Available online at www.sciencedirect.com

ScienceDirect

Electronic Notes in Theoretical Computer Science 329 (2016) 149–167

Electronic Notes in Theoretical Computer Science

www.elsevier.com/locate/entcs

Detecting Similar Areas of Knowledge Using Semantic and Data Mining Technologies

Xavier Sumba[a,1], Freddy Sumba[a,2], Andres Tello[a,3], Fernando Baculima[a,4], Mauricio Espinoza[a,5] and Víctor Saquicela[a,6]

[a] *Department of Computer Science, University of Cuenca, Cuenca, Ecuador*

Abstract

Searching for scientific publications online is an essential task for researchers working on a certain topic. However, the extremely large amount of scientific publications found in the web turns the process of finding a publication into a very difficult task whereas, locating peers interested in collaborating on a specific topic or reviewing literature is even more challenging. In this paper, we propose a novel architecture to join multiple bibliographic sources, with the aim of identifying common research areas and potential collaboration networks, through a combination of ontologies, vocabularies, and Linked Data technologies for enriching a base data model. Furthermore, we implement a prototype to provide a centralized repository with bibliographic sources and to find similar knowledge areas using data mining techniques in the domain of Ecuadorian researchers community.

Keywords: Data Mining, Semantic Web, Linked Data, Data Integration, Query Languages.

1 Introduction

The number of publications is rapidly increasing through online resources such as search engines and digital libraries, making more challenging for researchers to pursue a topic, review literature, track research history because the amount of information obtained is too extensive. Moreover, most of the academic literature is noisy and disorganized. Currently, certain information about researchers and their bibliographic resources are scattered among various digital repositories, text files or bibliographic databases.

[1] Email:xavier.sumba93@ucuenca.ec

[2] Email:freddy.sumbao@ucuenca.ec

[3] Email:andres.tello@ucuenca.edu.ec

[4] Email:fernando.baculima@ucuenca.edu.ec

[5] Email:mauricio.espinoza@ucuenca.edu.ec

[6] Email:victor.saquicela@ucuenca.edu.ec

http://dx.doi.org/10.1016/j.entcs.2016.12.009

When you need to propose projects with several researchers in a specific area belonging to different Higher Education Institutions (HEI), different questions arise. For instance, who works in similar areas of research? or, how can I create a network of researchers in a common knowledge area? Then, detecting similar areas based on the keywords it could help governments and HEI to detect researchers with interests in common, opening an opportunity to generate new research projects and allocate efforts and resources to them. In that case, we could detect potential collaboration networks.

The expansion of this knowledge base will allow our academic community to have a centralized digital repository which has information of Ecuadorian researchers based in bibliographic resources. The collaborators are identified through a semantic enrichment of scientific articles produced by researchers who publish with Ecuadorian affiliations. This work aims to encourage institutions to collaborate and obtain a semantic repository to identify researchers working in similar areas and, provide updated information accessible and reusable. Enhancing the generation of research networks with academic peers in the region could provide a greater opportunity for collaboration between the participating institutions.

Obviously, there are many tools and services currently available in the web which already provide a wide variety of functionalities to support the exploration of academic data. Each tool or service operates in different ways, that in some cases complicate the literature review or utilization data. These tools or services allow search publications using keywords, author names, conferences, authors affiliations through *Applications Programming Interface (APIs)*. They have started using semantic technologies that helps to describe their resources, but each source is different. Our approach use these characteristics, to retrieve and enrich bibliographic data from several bibliographic sources to detect similar areas.

The rest of this paper is organized in the following way: section 2 presents the related work. We outline the architecture in section 3, detecting similar areas in the domain of Ecuadorian Researchers and detecting potential networks of collaboration, using semantic technologies to enrich data extracted from different bibliographic sources in a common model. Conclusions and future work are presented in section 4.

2 Related Work

This section introduces tools and services used for searching publications, unification of publications, authors disambiguation, and approaches related to the identification of similar research areas.

Some bibliographical sources have tools that allow access to data, but others sources do not have. For example, *Google Scholar* does not have an API that allows an automatic retrieval of publications. *Microsoft Academics Search* provides an API to search for publications, and they also provides a variety of tools for visualizations such as co-authorship graphs, trending publications, and co-authorship paths between authors. However, they have data from 2013, which actually is out-

dated. Recently they released a new version where the main problem is ambiguity of authors. Scopus, also has *Elsevier API*, this source is accessible only under subscription and it has limited requests. *Digital Bibliography & Library Project (DBLP)* offers three available databases (*Trier1, Trier2, Dagstuhl*) through an API, and the data are available in several data formats such as JSON, XML or RDF.

Each bibliographic source have data that can be duplicated or inconsistent. In our case it is necessary to correct ambiguous data before it is stored. In [21], there are two methods of disambiguation of authors, the first one uses the names of the authors and their initials, and the second one is an advanced method that uses initial names and authors affiliation. In [9], presents a framework that uses a clustering method *DBSCAN* to identify the author according to their articles. We analyse the similarity between sets of publications of different authors. If the similarity between these resources is found, the correct author to a specific publication is established.

In [17] proposed the Rexplore System, which uses statistical analysis, semantic technologies, and visual analytics to provide scholarly research data and locate research areas. We use a similar idea, but we will dynamically add new data sources to improve authors information. A similar work is made by [18], which detect potential collaborative networks through the semantic enrichment of scientific articles. However this work has authors from a single source and Ecuadorian affiliation only; while we can present external information when it is needed from various sources. They find similar papers using SKOS [7] concepts, while we used data mining algorithms instead.

In the field of geoscience studies it has shown that is posible to improve data retrieval, reuse, and integrate data repositories through the use of ontologies. For instance in [10], the *Geolink* project, a part of *EarthCube* [8], integrates seven repositories using *Ontology Design Patterns (ODPs)* [6] defined manually. They have a set of ODPs as the overall scheme, rather than using a monolithic ontology. To obtain data they executed federated queries. Conversely, in our proposal all sources form a single repository and we do not use federated queries because the response time is endless. The data model Geolink is defined specifically for geodata, which differs from our proposal covering several domains according to the bibliographic source.

