Scientific Papers Retrieval with an Emphasis on Graph-based Structural Information

Farzaneh Norouzi
Department of Computer Engineering, University of Science & Culture
Tehran, Iran
Farzaneh71754@yahoo.com

Fatemeh Azimzadeh*
Ph.D, Scientific Information Database (SID), ACECR
Tehran, Iran
F.azimzadeh@gmail.com

Received: 2017/12/28 Revised: 2018/02/26 Accepted: 2018/03/26

Abstract— **With the increasing spread of science, various methods have been proposed to restore more and better scientific documents according to the needs and requests of users. Since there is no complete information for some documents, users have to access the metadata including the name of authors and their affiliation, the publication date, and references used for the document by accessing to the documents. Therefore, extraction of information based on the structural and geometrical characteristics of the document can be very helpful in retrieving relevant and required documents. In this paper, after extracting metadata using geometrical features of documents and graph-based model, the relationships between different entities such as documents, authors, journals, and conferences are modeled for more efficient information retrieval. The extracted and refined data, stored in the graph model, are available in a web-based user interface. To produce the results of each query, the related documents are retrieved based on the graph's relationships, the quality of each document, and their citation score. To evaluate the proposed method, the PubMed and D2SPR databases are used. The results from the experiments show that the number of retrieved documents in the proposed method is 60% higher than the PubMed database search engine and 80% higher than D2SPR. Moreover, nDCG with an average of 0.824 in the proposed approach has a significant distance with the average of 0.30 in Pubmed search engine. While the average of F-measure on D2SPR dataset is 0.834 for the suggested system, the value is 0.71 in the current study.**

Keywords— ***Metadata Extraction; Information Retrieval; Knowledge Graph Data Model; Structural Data***

1. INTRODUCTION

In recent years, along with introducing and developing concepts such as semantic web, machine learning, data mining, text processing and etc. some new approaches have been considered for information retrieval. In classical methods and interfaces relations, direct and indirect relationships between entities of each system are not considered. Neglecting these relationships in the past is generally due to the nature of interlinking databases and the lack of appropriate scientific models for modeling the meaning and semantic relationships between documents.

Nowadays, human knowledge is published in the form of scientific articles. These documents are generally in the form of documents in portable document format (PDF) format and are made available through scientific publication databases. Each scientific article, in addition to the specialized text and the content contained in, also has other valuable data called metadata. For example, in the case of a scientific article, factors such as article title, author's name, number of citations, release date, and many other similar information this are considered as metadata. Metadata play a key role in improving the quality of search results and providing better responses to users. Although there are many metadata available in PDFs, many publishing databases provide users with only a limited amount of this information. Another issue in this regard is that when we want to combine the information in different databases into a unified structure. There is no uniform standard for metadata by scientific databases. In these cases, it is very important and essential that source files are able to extract the desired metadata from the articles related to each database. In this case, the original file, usually in the PDF format, is scanned and the required metadata is extracted in several steps. This metadata will be evaluated and validated in the suggested method.

In order to achieve this goal, three main steps have been considered: data extraction, refinement, graph-based modeling, and the search and retrieval of documents based on the structural data of the document file.

The remainder of this study is organized as follows. In Section II, we review the related work done in the past. Section III fully addresses the proposed method and its details. Section IV involves the implementation and evaluation of the results of the experiments carried out using the proposed approach. Section V presents a summary of the whole research, conclusions, and suggestions for future work.

2. RESEARCH BACKGROUND

The issue of metadata extraction is always a challenging topic. For example, different documents with similar information can be presented with different styles and fonts and the PDF format does not store document structure information such as words and paragraphs, lists and measures, table structure, the hierarchy of sections, and the order of the reading text.

2.1. The Most Important Tasks Done in the Metadata Extraction Scope

Most methods focus only on extracting the metadata of the article and often do not process all input documents.

Proposed solutions are often based on heuristic rules or heuristic techniques or machine learning techniques. Giuffrida et al. [1] extracted content from PostScript files using a tool based on pstotext and then processed the metadata and extracted them based on a set of rules for these pieces of texts.

Another example of the rule-based systems is PDFX, which was presented by Constable et al. [2]. PDFX converts scientific documents from PDF format to XML descriptions. In this method, the components of the input documents were first marked and then the metadata, full text, and references were extracted from the marked sections.

Another category of methods, which are very common, includes methods based on machine learning. For example, Han et al. [3] extracted the metadata from the headings of scientific articles using a two-stage classifier, including support vector machines (SVMs) and text-related features. Another example of methods based on SVM is the CRIS system provided by Kovacevic et al. [4] to extract metadata based on geometric features and features related with the text.

Lopez presented the GROBID system for the analysis of scientific texts in PDF format [5]. GROBID used Conditional Random Fields (CRFs) to extract document metadata, full text, and a list of scanned references.

The parts related to references are usually located using the Heuristic [2] or machine learning [5]. The purpose of locating is to designate the place of the geometric file. Referral scanning means the extraction of metadata from referral strings, usually using regular expressions and knowledge-based methods [6] [7], which is done using more advanced machine learning techniques such as CRF [5], SVM [8], and HMM [9].

One of the best and most successful approaches in metadata extraction is the approach presented by Tkaczyk et al. named [CERMINE] to extract metadata from scientific references [10]. CERMINE is an open-code, modular, and machine-based learning system that can extract metadata from scientific documents with an acceptable accuracy. CERMINE has three basic subdivisions, including text area detection, metadata extraction from areas, and the identification of references from the area related to references.

2.2. Modeling Scientific Documents with Graph

Database problems occur when there is a defect or loss in this information. For example, the DBLP database [11] has collected significant information from computer science articles but is weak in terms of keywords and information referrals. Another defection of this set is that it does not support other trends and disciplines. Other sets also have the same bugs more or less.

Regardless of which approach is used, we should have a method for storing and presenting standard metadata for scientific documents. In 2012, Google used the term “Graph of Knowledge” for the first time [12]. Many authors refer to the graph of knowledge as the basic structure for the expression of human knowledge in the form of a graph with an emphasis on comprehensiveness [13]. Examples of graphs of human knowledge are YAGO [14] and FreeBase [15].

Our concern regarding the concept of the graph of knowledge is the problem of aggregating information or metadata from scientific sources, with metadata derived from documents in the form of a model of the graph. Here, we face some issues like matching and connecting.

Recent works in the field of constructing graphs of knowledge, such as NOUS [16], Knowledge Vault [17], or NELL [18],[19],[20],[21],[22], focus on the formation of graphs based on the inference of existing data relationships.

Knowledge Vault proposed by Dong et al. [17] is a method for generating an automated database in a large volume and dimension. This method is intrinsically a statistical method.

2.3. Search by Structure

In this part, the most important works on how to search and respond to queries are reviewed in the graph space.

Park [23] proposed the top-k algorithm for the search field based on the model of the graph in the documentation. In this method, the model of the graph was implemented to search quickly and efficiently between various related keywords.

Peddinti et al. [24] studied the creation and implementation of a constructed query in the computer network environment. For this purpose, they used a data processing system, the output of which contains structured queries that can be used as a tool to identify the geographical location.

