International Journal of Hybrid Information Technology
Vol.8, No.3 (2015), pp.255-272
http://dx.doi.org/10.14257/ijhit.2015.8.3.23

An Effective Academic Research Papers Recommendation for Non-profiled Users

Damien Hanyurwimfura[1,3], Liao Bo[1], Vincent Havyarimana[1], Dennis Njagi[2] and Faustin Kagorora[1]

[1] *Key Laboratory for Embedded and Network Computing of Hunan Province, the College of Computer Science and Electronics Engineering, Hunan University, Changsha, 410082, China*
[2]*College of Information Science and Engineering, Central South University, China*
[3]*College of Science and Technology, University of Rwanda, Kigali, Rwanda*
hadamfr@yahoo.fr

Abstract

With the tremendous amount of research publications online, finding relevant ones for a particular research topic can be an overwhelming task. As a solution, papers recommender systems have been proposed to help researchers find their interested papers or related papers to their fields. Most of existing papers recommendation approaches are based on paper collections, citations and user profile which is not always available (not all users are registered with their profiles). The existing approaches assume that users have already published papers and registered in their systems. Consequently, this neglects new researcher without published papers or profiles. In this paper, we propose an academic researcher papers recommendation approach that is based on the paper's topics and paper's main ideas. The approach requires as input only a single research paper and extracts its topics as short queries and main ideas' sentences as long queries which are then submitted to existing online repositories that contains research papers to retrieve similar papers for recommendation. Four query extraction and one paper recommendation methods are proposed. Conducted experiments show that the proposed method presents good improvement.

Keywords: *academic paper recommendation, topics extraction, paper relationship, multi words topics, cosine similarity*

1. Introduction

Researchers are spending amount of time on Internet in researching papers that cover their interest due to the high volume of available resources online. Many online scientific papers repository such as journals and conference proceedings arrange their published research papers according to the year of publication, volumes and numbers, which make it difficult to find related papers. Many of those journals and conferences are not indexed or abstracted in Google scholar (the most used search engine for many researchers) that might be easier to find related papers. This means that to find papers, a reader or researcher has to know the link to the journals and then one can search them by year of publication, volumes and numbers which consumes large amount of time. The most reliable solution for this problem is paper recommendation systems. Papers recommendation systems aim at recommending relevant papers to researchers with respect to their individual demands [1].

The most apparent goal of a recommender system is to satisfy its users' information needs. One user may be interested in the most recent research papers on his field, while another may be interested in the first publication in one area or

ISSN: 1738-9968 IJHIT

just related papers. In many cases, after reading an academic paper, users probably want to find more related papers which solve the same problem or use the same method. One user may also want to compare his/her results with the most recent paper that solve the same problem.

Most of existing papers recommendation systems are based on user profiles [5, 11-14] and citation relation [1]. These user profiles based systems require that the users are already registered in the systems with their profiles and the papers are recommended based on the similarity between their profiles. Users with similar profiles are recommended the same papers. These methods have some limitation as for unregistered researchers or just a fresh researcher cannot use or benefit from them. Other recommender systems require their users to provide keywords that represent their interests. In such a case, a research paper recommender system does not differ from a normal academic search engine where a user provides a search query to retrieve relevant papers. This is a powerful approach, but it is also limited. Forming queries for finding new scientific articles can be difficult as a researcher may not know what to look for; search is mainly based on content, while good articles are also those that many others found valuable; and search is only good for directed exploration [16]. The shortcoming of this approach is that it is then the responsibility of the users to translate their information needs and goals in the best possible query with the most suitable terms trying to retrieve all and only all papers they may actually need [2]. This paper proposes effective methods that can formulate those queries on behalf of the user to retrieve the related papers.

To solve the problem of non-registered or fresh users who have just read one paper and want similar papers related to it, Cristiano, *et al.*, [2] developed a scholarly paper recommendation system, in which they use the title to construct user profiles, and the title and abstract to generate feature vectors of candidate papers to recommend . It is a scholarly paper recommendation system based on content-based filtering which is the same approach as ours. Their method requires as input only a single research paper and generates several potential queries by using terms in that paper, which are then submitted to existing web information sources that hold research papers. They consider title, abstract and body as target section for queries generation and used title and abstract section for candidate papers generation. However, we feel that such a small span of text does not effectively represent a user's interest of the candidate paper. We propose an approach that can recommend related papers based on the topics the target paper is addressing and its main idea by considering the full paper content in queries generation. For paper recommendation; title, abstract, introduction and related works sections for candidate paper are considered.

In this paper, four algorithms are proposed and applied to the different parts of the target paper to generate topics as short or long queries. The first algorithm is applied to the title and references of the target paper to generate short queries. The second one is applied to the abstract to extract main idea the paper is talking about based on some cue words to generate long queries. The last two algorithms are applied to body and only few sentences or phrases that are more relevant to the paper's main idea are selected as long or short queries.We propose a new method for paper recommendation which is based on similarity of specific fields of both the target paper and candidate paper (Title, abstract, introduction and related works sections) and uses cosine similarity function to measure the relevance.

2. Related Works

2.1. Paper's Topics Extraction

Some approaches have been proposed for scientific paper's topics extraction. Buitelaar, *et al.,* [26] proposed a topic extraction method from scientific literature for competency management. It is based on the extraction of relevant competencies and semantic relations between them through a combination of linguistic patterns, statistical methods as used in information retrieval and machine learning and back-ground knowledge if available. This method uses the domain-specific linguistic patterns for the extraction of potentially relevant competencies, such as scientific topics and technologies, from publicly available scientific publications. The core assumption of this approach is that such topics will not occur in random fashion across documents, but instead occur only in specific scientific discourse contexts that can be precisely defined and used as patterns for topic extraction.

Latent Dirichlet allocation (LDA) [25] has been considered in previous methods to extract topics from text documents. The basic idea behind LDA is that documents are represented as random mixtures over latent topics, where each topic is characterized by a distribution over words. Griffiths, *et al.,* [24] used LDA to find scientific topics from abstracts of papers published in the proceedings of the national academy of sciences. but the main disadvantages of this model are that the topics are distribution over single words and thus the semantics is lost; and this method needs a pre-defined number of latent topics (people can choose a different number of topics, thus producing different results) and manual topic labeling, which is usually difficult for people.

