, commonly, a graph embedding is learned jointly from text samples and a knowledge graph. Similarly, *question answering* requires the extraction of an entity and a desired relationship from input text in order to retrieve the desired resulting entity. KGs organize this knowledge of entities and their relationships, and specialized embeddings are constructed such that queries and their responses are within proximity of one another in a vector space [37], [38]. In *recommender systems*, KG embeddings have been used to improve collaborative filtering to learn joint representations of heterogeneous information consisting of structural, textual, and visual knowledge [39].

By leveraging KG embeddings specifically as a solution to the problem of name disambiguation, we exploit the demonstrated potential of KG embeddings in a vast array of applications in a context where, to the best of our knowledge, the benefits of KGs are not yet fully exploited.

III. Background

This section describes in more detail the problem we tackle and introduces the main terms and concepts used throughout the rest of this paper.

A. Author Name Disambiguation

The problem tackled in this paper can surface in the scientific literature in two different ways: i) when two (or more) researchers publish under the same name (homonymy) and ii) when a given researcher is referenced using different names (i.e., surface forms) in scientific data. In both cases, this necessitates a correction such that research items, in the form of projects or publications, get associated to the correct researcher.

B. Core Concepts

Here are the core concepts used in this paper in the description of our problem (see Section III-C below) and method (Section IV).

a) Publication: An academic paper in the form of an article in a journal or conference or a book; commonly peer-reviewed. We denote it as P. For our purpose, P can be defined through the following metadata,

$$P = \{\text{authors, title, publication year, published in, abstract}\}. \tag{1}$$

b) Project: A research project that is funded through a grant in a one-to-one mapping, hence denoted as G. Similarly to P, we consider that G is defined through the following metadata:

$$G = \{\text{researchers, title, time period, abstract}\}. \tag{2}$$

While distinct in some features, we group together P and G under the common term documents D.

c) Author: We denote as A authors of scientific publications P and researchers associated with projects G. A refers to a unique author, that is, we may have multiple candidate authors appearing under the same name. We denote the set of candidate authors as $\bar{A} = \{A_1, A_2, \dots A_n\}$.

C. Problem Formulation

The problem we tackle is thus as follows: Given an author name and documents $\bar{D} = \{D_{A1}, \dots D_{An}\}$, partition $\bar{D}$ into clusters C where all D within C_A are associated with the same unique author A, and all D_A authored by a specific author A are within the same cluster C_A. In order to achieve this through the available features described above, we aim at learning efficient and effective document representations $\mathbb{R}(D)$ such that the similarity between two $\mathbb{R}(D_i)$ is maximized if they are written by the same author A.

IV. Method

This section presents the key components and methods of our proposed author name disambiguation framework. Figure 2 presents an overall view of all the components and their integration. Candidate documents D proceed through two parallel pipelines: the first pipeline extracts common metadata and learns their representation using word embedding techniques, while the second pipeline extracts entities and obtains their representation as a knowledge graph embedding. Finally, both of these features are used to obtain a combined representation of the input. The following explains these components in more detail.

A. Representation Learning for Document Clustering

Incoming documents D, consisting of P and G, along with their associated metadata, are used to learn a vector space representation through Word2Vec [40]. Recall that a document is comprised of features derived from its metadata, that is, title, publication year or project time period, authors, abstract, keywords, and wherever available publication venue. This metadata is treated as sentence input when training embeddings on the attributes. This learns a semantic representation that allows us to leverage the proximity of, for example, publication venues and the subjects of publications. It is important to note that we are not just looking for common representations of terms that are derived from text usage, but we are also seeking semantic links between attributes such as conference acronyms and title topics. Therefore, the use of embedding models that are pretrained on general-purpose corpora (such as BERT) is not a feasible alternative. Other, potentially more sophisticated word embedding algorithms could be used at this stage, but we found that Word2Vec is sufficient in the context of our task, as we experimentally show in Section V. As illustrated in Figure 3, we train a Word2Vec model on the entire corpus of all documents in the given dataset, then get the vectors associated with the features present in each document D.

In order to obtain one single lower-dimensional document vector $\epsilon(D)$, we additionally weigh its features by relevance to the document compared to the overall corpus of all documents using term frequency-inverse document frequency (tf-idf). As a result, each candidate document corresponds to a single vector representing its pertinent features.