3. THE PROPOSED METHOD

In the following, the proposed method is described and its various dimensions are expressed. This method has three main phases:

- Phase I: extracting information from source documents to PDF format
- Phase II: refining information, identifying and establishing relationships, creating a model of graph
- Phase III: sending queries and retrieving information based on the obtained model

These parts are from the operational point of view and implementation of the system. From a functional point of view, the system has two sections for feeding and building its database, as well as sending requests and receiving appropriate responses. In the first section, each document input to the system must be processed and its data added to the system.

After processing each document and inserting it into the system, that document is suitable for being retrieved by appropriate and relevant queries. Receiving and processing of queries and then restoring the results are conducted in the second operational part (equivalent to the third phase of the implementation).

Certainly, in all these steps, the relevant data are needed to feed different parts of the system. In the first section, for the training of the system, a number of scientific articles are needed for the open access section of the PubMed database [25].

After information extraction, based on the concept of the graph of knowledge, the relationships between the various components of each document are graphically mapped and stored using the Neo4j database, which is a graph-based database. In order to refine, validate, and convert information extracted from documents, an interface code is written in PHP. Using the written code, their information and relationships are created in the form of Neo4j database commands and the desired model of the graph [26].

Finally, in order to use the created model, a web interface is created to receive user query requests and retrieve related documents. In order to evaluate the system's performance, the system function has been compared in evaluation criteria like f-measure and quality rate of the available in the retrieval result with some previous scientific retrieval systems.

The distinction between proposed method and other methods is that, here, in addition to the coordination of words, the available relationships between documents, such as publication in a collection (journal, conference), and a collaborative author, are used to retrieve related documents. Also, since some of the related documents, which may not exactly match the words in the query, have the chance to be retrieved first through communications that are related to other related documents and secondly if they have more referrals, they will rank better in the final list. Another feature of the proposed system is that each section of the system can be optimized individually. This can be carried out in all three phases by setting parameters or optimizing algorithms.

3.1. Content Extraction by Using CERMINE Method

CERMINE accepts a scientific document in PDF format as input [27]. The extraction algorithm looks for the entire contents of the document and generates two types of outputs: document metadata and references. The CERMINE extraction process consists of three paths:

- The initial path of extraction received the structure of a PDF file as the input and then creates a re-representation of its geometric hierarchy.
- The metadata extraction path and metadata sections analyze the geometric hierarchy structure and extract a rich set of metadata of the document.
- The path of extracting information of references and bibliographic information analyzes and scans parts of the structure labeled as references.

What is being used in the process of the metadata extraction is the created contents of the XML file generated by CERMINE. This file contains different parts and types of required metadata, such as author's name, article title, year, volume number, journal title or conference, and authors' email address. In order to use this file, its contents must be scanned, evaluated, and validated. Through the assessment and validation process, it is tried to figure out whether the information extracted by CERMINE is valid.

Before using this value, it should be checked that the generated value of the program in the previous step is confirmed based on the standard patterns that are usually specified by regular expressions. In some cases, CERMINE cannot recognize many of the existing metadata in the document. In this case, the document is removed from the processing process and a copy of it is transferred to the preset path so that in the future, with CERMINE core training, the error rate can be reduced – especially for new format documents (Fig. 1).

The validation process for extracted data involves evaluating the data format based on predetermined principles. Format refers to the shape of related strings to each entity. These formats are as follows:

- The format of the authors' E-mails
- The format of the numbers, including page number, volume number, and year of publication
- The format of general texts such as abstract, title, names, and references
- The format of web links

Regular expressions are used to define and apply these formats. After evaluations at the level of strings, two conditions must be considered: First, does the document have a valid title? The string's valid title contains valid characters. If there is a valid title, the document must have at least one author, so that it can be considered as a document to be fed into the graph.

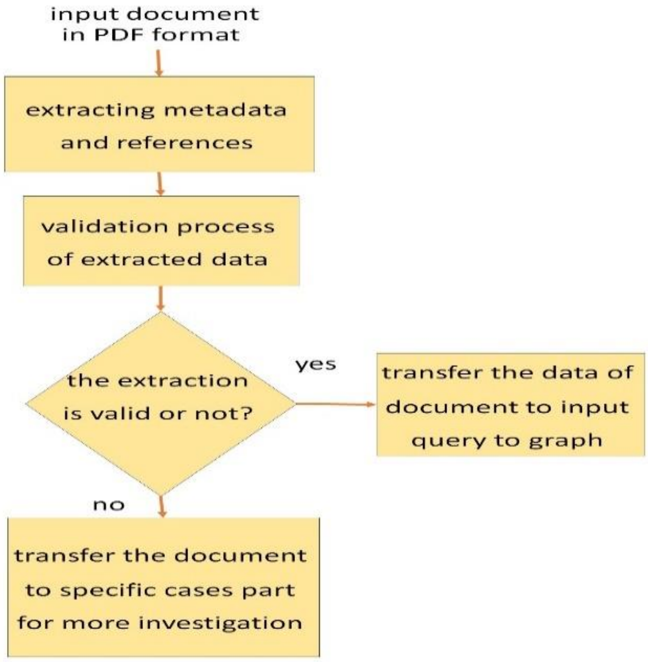

input document in PDF format
extracting metadata and references
validation process of extracted data
the extraction is valid or not?
yes
transfer the data of document to input query to graph
no
transfer the document to specific cases part for more investigation

Fig. 1. The Process of Metadata Validation

3.2. Building the Model of Graph

After generating XML files and initial validations, it is tried to extract the entities from these files and build the model of the graph. The list of entities in the proposed model is as follows: any document, any journal or conference, any author, any keyword, any affiliation.

The relationships between the above entities are defined as follows:

- Each document can be written by one or more authors.
- Each document can be cited by another document.
- Each document can belong to a journal or conference.
- Each keyword can be one of the keywords in a document.
- Each author can have his/her own specific affiliation (membership in a university, institute, etc.)

In addition to the above cases, each node and each edge can have its own specific attributes. For example, for a document, the following attributes are defined: the title, number of citations, abstract, volume number, title or general subject, page number, year.

Each of these features can be used in searches and queries performed on the model of the graph.

The mechanism for processing a document for insertion in the graph database is shown in Fig. 2. The mechanism includes updating document information, reference information, year of publication, page number, journal name or conference, volume number, DOI, and in general any information that may not have existed or been extracted in previous document processing. Any referral relationship added to the document increases one number the amount inside the document node.

3.3. Sending Query and Retrieving Documents

After creating the graph database and feeding it using a considerable number of documents, it is possible to retrieve the most relevant documents with the requested query based on different queries. The response generation algorithm for each query is shown in Fig. 3.

In the proposed method, the input request is examined followed by conducting a preprocessing. The purpose of this preprocessing is to fix the ambiguity and optimize the input request. At this point, pause words are recognized and deleted.