Hanyurwimfura, *et al.,* [9] proposed an unsupervised learning method for research papers organization. This method extracted topics based on the relationship between the paper's title, frequent sentences and most similar references to the paper's title. It means that only frequent sentences and cited references most related to the paper's title are only considered in topics extraction; and based only on this, some topics are not extracted. A better approach is to extract all paper's topics, it is proposed in this paper, because paper's title features of top frequent sentences relationship, keywords relationship and references relationship are not enough to get all topics that the paper is addressing.

Shubankar, *et al.,* [8] proposed a method that uses closed frequent keyword-set of the titles' phrases to form topics. In their proposed approach, they form closed frequent keyword-sets by top-down dissociation of keywords from the phrases present in the paper's titles on a user-defined minimum support [27]. In order to formulate topics in the paper's title, their method extracts substrings of paper' phrases as topics which require many steps and iterations as follows: from keyword-set to frequent keywords-set, from frequent keywords-set to closed frequent keywords-sets as topics. Their method considers only the paper's title and ignores the topics in the rest of the paper which means that many topics are not extracted. Our proposed methods also extract topics using paper' phrases but does not need the iterations, just keyword-set as topics in order to minimize running time and space. It considers also other parts of the paper and extracts multi co-occurring terms based on words adjacency.

2.2. Research Paper Recommendation

Recommendation systems for research articles are useful applications, which for instance help researchers keep track of their research field and recommend relevant papers with respect to their individual interests. There are two mainly approaches in filtering: collaborative filtering (CF) and content-based filtering

(CBF). The basic idea of collaborative filtering was that users like what like-minded users like. It is just filtering information based on user similarity. Two users are considered similar, when they like the same items, in our case here items are research papers. CBF is based on the idea that users are interested in items being similar to the ones they already are connected to. Each item is represented by a content model that contains the items' feature. The collaborative filtering is supposed to provide unexpected recommendations because recommendations are not based on item similarity but on user similarity [23].

Wang, *et al.,* [16] proposed a collaborative topic regression model which combines ideas from CF and content analysis based on probabilistic topic modeling. They developed an algorithm for recommending scientific articles to users of online archives where each user has a library of articles that he /she is interested in, and their goal was to match each user to articles of interest that are not in his/her library. They used the abstract and title of the paper to model a user and characterize candidate papers to recommend, which occasionally results in irrelevant recommendations. The abstract and title are not good enough to help know the content of the paper as some abstracts are not well written due to the expertness of the author or the abstract length limit(may be 100 words) suggested by journal or conference format. Sugiyama, *et al.,* [15] considered citation sentences, abstract, introduction and conclusion sections to get good recommendation results.

Cristiano, *et al.,* [2] introduced a source independent framework for research paper recommendation. Their method requires as input only a single research paper and generates several potential queries by using terms in that paper, which are then submitted to existing web information sources that hold research papers. They consider title, abstract and body as target section for queries generation and used title and abstract section for candidate paper generation, stating that title and abstract are only publicly available section for researcher. Their approach generates a 2-gram word and noun-phrases extracted using part of speech tagging as queries. However, we feel that such a small span of text does not effectively represent a user's interest of the candidate paper. Title and abstract are not enough to provide paper's information for recommendation. We believe that all researcher institutions are subscribed to those well known Web information sources providing full papers such as ACM Digital Library, IEEE Xplore, and Science Direct, *etc.*, and using the full candidate paper can improve the accuracy and provide relevant papers.

Their approach presents some disadvantages: (1) the 2-gram words generated as queries are short and probably many and consequently they can increase running time (as many queries will be submitted). (2) Only using title and abstract of candidate paper can lead to irrelevant recommendation papers and can not retrieve all relevant papers. The solution to this problem is to reduce the number of queries while keeping submitting the same information to get similar results and this is one of contributions of this paper. Different methods are applied to select important sentences containing main idea of the target paper and the Part Of Speech tagger is applied to the important selected sentences, and only sequence of two or more nouns or a sequence of adjectives followed by a sequence of nouns are selected as queries, thus the number of queries is reduced. This means that many noun-phrases form one query which is submitted at once. Another solution is considering the full content of the candidate paper for recommendation. The proposed method considers the full paper content as in [2] to generate queries and only main topics and main ideas sentences are extracted and submitted as queries.

Jiang, *et al.,* [3] presented recommending academic papers method via users' reading purposes. They are interested to satisfy user-specific reading purposes by recommending the most problem-related papers or solution-related papers to users separately. For a target paper, they use the paper citation graph to generate a set of

potential relevant papers. Once getting the candidate set, they calculate the problem-based similarities and solution-based similarities between candidates and the target paper through a concept based topic model, respectively. This method considers only abstract section. Unfortunately; on the other hand, many abstracts do not adequately describe all aspects of a paper's contribution [7] and considering it alone can lead to poor recommendation results.

3. Academic Research Papers Recommendation for Non-profiled Users

A scientific paper can deal with multiple topics, and the words that appear in that paper reflect the particular set of topics it addresses. Scientific paper topics are defined as phrases that capture the main topics discussed in a paper [24]. They offer a brief precise summary of the paper content; they can be very useful in paper recommendation process by retrieving or recommending the papers dealing with similar topics. In this paper we extract topics that the paper is addressing and those topics are considered as queries that can help retrieve similar papers for recommendation. This paper solves the same problem as [2] but different methods are applied to improve it. Figure 1 shows the architecture of the proposed approach which mainly consists of 2 stages: Candidate queries extraction and Paper recommendation.