B. Multilingual Named Entity Extraction

Entities extracted from text provide information about the subject matter of a text and as such are a valuable feature for similarity comparison and clustering. While the representation trained in the previous Section IV-A learns features from within the dataset, linking entities inside the data to an external knowledge graph improves representation by leveraging conceptual connections that may not have appeared within the dataset. For example, two concepts may only be few edges apart in a graph, but never co-occur in similar contexts in the input dataset. Using their graph representation to improve the learned document representation includes this conceptual similarity in the input to clustering.

We perform named entity recognition and linking on the textual data present in P and G, namely, their titles and abstracts. This procedure corresponds to the lower pipeline in Figure 2 and is illustrated in more detail in Figure 4.

1) OpenNMT: Since this textual data can appear in English but also in other languages potentially (e.g., in German, French or Italian, in the novel dataset we provide in Section V-A), we first find some common ground by applying neural machine translation from the detected language into English (step 1 in Figure 4). For this we leverage translation models provided in OpenNMT [41]. OpenNMT uses an attention-based encoder-decoder architecture approach to neural machine translation. Translating all documents into English allows for the following steps to use more efficient, single-language models, as opposed to requiring a model per language on each subsequent step. It also captures rarer concepts that may not be present in non-English Wikipedia.

2) FLAIR: The FLAIR framework [42] is a PyTorch-based natural language processing library. It provides a pretrained model for named entity recognition. As such, we use it to detect mentions of entities in text (step 2), i.e., titles and abstracts in our input data, for each document. The detected entity mentions and their contexts are pipelined into BLINK for disambiguation, such that they can be assigned a unique identifier in our chosen knowledge base.

3) BLINK: The BLINK library [43] provides efficient and accurate entity linking with Wikipedia as the target knowledge base. BLINK utilizes BERT architectures to consider the context of an entity mention and select the most suitable corresponding *Wikipedia* entry (step 3). For each *Wikipedia* entry there exists a corresponding node in *Wikidata*. Knowing the entity allows us to obtain *Wikidata* identifiers for detected entities, that is, unique strings of the format *Q123* (step 4).

4) Wikidata: As a store of structured data, *Wikidata* possesses an inherent graph structure capturing a large base of