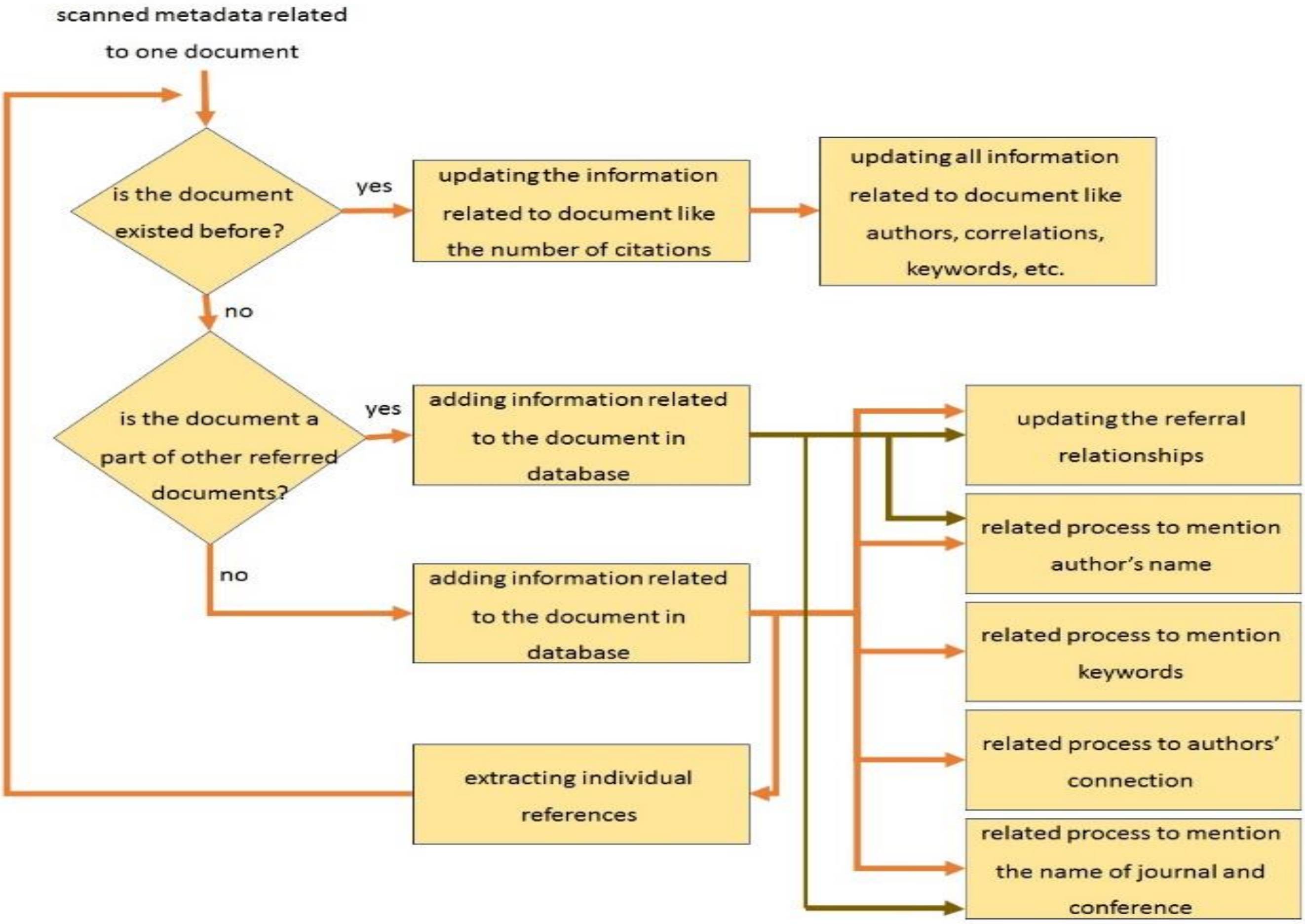

scanned metadata related to one document
is the document existed before?
yes
updating the information related to document like the number of citations
updating all information related to document like authors, correlations, keywords, etc.
no
is the document a part of other referred documents?
yes
adding information related to the document in database
no
adding information related to the document in database
extracting individual references
updating the referral relationships
related process to mention author's name
related process to mention keywords
related process to authors' connection
related process to mention the name of journal and conference

Fig. 2. Processing and Mentioning a Document in the Graph's Database

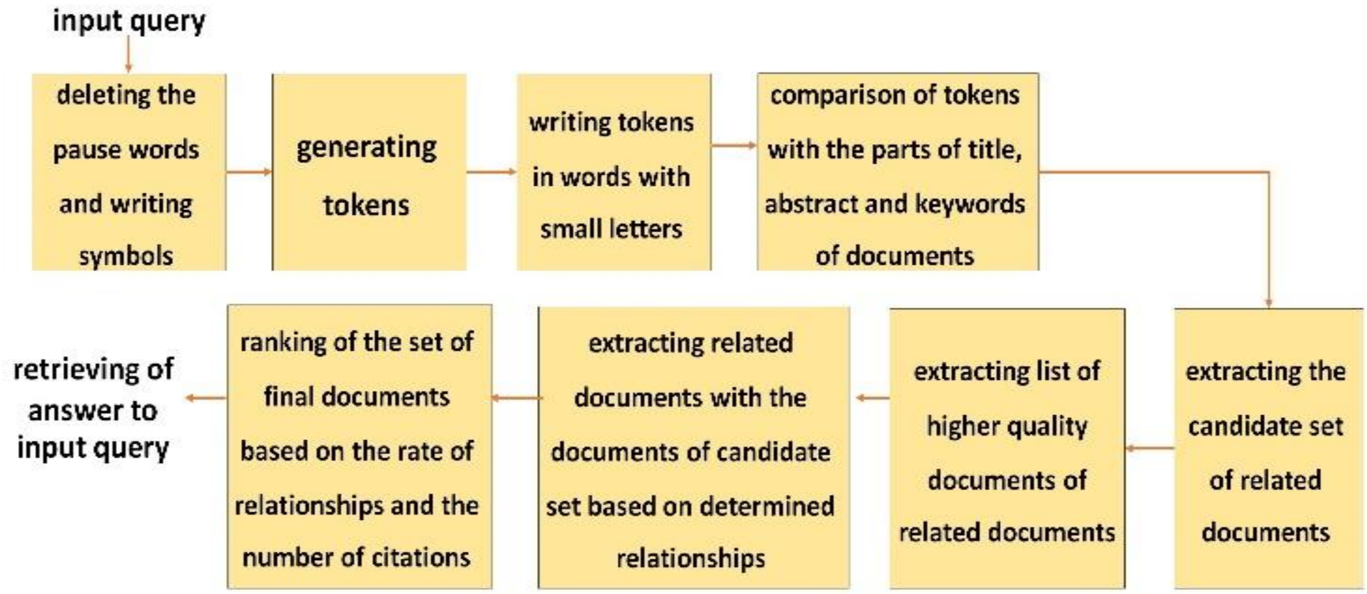

input query
deleting the pause words and writing symbols
generating tokens
writing tokens in words with small letters
comparison of tokens with the parts of title, abstract and keywords of documents
extracting the candidate set of related documents
extracting list of higher quality documents of related documents
extracting related documents with the documents of candidate set based on determined relationships
ranking of the set of final documents based on the rate of relationships and the number of citations
retrieving of answer to input query

Fig. 3. Response Generation for the Input Query

In the continuation of the preprocessor, the input string is divided into smaller pieces based on the distance between the words remaining in the text processing literature called token. For ease of comparison, all texts are converted into texts with small letters.

