English-Malayalam Cross-Lingual Information Retrieval – an Experience

Nikesh P.L, Sumam Mary Idicula, David Peter S

Abstract— **This paper describes about an English-Malayalam Cross-Lingual Information Retrieval system. The system retrieves Malayalam documents in response to query given in English or Malayalam. Thus monolingual information retrieval is also supported in this system. Malayalam is one of the most prominent regional languages of Indian subcontinent. It is spoken by more than 37 million people and is the native language of Kerala state in India. Since we neither had any full-fledged online bilingual dictionary nor any parallel corpora to build the statistical lexicon, we used a bilingual dictionary developed in house for translation. Other language specific resources like Malayalam stemmer, Malayalam morphological root analyzer etc developed in house were used in this work.**

Index Terms— **Cross-Lingual Information Retrieval, Vector space model, Malayalam, Document ranking, Bilingual dictionary, Content based retrieval.**

I. Introduction

Information Retrieval (IR) systems aim to retrieve relevant documents to a user query, where the query is a set of keywords. Cross-Lingual Information Retrieval (CLIR) involves the retrieval of documents in a language other than the query language. Since the language of query and the documents to be retrieved are different, the queries need to be translated in CLIR. But this translation step causes a reduction in the retrieval performance of CLIR as compared to monolingual IR system. The main reasons for this reduced performance are missing specified vocabulary, missing general terms and wrong translation due to ambiguity [1].

The three main approaches of query translation include dictionary-based machine translation; parallel corpora based statistical lexicon and ontology-based methods [2]. The basic idea of machine translation is to replace each term in the query with an appropriate term or a set of terms from the lexicon. This approach was used in our experiment.

This paper presents one Cross-Lingual IR and one Monolingual IR. The cross-lingual task involves Malayalam document retrieval in response to queries in English and the monolingual task involves Malayalam document retrieval in response to Malayalam queries.

India is a multilingual country. Malayalam is one of the 22 official languages of it, spoken by around 37 million people. It belongs to the family of Dravidian Languages. It is ranked as the 29th most popular language in the world. Recently the volume of Malayalam electronic data has increased very much. All major new papers, Government departments, public sector organizations have started websites in Malayalam language. The population of Internet users in India has also drastically increased. CLIR helps to break the barrier of languages and helps to access information in different languages.

II. Related Works

In Indian languages CLIR is still in its primitive state. The first major work involving Hindi occurred during TIDES Surprise Language exercise [3]. The objective of this work was to retrieve Hindi documents in responds to English queries. Similar work has been reported for Bengali [2] and Tamil [14]. But nothing has been reported for Malayalam, even though it is a prominent regional language. CLIR works have also been reported for Chinese-English [6], Arabic-English [5] and European languages like German-English [12], French- English [13] etc. Some of the language specific obstacles of CLIR are proprietary encoding of text, lack of availability of corpora and variability in Unicode encoding [11].

III. Resources Used

For processing English queries, we had used UMass's (University of Massachusetts) stop word list and KSTEM stemmer developed by Robert Kroevetz [7]. Certain resources like an English-Malayalam bilingual dictionary, a morphological processor and a list of 225 stop words developed in house [15] were used for processing Malayalam queries and documents. The vector space model (VSM) [8], [9], [10] was used for document ranking and retrieval.

IV. System Architecture

System architecture is described in Fig. 1. It supports both cross-lingual (English-Malayalam) and monolingual (Malayalam) Information retrieval. The individual modules

Manuscript received February 11, 2008. This work was supported by Indian Space Research Organization under RESPOND Scheme.

Nikesh P.L is with Department of Computer Science, Cochin University of Science & Technology, Cochin, Kerala, India email: nikesh.pl@gmail.com

Sumam Mary Idicula, Reader, Department of Computer Science, Cochin University of Science & Technology, Cochin, Kerala, India (corresponding author: Phone 0484-2577126, email: sumam123@yahoo.com)

David Peter S, Reader, Department of Computer Science, Cochin University of Science & Technology, Cochin, Kerala, India email: davidpeter123@yahoo.com

978-1-4244-2030-8/08/$25.00 ©2008 IEEE.