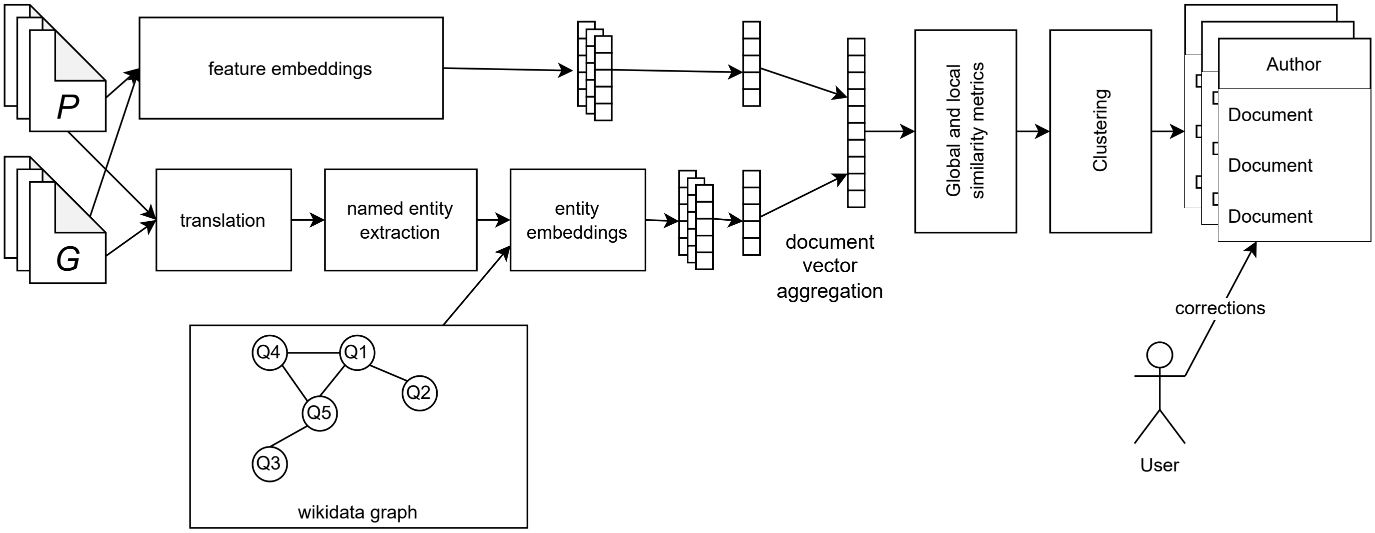


Fig. 2. Schematic overview of knowledge graph-powered author name disambiguation. Documents (P - publications and G - projects) are processed in two parallel pipelines. The upper pipeline yields document embeddings of textual features and is detailed further in Figure 3. The lower pipeline produces relevant knowledge graph embeddings for the documents. It first translates documents into English as standard representation, then identifies mentions of named entities in text, before linking them to the most appropriate candidate in Wikidata. It then takes the vector representation of the entity in the Wikidata graph and combines this with the vector obtained in the upper pipeline for document representation. From these documents the system learns similarity metrics that are then used in clustering in order to assign documents to unique authors.

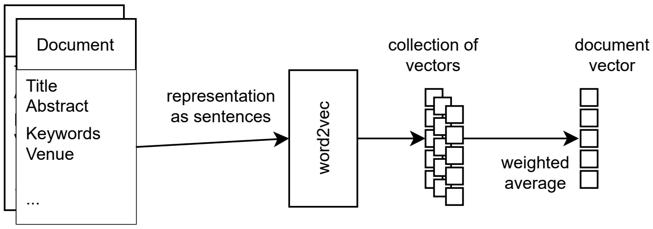


Fig. 3. Representing documents for learning. Documents—publications or projects—consist of several attributes, including textual data from title and abstract as well as keywords and journal/venue (for publications). These document data are treated as sentence input to word2vec, yielding a collection of word vectors for each document. Document relevance scores are used to combine the vectors in a weighted average, such that each document is represented by one vector.

knowledge contained within the Wikimedia projects. While the embeddings obtained in section IV-A are trained purely on data contained within the dataset at hand, by leveraging the Wikidata structure, we are able to obtain representations that are aware of information in the graph not contained within the dataset, such as distances and similarity between entities through several steps in Wikidata. This follows the assumption that one author will typically be working on similar—related—concepts across multiple papers and projects, even when not using the same terminology. As stated previously, graph traversal algorithms are computationally costly. We hence need a compact representation of the semantic relationships of concepts. In order to work with an efficient representation of this information, we decided to represent the Wikidata graph using PyTorch-BigGraph [44] as a collection of lower-dimensional embedding vectors. This learns a vector for each node in the Wikidata graph where concepts that are close in the graph and conceptually similar are closer in the vector space. In this way, we are able to efficiently calculate the distance between entities in the vector space, as opposed to finding the shortest path between two nodes via traversal in a graph. We obtain the vector for a concept via its Wikidata identifier. Figure 5 illustrates the motivation of using the condensed graph embedding representation. Finally, we summarize several entities associated with one document into a document embedding [45] in order to obtain a single entity representation for each document. The summarization follows the feature selection described in section IV-A: Entities' relevance to a given document in the context of the overall corpus is calculated using tf-idf such that entities that are highly specific to a given document have higher weight. From there, each document is associated with a single entity vector representation obtained as a weighted average from the set of entity vectors associated with a document.

In conjunction with the feature embeddings extracted in section IV-A, we use the concatenated document vectors from the two parallel pipelines for the global representation as well as in order to obtain similarity metrics between candidate documents for the creation of the local linkage graph.

C. Graph Representation and Clustering

For clustering documents by authors, we adapt the method proposed by Zhang et al. [4]: For each candidate set, that is, for each group of potentially homonymous authors $\bar{A}$, in addition to the global representation as described in IV-A, each pair of documents within the group is evaluated on their feature intersection, including the entity features extracted in