Previous studies finding a relationship between publications have shown that citation data is often used as an indicator of relatedness. Citations are used to measure the impact of documents [7]. Nevertheless, there are other approaches to find related papers, the work of [19] shows that digital records can be used as indicators as well. Collaborative filtering could be used to find related publications too; in the work of [13] they use the citation web between publications to create the rating matrix and recommend research papers. Additionally, relationships based on the citations gives an insight of the hierarchy distribution of publications around a given topic as shown by [1]. Although citations are an excellent indicator to express relatedness, we could not find work in the literature to use keywords as

[7] https://www.w3.org/2004/02/skos/

[8] EarthCube is a community-led cyberinfrastructure initiative for the geosciences; http://earthcube.org/

an indicator to find a relationship between publications and identify common areas using publication keywords.

After having analyzed the related work of approaches that deal with identifying research topics, we can state that the existing works do not automatically enrich bibliographic resources obtained from different sources, such as *Google Scholar* or *DBLP* . Furthermore, we propose the use of data mining algorithms to detect similar areas of knowledge and semantic ontologies to describe and re-use the data extracted and processed.

3 Architecture for detecting similar areas of knowledge

In this section, we describe the detailed aspects of our architecture proposed to enrich academic literature available on the web and find relationships between authors and their publications. Our approach relies on three different main modules, namely: 1) Data Extraction, which describes and stores authors and publications that have several data models. 2) Data Enrichment, which takes publications of each author and enriches them using semantic technologies and 3) Pattern Detection, which makes use of data mining algorithms to detect similar knowledge areas and potential networks of collaboration. The high-level modules of the architecture are illustrated in Fig. 1 and their features will be explained throughout this section. Finally, we provide an SPARQL endpoint [9] for querying authors, publications, knowledge areas, and collaboration networks.

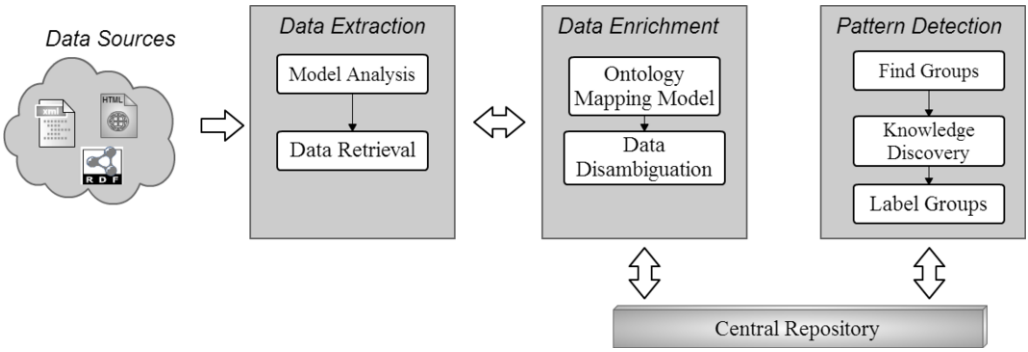

Fig. 1. General architecture to detect patterns from bibliographic data sources.

3.1 Data Sources

We use several data sources available on the web that support the exploration of scholarly data. Some of them provide an interface to a specific repository of bibliographic data, others integrate multiple data sources to provide access to a richer set of data, providing a richer set of functionalities. However, there are two types of bibliographic sources to retrieve data. First, the access is free and the information is available online. Second, access fees are required because they are

[9] http://redi.cedia.org.ec/sparql/admin/squebi.html

provided by major publishers of scientific literature. Then to solve access problem, we use the metadata available.

The different data sources represent repositories that contains information about authors and scientific publications of different areas. The sources about authors are distributed in different DSpace [10] repositories located in various HEI, and those records belong only to Ecuadorian authors. Each repository contains scientific papers, theses, dissertations, books, monographs of researchers or students.

The scientific publications are extracted from bibliographic sources such as Microsoft Academics, Google Scholar, DBLP and Scopus that make available their data via APIs. The data vary in their content due to each source has a different structure. Also, the access to the data is restricted in some cases, for example, in Scopus we can make a maximum of 5000 querys for each IP, then the source locks the access for seven days. Moreover, the sources of publications have not the same fields, for example, Scopus has the following fields: data affiliation of authors, tables, graphs of publications, authors study areas, but *DBLP* or *Microsoft academics* do not have these fields. Therefore, we see that, it is necessary to make a unification of these variety of data models in a common model that describe literature from different domains stored in a central repository.

It is necessary to process the data sources referred above to understand the structure and access of the data. These tasks are described in detail in the next subsection.

3.2 Data Extraction.

The *data extraction* module is responsible for extracting and describing bibliographic data from several sources using semantics technologies and Linked Data practices. The data extracted is analyzed in order to define a structure using the documentation available on the source and if it does not exist, the data model of the source is analyzed using web scraping techniques. After that, the model of data is established, the data is extracted and stored on a triple store, in this case Apache Marmotta [11] . Some sources have their data recorded with a bibliographical ontology defined by the source owner. If the data already is annotated then it is stored directly on the triple store. Otherwise, this data is annotated and stored with BIBO ontology. We use the bibliographic data sources to cover different scenarios and find the main problems involved in the process of the extraction and enrichment of bibliographic resources. Every time a new source is added, we analyze manually the data model and then extract the data. These two processes are encapsulated into components described below.

3.2.1 Model Analysis

The different bibliographic sources provide their resources with a logical structure or with a different data model having the same type of information. Bibliographic

[10] DSpace is the software of choice for academic, non-profit, and commercial organizations building open digital repositories; http://www.dspace.org

[11] http://marmotta.apache.org/

resources are not completely modeled by a standard or comprehensive model encompassing all properties as authors, appointments, conferences, knowledge areas, etc. Some features such as DOI, ISBN, format bibliographic references of resources are described by the International Standard Bibliographic Description (ISBD)[3], ISO 690[12]. *Functional Requirements for Bibliographic Records (FRBR)* [15] recommend a new approach for cataloging based on an entity-relationship model to a bibliographic resource. However, this is not enough to have a common description of bibliographic resources Then, one of the main challenges is to define a common data model to facilitate the processing of scientific publications.

The heterogeneity of models represents the challenge of integrating various sources. Therefore, before adding a new data source we must perform a manual analysis of the data with respect to the models already being used to define how it will perform the extraction of these data and how these is adapted to our common data model. In some cases, the sources do not publish documentation about the data model. We have three forms to find the data model of a source. First, the source provides documentation, second the data model is published in research works such as the data model of *DBLP* as is described in [11]. Finally, we perform HTTP requests by sending as parameters the author names to the source which helps us infer the data structure. The result of this component is data with a defined model for each source.