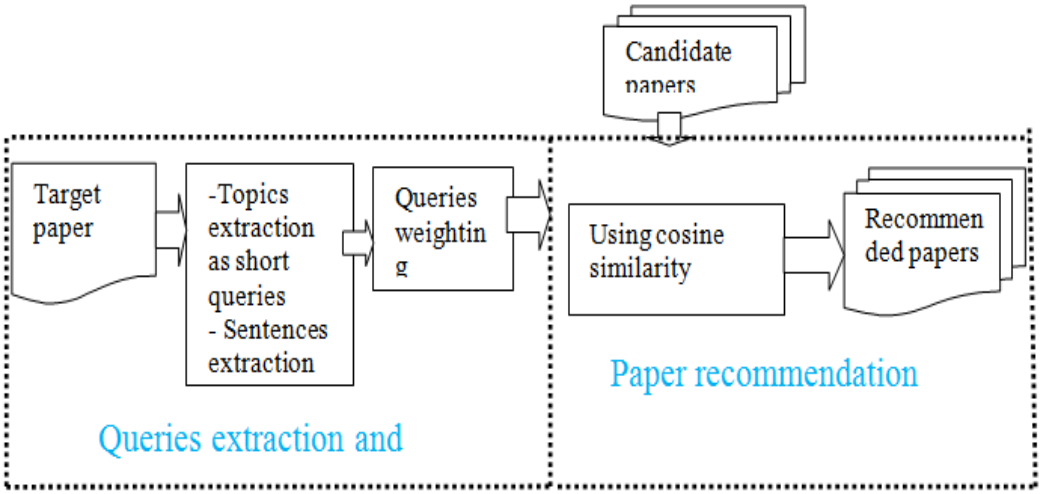

Figure 1. The Proposed Paper Recommendation Approach

Queries extraction stage and weighting: In this stage, different algorithms are proposed to extract main topics and main idea sentences the papers is addressing. The input to this stage is only the target paper from which the queries are generated. Main topics are extracted as short queries and main idea sentences are extracted as long queries. Because main topics and sentences can be extracted by our proposed algorithm, a queries weighting methods is applied to remove inadequate queries that can retrieve irrelevant papers or slow the retrieval. Only relevant queries are selected.

Papers recommendation stage: In this stage, one technique is proposed to retrieve related papers for recommendation. It is based on the similarity between the contents of both target and a candidate paper. If the two contents are similar then the paper is recommended. The inputs to this stage are selected queries and candidate papers from online paper repository databases.

3.1. Candidate Queries Rxtraction

Our aim is to extract from the paper the best queries that can retrieve the best results. Different techniques are proposed: some can extract topics the paper is addressing as short queries and others can extract main ideas sentences as long queries in the interest to capture the full content of the paper.

3.1.1. Extracting Long Queries: Firstly, This paper extracts important sentences that capture the main idea of the paper in the abstract as it is the main section of the paper read by many researchers. Second, sentences appearing in other sections that are very similar to the paper's title and those containing most frequent terms are also extracted.

Our method first generates sentences from abstract based on cue words. Selecting sentences with keywords that express the meaning of the sentence can usually represent theme of the document. They are indicative sentences and contain key phrases. For example in the abstract, the main idea or important points or the proposed solution is located in sentences containing cue words like " *in this paper*, "*approach*, *method*, *contribution*, *model*, *framework*, *study*, *algorithm*, *solution*, etc . Sentences containing these keywords are selected as candidate queries. Since extracted sentences contain some unnecessary words that are not useful, a Part of Speech (POS)[1] tagger is applied and remaining parts are considered as long queries.

For example in the abstract of [2], if the following sentence is extracted as main idea sentence:

"The framework requires as input only a single research paper and generates several potential queries by using terms in that paper, which are then submitted to existing Web information sources that hold research papers". A POS tagger is applied as follow:

The/DT framework/NN requires/VBZ as/IN input/NN only/RB a/DT **single/JJ research/NN paper/NN** and/CC generates/VBZ ***several/JJ potential/JJ queries/***NNS by/IN using/VBG terms/NNS in/IN that/DT paper/NN,/, which/WDT are/VBP then/RB submitted/VBN to/TO **existing/JJ** ***web/NNP information/NN sources/NNS that/WDT hold/VBP research/NN papers/NNS****"*

Then, only sequence of two or more nouns or sequences of adjectives followed by a sequence of nouns are considered as queries. The bolded sequences are selected as queries as follows:

"Single research paper several potential queries existing web information sources research paper".

Other sentences that our method considers as main content sentences (queries) are those sentences from other sections of the paper excluding tables, figures, equations, symbols and footnotes, experimental setup, discussion and references that are frequent and most similar to the paper's title.

Generally, sentences similar to the title contain important terms [4]. In Mock [21] terms that occur in the title have higher weights. But the effectiveness of this method depends on the quality of the title. For example, same paper's titles are short because some conferences and journals limit the paper' title to 8 words for example and this kind of title cannot capture all paper content.

For example, let us consider the following title; "*Query by document*" [10] .This title alone is too short to be used as a query in order to retrieve the best results, only two words are searched. Nevertheless, sentences with the term "query" should be handled importantly because they can have key terms about the title. Similar sentences to the title contain the important terms generally. Information in which we are interested in bears relationship with the paper's title and such information will frequently appear in a number of sentences, thus all sentences containing terms in paper's title are extracted and considered as main content sentences or queries

[1] http://nlp.stanford.edu:8080/parser/index.jsp

candidates. Other sentences that contain the main idea of the papers are those containing most frequent terms. This paper uses Term Frequency (TF) to get all paper terms frequency and terms that are most frequent are extracted.

After getting most frequent terms, sentences containing those terms are extracted as queries candidates. Terms like *paper*, *approach*, *method*, *contribution*, *model*, *results*, *experiments*, *algorithm*, *study*, *framework*, *figure*, *table*, *study*, *etc* are not considered as frequent because they are always used in almost all papers. The final sentences queries are obtained after measuring their similarity with the paper's title.

We measure the similarity between the title and each sentence, and then we assign the higher importance to the sentences with the higher similarity. The title and each sentence of the target paper are represented as the vectors of content words. The similarity value of them is calculated by the inner product. The similarity value between the title T and the sentence S_i in a target paper P is calculated by the following formula:

$$Sim(S_i, T) = \frac{\vec{S_i}.\vec{T}.}{Max_{S_j \in P}(\vec{S_j}.\vec{T})} \quad (1)$$

Where $\vec{T}$ denotes a vector of the title T, and S_i denotes a vector of sentence.

