Framework for Author Name Disambiguation in Scientific Papers Using an Ontological Approach and Deep Learning

Peer-reviewed author version

Lisandra Díaz-de-la-Paz; CONCEPCION PEREZ, Leonardo; Jorge Armando Portal-Díaz; Alberto Taboada-Crispi & Amed Abel Leiva-Mederos (2022) Framework for Author Name Disambiguation in Scientific Papers Using an Ontological Approach and Deep Learning. In: Villazón-Terrazas, Boris; Ortiz-Rodriguez, Fernando; Tiwari, Sanju; Sicilia, Miguel-Angel; Martín-Moncunill, David (Ed.). Knowledge Graphs and Semantic Web 4th Iberoamerican Conference and third Indo-American Conference, KGSWC 2022, Madrid, Spain, November 21–23, 2022, Proceedings, Springer, p. 216 -233.

DOI: 10.1007/978-3-031-21422-6_16
Handle: http://hdl.handle.net/1942/39033

Framework for Author Name Disambiguation in Scientific Papers Using an Ontological Approach and Deep Learning*

Lisandra Díaz-de-la-Paz[1,2][0000−0003−4281−1517], Leonardo Concepción-Pérez[1,2][0000−0002−4515−2038], Jorge Armando Portal-Díaz[1][0000−0003−1360−4930], Alberto Taboada-Crispi[1,2][0000−0002−7797−1441], and Amed Abel Leiva-Mederos[1,2][0000−0002−9144−5018]

[1] Universidad Central "Marta Abreu" de Las Villas (UCLV), Carretera a Camajuaní Km 5 1/2, Santa Clara, Cuba

[2] Centro de Investigaciones de la Informática, UCLV, Santa Clara, Cuba

{ldp,ataboada,amed}@uclv.edu.cu

{lcperez,jportal}@uclv.cu

Abstract. The aim of this paper is to solve the problem of disambiguation of authors' names in scientific papers. In particular, it focuses on the problem of synonyms and homonyms. Thus, we often find two or more names written in different forms denoting the same person. Moreover, there may be several authors using the same name. To address both the synonym and homonym problems in scientific papers, we propose a framework that uses a hybrid approach of an ontological model and a deep learning model. First, we describe the design of the ontology model, the automatic ontology creation process, and the construction of a weighted co-author network through a set of semantic rules and queries. Second, the selected features are preprocessed during the attribute engineering process to measure the similarity indicator for each feature. Third, the similarity indicators are reduced to a vector space model and used as input to the Deep Learning-based author name disambiguation method to model different types of features. Fourth, the proposed framework is tested on smaller groups of the gold standard large dataset of scientific papers from several international databases named LAGOS-AND and achieves promising results compared to other similar solutions proposed in the literature.

Keywords: Author name disambiguation · deep learning · framework · ontology · scientific papers.

* This work is partially supported by Project 3 "ICT supporting the educational processes and the knowledge management in higher education (ELINF)" of the NETWORK University Cooperation "Strengthening of the role of ICT in Cuban Universities for the development of the society". We thank Carlos Alberto Morell for his useful suggestions and ideas and the team of Li Zhang, Wei Lu and Jinqing Yang for providing the corpus used to train the Doc2Vec model of the gold standard dataset LAGOS-AND.

1 Introduction

Author name disambiguation (AND) can occur in two different forms: (1) when two or more names are written in different forms but represent the same person (name variety problem, also called synonyms), and (2) when multiple authors have the same designation name but represent different people (polysemy, also called homonyms). According to [1], the coincidence of both problems is called the name mixture problem and is most common in real-world datasets. In digital libraries, both problems occur together and manifest themselves in the description of scientific papers and in bibliographic metadata. Currently, AND is very common in scientific publication data. With the rapid growth of scientific publications and authors, AND is becoming increasingly important for data cleaning in scientific network analysis and mining [2]. Author names are ambiguous because they may be written in different forms, abbreviations may be used, typos may occur, a person's name may change after marriage in some countries, norms for author names in journals vary, and some people use the same name designation. All these aspects affect the search for information about these author names. To solve the AND problem, there are important contributions that use an author number or code. There are several databases (e.g., Scopus, PubMed, Web of Science), publishers (e.g., Elsevier, PLoS, Thomson Reuters, Nature, Wiley), manuscript submission systems (e.g., ScholarOne), research and professional associations (e.g., ACS, IEEE, AAAS), and others that use a unique researcher number ID to solve the problem of author ambiguity through proprietary identification systems. Some examples of these proprietary identification systems are Scopus Author ID, ResearcherID or Open Researcher and Contributor ID (ORCID). However, there is a large subset of author names that do not show up in any of these systems because a large number of publications and conferences do not yet ask authors for their ORCID ID (or other proprietary identification system), or they have not asked for it in the past (which is obvious for older publications). Authors may also be submitting incorrect metadata information to the system [1]. The AND problem is still being researched to improve the quality measurements of the new solutions. Numerous approaches have been used to solve the AND problem. In order to locate the main approaches in the literature, several authors AND conducted surveys and reviews that classified the different approaches as follows:

- In [3], the authors proposed a AND taxonomy to classify the AND techniques. The taxonomy is divided into two main categories: machine learning techniques and non-machine learning techniques. Machine learning techniques include supervised, unsupervised, and semi-supervised techniques. Non-machine learning techniques include graph-based and heuristic techniques.
- In [4], the authors classified the existing methods of AND into two different categories depending on their main approach: Author Grouping and Author Assignment. Author grouping methods attempt to group author records from the same author based on some sort of similarity in their attributes, including heuristic, graph-based, and methods that use string matching strategies.

Author attribution methods aim to directly attribute authorship to respective authors using either a classification or a clustering technique. Alternatively, the methods can be grouped according to the evidence studied in the disambiguation task, namely citation attributes (only), web information, or implicit data that can be extracted from the available information. The categories in this taxonomy are not completely disjoint; some methods use two or more types of evidence or mix approaches.

- In [1], the authors categorized the methods of AND into five types: (1) supervised learning, (2) unsupervised learning, (3) semi-supervised learning, (4) graph-based, and (5) ontology-based. They also explained the advantages and disadvantages of using these methods. The authors expressed that the two less explored methods are graph-based and ontology-based, especially the latter one that allows semantics to be added to the disambiguation process. The papers analyzed in this survey are from the period between 2004 and 2016, which means that some important contributions from the last five years are missing.