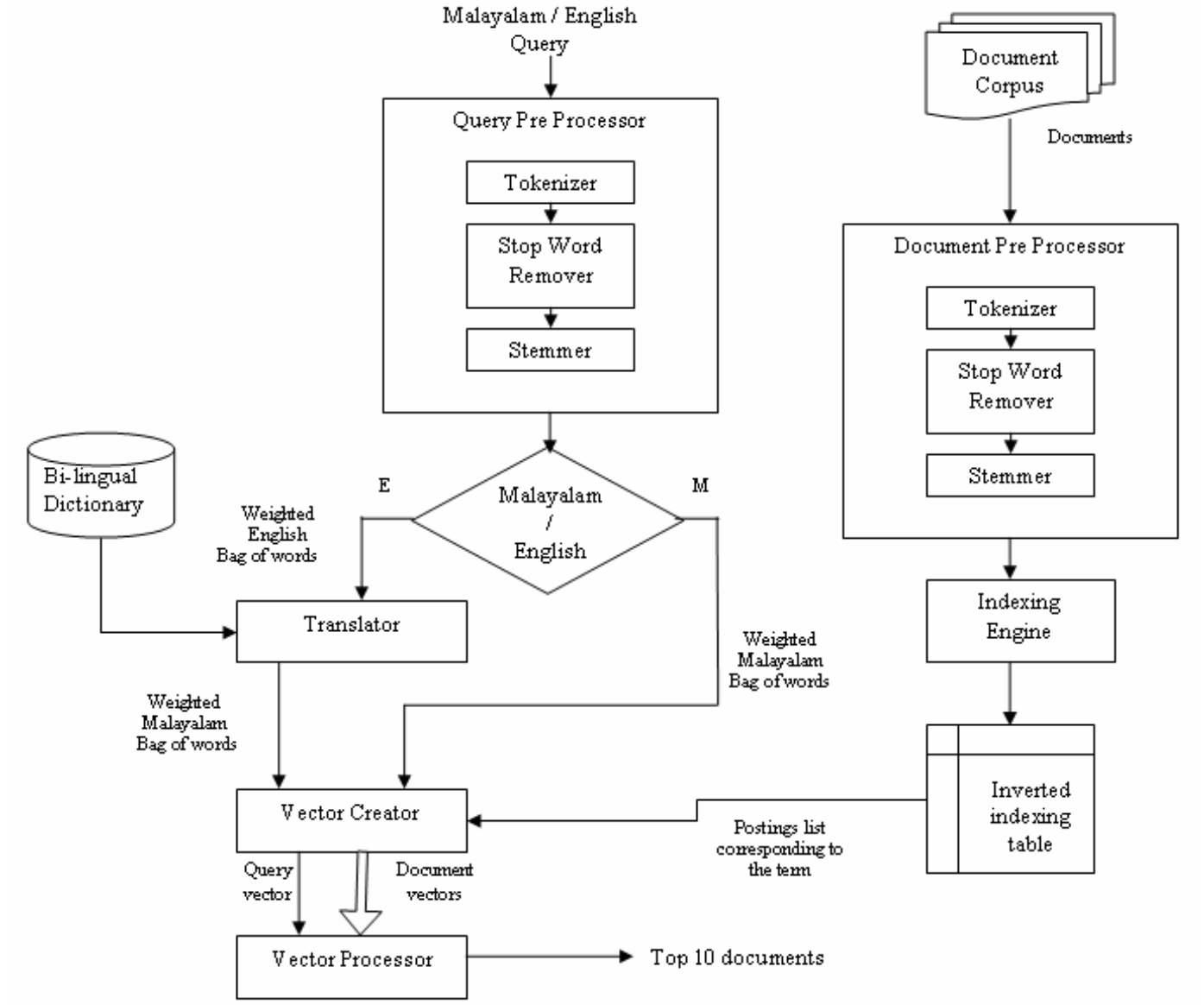

Fig. 1. System Architecture of English-Malayalam Cross-Lingual Information Retrieval System.

are explained in the subsequent sections.