After achieving tokens and defining the rules of comparison, the tokens are compared with titles, abstract, and article keywords one by one. In this part, a criterion of adaptation evaluation is applied in order to evaluate the quality of the compliance of the documents. The purpose of this evaluation criterion is to determine the degree of compliance of a document and to present a qualitative parameter for comparing the documents. To coordinate a linear composition with fixed weights (Eq. 1) is applied.

$$rm = \sum_{i=1}^{k} n_i w_i \tag{1}$$

where *rm* is the criterion of document relationship, *k* is the total number of parameters, and n_i is the number of adaptations for the parameter w_i ,*i* is the weight of the parameter *i*. These weights are considered according to the importance of adaptation in each part.

After calculating the ratio of documents to the input query, the documents related to the input query are the set of candidate documents.

In the next step, using graph relationships, other related documents are extracted that are not present in the initial candidate list but are indirectly related to the input query. To extract these documents, they are used in the candidate response set. The goal is to bring documents related to higher quality documents into the final list of documents. To do so, the mean and the variance of the criterion of the relationship between the documents in the candidate dataset are calculated using the following heuristic:

All documents whose differences are less than or equal to standard deviations are higher than the average or lower than the average.

After specifying the list of documents related to quality, the following relationships are considered for extracting related documents:

- Two documents are linked together if published in a journal or conference.
- The two documents are related if they have an author in common.

Based on these rules, related documents are extracted for each document in the list of higher quality candidate documents. A larger list of documents is obtained by merging the two lists of qualitative candidate documents and their related documents. Now, to represent the final results, we need to arrange the documents in the list based on their quality, which results from two parameters for referencing each document and its relation to the input query. For this purpose, the criterion is defined as the ratio of the relationship between the document and its final rank in the search results set (Eq.2).

$$dr_i = rm_i w_1 + c_i w_2 \tag{2}$$

Where dr_i is the final ranking criterion for a document *i*, rm_i is the criterion for a relationship for the document *i*, c_i is the number of citations to the document *i*, w_1 is the weight of the document’s compliance with the input query, and w_2 is the weight of the document’s compliance rate with the number of citations to the document. The weights of w_1,w_2 and component values of the system are empirically defined. Here, weights and values are considered as 1 and 10, respectively. The final result consists of documents arranged in descending order based on the *dr* measured values.

1) *Using Gephi for Data Visualization of Relationship Extraction*

The process of generating relationships between entities should be examined when the data scanned and the model of the graph is generated. For this purpose, using a graphical interface existing in Neo4j is impossible because of the weakness in displaying high volume graphs. The strong tool of Gephi has been used instead of this interface [28]. According to the definition of producers, Gephi is a software with an open code for analyzing and visualizing the network.

Here, Gephi is used for evaluating the number of citations and the relationships between authors. In other words, through developing the transition part of metadata files to Neo4j by Gephi, transition operations and probable defections can be seen visually in the clip of articles. Fig. 4 presents a sample of referring graph between documents created when 1156 input documents exist in the system in Neo4j. As can be noted, a graph with such a clutter gives inadequate data about the issue. Gephi provided a better representation of referring clusters by compound execution of algorithms of Fruchterman Reingold and Yifan Hu in various parts of the graph in Fig. 5

4. EVALUATION CRITERIA

The criteria of evaluation mean any criterion that can be used to quantitatively measure and achieve the desired goals in the research. The model proposed in this research has different parts that can be used for each of them, including parameters such as precision and accuracy for performance evaluation.

4.1. Offline Criteria

This criterion is generally used to judge the relevance of results or the same quality of search results.

1) *The ratio of the number of retrieved documents to the total number of document*

It represents a fraction of the retrieved document relative to the total available documents.

2) *Precision*

Accuracy is equal to the fraction of recovered documents that are related to the requested information.

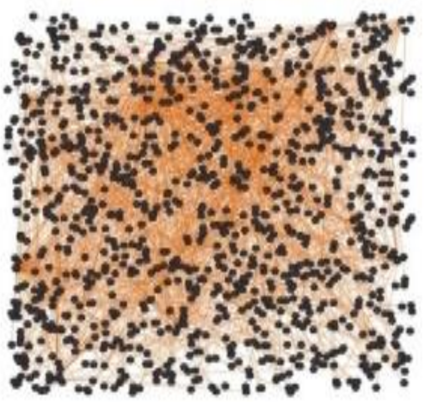


Fig. 4. . Input data from Neo4j to *Gephi*

$$Precision = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{retrieved documents}\}|} \quad (3)$$

3) *Recall*

The purpose of recovery is a fraction of the documents associated with the input query that was successfully retrieved.

$$Recall = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{relevant documents}\}|} \quad (4)$$

4) *F measure / F score*

The *F* index is defined as the mean harmonic weighted by precision and retrieval.

$$F = \frac{2.\text{precision.recall}}{(\text{precision} + \text{recall})} \quad (5)$$

5) *Discount Cumulative Interest*

Discount cumulative interest uses a gradation scale from the relevance of the documents used to evaluate the usefulness or interest of each document, based on its position in the result list.

$$DCG_p = \sum_{i=1}^{p} \frac{rel_i}{\log_2(i+1)} = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2(i+1)} \quad (6)$$

6) *Discount Cumulative Interest*

For normalization, all relevant documents in the set must be sorted according to their relationship, and the largest possible discounted cumulative gain (DCG) is searched for the position called IDCG.

$$nDCG_p = \frac{DCG_p}{IDCG_p} \quad (7)$$

$$IDCG_p = \sum_{i=1}^{|REL|} \frac{2^{rel_i} - 1}{\log_2(i+1)} \quad (8)$$

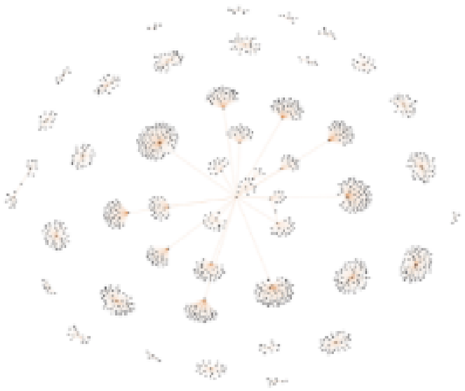


Fig. 5.. Compound Using Algorithms Yifan Hu Fruchterman Reingold for a Better Separation

4.2. Data set of PubMed

This set contains more than 30 million referrals in the medical field, scientific journals, online books, and etc. Some references to full-text documents are linked through PubMed or publisher websites [1]. A significant portion of the documents on this site has not been available without payments. In another part of the database, PDFs are some documents along with XML files that provide metadata and bibliographic information documents. These documents are available in various combinations.

Because there is a need for a significant number of articles in related areas in the construction of the graph and for the significance of referral relations and other relationships, in the present research, has used 10,000 documents for system training and 1,000 documents for testing system exploitation.

4.3. D2SPR Dataset

This dataset was presented by a team from Singapore National University [29]. This set contains the following information:

- The initial feature vector for articles
- The text of articles in the PDF and CSV formats in a separated and labeled form
- References and referral rate of articles
- Classification based on the interests of the top 50 researchers in the field of articles

4.4. Performance Evaluation

In order to evaluate the performance of the proposed model, the changes made to the results retrieved by the proposed system should be compared to the results retrieved by the other system. To this end, the PubMed database and its search capabilities, as well as the D2SPR database and the method provided by Dwaipayan et al. [30] were used. The D2SPR database and the method tested by Dwaipayan et al. include specific requirements and limited data. However, the use of PubMed database requires considering some certain points.