Despite the different classification methods of the AND techniques, all of them agree in their goal of grouping each author with their corresponding publications, dealing in some way with problems of both synonyms and homonyms. To our knowledge, we prefer a hybrid approach to solve the problem AND through a hybrid solution. Therefore, the main contribution of this work is to develop a framework that combines an ontological model to represent authors, publications, and a weighted co-author network created by semantic rules with deep learning techniques in smaller groups of the gold standard large dataset of scientific papers from several international databases called LAGOS-AND [5].

The rest of this paper is organized as follows. In Section 2, we provide an overview of related work. In Section 3, we describe our framework in detail and formalize the AND problem from the perspective of the ontological model and the deep learning techniques used. In Section 4, we then evaluate the overall framework and validate it in terms of its implications for research and practice. In Section 5, we conclude the paper and provide directions for future research.

2 Related Works

In information science, ontology is a set of concepts and categories in a subject area or domain, showing their properties and the relationships between them. In other words, it is the knowledge representation of a domain. Ontologies are a fundamental artificial intelligence tool for knowledge-based systems (KBS) development. With its formal and well-defined structure, an ontology provides a machine-understandable language that enables automatic reasoning for problem solving. Typical KBSs are expert systems and decision support systems [6].

2.1 Ontology-based AND

Ontology-based AND has been used by many researchers in various fields. Examples: Authority control of individuals and organizations [7], person identities in

linked open data [8], scale-free collaboration networks [9], ontology-based cross-language information retrieval system Tamil-English [10], ontological framework for information extraction with fuzzy rule base and word sense disambiguation [11], etc. Especially in digital libraries or databases, researchers have used this kind of method less. For example, in [1], the authors present a summary table that analyzes only two works that propose an ontology-based solution for author name disambiguation. The papers are [12] and [13]. According to [1], [12] focuses on entity disambiguation using an ontology-based method, background knowledge, and attributes such as authors, conferences, and journals. In [12], data from DBLP and a corpus from DBWorld were used to prove the results using a largely populated ontology. The main limitation of this work is that it needs to be tested on more robust platforms.

On the other hand, [13] addresses the problem of sharing names through OnCu ontology-based categories using the author ontology and the domain ontology of computer science. In [13], collected contributions from AAAI, ISWC, ESWC, and WWW conference websites were used to perform their evaluation based on category usage over the created ambiguity dataset. The main limitation of this work is that it does not consider property relations.

The publication year of both works is more than 10 years ago and their scope is limited. However, we have adopted some of the features proposed in [1] to build their summary table, and present in Table 1 with updated information about recent works that have used ontology-based methods to solve author name disambiguation in the last 10 years.

In summary, the models of ontologies that disambiguate author names have solved many of the problems in isolation and in more specific contexts (see Table 1). Following this analysis, we believe that the development of a new ontology is necessary that combines the context of research and authority control in libraries.

Table 1: Summary of ontology-based methods for AND in the last 10 years.

Ref.	Tool/Method	Features	Findings	Limitations
[14]	Ontology-based personal name disambiguation (OnPerDis) for Chinese personal names on the web	Name, basic information, introduction, contact, and personal relationship	The approach achieves good performance in the three categories of disambiguation of personal names. The F-scores of the approach improve by more than 4%, 5.51%, and almost 9.8%, respectively	More instances of person ontology need to be added to the knowledge base of OnPerDis. Also, the mapping relationships between English names and corresponding Chinese names need to be investigated. 3 experiments were conducted, but they focused more on information extraction than disambiguation of person names
[15]	Researcher Name Resolver (RNR) with a web resource	Researcher name and affiliation, external direct links and external search links	RNR constructs researcher URIs to display researcher pages with profiles and links to related external resources	Administrative staff appropriately engage with researcher profiles and maintain researcher profiles in their daily work as researchers. The method has not been compared to any other in the literature
[2]	A semi-supervised framework for AND in academic social networks that addresses both synonym and homonym issues	Co-author information, title, year, publisher, keywords, affiliation, and topic information	A self-learned method is proposed to solve the ambiguity of co-author information to improve the performance of other models	LDA topic inference is the most time-consuming method in the proposal, about 34 hours. The authors tested different combinations of the comparison methods, but were not compared with other similar AND works
[16]	PDF2TXT, Semantic Fingerprint Generator, Comparator, Claim Decision Maker, Publication Assignment, Arbiter	Metadata is used to extract information about co-authors and institutions, while text data is used to fingerprint	The method introduced semantic fingerprint integrated with co-author features and institution features to AND problem	The size of the dataset was too small and the recall index was low. The method may not work for two authors with the same name and research areas. The method was not compared with other methods from the literature. It was only tested with 7 Chinese author names. An ontology-based solution approach for collecting, displaying and managing researcher profiles

[17]	An ontology-based solution approach for capturing, displaying and managing researcher profiles	Publication title, author name, email, department, keywords, publication year, volume, etc.	Semantic rules implemented to find collaborations between professors	Similarity indicators between analyzed attributes are not considered, only exact matches. The ontology can be enriched with further semantic relations containing summaries and keywords of publications, researchers and topics of interest
[18]	Rule-based binary Classifier and hierarchical agglomerative clustering approach. Reclassification of existing publications from MAKG into a set of 19 disciplines	Author name, Affiliation, Co-authors, Title, Years, Journals and Conferences, References	The evaluation showed that ComplEx is the best large scale entity embedding method we could apply to the MAKG	for trained entity embedding, future research could generate embeddings with higher dimensionality. The main challenge of the task lies in the hardware requirements for training embedding at such a large scale
[19]	Framework Literally Author Name Disambiguation (LAND)	Author names used to get LNFI blocks sorted. Title and Publication date used for comparisons	Benchmark dataset that defines an SCC compliant with the Open Citations Data Model and another SCC (named AMiner-534K). LAND The draft addresses data within knowledge graphs	Includes author collaboration and network information along with the AND network information along with the topic of interest/expertise, which is obtained by processing the authors' publications extracted using deep learning approaches. With this additional data, they can test whether they can use the results for the task of AND

2.2 Deep learning-based AND

On the other hand, [20] investigated how deep neural network (DNN) results can be used to form new clusters and how contributions can be assigned to existing clusters. In the first case, the authors obtained an F1 value of 48.0% for the optimal cut-off in each cluster, while in continuous clustering they obtained an F1 value of 84.3%. The main drawback of their approach is the need for pairwise preprocessing, which scales in quadratic order with the number of contributions in each clustering. This caused most of the computation time in their study. Moreover, [20] focused only on disambiguation of homonymous authors, and when they also consider synonymy of names, the number of pairs increases further.