Since the method by the title depends on the quality of the title, it can be useless in the document with a meaningless title. Besides, the sentences, which do not contain important terms, need not be handled importantly although they are similar to the title. On the contrary, sentences with important terms must be handled importantly although they are dissimilar to the title.

Considering these points, we first measure the importance values of terms by Term Frequency value and then the sum of the importance values of terms in each sentence is assigned to the importance value of the sentence. In this method, the importance value of a sentence S_i in a target paper is calculated as follows:

$$Score(S_i) = \frac{\sum_{t \in p} tf(t)}{\max_{s_j \in P}\left\{\sum_{t \in S_j} tf(t)\right\}} \quad (2)$$

Where $tf(t)$ denotes the term frequency of term t

Then two kinds of sentence importance (similarity and importance value score) are simply combined by the following formula:

$$TotalScore(S_i) = k \times Sim(S_i, T) + j \times Score(S_i) \quad (3)$$

The constant k and j control the rates of reflecting two importance values.

Because some important sentences may be very similar to ones extracted in abstract section, the similarity between each extracted sentences and each extracted sentences in abstract section is calculated and only dissimilar sentences to those extracted in abstract section are considered. If sentence Si is selected as important sentence and sentences Sa was selected in abstract section, then the similarity is calculated by the following formula:

$$Sim(Si, Sa) = \arg\max(2(Si \cap Sa)/(Si \cup Sa) \quad (4)$$

A Part of Speech (POS) tagger is also applied to each top important sentence to remove unnecessary terms and only sequence of two or more nouns or a sequence of adjectives followed by a sequence of nouns remain in the sentences.

3.1.2. Extracting Short Queries: The proposed approach extracts semantic topics as short queries from research paper using:

1) Phrases in the papers' title and cited references,
2) Frequent adjacent words

Two algorithms to extract paper's topics which will be used as queries in paper recommendation phase are proposed. The first one uses phrases appearing in papers' title, keywords and cited references as topics. The second one uses statistical information to extract high frequent multi co-occurring words topics respecting the order the words appear in research papers. The extracted topics are multi co-occurring words in the order they appear in original research paper.

Algorithm 1. Topics Extraction based on Phrases in the Paper's Title and Cited References

This method uses phrases appearing in papers' title, keywords and cited references to be topics as queries.

Shubankar, *et al.,* [8] defined a phrase P as a run of words between two stop-words in the title of a research paper. They used a comprehensive list of 671Standard English stop-words. This means that all stop-words are used in defining phrase and this requires space and time.

In our algorithm, we select few stops words in the list and called them "*relationship terms"* because they relate phrases in the paper' title. Then we define phrase as *a run of words between two relationship words*. We believe that there is no paper's title containing stop words like "*meanwhile, may be, nevertheless, thanks, etc*.,", thus considering all stop words is wasting time and space.

The relationship words are those that appear mostly in the paper's titles and are often used to show relationship between phrases containing the paper's title.

They are: *a, an, based on, based, on, using, for, by, of, in, with, to, by, through, as, etc*.

This paper assumes that a paper's title is made of one or many phrases separated by relationship words.

Paper's keywords are already in the form of phrases as written by authors and this paper considers them as paper's topics.

Shubankar, *et al.,* [8] extracts closed frequent keyword-sets as topics after many steps and iterations. Firstly, phrases are formulated, from phrases, the method extracts keyword-sets, from keyword-sets; frequent keywords-sets are formed and finally from frequent keywords-set, closed frequent keywords-sets are extracted as topics. Looking at those steps and iterations executed to get the paper's topics it is clear that they consume a lot of execution time and space, we have improved this by considering paper's topics at first step just using phrases as topics, and this reduces automatically the execution time.

For example the title from [2] "A source independent framework for research paper recommendation" generates the following 2 queries*: source independent framework* and "*research paper recommendation*" but using n-gram extraction by Cristiano [2] generates 5 queries: "*source independent*"," *independent framework*", "*framework research*", "*research paper*" and "*paper recommendation*" which indicate that our method reduces the number of queries but the same information is submitted and same results are returned.

We considered also references section and extract phrases in reference titles as topics that are addressed in the paper. Our method reduces the number of stop words to be checked in phrases extraction to a small number and also the execution time to extract paper's topics is reduced. The algorithm is executed as follow:

Input: Paper and cited reference titles as sentences T_n

Relationship words, RW_m

Output: extracted topics

1. For every sentence T_i, $i = \overline{1,n}$
2. For every relationship word RW_j, $j = \overline{1,m}$
3. If T_i contains RW_j
4. Extract right side phrase T_{i+1} and left side phrase T_{i-1} on RW_j,
5. Save extracted phrases in file F
6. End if
7. End for j
8. End for i

Figure 2. Phrase Extraction Algorithm from Title and References Sections

The extracted phrases are considered as topics of the paper is addressing. Duplicate topics were removed.

Algorithm 2: Frequent Adjacent Words Extraction Algorithm

This one uses statistical information to extract high frequent multi co-occurring words topics respecting the order the words appear in the research paper. Top frequent terms at certain threshold are considered to extract their adjacent words. We believe that if a word is most frequent in a research paper, some of its adjacent words are important and together they can form co-occurring word topics. This idea is considered in this paper and frequent words and their adjacent words are extracted as co-occurring words to form topics.