After analysing the data models, we need to retrieve information from each of the sources. The component described in the section 3.2.2 is responsible for extracting scientific publications by each author.

3.2.2 Data Retrieval

The component retrieves authors and publications using different APIs, web pages or SPARQL endpoints from different bibliographical sources. This component is designed abstractly, with the aim to extract information from any bibliographic source. Listing 1 and 2 illustrates data responses from *Microsoft Academics* and *DBLP*, and those responses have a different format and structure despite being in the same publication. To extract data we use the LDClient[13] library from *Apache Marmotta* that offers several forms to consume XML data from web services or web pages such as *Google Scholar* which does not have an API. The data is processed in memory using a temporary triple store called *Sesame*[14] which adds information about authors and bibliographical sources that later use to discard erroneous information. Finally, the data is stored in the Apache Marmotta triple store.

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:bibtex
    ="http://data.bibbase.org/ontology/#" xmlns:dblp="http://dblp.dagstuhl.de
    /rdf/schema-2015-01-26#" xmlns:dcterms="http://purl.org/dc/terms/"
    xmlns:foaf="http://xmlns.com/foaf/0.1" xmlns:owl="http://www.w3.org
    /2002/07/owl#">
```

[12] ISO standard for bibliographic referencing in documents of all sorts.

[13] http://marmotta.apache.org/ldclient/

[14] Sesame is a powerful Java framework for processing and handling RDF data; http://rdf4j.org/

```
<dblp:Publication rdf:about="http://dblp.dagstuhl.de/rec/conf/icwe/
    SaquicelaVC10">
  <owl:sameAs rdf:resource="http://dblp.org/rec/conf/icwe/SaquicelaVC10"
      />
  <owl:sameAs rdf:resource="http://dx.doi.org/10.1007/978-3-642-16985-4
      _24" />
  <dblp:publicationLastModifiedDate>2010-11-11T07:32:58+0100</
      dblp:publicationLastModifiedDate>
  <dblp:title>Semantic Annotation of RESTful Services Using External
      Resources.</dblp:title>
  <dblp:bibtexType rdf:resource="http://data.bibbase.org/ontology/#
      Inproceedings" />
  <dblp:publicationType rdf:resource="http://dblp.dagstuhl.de/rdf/schema
      -2015-01-26#Inproceedings" />
  <dblp:authoredBy rdf:resource="http://dblp.org/pers/s/Saquicela:Victor"
       />
  <dblp:authoredBy rdf:resource="http://dblp.org/pers/b/Bl=aacute=
      zquez:Luis_Manuel_Vilches" />
  <dblp:authoredBy rdf:resource="http://dblp.org/pers/c/Corcho:=Oacute=
      scar" />
  <dblp:primaryElectronicEdition rdf:resource="http://dx.doi.org
      /10.1007/978-3-642-16985-4_24" />
  <dblp:publishedInBook>ICWE Workshops</dblp:publishedInBook>
  <dblp:pageNumbers>266-276</dblp:pageNumbers>
  <dblp:yearOfPublication>2010</dblp:yearOfPublication>
  <dblp:publishedAsPartOf rdf:resource="http://dblp.org/rec/conf/icwe
      /2010w" />
  <dcterms:license rdf:resource="http://www.opendatacommons.org/licenses/
      by/" />
</dblp:Publication>
</rdf:RDF>
```

Listing 1: DBLP response

```
{"type": "Publication:http:\/\/research.microsoft.com",
 "Title": "Semantic Annotation of RESTful Services Using External Resources
     ",
 "Abstract": "\u000a Since the advent of Web 2.0, RESTful services have
     become an increasing phenomenon. Currently, Semantic Web technologies
     are\u000a being integrated into Web 2.0 services for both to leverage
     each other strengths. The need to take advantage of data available\u00
     0a in RESTful services in the scope of Semantic Web evidences the
     difficulties to cope with syntactic and semantic description\u000a of
     the",
 "Author": ["Victor Saquicela", "Luis Manuel Vilches", "\?Ascar Corcho"],
 "CitationCount": 0,
 "Conference": "International Conference on Web Engineering",
 "HomepageURL": null,
 "ID": 46,
 "PublicationCount": 0,
 "ShortName": "ICWE",
 "StartYear": 0 "DOI": "10.1007\/978-3-642-16985-4_24",
  "FullVersionURL": [
      "http:\/\/www.springerlink.com\/content\/u352rt6422820447",
      "http:\/\/www.springerlink.com\/index\/u352rt6422820447.pdf",
      "http:\/\/dx.doi.org\/10.1007\/978-3-642-16985-4_24",
      "http:\/\/www.informatik.uni-trier.de\/~ley\/db\/conf\/icwe\/icwe2010
          w.html#SaquicelaVC10"],
  "ID": 39269940,
  "Journal": null,
  "Keyword": ["Domain Ontology", "Semantic Annotation", "Semantic
      Description", "Semantic Web", "Semantic Web Technology"]
  "ReferenceCount": 19,
  "Type": 1,
```

```
"Year": 2010}
```

Listing 2: Microsoft Academics response

Some sources do not have tools that allow access to data, it affects the quality of the data in the repository because the results need to be complemented and cleaned. The response from *Google Scholar* illustrated in Listing 3 has fewer fields with respect to the response from other sources illustrated in Listing 1 and 2, although it is the same publication. If a source do not have an API that allows access to the data, this can affect the consistency of the information in scientific publications. To resolve this problem, we use *String Metric Algorithms* such as *Cosine Similarity* and *Levenshtein* described in [22] to determine the correct value of a publication field. The correct value of a field is the one most repeated among all values from different sources. For example, we have the next values for a title of a publication from each data source: [*Linked sensor data*] from DBLP, [*Linked sensor data*] from Scopus, [*Publishing linked sensor data*] from Google Scholar, [*Linked sensor data*] from Microsoft Academics. We determine with this component that value[*Linked sensor data*] is the correct value for title, because it is the most common among all values extracted titles.