As the test conditions should be the same, the searches made in the PubMed database are set to contain only open-access documents. This is done, because, in the suggested system, there is only the access to the PDF file of these articles. Accordingly, in the Search Details section, the “open access[filter]” is added (Fig. 6).

4.5. The Used Queries

In the test performed for the D2SPR database, the made queries are completely identical with the feature vectors defined inside the data set. The reason for using these queries is to compare the results with those of other methods. This result is completely different for the PubMed database, which contains much more documents. In this case, to evaluate the system, it is tried to use as much as possible more general queries that are likely to be available in both of the sample sets (PubMed website and the sampled version). The queried phrases are as follows:

- Fatty Liver
- Medical Insurance
- Brain Tumor
- Breast Cancer
- Genetic Predisposition
- Hematologic Disease
- Respiratory Allergy
- Arthritis
- Down Syndrome
- Open Heart Surgery

4.6. A Measure of the Number of Retrieved Results

The achieved results for various queries for the proposed system and the PubMed database are shown in Fig. 7. The figure presents the GB tag associated with the proposed system with the graph-based data model and the PM tag for the PubMed dataset.

As shown in Fig. 7, the proposed method in 60% of the queries provided more relevant results than PubMed. The meaning of 60% here is that in 6 out of 10 given queries, the proposed system works better.

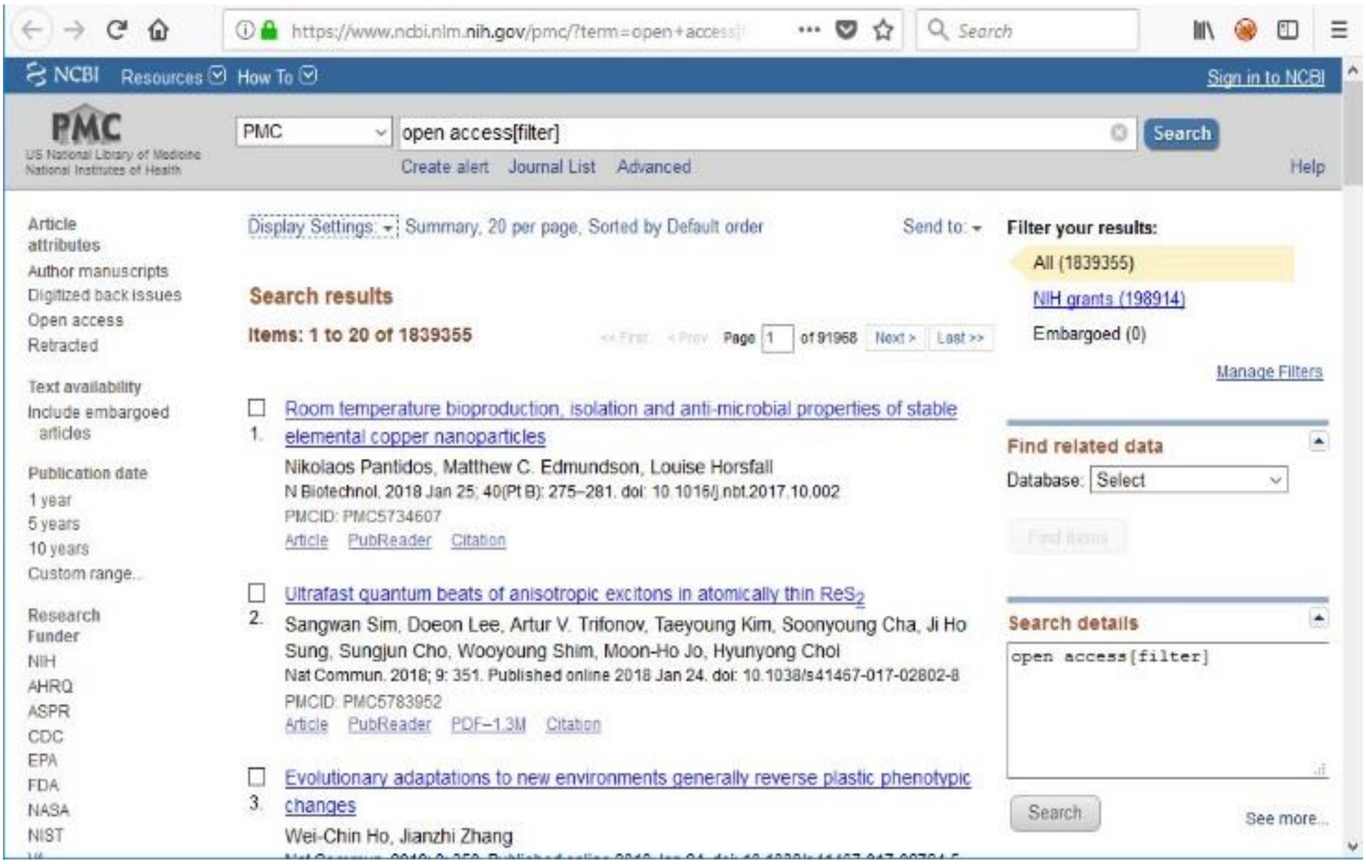
The achieved results for various queries for the proposed system and the D2SPR database are shown in Fig. 8.

As shown in Fig. 8, the proposed method in 80% of the queries provided more relevant results than D2SPR. By 80% here it is meant that in 4 out of 5 given queries, the proposed system works better.

4.7. The Measure of F Criterion

In order to evaluate the efficiency of the proposed system, the PubMed Publisher search engine was run using the *F-criterion* measure according to Eq. (5). The important point here is that to calculate accuracy and retrieval, there needs to be an expert in the search field to evaluate the retrieved documents in terms of relevance. Because of using the PubMed database in the medical field, two doctors have been requested to evaluate up to 50 first data retrieved for each query in the PubMed database as well as in the proposed system, compare based on the relevance of the request, and numbers between 0 and 10 for each document attributes to them. For calculating the *F* criterion and comparing the two systems, documents with a score above 5 are considered as relevant while those with a score less than 5 are considered unrelated documents. Based on this criterion, Fig. 9 presents the performance of the proposed system compared to the PubMed engine.

The proposed system outperformed all queries from the PubMed system for the first 50 data. This result is expected because both systems use a search and indexing mechanism in this section, except that the proposed method provides better data because of the better quality of the first 50 data. On average, the F index for PubMed for queries is 0.654 and for the proposed system is 0.826. This difference



https://www.ncbi.nlm.nih.gov/pmc/?term=open+access[f

Search

NCBI Resources How To Sign in to NCBI

PMC
US National Library of Medicine
National Institutes of Health

PMC | open access[filter] | Search

Create alert Journal List Advanced Help

Article attributes
Author manuscripts
Digitized back issues
Open access
Retracted

Text availability
Include embargoed articles

Publication date
1 year
5 years
10 years
Custom range...