Let consider D, be a set of words in the document and $D = \sum_{i=1}^{N} w_i$ where w_i represents each word. Assuming that $f(w_i)$ is the frequency of w_i in the document and S is a set of the sentences in the document where $\sum_{k}^{M} s_k = S(= D), \forall N > M$. Then, the algorithm is defined as follows:

1. Compute $f(w_i)$, the frequency of each word w_i such as $\sum_{i}^{IN} f(w_i) = \beta, (\beta \in IN)$. Given that λ is a certain threshold, select w_i such that $f(w_i) \geq \lambda$. Let call p_j this selected word where $P = \sum_{j=1}^{L} p_j$.
2. Extraction of multiword from s_k :
   For each sentence s_k
   For each frequent word p_j
   If $p_j \in S_k \ (j = \overline{1,L};\ k = \overline{1,M})$
   Check w_{j-1} and w_{j-1} (where w_{j-1} and w_{j+1} are the preceding and next term of p_j respectively)
   If $f(w_{j-1})_k \geq 2$
   Extract $w_{(j-1)k}$ and p_{jk} (noted $wp_{(j-1)jk}$)
   Concatenate $w_{(j-1)k}$ and $wp_{(j-1)jk}$
   If $f(w_{j+1})_k \geq 2$
   Extract p_{jk} and $w_{(j+1)k}$ (noted $pw_{j(j+1)k}$)
   Concatenate p_{jk} and $w_{(j+1)k}$
   End for p_j
   End For s_k
3. Save $wp_{(j-1)jk}$ and $pw_{j(j+1)k}$ in a given file F

Figure 3. Frequent Adjacent Words Extraction

Duplicate topics were removed and there might be some overlapping among the extracted multi-words where some short multi words are subset of long multi-words, the long frequent co-occurring terms are selected as topics. For example if the term *paper* is the top frequent term and the words "*research"* and "*recommendation"* are frequent and adjacent to the term "*paper*" then "*research paper recommendation"* and "*paper recommendation* " are extracted by our algorithm, only *research paper recommendation"* is preferred. This because the topics *research paper recommendation"* can retrieve more results than what *paper recommendation* can retrieve.

Note that because some topics can be extracted by both algorithms, repeated topics are removed and we are remaining with single topics. For example a topic can be in the title's phrases and at the same time is extracted by either algorithm1 or algorithm 2, in this case this topic is considered once.

3.2. Selecting Final Queries or Queries Weighting

All selected sentences from abstract were considered as final long queries but about sentences extracted from the body, we simply set both constant weights (a and b) to 1.0 in the experiment and sentence with score of 0.5 and above were considered as final query. For short queries, the extracted topics by our 2 proposed algorithms mentioned above in the target paper independently are mixed, the frequency of each extracted topics is computed and highly frequent topics at a given threshold were selected as queries. In additional, if two topics share the same word, this means that they have some similarity then they are combined to make one topic. The repeated word is removed in order not to keep the unnecessary information. For example if topics like "*knowledge discovery* and *knowledge management*, were selected as topics, their common term is "*knowledge*", their combined topic will be "*knowledge discovery management*".

This solves the problem of repeated words in generated queries as proposed in [2]. For example in the following 5 generated queries from the title : *source independent framework research paper recommendation are:* "*source independent*", "*independent framework*", "*framework research*", "*research paper*" and "*paper recommendation".* You can see that the term independent, framework, research and paper are repeated twice which is a waste of time and space when submitting them. On the centrally, using our method,

the repetition is removed as stated above, which results in only 2 queries such as "*source independent framework*", and "*research paper recommendation".*

For 2 word topics, we computed the frequency of each topic and only topics with 4 frequencies and above have been selected as queries.

3.3 Papers recommendation

We consider content based approach for recommendation, taking into account both content of the target paper and candidate papers. We differ with [2] that argue that the only publicly available metadata is the title and abstract because many researcher institutions normally subscribe to those online databases providing full research papers. Because the full-text of a document contains many parts that do not clearly describe or are marginally relevant to its main contribution [7] (This can include experimental setup, discussion of notation, tables, figure, *etc.*,), we select only few sections that contain important information.

We consider sections that adequately describe the main content of the paper while providing information of other related papers. We consider title, abstract, introduction and related works sections of both the target paper and candidate papers in this approach, the cosine similarity is used to measure their similarity. We argue that introduction section describes the problem, background of the problem and the proposed method to solve it and this should be taken into consideration when seeking papers in similar domain. Introduction field was also considered by Sugiyama, *et al.,* [15].

Related works sections provided enough background of previous related methods or related literature, how other research solve similar problem, therefore using it in paper recommendation can yield most relevant papers.

We know that research papers are normally semi structured documents, where many fields are followed each other in the order. As each specific field gives its own contribution to the paper, the similarity between the target paper specific section and its corresponding section in the candidate papers is calculated and linear combination of fields is used to judge the similarity between both papers. Similarly, Strohman, *et al.,* [19] experimented with a citation recommendation system where the relevance between two documents is measured by a linear combination of text features and citation graph feature. Cristiano, *et al.,* [2] adapted linear combination of the titles and abstracts to get the relevance between research papers. Because an abstract alone does not contain enough information to identify much of the related literature, their methods lead to poor results [20].

On the other hand, to capture much information of the paper as well as related literature, we consider paper's title, abstract, introduction and related work sections. A well-known similarity measure is the cosine function, which is widely used in document similarity and for papers ranking [2, 3, 14] as well. Cristiano, *et al.,* [2] used cosine similarity and applied it to each of two selected fields (title and abstract) with different important values. This paper on the hand treats all selected fields equally. The importance was given by just selecting them from other fields of the paper and not using the full paper, those fields are selected to represent the paper. Since each field presents its own contribution to the paper, we calculate similarity between the same fields and the linear combination results the whole similarity between two papers. In this paper, cosine similarity function is applied to each field and a linear combination is calculated as follow:

$$Cos(i,c) = a*\cos(t_i,t_c) + a*\cos(a_i,a_c) + c*\cos(\text{int}_i,\text{int}_c) + d*\cos(rel_i,rel_c) \quad (5)$$

Where i is input paper, c is candidate paper, t_i is the title of input paper, t_c is the title of candidate paper, a_i is the abstract of input paper , a_c the abstract of candidate paper, int_i introduction of input paper, int_c introduction of candidate paper, rel_i is the related

work of the input paper and rel_c is the related work of candidate paper. The constant a b, c and d control the rates of reflecting 4 importance values to the title, abstract, introduction and related works sections.