```
<?xml version="1.0" encoding="UTF-8"?>
<publication>
   <title>Enriching electronic program guides using semantic technologies and
        external resources.</title>
   <url>http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6965173</url>
   <year>2014 XL Latin</year>
   <citations>?</citations>
   <versions>..</versions>
   <clusterId>17749203648027613321</clusterId>
   <authors>V Saquicela, M Esponiza Mejia</authors>
   <abstract>Electrnic Program Guides (EPGs) describe broadcast programming
       information provided by TV stations. However, users may obtain more
       information when these guides have benn enriched. The main
       contribution of this work is to present an automation</abstract>
</publication>
```

Listing 3: Data retrieved from Google Scholar.

It is necessary to have materialized data about authors and publications in a repository to find correspondences between them locally. Other option is retrieving the publications when a user needs them, but time between making a request to an external source and mapping takes an average of eight to fifteen seconds depending on the API. Therefore, we have a unit repository to offer high availability and speed up to make queries of triples. With the data materialized the response time is short. If the result of a query is delayed, the data response and the query is stored in a graph, to give an immediate response the next time the query is executed. In this case, we do not run the query again, recovering only the result of this query that was stored in the repository when the query was executed the first time, .

Some publications has duplicated entities because these are extracted from several data sources. Also in some cases is ambiguous to determine publications of an author when they have similar names. So the data must be processed before to be stored, it is detailed in the 3.3 section.

3.3 Data Enrichment

The module of *Data Enrichment* unifies all data of publications and authors in a central repository using BIBO ontology. We find characteristics between publications and authors, assigning correspondences between the data model of the source and the common model that we have defined, through a component of *Mapping Ontology Model*. We have various entities of the same author or publication and this represent a problem of inconsistency. For this reason, we have a component called *Data Disambiguation* that solve this problem.

3.3.1 Mapping Ontology Model

In this component each data source with a different model is structured in a common model. This component find the correspondence between the properties of each source model to a common data model. Using *String Metrics Algorithms* mentioned in section 3.2.2. The common model is annotated using RDF [15] with a structure based in triples. The common model is illustrated in the Fig. 2, that shows the architecture used. The process of mapping is manual, using a file that contains the correspondences between the models. An axample of mapping between *DBLP* model and common model is illustrated in the table 1, it shows the mapping between the data model of a source and a common data model that we have defined. An alternative for this process is a study to automatic annotation of RESTful Web Services described in [16], that argues that we can do this process automatically.

The common model proposed is described using *BIBO Ontology* [8], which is an ontology used to describe bibliographic entities as books, magazines, etc. The authors are described using ontology FOAF (Friend of a Friend), it is an ontology used to describe people, their activities and relationships with other people and objects [5].

Data in the central repository is stored using a model of storage based in graphs. We have defined a graph for each data source (*Providers graph*), a graph for authors(Author graph) and a central graph (*Wkhuska graph*) that stores unified information of publications and authors. To make the unification of publications and authors, the data must be analyzed previously to establish correspondence and eliminate duplication.

Listing 4 illustrates the publication described using BIBO ontology. It is the same publication illustrated in the Listing 1 and 2, but enriched with data from different sources into a common data model. The publications are stored in a central repository. However, it is a problem to identify the correct author of a publication if there are multiple authors with the same or similar names. These problem is solved in the component data disambiguation.

```
<?xml version="1.0" encoding="utf-8" ?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
```

[15] Resource Description Framework; https://www.w3.org/RDF/
[16] <http//dblp.dagstuhl.de/rdf/schema/-2015/-01/-26/#>
[17] <http//purl.org/dc/terms/>
[18] <http//purl.org/ontology/bibo>

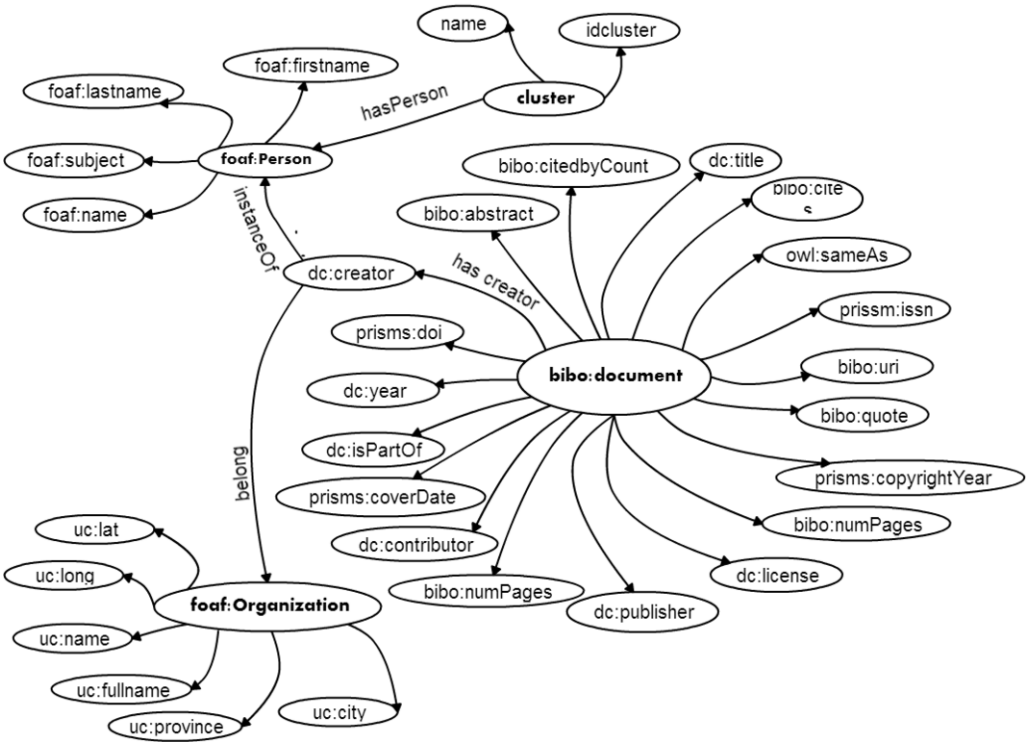

Fig. 2. Common model based on BIBO Ontology.