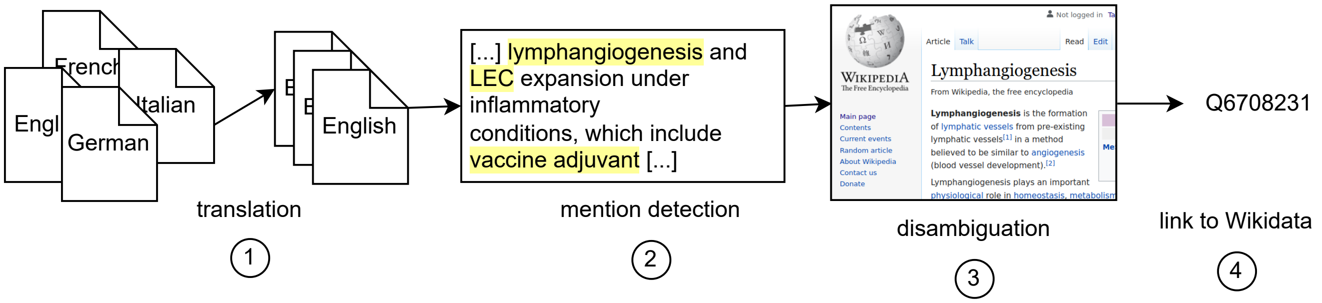

French
Italian
English
German
English
translation
1
[...] lymphangiogenesis and LEC expansion under inflammatory conditions, which include vaccine adjuvant [...]
mention detection
2
Wikipedia
The Free Encyclopedia
Lymphangiogenesis
From Wikipedia, the free encyclopedia
Lymphangiogenesis is the formation of lymphatic vessels from pre-existing lymphatic vessels[1] in a method believed to be similar to angiogenesis (blood vessel development).[2]
Lymphangiogenesis plays an important physiological role in homeostasis, metabolism
disambiguation
3
Q6708231
link to Wikidata
4

Fig. 4. Obtaining Wikidata identifiers for entities mentioned in input documents P and G. Documents arrive in a variety of languages and are translated to English in a first step in order to simplify subsequent processes. In a second step, we parse the title and abstract of the documents in order to detect mentions of entities. For each entity, among its candidate matches in Wikipedia, we pick the most suitable (3). Finally, we obtain the corresponding Wikidata identifier, pointing us to the representative node in the graph (4).

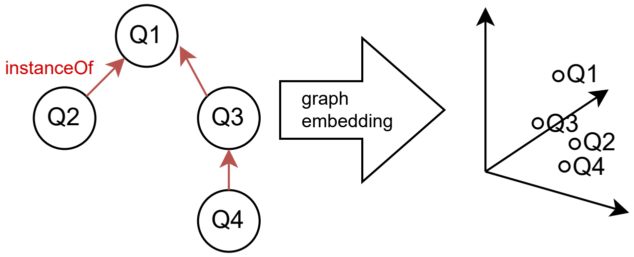


Fig. 5. Learning knowledge graph embeddings to perform efficient computation on their vector space representations. Instead of having to traverse the graph (left) in order to discover that both Q2 and Q4 are instances of the same concept Q1, we can obtain their Euclidean distance in the vector space (right) in a single operation. We use this representation when learning the document similarity measures and when clustering documents.

section IV-B, in an unsupervised manner in order to construct a local linkage graph where edges exist between sufficiently similar documents $D_{\bar{A}}$. As such, adjacency in this graph signifies high similarity. For efficiency, these graphs—one per $\bar{A}$ with the associated set of candidate documents $D_{\bar{A}}$—are then compressed using a variational graph auto-encoder that contains sufficient information to allow for the reconstruction of the adjacency matrix of the original graph. This generates, for each document, a new latent representation. The document embedding $\mathbb{R}(D)$— consisting of global and local network information—is finally used in a hierarchical agglomerative clustering (HAC) [46] algorithm implemented in scikit-learn in order to obtain clusters of documents written by the same author.

D. Continuous User Improvements

Due to the imperfect nature of author name disambiguation, we expect users of our system to identify mistakes and report them through a user interface. To support this additional feature, all data in our system, including author data, are represented as linked data, for which we maintain data provenance using PROV-O[5]. As such, any modifications to the data are documented and we are able to trace the lineage of author profiles. This can further serve as training data in the future by identifying various cases for author name disambiguation. Specifically, we allow feedback that

- merges author profiles,
- splits author profiles,
- corrects author profiles, and
- adds missing records.

V. Experimental Validation

In order to evaluate our approach, we propose a novel multilingual publication and project dataset obtained from the Swiss National Science Foundation (SNSF) and also test our method on the state-of-the-art AMiner benchmark dataset [4].