4. Evaluation of the proposed method

The proposed approach was evaluated in order to know its efficacy. Two types of evaluation techniques were used to measure the effectiveness of the proposed approach, researcher intervention evaluation for topics extraction and performance metric evaluation for research paper recommendation. The goals of our experiments were to verify: (1) whether the automatically extracted topics and main idea sentences are relatively reliable compared to the author's suggestion. (2) Whether the generated queries can retrieve the best papers for recommendation compared to [2].

4.1 Topics Extraction Evaluation

The same evaluation method as [26, 28] was used to evaluate our topics extraction method. The primary question we address in the experiments of this section is whether the automatically extracted topics are relatively reliable compared to the author's topics suggestion. We want to verify also that the extracted main ideas sentences really describe the paper main contents.

Due to the fact that it requires much time to know the topics of someone else paper or main idea sentences, our evaluation involves our fellow classmates and co-workers who have previously published their papers in the journal. A group of 20 researchers has shown their willingness in evaluation of our method. They provided us their publication papers and used our method to extract the main topics and main idea sentences. We then asked each one to evaluate the topics and main ideas sentences extracted from theirs papers. The evaluation consisted of extracted topics and main ideas sentences, for which the researcher in question was asked simply to accept or decline each of the topic.

Some topics were accepted in a great number and others rejected in small number. Similarly some paper main idea sentences were accepted while few of them were rejected. The overall acceptance ratio show that 20 researchers that agreed to work with us covered 1132 extracted topics, out of which 774 were accepted as appropriate (68.3%) which is a very good indication that our topics extraction is very good. Similarly for 468 main sentences extracted, 314 of them were accepted as appropriate (67%) and this indicates that the sentences extraction methods are also good.

4.2. Paper Recommendation Evaluation

The accuracy of the proposed solution has been evaluated by using the well-known Recall [22] and Normalized Discounted Cumulative Gain [23]. The proposed approach is only useful if it is accurate and present good performance as expected.

4.2.1. Experiment Setup: The same digital libraries as used by previous approaches were considered: ACM Digital Library[2] in [2, 14, 16], IEEE explore[3] and science direct[4] [2] were used to evaluate the proposed approach. All those sources allow searching for publications in Computer Science. Similarly as in Section 4.1, twenty researchers, all of them Computer Science participated in our experiments, which were performed as follows: (1) we take each one paper provided to us as input paper of their interest; (2) we submitted each provided paper to our approach to extract queries that are submitted to retrieve a ranked list of recommended papers; (3) each researcher evaluated his/her list of recommended papers as being *strongly related*, *related*, or *not related* at all to the paper

[2] http://portal.acm.org/
[3] http://ieeexplore.ieee.org/
[4] http://www.sciencedirect.com/

given as input. The papers in the list to be evaluated were presented in a random order so that we avoided any bias (users tend to trust top-ranked documents).

The following configurations were used in our experiments: For long queries extraction, the similarity between the candidate sentence and title, the value of 0.5 minimum was found to produce better results. For the importance of sentence, the 10 top sentences were selected. The constant weights k and j for sentences score were simply set to 1. We also simply set both constant weights a , b , c and d in similarity calculation to 1.0 respectively. A minimum of 4 times was found to select better topics to be used in paper ranking method. For each paper, the extracted topics by each of the query extraction methods were submitted to the selected sources and we took first 20 top papers for each source resulting 20*3*4*4= 960 papers where 20 is the number of top returned papers, 3 is number of sources, 4 is the number of queries extraction methods and 4 is the number of fields considered (title, abstract and body, the body considered 2 times, one time for short queries and one time for long queries) respectively. In the case where duplicate papers were returned by different queries, duplicate papers were removed resulting in small number of paper to be evaluated, 380 in average. The same as in topics extraction evaluation, 20 researchers have evaluated the paper recommendation approach showing their agreement on the recommended papers as *strongly related*, *related* and *not related* to their input paper; we use their papers because it is easy for them to know the related papers as it is in their research areas. Only those topics selected as *strongly related* and *related* as queries are submitted to the three information sources. Table 1 shows number of selected and submitted queries for each queries extraction method.

Table 1. Number of Submitted Queries in Each Information Sources

Fields	**Algorithm**	**Number Short queries**	**Number of Long queries**
Title and references	Stop word based	10	-
abstract	Cue word based	-	All extracted sentences (queries)
Body	Algorithm 2	10	-
Body	Similarity and importance based	-	Top 10 important sentences

4.2.2. Evaluation Metrics : We adopted two well and widely used metrics for evaluating ranking methods in information retrieval as Recall [22] and NDCG [18] as they were considered in [2, 3, 14, 15]. Recall is the fraction of positive (related or relevant) items retrieved by a query (considering that an input document is a query in our scenarios).

The same author's papers used in topics and main idea sentences evaluation were considered in paper recommendation evaluation. We take strongly related and related as positive and not related as negative when calculating Recall. Indeed, the Recall generated is a relative Recall where we consider the sum all positive items retrieved by all queries as the total universe of positive items.

Discounted Cumulative Gain (**DCG**) is a measure of ranking quality [18].

DCG is calculated as

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

where p is the position where DCG is calculated and rel_k is the relevance value of the item in position k. We set the relevance value in the following way: items evaluated as *strongly* related get 2 points, as *related* 1 and *not related* 0. NDCG is the Normalized DCG. Normalized discounted cumulative gain **(NDCG)** measures the performance of a

recommendation system based on the graded relevance of the recommended entities and it is calculated as

$$NDCG_p = \frac{DCG_p}{IDCGP_p} \quad (7)$$

Where $IDCG_p$ is the DCG calculated over the ideal ranking in which the most relevant items are ranked on the top and p is the position where the DCG is computed.

4.2.3 Experiment Results: The 4 topics extraction methods were evaluated against each source to show how the generated queries can retrieve similar papers. The results are shown in Table 2 – 4. We considered only those queries marked as *strongly related* and r*elated* queries by our annotators. In both tables, we show which Recall obtained when queries generated by 4 different methods are submitted to each source and also to all sources.