```
        xmlns:bibo="http://purl.org/ontology/bibo/"
        xmlns:dc="http://purl.org/dc/terms/"
        xmlns:foaf="http://xmlns.com/foaf/0.1/"
        xmlns:owl="http://www.w3.org/2002/07/owl#">
    <bibo:Document rdf:about="http://ucuenca.edu.ec/wkhuska/publication/
        semantic-annotation-of-restful- services-using-external">
        <dc:title>Semantic Annonation of RESTful Services Using External
            Resources.</dc:title>
        <foaf:Organization rdf:resource="http://dblp.uni-trier.de/" />
        <dc:contributor rdf:resource="http://dblp.dagstuhl.de/peers/s/
            Saquicela:Victor" />
        <dc:contributor rdf:resource="http://dblp.dagstuhl.de/peers/c/Corcho:
            =Oacute=scar" />
        <dc:contributor rdf:resource="http://dblp.dagstuhl.de/peers/b/BL=
            aacute=zquez:Luis_Manuel_Vilches" />
        <owl:sameAs rdf:resource="http://dblp.dagstuhl.de/rec/conf/icwe/
            SaquicelaVC10" />
        <bibo:uri rdf:resource="http://dx.doi.org/10.1007/978-3-642-16985-4
            _24" />
        <bibo:numPages>266-276</bibo:numPages>
        <dc:license rdf:resource="http://www.opendatacommons.org/licenses/by/
            " />
        <dc:publisher>ICWE Workshops</dc:publisher>
        <dc:isPartOf rdf:resource="http://dblp.dagstuhl.de/rec/rdf/conf/agile
            /conf/agile/config/iwce/2010w" />2
    </bibo:Document>
</rdf:RDF>
```

Listing 4: Publication described using BIBO Ontology.

Table 1
Mapping fields between DBLP model and the common data model. Prefixes: dblp [16], dc [17], and bibo [18].

DBLP fields	Common model fields
dblp:primaryElectronicEdition	bibo:uri
dblp:publishedAsPartOf	dc:isPartOf
dc:license	dc:license
rdf:type	rdf:type
dblp:publishedInBook	dc:publisher
dblp:authoredBy	dc:contributor
dblp:title	dc:title
dblp:pageNumbers	bibo:numPages
dc:contributor	dc:contributor
dc:title	dc:title
bibo:numPages	bibo:numPages
bibo:uri	bibo:uri
dc:publisher	dc:publisher
dc:isPartOf	dc:isPartOf

3.3.2 Data Disambiguation

Data about authors and publications come from different literature sources, have duplication and inconsistency especially when they have similar authors. For example, Table 2 illustrates a problem, the author *Victor Saquicela* is in multiple repositories DSpace, because he collaborates on several projects in different HEI. Therefore, it is necessary to discover authors that are the same entity between various sources. This component allows to define a single record of an author in a central repository using characteristics of the author and characteristics of their publications, taking advantage of ontological descriptions such as *OWL:SameAs*, that allow establish that *"Saquicela, Víctor"* is the same of *"Saquicela Galarza, Víctor Hugo"*

Searching publications of an author in the bibliographical sources such as *Microsoft Academics*, we have as parameter only the names of the author. Each bibliographic source has several authors with similar names and different publications. For example, when we search the author *Mauricio Espinoza* in *Microsoft Academics*, we get the response data illustrated in Table 3. In this case we have six authors that meet the search parameters. So, it is necessary to identify which author is the one that corresponds to the data of the Ecuadorian author, take account special characters.

We have defined a priority among bibliographical sources according to the quality

Table 2
Author search results in Microsoft Academics.

Author URI	Author name
ucuenca:SAQUICELA__V!!CTOR	Saquicela, V??ctor
ucuenca:SAQUICELA_GALARZA__VICTOR_HUGO	Saquicela Galarza, VÃÂctor Hugo
cedia:SAQUICELA__VICTOR,"Saquicela	Saquicela, Victor
ucuenca:SAQUICELA_GALARZA__V!!CTOR_HUGO	Saquicela Galarza, V??ctor Hugo
ucuenca:SAQUICELA__VICTOR	Saquicela, VÃctor
ucuenca:SAQUICELA__V	Saquicela, V

Table 3
Author search results in Microsoft Academics.

Author name	Affiliation	Fields
Mauricio Espinoza	Universidad Politcnica de Madrid	Databases, Engineering
Mauricio I. Espinoza	-	Pharmacology, Diseases, Ophthalmology
Mauricio Espinoza	-	Medicine
Mauricio Alfredo Rettig Espinoza	-	Law Criminology
Mauricio Espinoza R	-	Medicine
Andres Mauricio Espinoza Rivas	Universidad N. A. de Mexico	-

of the data. The most reliable source is Scopus, because it is the most consistent to searches, for example searches such as *Juan Pablo Carvallo Vega* and *Juan Pablo Carvallo Ochoa*, identify in some cases the difference. Other data sources such as DBLP does not keep complete records of the author and only use the first name and last name, causing scientific publications are assigned to other authors.

The component *Data Disambiguation* create a single record for an author and eliminates publications that do not belong to an author. Fig. 3, illustrates the process which extract an author from the graph of authors, but also extract their publications from each source graphs in the central repository. If some properties such as title, year of publication or conference between publications is found, the *OWL:SameAs* property is set between these publications. If the author of these publications is not yet in the central repository a new record is created with the publications processed. This process is iterative for each data source, and each author publication.