In contrast to [20], [21] proposed an author identification method in bibliographic data that uses DNN to solve both synonyms and homonyms. The method solves the synonym problem better than the homonym problem; moreover, its performance on the combined synonym-homonym problem is not yet satisfactory. The complexity of detecting and assigning publications to their respective authors is not an easy task. The results show that neural networks with one layer significantly outperform other classical machine learning methods such as Naïve Bayes (NB), Random Forest (RF), and Support Vector Machines (SVM) in average accuracy. Moreover, for homonyms and homonym synonyms, a suitable method should be implemented in other datasets to improve the performance. The use of feature engineering based on a semantic approach for title attributes could improve performance in all cases. In addition, [21] authors confirm that the use of deep neural networks is usually very helpful for working with larger datasets.

Although some work has been developed using the deep learning approach, we believe that this approach has not yet been sufficiently exploited, as well as the possibilities that this approach offers in combination with knowledge extracted from an ontology to solve the AND problem. Therefore, in this paper we describe the development process of the author's name disambiguation ontology (AND ontology) that enables the representation of intelligent system elements using Deep Learning.

3 Materials and Methods

In this paper, we propose a framework for solving the problem of author disambiguation in scientific papers. Fig. 1 shows our framework. In the first phase (1-3), the information from the dataset is imported in CSV format and using R2RML mapping language, the data is transformed into its semantic type corresponding to the ontology model AND previously imported in the W3C Web Ontology Language (OWL). Then, the URI are created and finally the generated triplets are presented in Resource Description Framework (RDF) format as output to be loaded into the AND ontology with all the injected data. The second stage (4) deals with the ontology model AND, the construction of co-author relationships through semantic rules. We also note that attributes such as co-author

frequency, total co-author frequency, and normalized weight can be very helpful in supervised disambiguation. The author's full name, title, location, organization, abstract, and co-author information play an important role in solving the problem AND. The third stage (5-6) is used to preprocess the data to calculate the similarity indicators for each attribute. Finally, the fourth stage (7) uses the similarity indicators and other vectorized data to train the deep learning model, which in turn feeds back the AND ontology through a data transformation tool. The process concludes with the feedback of the AND ontology, and the cycle in the framework is run once.

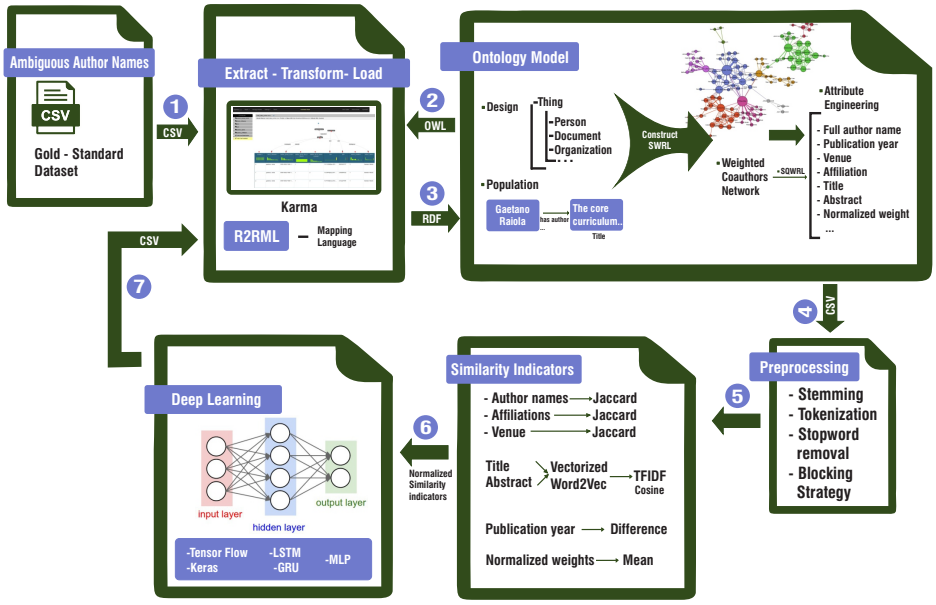

Fig. 1: Framework for Author Name Disambiguation.

3.1 Dataset

We use the LAGOS-AND [5] to support our results, as it accounts for problems with synonyms as well as homonyms, while other datasets are unlikely to provide the former. LAGOS-AND is a recompilation of the following 13 known datasets: Aminer-Rich, Aminer-Simple, Aminer-WhoisWho, Aminer-Zhang, BDBComp-Cota, DBLP-CiteSeerX, DBLP-GESIS, DBLP-Kim, DBLP-Qian, PubMed- GS, PubMed-Kim, REXA-Cullota, and SCAD -zbMATH-Muller. In [5], the authors present a method to automatically generate a large labeled dataset for author name disambiguation (AND) in academia by using authoritative sources, ORCID and DOI. This dataset contains 7.5 million citations from 797,000 unique authors

and shows great similarities to the entire Microsoft Academic Graph (MAG) for six gold standard validations. All datasets included in this large gold standard dataset are freely available for academic use without additional restrictions. In this work, the authors investigated the long-standing problem of name synonyms and showed for the first time the degree of variation in surnames. To accurately capture author similarity, the authors in [5] converted "block-based datasets" to "author-pair datasets", and the signal monitored (0/1) is whether the paired instance represents the same author (1) or not (0). Some homonymous authors (0) and synonymous authors (1). Consequently, the "paired dataset" consists of 500K instances, half of which are positive.

3.2 Ontology Model

In this subsection, the ontology model development process is explained with the phases of requirements definition, vocabulary selection for reuse, ontology implementation and integration, ontology evaluation, documentation, and maintenance. The ontology building process followed the regulations for ontology development and the Neon specifications [23], [24] through an application corresponding to an ontology for disambiguation of author names in scientific papers called AND Ontology. One of the goals of the AND ontology is to provide a common representation of data for author name disambiguation. Each element of the ontology (class, property) must be named with only one term to avoid semantic heterogeneity. The language of the ontology will be OWL-2. Scope of ontology. One of the ways to determine the scope of ontology is to make a list of questions that such a system should answer [25]. In the thematic domain of author name disambiguation, some of the possible questions that should be answered by the ontology are the following:

1. In which places has author X published?
2. What is the alias of author X?
3. Which authors collaborate with author X?
4. How many contributions has author X made?
5. What is the affiliation of author X?
6. What is the preferred name of author X?
7. How many authors write a publication P?
8. Did author X write the publication with the title T?
9. What is the exclusivity of publication P?
10. Which authors have a weighted co-author ratio greater than a threshold X?