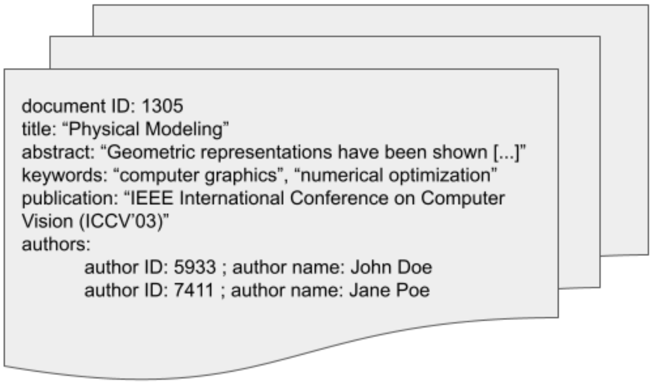
A. Swiss National Science Foundation Dataset

The raw data was obtained from the SNSF[6]. This contains, among other data, separate tables for grants (associated with projects), persons, and publications. While it does provide links between persons and projects via their identifiers, the original data contains no explicit links between a person and their publications. Prior to manual annotation, therefore, the profile for each person would only include their associated research projects. The data was manually annotated with unique person identifiers when absent, in order to correct inaccuracies that are present in the download as-is. As previously mentioned, the data contained two aspects of ambiguity: (1) when multiple authors publish under the same name, such as in the publications document, where only author names are provided without identifiers, and (2) when one author appears under multiple names, e.g, due to not reusing person identifiers between distinct projects or because of name changes, misspellings, or shortening of names with initials. In order to tackle issue (1), we manually inspected the area of research of the candidates as well as other online sources in order to determine whether two authors are distinct, and assigned unique identifiers in the case of distinct authors. Issue (2) was addressed by finding similarly spelled names and inspecting

[5] https://www.w3.org/TR/prov-o/

[6] https://data.snf.ch/datasets: grants with abstracts, persons, Output data: scientific publications

associated documents such that we could determine when two seemingly distinct author profiles should be merged.[7]



document ID: 1305
title: “Physical Modeling”
abstract: “Geometric representations have been shown [...]”
keywords: “computer graphics”, “numerical optimization”
publication: “IEEE International Conference on Computer Vision (ICCV’03)”
authors:
author ID: 5933 ; author name: John Doe
author ID: 7411 ; author name: Jane Poe

Fig. 6. Sample of documents in the SNSF dataset. Each document is denoted by a unique identifier and contains associated metadata. Metadata may vary depending on the type of document and availability. Authors are linked to the documents and have been assigned a unique ID for each distinct real-world person.

The dataset is available in a JSON format, consisting of documents (projects and grants) and their associated metadata. Figure 6 illustrates a sample of the dataset: Documents have a unique identifier assigned to them. They are also linked to their respective authors. Each unique author has been assigned a unique ID. During training, these identifiers are removed, but later used for validation of the resulting clusters.

TABLE I
BENCHMARK DATASET DESCRIPTION

Dataset	Documents (publications and projects)	Authors	Size
SNSF annotated	48 029	7 191	135MB
AMiner DB	70 258	12 798	351MB

Compared to AMiner, this dataset is smaller in training data and contains smaller candidate clusters, i.e., fewer candidates per ambiguous name. While the small size poses a challenge to training the representation learning component (section IV-A), the smaller clusters can be attributed to the different cultural context—European person names are less frequently homonymous [5]. While the data set does contain researchers from all around the world, it is biased more towards the Swiss research landscape. It would be expected that there is some overlap between the datasets; namely, when it comes to publications that are included in the available AMiner dataset. However, the dataset we propose is biased heavily towards project metadata that has not been published elsewhere, and also includes publications in languages other than English. Table I summarizes the two datasets used in the evaluation.

TABLE II
PERFORMANCE COMPARISON OF THE BASELINE AMINER AUTHOR NAME DISAMBIGUATION (AMINER AND) AND OUR KNOWLEDGE GRAPH-POWERED AUTHOR NAME DISAMBIGUATION (KG-AND) METHODS ON THE AMINER DB AND THE NOVEL SNSF DATASETS.