Until now, we have a central repository using Ontologies and Linked Data but it is necessary to extract knowledge from this data. In the following section we show how the pattern detection module was applied to detect similar areas between researchers.

3.4 Pattern Detection

In this section, we outline the purpose of this module that identifies communities or collaborators networks who have been working in similar areas. Detecting communities of collaborators allowed us to recommend a particular author, publications

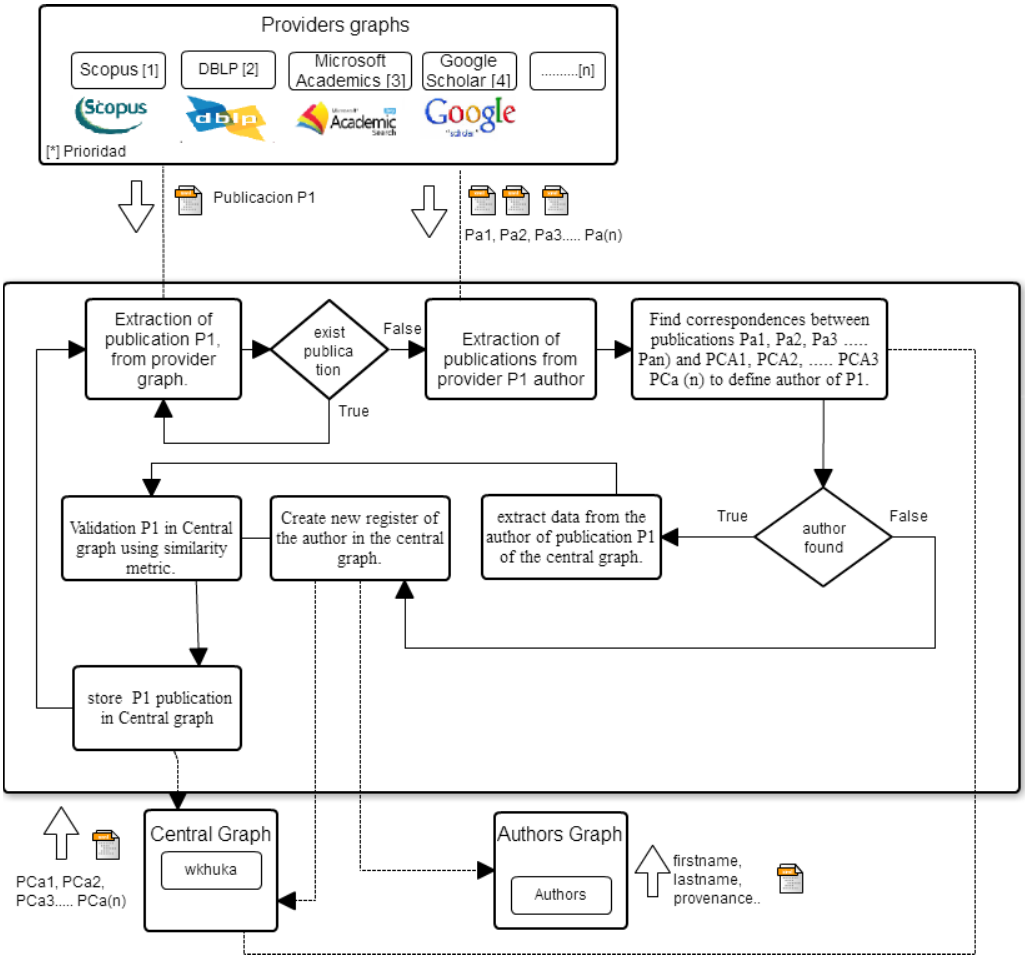

Fig. 3. Disambiguation Process.

from a specific area and colleagues with similar interests, who could be interested in working together. The module has three components to detect patterns from the data collected. First, all the data is taken from the *Central Repository*, it is used in the *Find Groups* component to detect some patterns in the data set. The *Knowledge Discovery* component is used to extract knowledge from the associated groups. To speed up queries and have organized groups, each group is labeled in the *Label Groups* component. Finally, results are stored in the *Central Repository* for further queries.

We use clustering algorithms to automatically discover similarities, but the computation complexity grows exponentially with the length of authors and publications. For a large text corpus not only is the complexity very high but the memory requirement is also very large, possibly the data cannot fit in main memory. We would rather use Apache Mahout [19] to execute algorithms of machine learning. We chose Mahout for the ability to deal with massive datasets, it is a scalable Java

[19] http://mahout.apache.org

library and we could profit of the distributed computation, because it is built upon Apache Hadoop [20] .

Keywords are index terms that provide most important information about the content of a publication. Broadly, keywords of academic literature talk about a certain topic area or methodology, allowing to detect similar areas based on those keywords as a result we could detect potential collaboration networks. Then, instead of use citations as an indicator of relatedness we used keywords of each author's publications.

3.4.1 Find Groups

Data preprocessing is accomplished before clustering. Our aim is to discover similarities with keywords of publications. *Keywords* along with other fields such as *title, URI and author* are extracted from the *Central Repository*, creating a document that contains keywords associated with other fields. After that, we separate keywords from the remaining fields in different files, because we need just keywords to identify common areas. Nonetheless, the remaining fields are necessary to later processing as you will see later on subsection 3.4.2. Both files are converted to a specific Hadoop file format that is SequenceFile [21] . Those files stores key/value pairs, where the first file contains a key with a unique identifier and a group of keywords that belongs to a paper is stored as a value (Table 4). Same happens in the second file with the difference that in the value pair it stores the remaining fields (Table 5).

Table 4
File with keywords.

Id	Keywords
1	keyword 1, keyword 2
2	keyword 2, keyword 3
...	...

Table 5
File with remaining fields.

Id	Author	Title	URI
1	author 1	title 1	http://ucuenca.edu.ec/id#1
2	author 2	title 2	http://ucuenca.edu.ec/id#2
...	...	...	...

Using the keywords of papers, we proceed to apply techniques of text clustering [2] for the *find groups* module. We use the file with keywords (Table 4), the groups

[20] https://hadoop.apache.org

[21] Mahout also use Sequence files to manage input and outputs of MapReduce and store temporary files.

of keywords in each line are considered as a document. Before clustering the data into Mahout, it is necessary to preprocess the data. Data has been preprocessed to convert text to numerical values, but not all the keywords have the same relevance. The weighting technique used to magnify the most important words and create vectors is Term frequency-inverse document frequency (TF-IDF) [20]. TF-IDF helps us to get a small weight for the stop words (*a, an, the, who, what, are, is, was, and so on*) and terms that occur infrequently get a large weight. Topic words have more importance in the vector produced, because those words usually have a high TF and a somewhat large IDF, by the product of the two. For example, if we have a fictional collection of documents with the words *the* and *semantic*, the IDF for the stopword *the* appearing in all documents is zero and a very specific term like *semantic*, which appears in a few documents, gets assigned a comparatively high IDF. Then, the product of TF-IDF gives us a larger value for *semantic*. The weighted values are used to generate the Vector Space Model (VSM) where words are dimensions. The problem with this VSM generated is that words are entirely independent each other and this is not always true. Sometimes words have some kind of dependency such as *Semantic* with *Web*. In order to identify this dependency, we use collocations [12]. At the time of writing, we are executing our experiment using bi-grams and an Euclidean norm (2-norm), which can change. In future experiments, it will be interesting to generate vectors using Latent Semantic Indexing (LSI) or apply a log-likelihood to take words that mostly have the chance to go together. So in the long run, we have our vectors completed to start clustering.