Judging by this list of questions, the ontology must contain information about the different types of documents and publications, as well as about the people or actors who authored these documents, their organization, and their collaboration through co-author relationships, among others. The development of the AND ontology is based on different vocabularies that allow the use and reuse of classes and properties from other ontologies. In the field of author name disambiguation in scientific papers, there are many applications based on ontologies with few design values. For this reason, languages that provide clues for formalizing ontologies have been selected and are listed below:

- FOAF[3]. It describes the characteristics of people and social groups that are independent of time and technology. FOAF defines the classes for person, organization, and the subclass OrgUnit refers to Department, Faculty, Research center, and College.
- SKOS[4]. It describes simple knowledge organization for the web. SKOS defines the class Concept.
- GEO[5]. It describes the vocabulary for building geographic ontologies and geospatial data. GEO defines the Country class and the Place subclass.

We also reuse several ontology systems developed for the domain, whose conceptual quality is sufficient for the development of other ontological schemes. The ontologies that best describe the domain and are most complete were used for this design. The ontologies selected for reuse are:

- BIBO[6]. It describes the characteristics of bibliographic records such as the class Document and the subclasses Conference Proceeding, Event, Journal, and Publication.
- VIVO[7]. It presents researchers in the context of their experiences, outcomes, interests, accomplishments, and related institutions.
- GND[8]. Used to describe the name of the person and variant names of the person. This ontology takes into account the Authority Resource and the Anglo-American Cataloguing Rules (AACR2).

The concepts we have declared in this section were selected using AgreementMaker, a software tool that allows us to map ontologies and determine whether there is similarity of terms and equality in the order of the hierarchical structure of classes. AgreementMaker helped us to find not only similar classes, but also properties related to specific concepts that appear in an ontology. This tool allowed us to identify unique terms that are not polysemous or homonymous. The following criteria were used to select the terminological concept base of the ontology:

1. Selection of the class whose hierarchy best describes each concept associated with disambiguation of author names in scientific papers.
2. Selection of classes whose annotations and definitions were accepted by IEEE.
3. The terms of other ontologies associated with the domain are used to establish synonymy relations within the ontology.
4. A base ontology is taken to integrate ontologies into it to build the domain.

This approach to ontology organization uses a mixed solution: symmetric and asymmetric.

[3] http://xmlns.com/foaf/0.1/
[4] http://www.w3c.org/2004/02/skos/
[5] http://www.w3c.org/2003/01/geo/
[6] https://purl.org/ontology/bibo/
[7] https://bioportal.bioontology.org/ontologies/VIVO
[8] https://d-nb.info/standards/elementset/gnd

Semantic rules defined Following [22], we construct a co-authorship relation between the individuals whose names are the same or very similar to determine whether or not they are the same person. We examined the six semantic rules proposed by [22] and found several drawbacks, such as that Rule 2, Rule 4, and Rule 5 have conceptual flaws because they use SQWRL to satisfy the rule and then assign the result to an object property or a data property of their ontology, which is not allowed in this language. The result could be a query but not a rule and they are expressed like semantic rules. If we type it in SWRL tab in Protégé, the system immediately recognizes it as a query. Taking this into account, we adapt their idea and change some rules and queries to construct the co-authorship relations.

Rule 1. Calculate the co-authorship relationship.

$bibo : Document(?d) \wedge and{:}nrAuthors\ (?d, ?nr) \wedge swrlb{:}greaterThanOrEqual\ (?nr, 2) \wedge and{:}hasAuthor(?d,\ ?a1) \wedge and{:}hasAuthor(?d,\ ?a2) \wedge and{:}hasURI(?a1,\ ?a1URI) \wedge and{:}hasURI(?a2,\ ?a2URI) \wedge swrlb{:}notEqual(?a1URI,\ ?a2URI) \wedge sameAs(?a1,\ ?a1) \wedge sameAs\ (?a2, ?a2) \wedge swrlx{:}makeOWLThing(?rel,\ ?a1,\ ?a2) \rightarrow and : Co-authorRelation(?rel) \wedge and{:}hasCoauthor\ (?\ a1,\ rel) \wedge and{:}hasCoauthorValue(?rel,\ ?a2)$

Rule 3. Calculate the exclusive co-authorship for a given document.

$bibo : Document(?d) \wedge and{:}nrAuthors(?d,\ ?nr) \wedge swrlb{:}greaterThanOrEqual(?nr,\ 2) \wedge swrlm{:}eval(?e,\ "1/(nr\text{-}1)",\ ?nr) \rightarrow and : hasExclusivity(?d, ?e)$

Rule 6. Calculate the co-authorship weight.

$and : hasCoauthorValue(?rel,\ ?a2) \wedge swrlm{:}eval(?w,"a12f/totalFa1",\ ?a12f, ?totalFa1) \wedge and{:}hasCoauthor(?a1,?rel) \wedge and{:}hasTotalFrequency(?a1,?totalFa1) \wedge and : hasCoauthorFrequency(?rel, ?a12f) \rightarrow and : hasCoauthorWeight(?rel, ?w)$

Query 1. Show the co-authorship relationship between pairs of authors.

$bibo : Document(?p) \wedge and{:}nrAuthors(?p,\ ?nr) \wedge swrlb{:}greaterThanOrEqual\ (?nr, 2) \wedge and{:}hasAuthor(?p,\ ?a1) \wedge and{:}hasAuthor(?p,\ ?a2) \wedge and{:}hasURI(?a1,\ ?a1URI) \wedge and{:}hasURI(?a2,\ ?a2URI) \wedge swrlb{:}notEqual(?a1URI,\ ?a2URI) \wedge sameAs(?a1,\ ?a1) \wedge sameAs(?a2,\ ?a2) \wedge swrlx{:}makeOWLThing(?rel,\ ?a1,\ ?a2) \rightarrow sqwrl : select(?a1, ?a2, ?rel)$

Query 2. Show the number of authors for each document.