They show the Recall results when extracted queries by both methods are submitted to each source and to all sources. Table 2 shows obtained Recall results when short queries are submitted as well as long queries. First all queries extracted from each method were evaluated against each source and then against all sources. Second, all extracted queries (long and short) were submitted to each source and then to all sources to verify if having many queries and many sources can improve the performance.

Considering all extracted short queries and all sources, the Recall was increased. The reason is that many different queries are submitted to a large database with many papers resulting in retrieving many relevant papers.

The results show that having many queries and many sources increases Recall. The best performance was due to the well queries formation methods considering phrases in paper's title and references.

Table 2. Recall Obtained for Both Short and Long Queries Generation

Fields	**Sources**			
	ACM	**IEEE**	**Science Direct**	**All sources**
Title and references	0.18	0.08	0.10	0.24
Body	0.22	0.12	0.14	0.30
Both fields for short queries	**0.29**	**0.16**	**0.18**	**0.61**
Abstract	0.28	0.16	0.22	0.31
Body (top and important sentences)	0.27	0.14	0.22	0.30
Both fields for long queries	**0.33**	**0.20**	**0.27**	**0.64**

The same for long queries extraction evaluation, Table 2 shows that the best performance in general was obtained on the ACM Digital Library (which usually returns more relevant documents), followed by Science Direct and IEEE Xplore. Using both fields (abstract and the body) for long queries extraction, it produces best Recall. Similarly, using all sources together, Recall values is increased due to the larger number of resources. There is a dramatic increase of Recall when all sources were used.

Using all sources together and both methods increases Recall values dramatically due to the larger number of resources and many queries. The comparison of our proposed approach with previous related method [2], results show that the proposed method outperforms it in 2 out of 3 sources (ACM and Science Direct) as shown in Table 3. We believe that the best performance was based on the best query extraction methods considering title, abstract, introduction and related works sections, using both short queries and long queries, sentences similarity to the title and importance of sentences.

Table 3. Recall Obtained when Both Methods are used Compared with Previous Approach

Approach	Sources			
	ACM	IEEE	Science Direct	All Sources
Proposed (Both short and long queries)	0.71	0.63	0.68	**0.71**
Nascimento, *et al.*, [2]	0.69	0.64	0.60	**0.69**

The evaluation of the proposed paper recommendation methods focuses on the analysis of how well the proposed recommendation method can order the retrieved papers is such way that the most relevant papers appear on the top. NDCG is used as an evaluation metric for each combination of the query generation and ranking strategies. Since users may just notice the top items, we concern mainly about whether the top ranked papers are relevant or not. Therefore, in this work, we use NDCG@N (N = 10) for evaluation where N is the number of top-N papers recommended by our proposed approaches.

Table 4. NDCG Obtained Results and Comparison with Previous Method

Approach	Sources			
	ACM	**IEEE**	**Science Direct**	**All sources**
Proposed	0.78	0.69	0.72	**0.80**
Cristiano et al. [2]	0.77	0.71	0.55	**0.78**

As we can see in Table 4, our recommendation method is able to put relevant documents on the top of the raking. The reason is that the considered fields contribute much to the paper main content as follows: the abstract summarizes the abstract summarize the paper, introduction gives more details of the problem to be solved and the solutions, thus producing best queries and lastly, the related works section as it points to other similar previous approaches makes it produce better queries that will retrieve those similar papers. The best performance was due to considering those important parts of the paper. In most of the cases we have an NDCG over 0.72, which is a good indication that the proposed ranking methods perform well. As it is shown in Table 4, NDCG increases when we have many sources or more documents, this because different papers are available in different sources and considering all sources increase the relevance. Comparing the proposed paper recommendation approach against our baseline [2], it is clear that there is an improvement of NCDG results when using our method as shown in Table 4. The best performance is due to the best queries selection methods that lead to paper ranking methods.

5. Conclusion

This paper proposes an effective academic papers recommendation approach without user profiles. The approach takes one single paper as input and then main topics and main idea sentences are extracted and submitted as queries to online paper repository databases to retrieve the similar papers. It is content-based filtering approach in which the content of both target and candidate papers are considered in selecting best papers to recommend. Four methods are proposed to generate those queries. The generated queries are submitted to online information repository to retrieve candidate papers from which we select best similar papers to recommend. Cosine similarity between specific selected fields of the paper is calculated to select related papers to recommend. Compared to other previous proposed methods solving similar problem, the results of experiments show the performance of the proposed method.

The best performance of our paper recommendation approach is based on best queries generation methods considering the full paper content as short or long queries and the best paper ranking methods considering full content of the candidate paper. We are planning to improve the propose method by considering indexing features.