A. Query Pre processor

It accepts a query either in English or Malayalam. This query is passed through processes like tokenizing, stop word removal and stemming. The output is a bag of weighted query words. Proper nouns and nouns weigh the highest. If the input is in English then in addition to the above processes, it is passed through a translation process also. For that a bilingual English-Malayalam dictionary is used. This process converts the weighted query terms into corresponding Malayalam words. The StringTokenizer class of java.util was used for tokenizing. High frequency words or function words such as articles, prepositions, and conjunctions are excluded from the tokenized text. Stop words are language dependent and we have used UMass's stop word list for English and a stop word list developed in house consisting of 225 words for Malayalam.

The stemming process reduces tokens to their corresponding root form. The stemming algorithm used in our test is Robert Krovetz's KSTEM. It uses a list of words and a set of rules for handling inflectional and derivational morphology. Linguistic resources developed in house were used for Malayalam.

B. Document Preprocessor

The documents from the document corpora are subjected to processes like tokenizing, stop word removal and stemming. This helps in reducing the number of words to be stored. These words are indexed and stored in a hash table called the inverted index file. This file contains an index of term and a postings list for each term. The postings list contains the

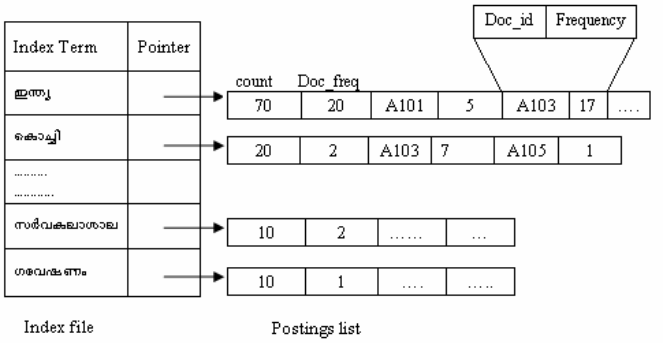

Fig. 2. Structure of Inverted Index file and postings list

documents id and the frequency of occurrence of the term in the document for all documents containing the term. The Count Field in the postings list contains the total number of times that term appear in all the documents of the collection and the Document Frequency is the number of documents containing that term. The structure of inverted index file and postings list is shown in Fig. 2.

C. *Vector Creator and Processor*

Vector Space Model was used for document ranking and retrieval. Each document is represented as a set of terms. These terms are the words remaining after operations like stop word removal and stemming. A union of all these set of terms, each set representing a document, forms the "document Space" of the Corpus. Each distinct term in the union set, represents one dimension in the document space.

Proper term weighting can greatly improve the performance of the vector space method. We can assign a numeric weight to each term in a given document, representing an estimate of the usefulness of the given term as a descriptor of the given document. A given term may receive different weights in each document it occurs; a term may be a better descriptor of one document than of another. A term that is not in a given document receives a weight of zero in that document. The weights assigned to the terms in a given document *D1* can then be interpreted as the coordinates of *D1* in the Document Space. A weighting scheme is composed of three different types of term weighting: local, global, and normalization. The term weight is given by $L_{i,j}G_iN_j$, *where* $L_{i,j}$ is the local weight for term i in document j, G_i is the global weight for term i, and N_j is the normalization factor for document j. Local weights are functions of how many times each term appears in a document, global weights are functions of how many times each term appears in the entire collection, and the normalization factor compensates for discrepancies in the lengths of the documents.

Local Term Weighting is a document-specific measure. It varies from one document to another. This formula depends only on the frequencies within the document and they do not depend on inter-document frequencies [16]. We are using a term weighting from the question analysis component for the query terms and a modified form of augmented normalized term frequency for the terms in the document.

The equation used for calculating the term weighting is given below [17].

$$tf_{i,j} = C * \chi(k_i, A_d) + (1-C) * \frac{\log(freq(k_i, A_d))}{\log(\max\{freq(k_1, A_d)..., freq(k_t, A_d)\})} \quad (1)$$

Where $\chi(k_i, A_d)$ is equal to one if that term is present and is equal to zero if not present, C is a constant set as 0.4, $freq(k_i, A_d)$ is the term frequency in the document, and $\max\{freq(k_1, A_d)....., freq(k_t, A_d)\}$ is the maximum term frequency in any document.