$bibo : Document(?d) \wedge and{:}hasAuthor\ (?d, ?a) \cdot sqwrl : makeSet(?s, ?a) \wedge sqwrl{:}groupBy(?s,?d)\ \cdot sqwrl : size(?size, ?s) \rightarrow sqwrl : select(?d, ?size)$

Query 3. Show the exclusive co-authorship for a given document.

bibo : *Document(?d)* ∧ *and:nrAuthors* (*?d*, *?nr*) ∧ *swrlb:greaterThanOrEqual* *(?nr, 2)* ∧ *swrlm:eval(?e," 1/(nr-1)",?nr)* → *sqwrl* : *select(?d ?e)*

Query 4. Show the frequency of co-authorship.
bibo:Document(?p) ∧ *and:nrAuthors* (*?p*, *?nr*) ∧ *swrlb:greaterThanOrEqual(?nr, 2)* ∧ *and:hasAuthor(?p,?al)* ∧ *and:hasAuthor* (*?p,?a2*) ∧ *and:hasURI(?a1, ?a1URI)* ∧ *and:hasURI(?a2, ?a2URI)* ∧ *swrlb:notEqual(?a1URI, ?a2URI)* ∧ *sameAs(?a1, ?a1)* ∧ *sameAs(?a2, ?a2)* ∧ *and:hasExclusivity(?p, ?e)* ·*sqwrl* : *makeSet*(*?s*, *?e*) ∧ *sqwrl:groupBy* (*?s*, *?a*1, *?a*2) · *sqwrl:sum* (*?f*, *?s*) → *sqwrl* :*select*(*?a*1, *?a*2, *?f*)

Query 5. Show the overall frequency of co-authorship.
foaf : *Person(?a1)* ∧ *and:hasCoauthor*(*?a*1, *?rel*) ∧ *and:hasCoauthorValue* (*?rel*, *?a*2) ∧ *and:hasCoauthorFrequency(?rel,?f)*·*sqwrl* : *makeBag*(*?s*, *?f*) ∧ *sqwrl:groupBy(?s,?a1)*·*sqwrl* : *sum*(*?totalFa*1, *?s*) → *sqwrl* : *select*(*?a*1, *?totalFa*1)

Query 6. Show the co-authorship weight.
and : *hasCoauthor(?a1, ?rel)* ∧ *and:hasCoauthorValue* (*?rel*, *?a*2) ∧ *and:hasTotalFrequency* (*?a*1, *?totalFa*1) ∧ *and:hasCoauthorFrequency(?rel, ?a12f)* ∧ *swrlm:eval(?w,"a12f/totalFa1", ?a12f, ?totalFa1)* → *sqwrl* : *select*(*?rel?*, *?w*)

For the construction of the graph of co-authorship relations, we refer to the definitions of the directed weighted co-authorship graph model presented in [22], which we have adopted for our framework to determine the weights and other key metrics of co-authorship relations. Let $A = a_1, ..., a_n$ denote the set of n authors. Let the set of m publication be denoted as $P = p_1, ..., p_k, ...p_m$. Let $f(p_k)$ define the number of authors of publication p_k. Then we used the definitions introduced by [22].

Definition 1 (Exclusivity per publication). If authors a_i and a_j are co-authors in publication p_k, then $g_(i, j, k) = 1/(f(p_k) - 1)$. $g_(i, j, k)$ represents the degree to which authors a_i and a_j have exclusive co-authorship for a given publication. In this definition, the relationships between co-authors are weighted more heavily for publications with a smaller total number of co-authors than for publications with a large number of co-authors.

Definition 2 (Co-authorship frequency). Another important metric for a pair of authors a_i and a_j, is the frequency of co-authorship: $c_{(i,j)} = \sum_{(k=1)}^{m} g_{(i,j,k)}$ it sums the exclusivity values $g_{(i,j,k)}$ for the same pair i, j over all publications k, $(k = 1..m)$ in which they appear as co-authors. This gives more weight to authors who publish more publications jointly and exclusively.

Definition 3 (Total co-authorship frequency). Consists of the sum of all co-authorship frequency values c_{ik} over a given author a_i and all his co-authors a_k $(k = 1..n)$ in in all publications in which a_i appears as an author $c_i = \sum_{(k=1)}^{n} c_{ik}$.

Definition 4 (Normalized weight). To obtain a normalized value for the weight of co-authorship between two authors, the following normalization step should be performed, in which the total co-authorship frequency of a given author is

taken into account when calculating the co-authorship frequency between that author and every other co-author by him: $w_{ij} = c_{ij}/c_i$. This ensures that the weights of an author's relationships sum to one.

Weighted Co-authorship Network We have defined the co-author network as a set of nodes vi and edges ei, where the node vi represents an author or co-author name and ei represents the co-author relationship in each document. The co-author network is constructed using the semantic rule R1. Then, we calculate the weight of each ei using the semantic rule R6 which depends on the other rules (R2, R3, R4 and R5). Since R2, R4 and R5 are queries, we need to compute these values using an external programming language or shop the results in a CSV file and insert the results into the ontology using the Karma integration tool. Either variant is a viable option. Part of the co-author relationship in the AND Ontology graph is shown in Fig. 2.

3.3 Preprocessing

The preprocessing step aims to prepare the data for the next steps. This usually includes standardization of author names as well as removal of stop words, tokenization, and stemming of work titles and abstracts. Then, the attributes are vectorized following the procedure of [5], using the Doc2Vec model of [27]. A similar procedure is performed for the venue and affiliation attributes: First, they are converted to lowercase, stop words and special characters are removed, and the Bag of Words is extracted. In the LAGOS-AND dataset, all ambiguous authors in a block have the same Credible Full Names (CFN), regardless of which identified author groups he/she belongs to. This rule makes the dataset more challenging than other LN (Last Name) - or LNFI (Last Name First Initial)-based datasets.
In [5] the authors considered using the CFN instead of LNFI or FN to further aggregate the identified author group into blocks. This dataset has a similar structure to the existing block-based datasets. However, unlike them, blocks arranged in this way have two major advantages. First, it is more challenging to disambiguate CFN blocks than LNFI blocks. In LNFI blocks, the authors may largely be known by completely different full names; for example, in the "Freyman_R" block, different authors "Richard Freyman" and "Robin Freyman" may exist, which would simplify the dataset and lead to unnecessary computations. For CFN blocks, e.g., "Freyman_Richard", it is usually more difficult to disambiguate, since all authors in this block can be named "Richard Freyman". Second, unlike FN blocks, CFN is more authoritative for representing the block. In the ORCID system, CFN can only be maintained by the author, which is displayed directly in the ORCID interface without intermediate processes. Note that there seems to be no better way to accurately identify an actual name than to retrieve the author-maintained name (CFN) [5].