Dataset	Method	Precision	Recall	F1
AMiner DB	AMiner AND	0.77	0.62	0.69
AMiner DB	KG-AND	0.77	0.65	0.70
SNSF	AMiner AND	0.79	0.63	0.70
SNSF	KG-AND	0.81	0.66	0.73

B. Results

Table II gives the precision, recall and F1-scores of baseline method we consider on both the AMiner[8] and the SNSF datasets, along with the results of our knowledge graph-powered author name disambiguation approach.

The reported precision and recall scores are the results of an evaluation on sets of 100 ambiguous names for each dataset, selected from the ambiguous names in the SNSF dataset and distinct from those used in training, and choosing the same test and training cluster sets for the AMiner set as in the original paper. These are not the same names, as the SNSF dataset contains more western names as AMiner, and the ambiguous names in the AMiner set are not present in the same manner in the SNSF dataset. Due to the nature of our use case, the performance of the algorithms on ambiguous names in the SNSF dataset is of great importance to our research data platform. Precision and recall are computed for each ambiguous name cluster, the reported numbers are the averages across 100 names. An evaluation on a fixed number ambiguous names, in the order of magnitude of 100, is common in the literature on the problem of name disambiguation [4], [17], [18].

C. Discussion

Looking at the performance of the baseline AMiner method, we see in table II that the SNSF dataset has higher performance (in terms of F1). This can be attributed to the smaller size in homonymous clusters—fewer candidates within the same cluster yields to less potential for error.

Accuracy did not improve between the baseline AMiner AND and KG-AND on the AMiner dataset. We speculate that the larger dataset size used in the first step of training embeddings on the corpus of all documents leads to higher quality Word2Vec embeddings. Recall increases between the methods and so does the overall performance (indicated by the F1-score) and as such we can conclude that the inclusion of concepts and their similarity from an external knowledge graph introduces helpful information to guide the disambiguation process.

The smaller SNSF datasets benefits more from the inclusion of the KG structure and we observe an increase in both

[7] The annotated SNSF dataset is publicly available at https://github.com/eXascaleInfolab/SNF_disambiguation

[8] AMiner results deviate from those reported in [4] due to the publicly available training set being smaller.

accuracy and recall. The smaller input data used in training the original Word2Vec embedding may not provide an equally good representation of terms. As such, additional data sources provide structural information on concept relatedness that are absent in the smaller document corpus.

VI. Conclusion

In this work, we presented a method leveraging external structural information from knowledge graphs in order to extend state-of-the-art methods for author name disambiguation. To the best of our knowledge, external knowledge graph sources have not been previously utilized in tackling this important and common problem. We evaluated our approach against a strong baseline system on a large dataset. In addition, we presented a novel multilingual dataset for author name disambiguation, which we also used in our evaluation of both the baseline and our proposed method. We showed that leveraging KGs aided in the performance of author name disambiguation, particularly when datasets are of smaller size and initial word embeddings are of lower representation quality. In terms of future work, we intend to extend our approach using further word embedding technologies trained on larger external corpora to become independent from the size of the input corpus, in order to obtain better representations of the input terms in addition to giving greater weight to the KG embedding features.

References

[1] J. Beall and K. Kafadar, "Measuring typographical errors' impact on retrieval in bibliographic databases," *Cataloging & Classification Quarterly*, vol. 44, pp. 197–211, 07 2007.

[2] C. P. Bourne, "Frequency and impact of spelling errors in bibliographic data bases," *Inf. Process. Manag.*, vol. 13, pp. 1–12, 1977.

[3] M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J.-W. Boiten, L. B. da Silva Santos, P. E. Bourne *et al.*, "The fair guiding principles for scientific data management and stewardship," *Scientific data*, vol. 3, no. 1, pp. 1–9, 2016.

[4] Y. Zhang, F. Zhang, P. Yao, and J. Tang, "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*, ser. KDD '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 1002–1011. [Online]. Available: https://doi.org/10.1145/3219819.3219859

[5] S. Sun, H. Zhang, N. Li, and Y. Chen, "Name disambiguation for chinese scientific authors with multi-level clustering," in *2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*, vol. 1, 2017, pp. 176–182.

[6] A. Correia, D. Guimarães, D. Paulino, S. Jameel, D. Schneider, B. Fonseca, and H. Paredes, "Authcrowd: Author name disambiguation and entity matching using crowdsourcing," in *2021 IEEE 24th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, 2021, pp. 150–155.