Global Term Weighting tries to give a "discrimination value" to each term. The use of global weighting can, in theory, eliminate the need for stop word removal since stop words should have very small global weights. In practice, however, it is easier to remove the stop words in the preprocessing phase so that there are fewer terms to handle. If every document in the collection contains the given term, the Inverse Document Frequency *(idf*) *is* zero. Many schemes are based on the idea that the less frequently a term appears in the whole collection, the more discriminating it is. We have used a variation of *idf* called Probabilistic Inverse Document Frequency *(pidf)* for calculating global term weight.

It assigns weights ranging from -∞ for a term that appears in every document to log (n-1) for a term that appears in only one document. In implementation we change -∞ to zero, ie that particular term is treated as a stop word. It differs from *idf* because probabilistic inverse actually awards zero weight (in practice) for terms appearing in more than half of the documents in the collection.

$$pidf_i = \log \frac{N - df_t}{df_t} \quad (2)$$

The third component of the weighting scheme is the normalization factor, which is used to correct discrepancies in document lengths. It is useful to normalize the document vectors so that documents are retrieved independent of their lengths. If it is not done, short documents may not be recognized as relevant. Two main reasons that necessitate the use of normalization in term weights are given next.

Higher term frequencies: long documents usually use the same terms repeatedly. As a result, the term frequency factors may be large for long documents.

Number of terms: long documents also have different numerous terms. This increases the number of matches between a query and a long document, increasing the chances of retrieval of long documents in preference over shorter documents.

Cosine Normalization is the most commonly used and popular normalization technique. It resolves both the reasons for normalization (Higher term frequencies, number of terms) in one step. With an inverted file, the number of postings lists accessed equals the number of query terms. The computational cost is acceptable for queries of reasonable size. Unfortunately, when we use cosine normalization the computation of the normalization factor is extremely expensive because the term $\sqrt{\Sigma_{j=1}^{m}(w_{i,j})^2}$ in the normalization factor requires access to all participating document's terms,