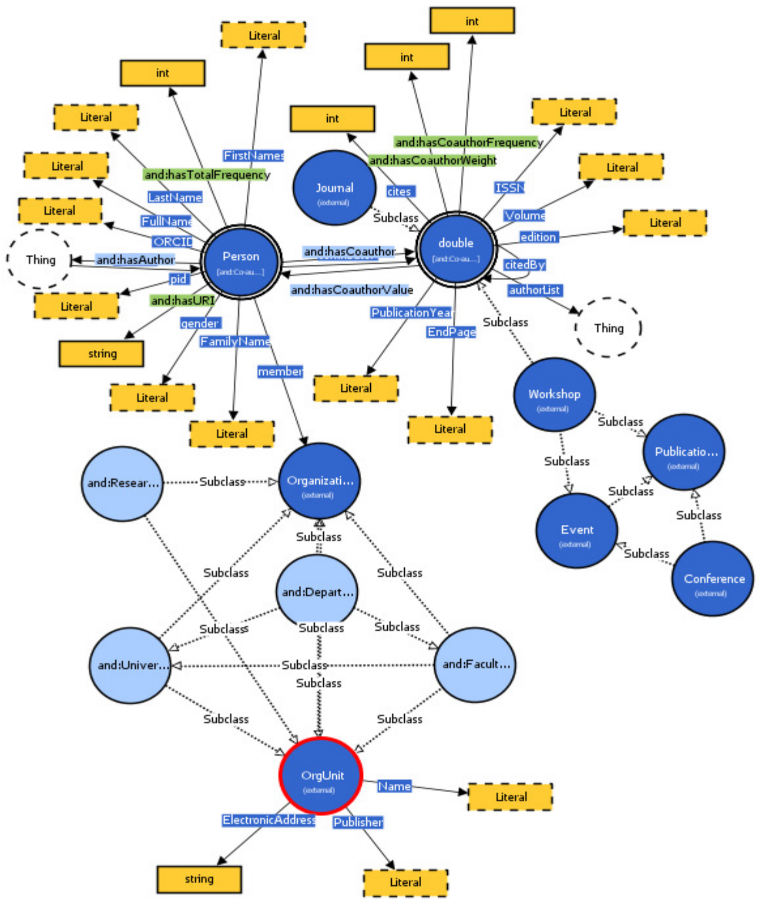

Fig. 2: Part of the AND Ontology Graph.

3.4 Similarity Indicators

The similarity indicator for each selected attribute is shown in Table 2. The first six attributes are the same used in [5], and we add the normalized weight attribute related to the network of co-author relationships. An appropriate measure or model is used to determine a similarity value for each attribute. For the features full author name, publication year, venue, and affiliation, a common word-level measure, i.e., Jaccard, is used. For content-based features such as title and abstract, TFIDF and a representation learning model are used in addition to Jaccard, i.e., Doc2vec [27] to determine the similarities.

Table 2: Similarity Indicator of each attribute selected for AND. Based on [5].

Attribute	**Data Type**	**Similarity Indicator/Dependent Model**
Full author-name	String	Jaccard (2- gram) char-level
Publication year	Integer	Normalized Absolute difference
Venue	String	Jaccard word-level
Affiliation	String	Jaccard word-level
Title	String	Jaccard word-level, TFIDF, Doc2vec, neural network
Abstract	String	Jaccard word-level, TFIDF, Doc2vec, neural network
Normalized weight	Double	Mean

3.5 Deep Learning

Deep Neural Networks (DNNs) can represent higher complexity functions, and according to the results reported in [5], DNNs seem to be a great solution to the problem AND. In this work, Gated Recurrent Unit (GRU), Long Short-Term Memory (LSTM) and Multilayer Perceptron (MLP) models are compared to select the best performing algorithm to be included in the proposed framework.
Since the LAGOS-AND dataset is huge and a solution with DNNs is computationally expensive, the simulations are performed with small datasets. We start by randomly forming four disjoint groups with 250, 500, 1000, and 1500 author names, respectively, from the "pairwise dataset". Note that the following steps are performed for each set so that a solution is given for each problem and then we compare the results. We construct the co-authorship graph according to the "block-based dataset", considering not only the previously selected authors, but all co-authors who share at least one publication with them. We then calculate the exclusivity per publication, the co-authorship frequency and the total co-authorship frequency. These metrics are needed to calculate the normalized weight representing the co-authorship relationship between each pair of authors. Since the relationship is bidirectional, there are two values for each pair of authors. Therefore, a single value is used to represent the entire relationship, and this is added to the similarity indicators. For each pair of authors, the mean value

between the normalized weights in both directions is selected as the representative value (see Table 2). The inputs to the classifier come from the preprocessed features of the previous stage. The goal is to determine whether two authors are the same or not for each instance in the training dataset.

4 Results and Discussion

We implement the proposed framework in Python using our data, which is partially based on [5], but the proposed framework and the model used are different. Protégé is used to design the AND ontology and to test the semantic rules and queries implemented in a small part of the LAGOS-AND dataset. The Karma data integration tool is used to convert the LAGOS-AND dataset, previously in CSV format, into RDF format using the R2RML mapping language. Then, Stardog triple store is used to host the data portions of the LAGOS-AND dataset in RDF format. Later, the similarity indicators need to be computed to feed the deep learning model, which is trained with the ultimate goal of determining whether two presented authors are the same or not. All experiments were run on a cluster with two x Intel Xeon E5-2630 v3 (Haswell) 16 cores 2.4 GHz CPU and 128 GB RAM 4 x 1 Ethernet GB.

4.1 Experimental results

We designed our experiments with four data partitions of 250, 500, 1000, and 1500 author names, extracting instances from the "pairwise dataset" that includes authors in the co-authorship graph. In each set of instances, we use 60% for training and 20% for validation. During training, we set some hyper-parameters for each model (GRU, LSTM and MLP). We tune the number of hidden layers (1, 2, 3), the number of input units for each layer (32, 64, 128), the learning rates (0.1, 0.01, 0.001), the activation functions for the hidden layers (Rectified Linear Unit (ReLU), Scaled Exponential Linear Unit (SELU), hyperbolic tangent), and the use or non-use of dropout (0, 1, 0.5). This tuning of hyperparameters is performed in the training phase, and the configuration with the best performance for each model is selected for re-training, where the training and validation sets (80% of all data) are merged. Finally, testing is performed on the remaining 20% of the data to select the best algorithm. It is worth noting that there are some aspects in the models that we do set as fixed values, such as the optimizer (Adam), the loss function (binary cross entropy), the metric (accuracy), the activation function in the last layer (softmax), the stack size (64), and the number of epochs (100).
According to the different combinations of these hyperparameters, the LSTM, GRU and MLP models are trained and their validation performance is compared to select the best representative of each model. It should be noted that this procedure is performed for each data partition. Table 3 shows the evaluation results of the four experiments and the three models (using the best hyperparameter setting for each), specifying the different partitions of the dataset LAGOS-AND.

The metrics used are in percentages and include accuracy (Acc), precision (Pre), recognition (Rec), and F1 score (F1). The comparison table also uses the Name similarity results presented in [5], MAG author ID and the best model from LAGOS-AND as reference models.