References

[1] Y. Liang, Q. Li and T. Qian, "Finding Relevant Papers Based on Citation Relations", Springer-Verlag Berlin Heidelberg, **(2011)**, pp. 403–414.
[2] C. Nascimento, A. H. F. Laender, A. S. da Silva and M. A. Gonçalves, "A Source Independent Framework for Research Paper Recommendation". ACM, **(2011)** June 13–17, Ottawa, Ontario, Canada.
[3] Y. Jiang, A. Jia, Y. Feng and D. Zhao, "Recommending Academic Papers via Users' Reading Purposes", ACM Conference on Recommender Systems, **(2012)** September 9–13, Dublin, Ireland.
[4] B. Endres-Niggemeyer, K. Haseloh, J. M€uuller, S. Peist, I. S. Sigel, A. Sigel, E. Wansorra, J. Wheeler and B. Wollny, "Summarizing information", Berlin: Springer-Verlag, **(1998)**, pp. 307–338.
[5] K. W. Hong, H. Jeon and C. Jeon, "User Profile-Based Personalized Research Paper Recommendation System", 8th International Conference on Computing and Networking Technology (ICCNT), IEEE (**2012**), pp. 134-138.
[6] T. Griffiths and M. Steyvers, "Finding scientific topics. In Proceedings of the National Academy of Sciences", **(2004)**, pp. 5228–5235.
[7] Q. He, D. Kifer, J. Pei, P. Mitra and C. L. Gilee, "Citation recommendation without Author Supervision", In the Proceedings of the fourth ACM international conference on Web search and data mining, **(2011)** February 9–12, Hong Kong, China.
[8] K. Shubankar, A. Singh and V. Pudi," A Frequent Keyword-Set Based Algorithm for Topic Modeling and Clustering of Research Papers", In the proceeding of the 2011 third Conference on Data Mining and Optimization (DMO) , IEEE, **(2011)** June 28-29, pp. 96-102, Selangor, Malaysia.
[9] D. Hanyurwimfura, L. Bo, D. Njangi and J. P. Dukuzumuremyi, "A Centroid and Relationship based Clustering for Organizing Research Papers", International Journal of Multimedia and Ubiquitous Engineering, vol. 9, no. 3, **(2014)**, pp. 219-234.
[10] Y. Yang, N. Bansal and W. Dakka," Query by document", WSDM, **(2009)** February 9–12, Barcelona, Spain.
[11] K. Hong, H. Jeon and C. Jeo, "Personalized Research Paper Recommendation System using Keyword Extraction Based on User Profile", Journal of Convergence Information Technology, vol. 8, no. 16, (**2013)**, pp. 106-116.
[12] T. Bogers and A. van den Bosch, "Recommending Scientific Articles Using CiteULike", In Proceedings of the 2008 ACM conference on Recommender systems, (**2008**), pp. 287-290
[13] Y. Wang, J. Liu, X. L. Dong, T. Liu and Y. Huang, "Personalized Paper Recommendation Based on User Historical Behavior", NLPCC**,** CCIS, vol. 333, **(2012)**, pp. 1-12, Springer-Verlag Berlin Heidelberg.
[14] K. Sugiyama and M.-Y. Kan, "Scholarly Paper Recommendation via User's Recent Research Interests", In Proc. of the 10th ACM/IEEE Joint Conference on Digital Libraries , **(2010)**, pp. 29–38.
[15] K. Sugiyama and M.-Y. Kan, "Exploiting Potential Citation Papers in Scholarly Paper Recommendation", **(2013)** July 22–26, pp. 153-162, Indianapolis, Indiana, USA.
[16] C. Wang and D. M. Blei, "Collaborative Topic Modeling for Recommending Scientific Articles", In Proc. of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, **(2011)**, pp. 448–456.
[17] V. Qazvinian and D. R. Radev," Scientific Paper Summarization Using Citation Summary Networks", Proceedings of the 22nd International Conference on Computational Linguistics (Coling), **(2008)**, pp. 689–696.
[18] K. Jarvelin and J. Kekalainen, "Cumulated gain-based evaluation of IR techniques", ACM Transactions on Information Systems, vol. 20, no. 4, (**2002**), pp. 422–446.
[19] T. Strohman, B. Croft and D. Jensen, "Recommending citations for academic papers". In SIGIR, **(2007)**.
[20] Q. He, J. Pei, D. Kifer, P. Mitra and L. Giles, "Context-aware citation recommendation", In WWW, **(2010)**.
[21] K. J. Mock, "Hybrid hill-climbing and knowledge-based techniques for intelligent news filtering", In Proceedings of the national conference on artificial intelligence AAAI, **(1996).**
[22] R. A. Baeza-Yates and B. Ribeiro-Neto, "Modern Information Retrieval", Addison-Wesley Longman Publishing Co., Inc., **(1999)**, Boston, MA, USA.
[23] J. L. Herlocker, J. A. Konstan, A. Borchers and J. Riedl, "An algorithmic framework for performing collaborative filtering," in Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, **(1999)**, pp. 230–237.
[24] T. Griffiths and M. Steyvers, "Finding scientific topics. In Proceedings of the National Academy of Sciences", **(2004)**, pp. 5228–5235.

[25] D. M. Blei, A. Y. Ng and M. I. Jordan, "Latent Dirichlet Allocation", Journal of Machine Learning Research, vol. 3, **(2003)**, pp. 993–1022.
[26] P. Buitelaar and T. Eigner, "Topic Extraction from Scientific Literature for Competency Management", Proceedings of the 3rd Expert Finder Workshop on Personal Identification and Collaborations: Knowledge Mediation and Extraction PICKME, **(2008)** October 27, Karlsruhe, Germany.
[27] R. Agarwal, T. Imielinski and A. Swami, "Mining Association Rules between Sets of Items in Large Databases," Proceedings of the 1993 ACM SIGMOD Conference, **(1993).**
[28] S. P. Parambath, "Topic Extraction and Bundling of Related Scientific Articles", cs. IR., **(2008)**.

Authors

Damien Hanyurwimfura, He is currently a PhD candidate at the College of Computer Science and Electronic Engineering, Hunan University, China. He is also teaching at the College of Science and Technology, University of Rwanda, Rwanda. He received his Masters degree of Engineering in Computer Science and Technology from Hunan University in 2010. His current research interests include data mining and information security.

Liao Bo, He received the PhD degree in computational mathematics from the Dalian University of Technology, China, in 2004. He is currently a Professor at Hunan University. He was at the Graduate University of Chinese Academy of Sciences as a post doctorate from 2004 to 2006. His current research interests include bioinformatics, data mining and machine learning.

Vincent Havyarimana, He received his B.S. degree in Mathematics from University of Burundi, Bujumbura in 2007 and the M.E. degree in Computer Science and Technology from Hunan University, Changsha, China in 2011. Currently, he is a PhD candidate in the College of Computer Science and Electronic Engineering, Hunan University, China. He is also a lecturer at one of the Universities of Burundi,"Ecole Normale Superieure". His research interests include wireless communication and mobile computing.

Faustin Kagorora, He is currently a Master student at the College of Computer Science and Electronic Engineering, Hunan University, China. His research includes Database Application Testing and Web Security Assessment.

Dennis Njagi, He received his Bachelors of Education degree in Mathematics from Egerton University, Kenya in 2000 and a MSc. in Computer Applications Technology from Central South University (CSU), China in 2004. He is currently a lecturer at Jomo Kenyatta University of Agriculture and Technology (JKUAT), department of Information Technology and a doctorate student at CSU. His current research interests include text mining and data fusion.