Research Funder
NIH
AHRQ
ASPR
CDC
EPA
FDA
NASA
NIST

Display Settings: Summary, 20 per page, Sorted by Default order Send to:

Search results

Items: 1 to 20 of 1839355 << First < Prev Page 1 of 91968 Next > Last >>

□ 1. Room temperature bioproduction, isolation and anti-microbial properties of stable elemental copper nanoparticles
Nikolaos Pantidos, Matthew C. Edmundson, Louise Horsfall
N Biotechnol. 2018 Jan 25; 40(Pt B): 275–281. doi: 10.1016/j.nbt.2017.10.002
PMCID: PMC5734607
Article PubReader Citation

□ 2. Ultrafast quantum beats of anisotropic excitons in atomically thin ReS_2
Sangwan Sim, Doeon Lee, Artur V. Trifonov, Taeyoung Kim, Soonyoung Cha, Ji Ho Sung, Sungjun Cho, Wooyoung Shim, Moon-Ho Jo, Hyunyong Choi
Nat Commun. 2018; 9: 351. Published online 2018 Jan 24. doi: 10.1038/s41467-017-02802-8
PMCID: PMC5783952
Article PubReader PDF–1.3M Citation

□ 3. Evolutionary adaptations to new environments generally reverse plastic phenotypic changes
Wei-Chin Ho, Jianzhi Zhang

Filter your results:
All (1839355)
NIH grants (198914)
Embargoed (0)
Manage Filters

Find related data
Database: Select
Find items

Search details
open access[filter]
Search
See more...

Fig. 6. . Search for the Open-access Recourses in the PubMed Database

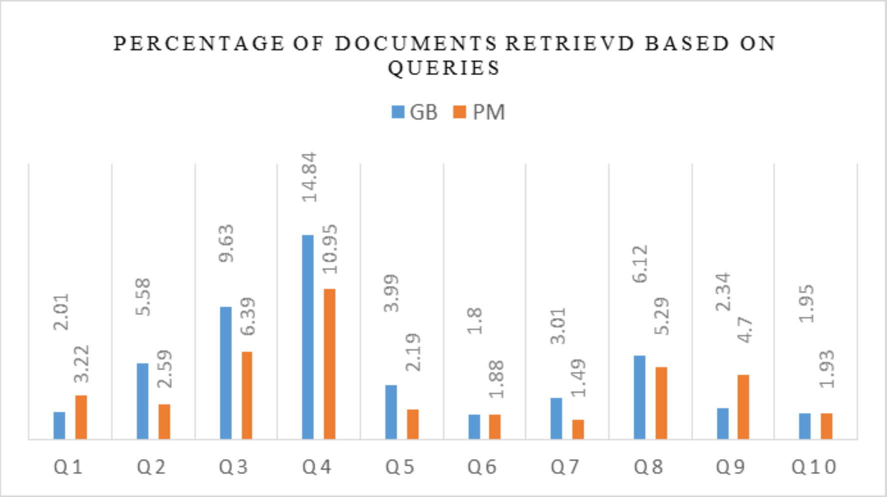


Fig. 7. Sample Query Results from the Proposed System and the PubMed database

reflects the better performance of the proposed system than the PubMed database engine.

Fig 10 compares the performance of the proposed system with D2SPR database.

As shown in Fig. 10, the proposed method in 80% of the queries provides more relevant results than D2SPR. The meaning of 80% here is that in 4 out of 5 given queries, the proposed system works better.

4.8. Normalized Discount Cumulative Interest Rate Measure

Since one of the main benefits of the proposed system is to provide more relevant documents based on the level of relevance to the query and the number of citations, we need a measure to assess the presentation quality of the related documents based on their presentation order. Here, a human expert agent is used to evaluate the relevance of documents recovered by each system, which assigns each numeric

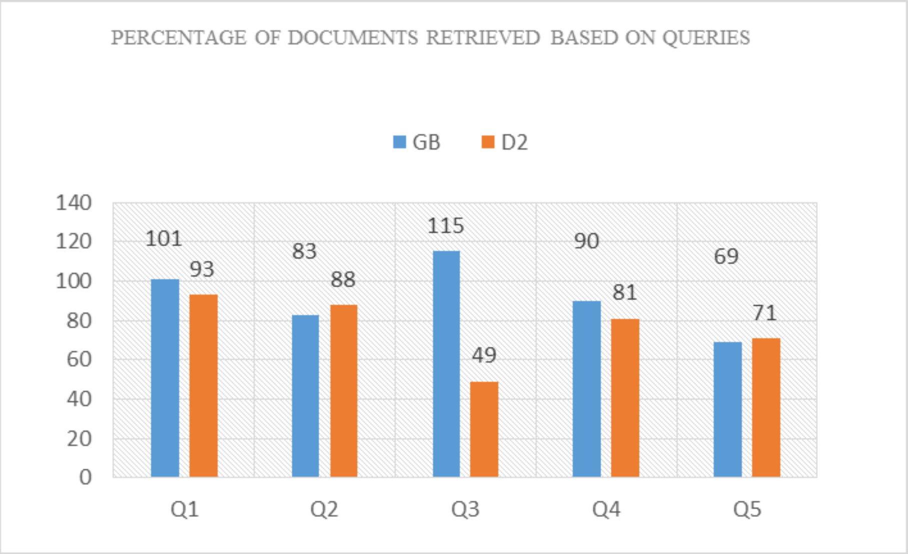


Fig. 8. . Sample Query Results from the Proposed System and the Dwaipayan System on the D2SPR database

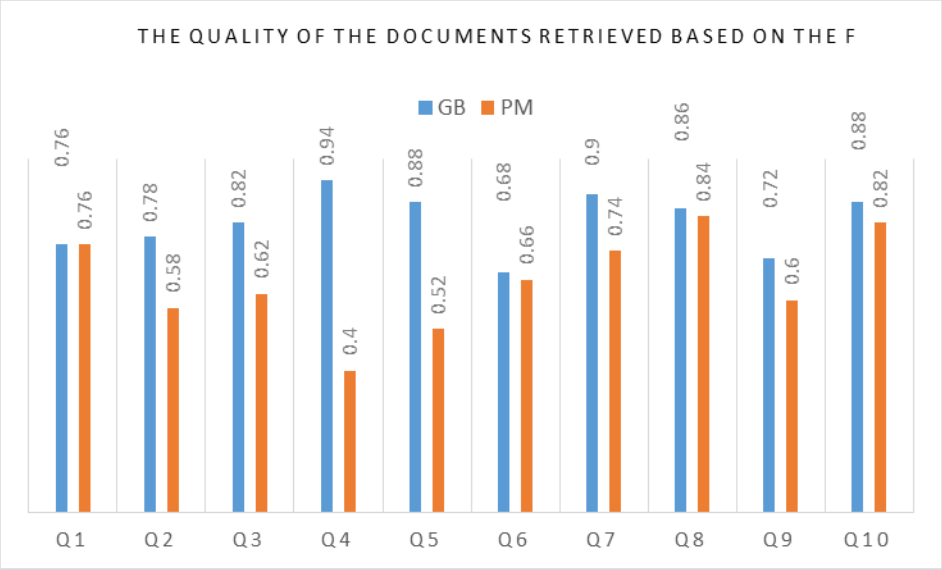


Fig. 9.. Results from Applying Sample Queries to the Proposed System and the PubMed Database Based on *F* Score Measure