- Name similarity: It is a basic method that uses only name differences to disambiguate authors [5].
- MAG Author ID: The ID system is disambiguated by the Microsoft Academic research team for its over 560 million authorships [5].
- LAGOS-AND: The best model identified in [5] is based on features and content characteristics (bf+cfnn)) that are used in the content similarity score by the neural network.

Table 3: Evaluation results of methods for AND. Based on [5].

Method-#Authors	**Accuracy(%)**	**Precision(%)**	**Recall(%)**	**F1 score(%)**
MLP-250	80.68	83.32	62.24	71.25
MLP-500	81.02	84.63	62.56	71.94
MLP-1000	81.25	85.05	64.12	73.12
MLP-1500	82.37	86.56	65.71	74.71
GRU-250	79.68	89.97	72.21	80.12
GRU-500	83.25	89.93	74.54	81.51
GRU-1000	85.54	91.72	77.56	84.05
GRU-1500	85.66	**92.86**	**82.21**	**87.21**
LSTM-250	83.69	87.54	78.02	82.51
LSTM-500	84.24	88.60	78.16	83.05
LSTM-1000	**86.35**	89.20	79.25	83.93
LSTM-1500	89.20	92.03	80.36	85.80
Name similarity	54.79	64.39	21.46	32.19
MAG Author ID	81.87	**98.49**	64.74	78.13
LAGOS-AND	**90.08**	93.23	**86.44**	**89.71**

As it is shown in Table 3, GRU wins on precision, recall, and F1 score, but LSTM has a higher value on average accuracy. GRU wins on precision, recall, and F1 score, but LSTM has a higher value on average accuracy. GRU wins slightly over LSTM, and MLP lags further behind, but LSTM's performance improves as the number of data increases. The best performers for MLP, GRU, and LSTM in the test phase are those with 1500 author names, as shown in Fig. 3. At first glance, we can see a relative proportionality function between the results: As the number of authors increases, the performance of the methods increases slightly.

4.2 Findings

Our method behaves similarly to the method in (LAGOS-AND), but we add semantic rules and queries in our framework to increase the semantic rigor of

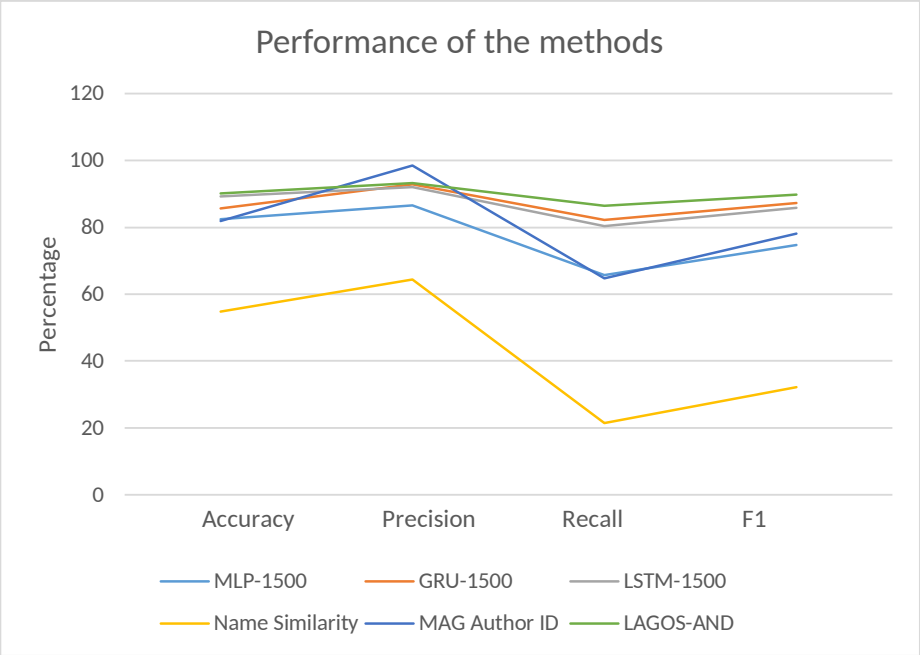

Fig. 3: Performance of the methods based on accuracy, precision, recall and F1.

the weighted co-authorship network. Our contribution helps disambiguate author names in scientific papers, which was not considered in [5]. Although the comparison was performed on different parts of the LAGOS-AND dataset, the results presented are not significantly different from those in [5] (see Fig. 3). The LAGOS-AND dataset is more difficult to disambiguate than other datasets, not only because of its dimensionality, but also because it contains synonyms and homonyms of author names at the same time. In addition, the block technique used is more challenging. However, the results obtained with GRU and LSTM are very close to those presented in [5], with GRU achieving slightly better results.

5 Conclusions

In this work, we have presented a framework for author name disambiguation using a hybrid approach of an ontological model and a deep learning model. We have described the design of the ontology model, the process of automatic ontology generation, and the construction of a weighted co-author network through a set of semantic rules and queries. Then, we preprocessed the selected features during the attribute engineering process to measure the similarity indicator of each feature. The proposed framework was evaluated on four different data portions of the LAGOS-AND dataset using three different deep learning models, which show similar results to those presented in [5], with the GRU model per-

forming slightly better in terms of precision, recall and F1 score. In future work, we will repeat the experiment with the entire LAGOS-AND dataset under better hardware conditions. We will also include other deep learning models in the comparison and apply the framework in other real-world scenarios.