We use the vectors generated to execute K-Means algorithm in Mahout. It was executed using a Cosine distance measure as similarity measure. To seed the initial centroids, we use RandomSeedGenerator, which is used to generate random centroids in Mahout. The experiment has 100 iterations as maximum and the number of clusters (k) varies according to the amount of data extracted from the different bibliographic sources, because when new publications or authors are extracted the k must be adjusted. Once the algorithm is executed we have our VSM clustered, where each vector belongs to a cluster. In order to make this information readable, we process again this results as you will see next.

3.4.2 Knowledge Discovery

The result of this component replace the original keywords in the vectors clustered, for example, instead of the vector C_1=(182.12, 334.32, 324.43) the vector is translated to C_1=(“*Web Semantic*”, “*Ontology*”, “*Linked Data*”). Once detect the clusters of common areas, we use the keywords of each author in order to detect potential collaboration networks. Authors belonging to this network may not have published together, so they are not necessarily co-authors.

Fig. 4 illustrates the process to discover knowledge areas and collaboration networks using the clustered data. We developed a MapReduce model that is formed by two joints. In (a), we put together the words before processing (Table 4) and the clustering results. The field *ClusterId* helps us to identify similar knowledge areas, thus, when we have the same *ClusterId* it means that these keywords belong to

the same area or topic. While in (b), these knowledge areas are taken to pinpoint potential collaboration networks. Joining the data about authors (Table 5) with the knowledge areas detected help to identify a network of researchers. All authors belonging to the same cluster can work together because they are interested in similar topics. Besides the *author* field, there are other fields such as publication *title* or *URI* (for authors and publications) that they are not necessary for the current analysis but are important for further storage.

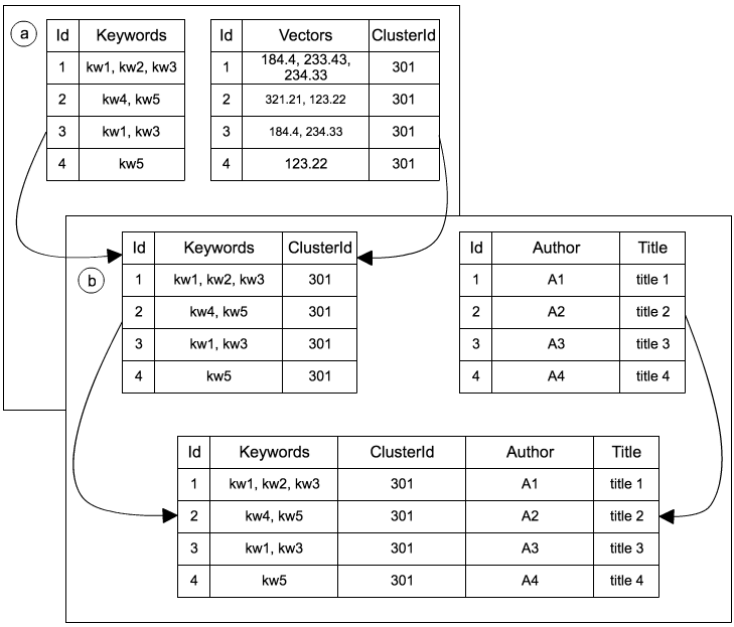

Fig. 4. Joins made to discover knowledge areas (a) and collaboration networks (b).

In most of the clusters, results are correct. Fig. 5 illustrates a sample clustering result taken of the previous process. On the left there are common topics, which means all keywords are related with each other, meaning that the authors (in the right side) can form a collaborative network and consequently work together on future projects. Finally, we not only identify similar areas and collaborative networks but also we could recommend papers based on the title of a publication.

Each cluster belongs to a general topic area. So in the following subsection, we label each cluster in agreement to the keywords that contains it.

3.4.3 Label Groups

We have many keywords that belong to complex domains and hand-label each cluster it is a tedious and expensive task. Keywords help us to handle searches effectively and search engines could increase performance in searches by finding a general topic area based on the words that belong to a cluster. Labeling clusters help to respond to specific queries (i.e.: show all researchers working in a specific area or all subareas belonging to a general topic area).

In order to achieve the purpose of labeling, the keywords associated with pub-

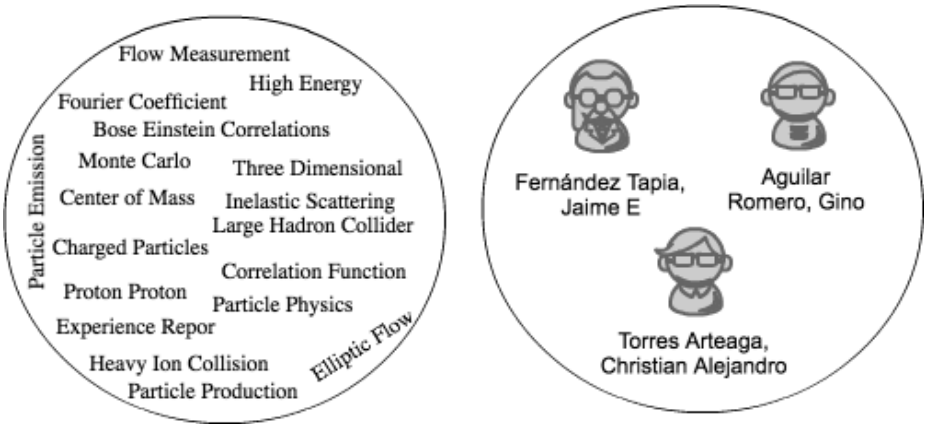

Fig. 5. Knowledge areas (left) and collaboration networks (right).

lications was enriched. We use WordNet[22] [14] to find synonyms, hypernyms, hyponyms and the concept of a word for all keywords in each cluster. That helps to find a common meaning in the way that words could occur together and find similar meanings. In other words, with the group of words set up, we could find a concept or a topic for each cluster.

We applied Collapsed Variational Bayes (CVB) algorithm that is an implementation for Latent Dirichlet Allocation (LDA) [4] in Mahout. We use all the words generated by WordNet plus the title and keywords of each publication to find a broader topic based in multiple subtopics described by the keywords. First, we generate vectors using Term Frequency (TF) and after that, we use Mahout RowId to convert TF vectors into a matrix. Once generated the matrix the CVB algorithm was executed with the following parameters: 1 for the number of latent topics and 20 maximum interactions. This job was applied to each cluster. We got good results using the topic model technique, for example, the cluster keywords of Fig. 5 was labeled as *physics*.

All results are stored in the Central Repository using an RDF model. Fig. 6 illustrates the concepts and relationships used to store the results. The full arrow symbolizes a relationship between classes and the dashed arrow symbolize a common relationship. A new URI is created for clusters, for example, `http://ucuenca.edu.ec/resource/cluster#1`. Authors and publications are already stored in the central graph. So we link clusters with authors with the property *hasPerson* also, clusters has a property *rdfs:label* to store the label of each cluster.