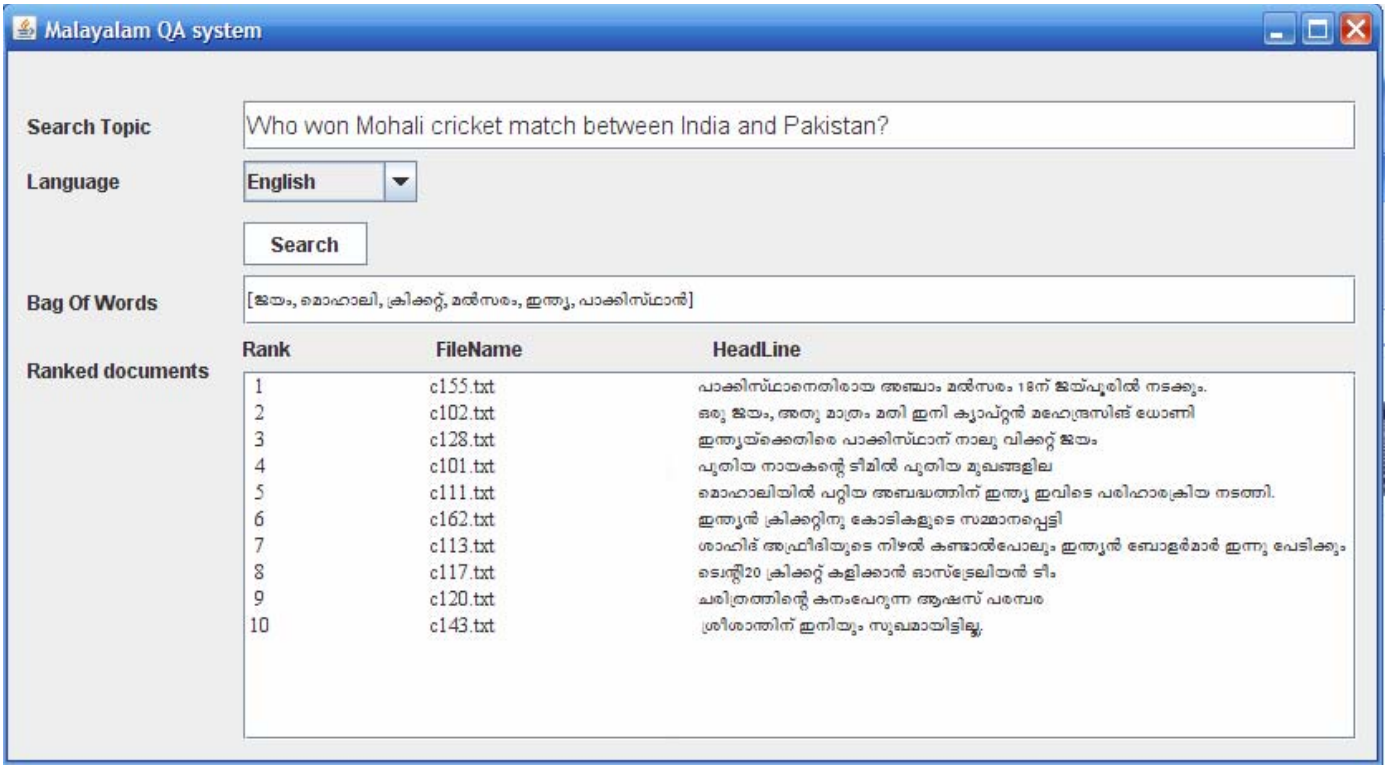

Fig. 3. Screen shot of English-Malayalam CLIR system.

not just the terms specified in the query.

To approximate the effect of normalization the square root of the number of terms in a document was used as the normalization factor. This normalization factor is much easier to compute than the original one; also, pre computation is possible. At the time of document processing, a log file that contains the document id and the corresponding square root of the number of terms in the document is prepared. This is used at the time of normalization. With this approximation, the formula for similarity betwenn document i and query Q is

$$Sim(Q, D_i) = \frac{\sum_{j=1}^{T}(q_j * w_{j,i})}{\sqrt{number\ of\ terms\ in D_i}} \qquad (3)$$

Where, numerator is the dot product of the query and document i and denominator is the normalization factor used as document length. T is the total no: of keywords in the query Q.

V. User Interface

Netbeans 6 with JDK 1.6 was used for developing the user interface. The user can run the system either monolingual or cross-lingual by selecting the language of query either English or Malayalam. When the search button is pressed, the list of keywords in the query will be displayed first which will be followed by the ranked documents containing the information. The user interface of the CLIR system displaying answer is shown in Fig. 3.

VI. Evaluation Results

We ran the system for 25 questions and evaluated the result. Table 1 shows some examples of query given to the system.

TABLE I
EXAMPLE ENGLISH QUERIES AND RANK OF THE MOST RELEVANT DOCUMENT RETRIEVED

1	When did Harry Potter come to India?
1	About Indo-US nuclear deal?
1	How may SEZ in India?
1	At the end of this year what will be the investment in SEZ?
9	Reduction of interest rate in America and world market
5	Who is the Indian test cricket captain?
1	Chickun Guniya treatment
1	About Chethi bridge construction
5	What is Supreme Court's new decision on marriage registration?
2	Who is the director of cricket academy?
9	Who are the famous players of ICL?
3	Who won Mohali cricket match between India and Pakistan?
2	How many centuries have Gilchrist scored in test cricket?
:-	Who is the Indian president?
1	How old is Sachin?
4	Who won under 20 football world cup?
:-	What is the government decision against strike of doctors?
1	What is scientists' opinion about population?
1	Kongani Language
2	Different types of Yogas?
7	List Ambika's poems
6	India Pakistan War.