References

1. Shoaib, M., Daud, A. and Amjad, T., "Author Name Disambiguation in Bibliographic Databases: A Survey," arXiv Prepr. arXiv2004.06391, pp. 1–24 (2020).
2. Wang, P., Zhao, J., Huang, K., and Xu, B., "A Unified Semi-Supervised Framework for Author Disambiguation in Academic Social Network". In: Decker, H., Lhotská, L., Link, S., Spies, M., Wagner, R.R. (eds.) Conference 2014, LNCS, vol. 8645, pp. 1–16. Springer International Publishing (2014). https://doi.org/10.1007/978-3-319-10085-2
3. Hussain, I. and Asghar, S., "A Survey of Author Name Disambiguation Techniques: 2010–2016," The Knowledge Engineering Review, vol. 32, pp. 1—24 (2017).
4. Ferreira, A. A., Gonçalves, M. A., and Laender, A. H. F., "Automatic Disambiguation of Author Names in Bibliographic Repositories," In: Synthesis Lectures on Information Concepts, Retrieval, and Services, vol. 12 (1), pp. 1—146. Morgan & Claypool Publishers (2020). https://doi.org/10.2200/S01011ED1V01Y202005ICR070
5. Zhang, L., Lu, W., and Yang, J., "LAGOS-AND: A Large, Gold Standard Dataset for Scholarly Author Name Disambiguation," arXiv Prepr. arXiv2104.01821, pp. 1—27 (2021).
6. Fiannaca, A., La Rosa, M., Gaglio, S., Rizzo, R., and Urso, A., "An ontological-based knowledge organization for bioinformatics workflow management system," EMBnet.journal, vol. 18 (B), pp. 110-–112 (2012).
7. Kurki, J., and Hyvönen, E.: "Authority Control of People and Organizations on the Semantic Web," In: International Conferences on Digital Libraries and the Semantic Web Proceedings, vol. 2 (009). (2009).
8. Pattuelli, M. C.: "From uniform identifiers to graphs, from individuals to communities: what we talk about when we talk about linked person data," In: Challenges and Opportunities for Knowledge Organization in the Digital Age, pp. 571—580. Ergon-Verlag (2018).
9. Kim, J., "Scale free collaboration networks: An author name disambiguation perspective," Journal of the Association for Information Science and Technology, vol. 70 (7), pp. 685–700 (2019).https://doi.org/10.1002/asi.24158
10. Thenmozhi, D., and Aravindan, C., "Ontology-based Tamil-English cross-lingual information retrieval system," Sadhana, vol. 43 (157), pp. 1–14 (2018).https://doi.org/10.1007/s12046-018-0942-7
11. Zaman, G., Mahdin, H., Hussain, K., Rahman, A.-U., Abawajy, J., and Mostafa, S. A., "An Ontological Framework for Information Extraction from Diverse Scientific Sources," IEEE access, vol. 9, pp. 42111–42124 (2021). https://doi.org/10.1109/ACCESS.2021.3063181
12. Hassell, J., Aleman-Meza, B., and Arpinar, I. B.: "Ontology-Driven Automatic Entity Disambiguation in Unstructured Text," In: International Semantic Web Conference Proceedings, pp. 44—57 (2006). https://doi.org/10.1007/11926078_4
13. Park, Y.-T., and Kim, J.-M.: "OnCU System: Ontology-based Category Utility Approach for Author Name Disambiguation," In: 2nd International Conference on Ubiquitous Information Management and Communication Proceedings, pp. 63–68. New York, USA (2008). https://doi.org/10.1145/1352793.1352807

14. Lu, Z., Yan, Z., and He, L.: "OnPerDis: Ontology-based Personal Name Disambiguation on the Web," In: 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT) Proceedings, vol. 1, pp. 185–192. IEEE (2013).https://doi.org/10.1109/WI-IAT.2013.28
15. Kurakawa, K., Takeda, H., Takaku, M., Aizawa, A., Shiozaki, R., Morimoto, S., and Uchijima, H., "Researcher Name Resolver: identifier management system for Japanese researchers," International Journal on Digital Libraries, vol. 14 (1-2), pp. 39–58 (2014). https://doi.org/10.1007/s00799-014-0109-z
16. Han, H., Yao, C., Fu, Y., Yu, Y., Zhang, Y., and Xu, S., "Semantic fingerprints-based author name disambiguation in Chinese documents," Scientometrics, vol. 111 (3), pp. 1879–1896 (2017). https://doi.org/10.1007/s11192-017-2338-6
17. Bravo, M., Reyes-Ortiz, J. A., and Cruz, I.: "Researcher Profile Ontology for Academic Environment," Book section of Advances in Intelligent Systems and Computing, vol. 943, pp. 799–817 (2019). https://doi.org/10.1007/978-3-030-17795-9_60
18. Färber, M., and Ao, L., "The Microsoft Academic Knowledge Graph enhanced: Author name disambiguation, publication classification, and embeddings," Quantitative Science Studies, vol.3 (1), pp. 51–98 (2022). https://doi.org/10.1162/qss_a_00183
19. Santini, C., Gesese, G. A., Peroni, S., Gangemi, A., Sack, H., and Alam, M., "A Knowledge Graph Embeddings based Approach for Author Name Disambiguation using Literals," Scientometrics, vol. 127 (8), pp. 4887-4912 (2022). https://doi.org/10.1007/s11192-022-04426-2
20. Gnoyke, P., and Matta, K., "Author Name Disambiguation by Clustering based on Deep Learned Pairwise Similarities," pp. 0—12, no. May (2020).
21. Firdaus, F., Nurmaini, S., Malik, R. F., Darmawahyuni, A., Rachmatullah, M. N., Juliano, A. H., Nugraha, T. A., and Putra, V. O. K., "Author identification in bibliographic data using deep neural networks," TELKOMNIKA (Telecommunication Computing Electronics and Control, vol. 19 (3), pp. 911–919 (2021). https://doi.org/10.12928/telkomnika.v19i3.18877
22. Ahmedi, L., Abazi-Bexheti, L., and Kadriu, A.: "A Uniform Semantic Web Framework for Co-Authorship Networks," In: IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing Proceedings, no. 2, pp. 958–965 (2011). https://doi.org/10.1109/DASC.2011.159
23. Gómez-Pérez, A., and Suárez-Figueroa, M. C., "NeOn Methodology for Building Ontology Networks: a Scenario-based Methodology," (2009).
24. Suárez-Figueroa, M. C., Gómez-Pérez, A., and Mariano, F.-L., "The NeOn Methodology framework: A scenario-based methodology for ontology development," Applied Ontology, vol. 10 (2), pp. 107–145 (2015).
25. Leiva-Mederos, A., García-Duarte, D., Gálvez-Lio, D., Hidalgo-Delgado, Y., and Senso-Ruíz, J. S: "An Ontological Model for the Failure Detection in Power Electric Systems," In: Iberoamerican Knowledge Graphs and Semantic Web Conference Proceedings, pp. 130–146 (2020). https://doi.org/10.1007/978-3-030-65384-2_10
26. Díaz-de-la-Paz, L., Riestra-Collado, F. N., García-Mendoza, J. L., González-González, L. M., Leiva-Mederos, A. A., and Taboada-Crispi, A., "Weights Estimation in the Completeness Measurement of Bibliographic Metadata," Computación y Sistemas, vol. 25 (1), pp. 117-128 (2021). https://doi.org/10.13053/cys-25-1-3355
27. Le, Q. V., and Mikolov, T.: "Distributed representations of sentences and documents," In: International Conference on Machine Learning Proceedings, arXiv Prepr. arXiv:1405.4053, vol. 32 (2), pp. 1188–1196 (2014).