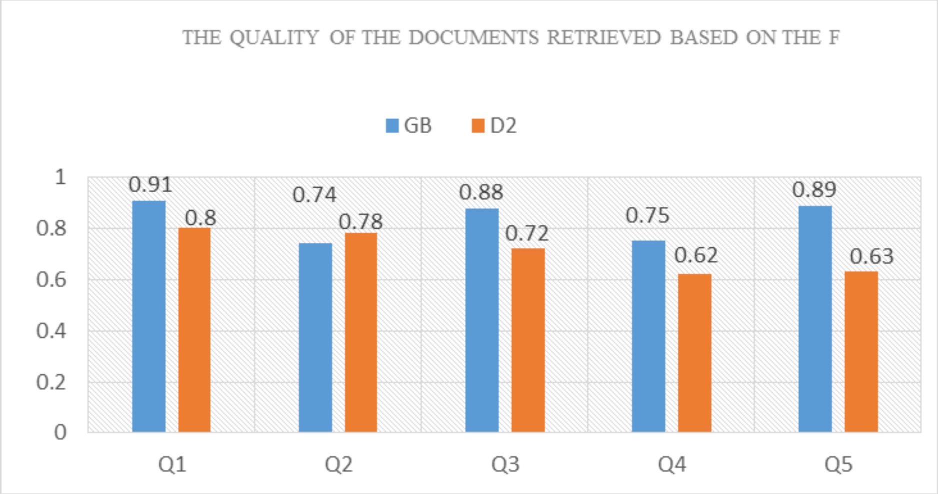


Fig. 10. Results from Applying Sample Queries to the Proposed System and the D2SPR Database Based on *F* Score Measure

document between 0 and 10 based on its relevance to the input query. The function of the proposed system is shown in Fig. 11 compared to the PubMed system engine.

The difference in the proposed method with the PubMed search engine's default method is more pronounced in this measure. In the PubMed engine, documents are mostly retrieved according to the date and adaptation of words while referrals relations are not included in them. However, in the proposed method, because of the presence of referral relations in the ranking, documents with a higher quality show better rankings. The average nDCG for the proposed method is 0.828 and for the PubMed engine is 0.30. These numbers by themselves indicate the difference in quality and the order in which the documents are presented.

Fig. 12 compares the performance of the proposed system with the D2SPR database. As can be noted, the difference in the proposed method with the D2SPR database default method is more pronounced in this measure. In the proposed method, because of the presence of referral relations in the ranking, documents with quality higher are displayed in better rankings.

The average nDCG for the proposed method and the D2SPR database is 0.828 and 0.702, respectively. These numbers by themselves indicate the difference in quality and the order in which the documents are presented.

5. . CONCLUSION AND FUTURE WORKS

The volume of scientific information produced in recent years and the increasing need for researchers to provide more and better access to scientific documents have motivated them to provide new models and approaches in providing and retrieving information. One of these approaches is the use of existing geometric information in scientific documents, mainly published in PDF format. In this research, we tried to provide a systematic method of extraction of metadata from scientific documents and then using a model of the graph to use this data in order to retrieve more and better scientific documents.

By evaluating different approaches, the CERMINE framework, which is currently the most powerful information extraction framework for scientific documents, was selected as the base model for the extraction part. This framework receives a scientific document in PDF format as.

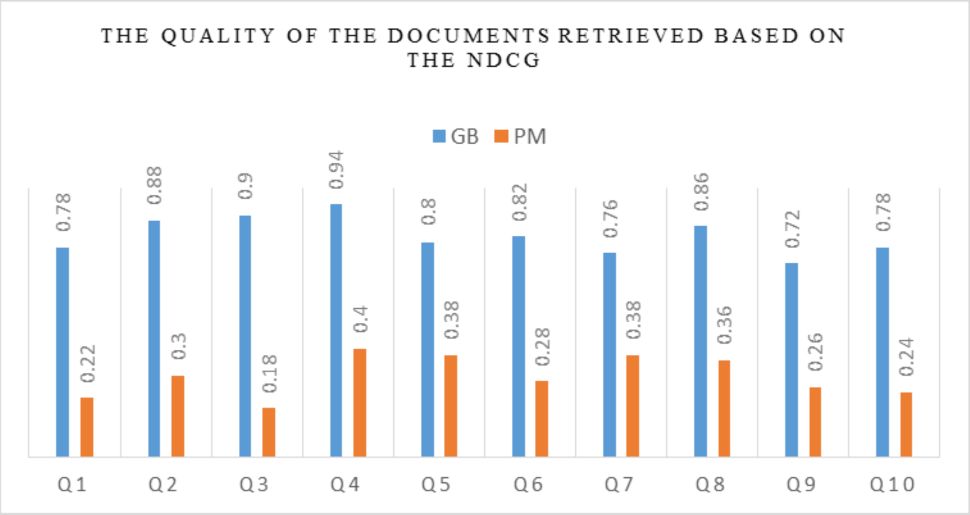


Fig. 11. Results of Applying Sample Queries to the Proposed System and the PubMed Database Based on the nDCG measure

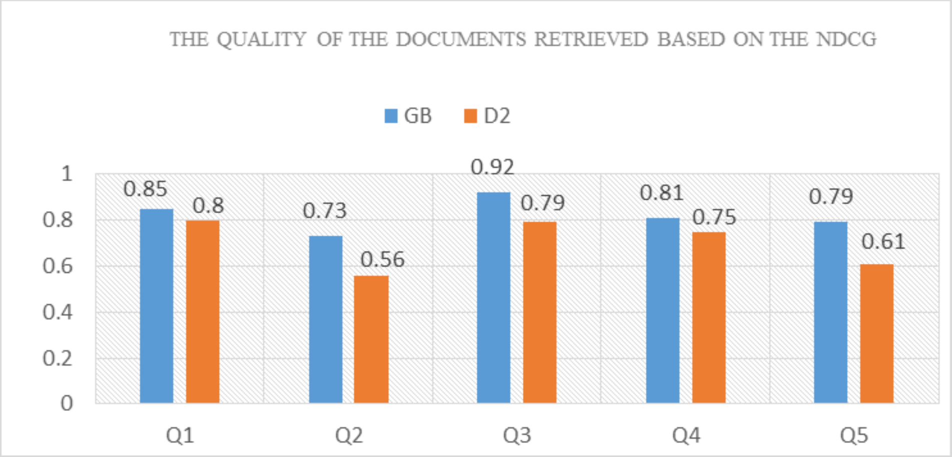


Fig. 12. Results of Applying Sample Queries to the Proposed System and the D2SPR Database Based on the nDCG measure

an input. The extraction algorithm looks for the entire contents of the document and generates two types of outputs: i.e., document metadata and references

The second phase of the proposed method is the scanned phase and the construction of the model of the graph. The scanning is done in order to evaluate the information obtained from the extraction stage before entering into the graph structure. The scanning operations include examining inconsistencies and removing them based on a number of rules.

After the scanning, it is tried to model the data as a graph. In the next step, the most commonly used methods are used. After reviewing the required parameters based on parameters such as flexibility, ease of use, being up-to-date, and availability, the Neo4j-based database is selected as the modeling infrastructure of the graph.

In the third phase of the system, a web interface for designing and retrieving documents to the system is designed and implemented. The document retrieval mechanism is based on the model of the graph and also is based on weighing on various parameters such as the number of citations, number of adaptation of keywords, and so on.