4 Conclusion and Future Work

We have presented an architecture to identify common research areas among Ecuadorian authors. This architecture encompasses a process to extract, enrich, and represent bibliographical resources to discover patterns using data mining algorithms. The different components implemented and techniques used to define the

[22] It is a lexical database for the English language that is used for text analysis applications.

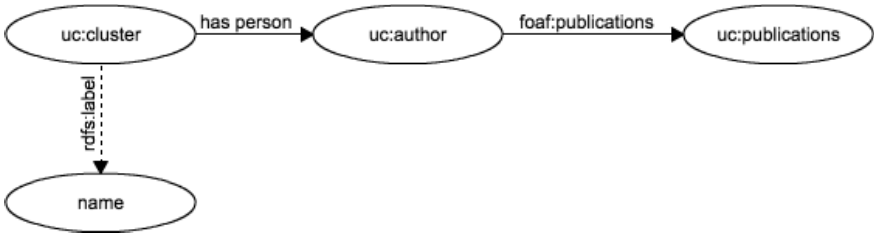

Fig. 6. Concepts and relationships of RDF model.

architecture shows the potential of this proposal; currently, these components are being used on the REDI project [23]. Future work will focus on analyzing alternatives to semantics techniques with the aim of improving data disambiguation. Furthermore, we also plan to make improvements to the clustering and labeling process. In the same line of thought, this work could be adapted to recommend literature based on user searches history. Finally, new Higher Education Institutions' bibliographical data sources will be added to the system improving the generation of potential collaboration networks.

Acknowledgement

This research was supported by CEDIA [24] in the project "Ecuadorian Repository of Researchers". We thank our colleagues who provided insight and expertise that strongly assisted the research.

References

[1] Alfraidi, H., "Interactive System for Scientific Publication Visualization and Similarity Measurement based on Citation Network," Master's thesis, University of Ottawa (2015).

[2] Andrews, N. O. and E. A. Fox, *Recent developments in document clustering*, Department of Computer Science, Virginia Tech, 2007.

[3] Barbarić, A., *Isbd: International standard bibliographic description* (2014).

[4] Blei, D. M., A. Y. Ng and M. I. Jordan, *Latent dirichlet allocation*, J. Mach. Learn. Res. **3** (2003), pp. 993–1022.

[5] Brickley, D. and L. Miller, *Foaf vocabulary specification 0.98*, Namespace document **9** (2012).

[6] Gangemi, A., "The Semantic Web – ISWC 2005: 4th International Semantic Web Conference, ISWC 2005, Galway, Ireland, November 6-10, 2005. Proceedings," Springer Berlin Heidelberg, Berlin, Heidelberg, 2005 pp. 262–276.
URL http://dx.doi.org/10.1007/11574620_21

[7] Garfield, E. et al., *Citation analysis as a tool in journal evaluation*, American Association for the Advancement of Science, 1972.

[8] Giasson, F. and B. D'Arcus, *Bibliographic ontology specification* (2009).
URL http://bibliontology.com/specification

[9] Huang, J., S. Ertekin and C. L. Giles, *Fast author name disambiguation in citeseer*, ISI technical report **66** (2006).

[23] http://redi.cedia.org.ec/

[24] www.cedia.org

[10] Krisnadhi, A. A., Y. Hu, K. Janowicz, P. Hitzler, R. A. Arko, S. Carbotte, C. Chandler, M. Cheatham, D. Fils, T. Finin, P. Ji, M. B. Jones, N. Karima, K. Lehnert, A. Mickle, T. Narock, M. OBrien, L. Raymond, A. Shepherd, M. Schildhauer and P. Wiebe, *The geolink framework for pattern-based linked data integration*, in: *SEMWEB*, 2015.

[11] Ley, M., *Dblp xml requests* (2009).

[12] Manning, C. D. and H. Schütze, "Foundations of Statistical Natural Language Processing," MIT Press, Cambridge, MA, USA, 1999.

[13] McNee, S. M., I. Albert, D. Cosley, P. Gopalkrishnan, S. K. Lam, A. M. Rashid, J. A. Konstan and J. Riedl, *On the recommending of citations for research papers*, in: *Proceedings of the 2002 ACM Conference on Computer Supported Cooperative Work*, CSCW '02 (2002), pp. 116–125.
URL http://doi.acm.org/10.1145/587078.587096

[14] Miller, G. A., *Wordnet: A lexical database for english*, Commun. ACM **38** (1995), pp. 39–41.
URL http://doi.acm.org/10.1145/219717.219748

[15] O'Neill, E. T., *Frbr: Functional requirements for bibliographic records*, Library resources & technical services (2002).

[16] Ortiz Vivar, J. E. and J. L. Segarra Flores, *Plataforma para la anotación semántica de servicio web restful sobre un bus de servicios* (2015).

[17] Osborne, F., E. Motta and P. Mulholland, *Exploring scholarly data with rexplore*, in: *The Semantic Web–ISWC 2013*, Springer, 2013 pp. 460–477.

[18] Piedra Nelson, N., J. Chicaiza, E. Cadme, R. Guaya et al., *Una aproximación basada en linked data para la detección de potenciales redes de colaboración científica a partir de la anotación semántica de producción científica: Piloto aplicado con producción científica de investigadores ecuatorianos* (2014).

[19] Pohl, S., F. Radlinski and T. Joachims, *Recommending related papers based on digital library access records*, in: *Proceedings of the 7th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '07 (2007), pp. 417–418.
URL http://doi.acm.org/10.1145/1255175.1255260

[20] Robertson, S., *Understanding inverse document frequency: On theoretical arguments for idf*, Journal of Documentation **60** (2004), pp. 503–520.
URL http://research.microsoft.com/apps/pubs/default.aspx?id=67744

[21] Torvik, V. I. and N. R. Smalheiser, *Author name disambiguation in medline*, ACM Transactions on Knowledge Discovery from Data (TKDD) **3** (2009), p. 11.

[22] Xiao, C., W. Wang, X. Lin, J. X. Yu and G. Wang, *Efficient similarity joins for near-duplicate detection*, ACM Trans. Database Syst. **36** (2011), pp. 15:1–15:41.
URL http://doi.acm.org/10.1145/2000824.2000825