[7] X. Sun, J. Kaur, L. Possamai, and F. Menczer, "Ambiguous author query detection using crowdsourced digital library annotations," *Information Processing & Management*, vol. 49, no. 2, pp. 454–464, 2013.

[8] H. Cai, V. W. Zheng, and K. C.-C. Chang, "A comprehensive survey of graph embedding: Problems, techniques, and applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 9, pp. 1616–1637, 2018.

[9] C. Zhou, Y. Liu, X. Liu, Z. Liu, and J. Gao, "Scalable graph embedding for asymmetric proximity," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, no. 1, 2017.

[10] H. Wang, J. Wang, J. Wang, M. Zhao, W. Zhang, F. Zhang, W. Li, X. Xie, and M. Guo, "Learning graph representation with generative adversarial nets," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 8, pp. 3090–3103, 2019.

[11] Y.-A. Lai, C.-C. Hsu, W. H. Chen, M.-Y. Yeh, and S.-D. Lin, "Prune: Preserving proximity and global ranking for network embedding," *Advances in neural information processing systems*, vol. 30, 2017.

[12] X. Ren, W. He, M. Qu, C. R. Voss, H. Ji, and J. Han, "Label noise reduction in entity typing by heterogeneous partial-label embedding," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016, pp. 1825–1834.

[13] Q. Wang, Z. Mao, B. Wang, and L. Guo, "Knowledge graph embedding: A survey of approaches and applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 12, pp. 2724–2743, 2017.

[14] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge graph embedding by translating on hyperplanes," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 28, no. 1, 2014.

[15] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, "Learning entity and relation embeddings for knowledge graph completion," in *Twenty-ninth AAAI conference on artificial intelligence*, 2015.

[16] F. Ilievski, P. Szekely, G. Satyukov, and A. Singh, "User-friendly comparison of similarity algorithms on wikidata," *arXiv preprint arXiv:2108.05410*, 2021.

[17] X. Fan, J. Wang, X. Pu, L. Zhou, and B. Lv, "On graph-based name disambiguation," *J. Data and Information Quality*, vol. 2, no. 2, feb 2011. [Online]. Available: https://doi.org/10.1145/1891879.1891883

[18] B. Zhang and M. Al Hasan, "Name disambiguation in anonymized graphs using network embedding," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, ser. CIKM '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 1239–1248. [Online]. Available: https://doi.org/10.1145/3132847.3132873

[19] D. R. Amancio, O. N. Oliveira Jr, and L. d. F. Costa, "Topological-collaborative approach for disambiguating authors' names in collaborative networks," *Scientometrics*, vol. 102, no. 1, pp. 465–485, 2015.

[20] G. Louppe, H. T. Al-Natsheh, M. Susik, and E. J. Maguire, "Ethnicity Sensitive Author Disambiguation Using Semi-supervised Learning. Ethnicity sensitive author disambiguation using semi-supervised learning," *Knowledge Engineering and Semantic Web*, vol. 649, pp. 272–287. 16 p, Aug 2015. [Online]. Available: http://cds.cern.ch/record/2222878

[21] Y. Chen, H. Yuan, T. Liu, and N. Ding, "Name disambiguation based on graph convolutional network," *Scientific Programming*, vol. 2021, pp. 1–11, 05 2021.

[22] H. Wang, R. Wang, C. Wen, S. Li, Y. Jia, W. Zhang, and X. Wang, "Author name disambiguation on heterogeneous information network with adversarial representation learning," 2020.

[23] B. Chen, J. Zhang, J. Tang, L. Cai, Z. Wang, S. Zhao, H. Chen, and C. Li, "Conna: Addressing name disambiguation on the fly," *IEEE Transactions on Knowledge and Data Engineering*, 2020.

[24] Y. Song, J. Huang, I. G. Councill, J. Li, and C. L. Giles, "Efficient topic-based unsupervised name disambiguation," ser. JCDL '07. New York, NY, USA: Association for Computing Machinery, 2007, p. 342–351. [Online]. Available: https://doi.org/10.1145/1255175.1255243

[25] S. Jiang, Y. Xian, H. Wang, Z. Zhang, and H. Li, "Representation learning with lda models for entity disambiguation in specific domains," *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol. 25, no. 3, pp. 326–334, 2021.