:- not in top 10 documents

Each question is preceded by the rank of the most relevant document retrieved. A":-" indicates that the most relevant

document was not in the top 10 documents returned by the system for that query. In Table 2 the number of queries and the rank of the most relevant document corresponding to them

TABLE 2
QUERIES AND RANK OF THE MOST RELEVANT DOCUMENT RETRIEVED

Rank	Malayalam	English
1	12	9
2	2	3
3	0	1
4	2	1
5	1	2
6	2	1
7	0	1
8	0	0
9	2	2
10	0	0
total	21	20

are shown. It can be seen from the table that for both monolingual and cross-lingual IR, the probability of getting the most relevant document is almost the same. This shows the efficiency of the translation mechanism used. We have used an English-Malayalam dictionary developed in house for that.

VII. CONCLUSION

We have developed an English Malayalam CLIR system in three months. We believe that we showed that cross-lingual IR for English-Malayalam is viable and that a basic system can be constructed quickly once the linguistic tools become available.

.

REFERENCES

[1] Anna R. Diekema., "Translation Events in Cross-Language Information Retrieval," In ACM SIGIR Forum, vol.3,No. 1, June 2004.

[2] Debasis Mandal, Sandipan Dandapat, Mayank Gupta, Pratyush Banerje and Sudeshna Sakar, "Bengali and Hindi to English Cross-language Text Retrieval under Limited Resources," CLEF 2007. Available at http://www.clef-campaign.org/2007/working_notes/ mandalCLEF2007.pdf

[3] Leah S. Larkey, Margaret E. Connell and Nasreen Abduljaleel, "Hindi CLIR in thirty days," ACM Transactions on Asian Language Information Processing (TALIP), vol. 2, Issue 2. June 2003, pp. 130-142.

[4] Ballesteros, L. and Croft, B., "Dictionary Methods for Cross-Lingual Information Retrieval," In the Proceeding of the 7th International DEXA Conference on Database and Expert Systems Applications, pp. 791-801. 1996.

[5] Mohammed Aljlayl, Ophir Frieder and David Grossman, "On Arabic-English Cross-Language Information Retrieval: A Machine Translation Approach," Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'02) 8-10 April 2002 pp. 2- 7.

[6] Feng Yu, Dequan Zheng, Tiejun Zhao, Sheng Li and Hao Yu, "Chinese-English Cross-Lingual Information Retrieval based on Domain Ontology Knowledge," Int. Conf. on Computational Intelligence and Security, 2006 vol. 2, 3-6 Nov. 2006, pp. 1460 - 1463

[7] Robert Krovets, "Viewing morphology as an inference process," In Proceedings of the 16th Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval., pp. 191-202, 1993.

[8] G. Salton, A. Wong and C.S. Yang "A Vector space Model for Automatic indexing", Communication of the ACM, vol. 18 , Issue 11, November 1975, pp. 613 – 620.

[9] Jorg Becker "Topic based VSM", Business information systems, proceedings of BIS 2003, Colorado Springs, USA.

[10] Dik L. Lee, Huei Chuang and Kent Seamons, "Document Ranking and the Vector-Space Model", IEEE Software3, vol. 14, Issue 2, March 1997, pp. 67 – 75.

[11] Seetha, Anurag Das, Sujoy and Kumar, M "Evaluation of the English-Hindi Cross Language Information Retrieval System Based on Dictionary Based Query Translation Method", 10th Int. Conf. on Information Technology, (ICIT 2007). 17-20 Dec. 2007, pp. 56-61.

[12] D. Neumann.,"A cross-language question answering system for German and English," CLEF –2003.

[13] A. O'Gorman, I. Gabby and Sutcliffe," French in the cross-language task," CLEF-2003.

[14] Prasad Pingali, Jagadeesh Jagarlamudi and Vasudeva Varma, "Webkhoj: Indian language IR from Multiple Character Encodings," In Int. World Wide Web Conf., May 23-26, 2006.

[15] Sumam Mary Idicula and David peter, "A morphological processor for Malayalam Language," Journal of south Asian research vol. 27(2), pp. 173-186.

[16] Nicola Polettini, "The vector Space Model in Information Retrieval-Term Weighting Problem," Department of information and communication Technology, University of Trento, Italy, 2004.

[17] W. B. Croft and Raj Das, "Experiments with representation in a document retrieval system," Proceedings of the 13th annual int. ACM SIGIR conf. on Research and development in information retrieval, Brussels, Belgium, 1989, pp. 349 - 368.