To assess the performance of the proposed system, the PubMed database is selected. The results show that even with a smaller number of documents in the set of used systems, in 60% of the queries, there are more related records retrieved by the proposed system. In addition, retrieved documents have a higher quality than the PubMed database; based on an *F* score measure with the improvement of 0.170 and the nDCG measure with an improvement of 0.520.

Although the proposed method has good performance in the experimental dataset, it also has some shortcomings that can be improved by some fixing methods, presented in the following:

- Setting system parameters including SVM classification parameters, the type of used classification algorithms, the type of clustering algorithm in the reference extraction part, the mechanism for its detection in the system and the strategies for exiting from it
- Supporting documents containing Persian and Arabic texts: By adding a coding recognition unit and language recognition, it is first possible to identify the dominant language on each page of the document and then determine the order of the readings of the different areas on the page.
- Adding an OCR unit for the scanned documents: In order to recognize the characters and extract geometric features from the image in a text, we need an optical text recognition unit or OCR to use the infrastructure created for scanned documents.
- Enriching the system with more samples: As the number of samples in the system increases, the generated graph will be richer in terms of the number of documents and existing relationships.
- Adjusting the weights associated with the rankings of the documents using the methods of optimization or machine learning to minimize the error and achieve ideal weights in the rating.

References

[1] G.Giuffrida , EC. Shek , J.Yang, "Knowledge-based metadata extraction from postscript files.," ACM DL, p. 77–84,San Antonio, Texas, USA — June 02 - 07, 2000.

[2] A.Constantin , S.Pettifer , A.Voronkov, "PDFX: fully-automated pdf-to-xml conversion of scientific literature.," in ACM Symposium on Document Engineering, 2013.

[3] H.Han , C.L. Giles. E. Manavoglu, "Automatic document metadata extraction using support vector machines.," in ACM/IEEE 2003 Joint Conference on Digital Libraries, Houston, Texas, 2003.

[4] A. Kovačević , D.Ivanović. , B.Milosavljević , Z.Konjović , D.Surla, "Automatic extraction of metadata from scientific publications for CRIS systems.," in Program 45(4), 2011.

[5] P.Lopez, "GROBID: combining automatic bibliographic data recognition , term extraction for scholarship publications.," in Research , Advanced Technology for Digital Libraries, 13th European Conference, Berlin, Heidelberg, 2009.

[6] MY.Day , RT.Tsai , CL.Sung ,CC.Hsieh ,CW.Lee , SH.Wu , KP.Wu , CS.Ong , WL.Hsu, "Reference metadata extraction using a hierarchical knowledge representation framework.," Decision Support System, vol. 43, no. 1, p. 152–167, 2007.

[7] E.Cortez , ASd.Silva , MA.Gonçalves , F.Mesquita , ESd.Moura, "FLUX-CIM: flexible unsupervised extraction of citation metadata.," in ACM/IEEE Joint Conference on Digital Libraries, Vancouver, BC, Canada, 2007.

[8] X.Zhang , J.Zou , DX.Le , GR.Thoma, "A structural SVM approach for reference parsing.," BMC Bioinform., pp. 12(S–3), S7, 2011.

[9] E. Hetzner, "A simple method for citation metadata extraction using hidden markov models.," in ACM/IEEE Joint Conference on Digital Libraries, Pittsburgh PA, PA, USA,2008.

[10] CERMINE. [Online]. Available: http://cermine.ceon.pl . [Accessed 27 1 2018].

[11] DBLP. [Online]. Available: http://dblp.uni-trier.de/. [Accessed 27 1 2018].

[12] A.Singal, "Introducing the knowledge graph: Things, not strings," 2012.

[13] L.Ehrlinger , W.Wöß, "Towards a definition of knowledge graphs.," in SEMANTiCS, 2016.

[14] J.Hoffart , FM.Suchanek , K.Berberich , G.Weikum , "YAGO2: A spatially , temporally enhanced knowledge base from Wikipedia," Artificial Intelligence, vol. 194, pp. 28-61, 2013.

[15] Google, "FreeBase," [Online]. Available: https://developers.google.com/freebase/. [Accessed 27 1 2018].

[16] S.Choudhury , Kh.Agarwal , S.Purohit, B.Zhang , M.Pirrung , W.Smith , M. Thomas, "NOUS: construction , querying of dynamic knowledge graphs.," in ICDE, 2017.

[17] X.Dong , E.Gabrilovich , G.Heitz , W.Horn , N.Lao , K.Murphy , T.Strohmann , Sh.Sun , W.Zhang, "Knowledge vault: a web-scale approach to probabilistic knowledge fusion.," in SIGKDD, 2014.

[18] A.Carlson , J.Betteridge , B.Kisiel , B.Settles , ER. Hruschka Jr , TM. Mitchell, "Toward an architecture for never-ending language learning.," in Proceedings of the 28th AAAI, 2010.

[19] C. L. M.-E. V. S. A. Afshin Sadeghi, "Integration of Scholarly Communication Metadata Using Knowledge Graphs," in Research and Advanced Technology for Digital Libraries. New York, NY, USA,TPDL 2017, 2017.

[20] I. P. G. F. A. V. M.-E. Traverso-Rib´on, "Considering semantics on the discovery of relations in knowledge graphs.," in Blomqvist, E., Ciancarini, P., Poggi, F., Vitali, F. (eds.) EKAW 2016. LNCS (LNAI),.

[21] N. S. B. a. V. M. Aggarwal, "Connecting the Dots: Explaining Relationships Between Unconnected Entities in a Knowledge Graph.," 2016.

[22] X. e. a. Cai, "Greta: Graph-based tag assignment for github repositories.," in IEEE Computer Software and Applications Conference (COMPSAC), IEEE 40th Annual, Atlanta, GA, USA, 2016.

[23] C. Park, "Keyword Search over Graph-structured Data for Finding Effective and Non-redundant Answers.," in SEKE, 2016.

[24] R. a. L. D. Peddinti, "Structured search query generation and use in a computer network environment.". Patent Google Patents, 2017

[25] PubMed. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed. [Accessed 27 1 2018].

[26] TutorialsPoint, "Neo4j Introduction," [Online]. Available: https://www.tutorialspoint.com/neo4j/index.htm. [Accessed 27 1 2018].

[27] D. S. P. F. M. e. a. Tkaczyk, "CERMINE: automatic extraction of structured metadata from scientific literature ," IJDAR, vol. 18, no. 4, p. 317–335 , 2015.

[28] Gephi. [Online]. Available: https://gephi.org/about/. [Accessed 27 1 2017].

[29] "Dataset 2 for Scholarly Paper Recommendation," [Online]. Available: http://www.comp.nus.edu.sg/~sugiyama/Dataset2.html . [Accessed 12 1 2018].

[30] K . Sugiyama , M.-Y. Kan, "A comprehensive evaluation of scholarly paper recommendation using potential citation papers", "cc," International Journal on Digital Libraries, vol. 16, no. 2, pp. 91-109, 2015.

Farzaneh Norouzi is a graduated student in master of software engineering at the university of Science and Culture, Tehran, Iran. Her research interest is information retrieval.

Fatemeh Azimzadeh received PhD degree in Information Technology from the University Putra Malaysia in 2012. Currently she is an assistance professor in ACECR, Tehran, Iran. She is also the director of SID (Scientific Information Database) in Iran. Her research interests include information retrieval and information quality.