[26] L. Shu, B. Long, and W. Meng, "A latent topic model for complete entity resolution," 03 2009, pp. 880–891.

[27] K. Pooja, S. Mondal, and J. Chandra, "A graph combination with edge pruning-based approach for author name disambiguation," *Journal of the Association for Information Science and Technology*, vol. 71, 04 2019.

[28] S. Cucerzan, "Large-scale named entity disambiguation based on Wikipedia data," in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*. Prague, Czech Republic: Association for Computational Linguistics, Jun. 2007, pp. 708–716. [Online]. Available: https://aclanthology.org/D07-1074

[29] Q. Vu, A. Takasu, and J. Adachi, "Improving the performance of personal name disambiguation using web directories," *Information Processing & Management*, vol. 44, pp. 1546–1561, 07 2008.

[30] M. Song, E. H.-J. Kim, and H. J. Kim, "Exploring author name disambiguation on pubmed-scale," *J. Informetrics*, vol. 9, pp. 924–941, 2015.

[31] M. Färber and D. Lamprecht, "The data set knowledge graph: Creating a linked open data source for data sets," *Quantitative Science Studies*, pp. 1–30, 2021.

[32] C. Xiong, R. Power, and J. Callan, "Explicit semantic ranking for academic search via knowledge graph embedding," in *Proceedings of the 26th international conference on world wide web*, 2017, pp. 1271–1279.

[33] J. Weston, A. Bordes, O. Yakhnenko, and N. Usunier, "Connecting language and knowledge bases with embedding models for relation extraction," *arXiv preprint arXiv:1307.7973*, 2013.

[34] S. Riedel, L. Yao, A. McCallum, and B. M. Marlin, "Relation extraction with matrix factorization and universal schemas," in *Proceedings of the 2013 conference of the North American chapter of the association for computational linguistics: human language technologies*, 2013, pp. 74–84.

[35] M. Fan, D. Zhao, Q. Zhou, Z. Liu, T. F. Zheng, and E. Y. Chang, "Distant supervision for relation extraction with matrix completion," *arXiv preprint arXiv:1411.4455*, 2014.

[36] A. Smirnova and P. Cudré-Mauroux, "Relation extraction using distant supervision: A survey," *ACM Computing Surveys (CSUR)*, vol. 51, no. 5, pp. 1–35, 2018.

[37] A. Bordes, J. Weston, and N. Usunier, "Open question answering with weakly supervised embedding models," in *Joint European conference on machine learning and knowledge discovery in databases*. Springer, 2014, pp. 165–180.

[38] A. Bordes, S. Chopra, and J. Weston, "Question answering with subgraph embeddings," *arXiv preprint arXiv:1406.3676*, 2014.

[39] F. Zhang, N. J. Yuan, D. Lian, X. Xie, and W.-Y. Ma, "Collaborative knowledge base embedding for recommender systems," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016, pp. 353–362.

[40] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013.

[41] G. Klein, Y. Kim, Y. Deng, J. Senellart, and A. Rush, "OpenNMT: Open-source toolkit for neural machine translation," in *Proceedings of ACL 2017, System Demonstrations*. Vancouver, Canada: Association for Computational Linguistics, Jul. 2017, pp. 67–72. [Online]. Available: https://www.aclweb.org/anthology/P17-4012

[42] A. Akbik, T. Bergmann, D. Blythe, K. Rasul, S. Schweter, and R. Vollgraf, "Flair: An easy-to-use framework for state-of-the-art nlp," in *NAACL 2019, 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, 2019, pp. 54–59.

[43] L. Y. Wu, F. Petroni, M. Josifoski, S. Riedel, and L. Zettlemoyer, "Scalable zero-shot entity linking with dense entity retrieval," in *EMNLP*, 2020.

[44] A. Lerer, L. Wu, J. Shen, T. Lacroix, L. Wehrstedt, A. Bose, and A. Peysakhovich, "Pytorch-biggraph: A large-scale graph embedding system," *arXiv preprint arXiv:1903.12287*, 2019.

[45] J. Coates and D. Bollegala, "Frustratingly easy meta-embedding – computing meta-embeddings by averaging source word embeddings," 2018.

[46] A. Lukasová, "Hierarchical agglomerative clustering procedure," *Pattern Recognition*, vol. 11, no. 5-6, pp. 365–381, 1979.