International Journal of Scientific Research in Computer Science and Engineering
Vol.4, Issue.1, pp.1-5, February (2016)

Review Paper

E-ISSN:2320-7639

Precision Improvement in Information Storage and Retrieval System by Document Length Normalization

D. Sharma[1*], H. Nagar[2]

[1*,2] Department of Computer Science and Engineering, Mewar University, Chittorgarh, India
[1*,2] Department of Computer Science and Engineering, Mewar University, Chittorgarh, India

Corresponding Author: dharmu411180@gmail.com

Available online at www.isroset.org

Received: Jan/12/2016, Revised: Jan/24/2016, Accepted: Feb/15/2016, Published: Feb/28/2016

Abstract— Huge amount of information are available over the internet in electronic document format but retrieving the correct document according to users information need is very critical task. The relevancy of the document can vary according to the length of document. Automatic information storage and retrieval system have to deal with documents of varying length in text collection. In this paper we are presenting a document term weighting scheme based on length of document. Our method increases the rank of relevant document in the retrieved ordered document set. From the result we have seen that our method increase the document rank from 0.83 precision to 0.16 precision.

Keywords- Document length, Normalization, Rank, Storage system, Precision, Information Storage, Retrieval System, Normalization

I. INTRODUCTION

With the growth of the World Wide Web (WWW), it has become the most popular place to retrieve the information. However the size of the WWW makes it difficult for people to retrieve relevant documents [1][2]. About 85% of all Web users use information retrieval system to retrieve the information from the web [3][4]. However, existing information retrieval system do not return relevant document. Many Web users have been dissatisfied with using information retrieval system. The main reasons for dissatisfaction are the inability to retrieve the relevant document. Many IR systems are based on a vector space model, where documents and queries are represented as vectors of terms and their similarity scores are computed to use an inner product. Today, much research has been carried out on similarity measures and weighting schemes, and on variations of their implementation to enhance retrieval performance. Most of the similarity measures and weighting schemes based on the inner product and the cosine measures. Term weighting is a crucial part of any information retrieval system. The purpose of a term weighting scheme is to classify the indexing terms by assigning them weights corresponding to how well they are in improving the precision of the retrieval. For those information retrieval system based on VSM, every component of the document vector and query vector are all weight values computed from" certain formula.

This paper provides an improved term weighting scheme based on the detail analysis to the nature of vector space model. In our method, we take into account the following factors: document length, term frequency. The experimental results show the success of the weighting scheme in terms of precision.

The remaining sections of this paper consist of four different parts. In related work section we focus on the work done in the performance improvement of the information storage and retrieval system. The methodology section describes the process of implementing the system. The result and discussion section focus our findings and significant. In conclusion we conclude our work with limitation and future work.

II. RELATED WORK

The relevancy measures between queries and information resources relevant to users' information needs have been studied for a long time. The most popular and basic method is vector space model [1]. Feature reduction techniques of vector space model have been used for developing traditional vector space models such as latent semantic indexing [2] and the mathematical model of meaning [3][4]. These techniques are applied to information resources, characterized by elements in a flat domain. However, it is to be noted that when the elements have a tree structure, all the elements are not orthogonal to each other. A few studies have used computational measures of feature relationships [5] in an orthogonal vector space.

The mathematical model of meaning realizes a context-based dynamic semantic computation. However, it has to prepare a space for the semantic commutations before There have been studies defining similarity metrics for hierarchical structures such as WorldNet [6]. Rada et al. [7] have proposed a "conceptual distance" that indicates the similarity between concepts of semantic nets by using path lengths. Some studies [8][9] have extended and used the conceptual distance for information retrieval. Resnik [10] has proposed an alternative similarity measure based on the concept of information content. Ganesan et al. [11] have presented new similarity measures in order to produce more intuitive similarity scores based on traditional measures. On the other viewpoints, the reference [12] is surveyed. This survey [12] shows common architecture and general functionality as OBIE from various ontology-based information extraction researches. It consists of "information extraction module", "ontology generator", "ontology editor", "semantic lexicon" and some preprocessors. Their researchers are working for both of various researches of OBIE system implementation and research focusing on each module. In this paper, we will mainly introduce research of OBIE system implementation. The OBIE system implemented by Saggion et al. [13] operates the application of information extraction for a practical e-business. The system implemented by Wu et.al. [14] Is one of the OBIE because of using structure of "inforboxes" of Wikipedia as ontology? The Cimiano et al. system [15] is pattern-based approach to categorize instances with regard to ontology. OntoSyphon [16] uses the ontology to specify web searches that identify possible semantic instances, relations, and taxonomic information. The Maedche et al. system [17] is bootstrapping approach that allows for the fast creation of an ontology-based information extracting system. The TEXT-TO- ONTO Ontology Learning Environment [18] is based on a general architecture for discovering conceptual structures and engineering ontologies from text. SOBA [19] is a component for ontology-based information extraction from soccer Web pages. Embley [20] proffers the use of information-extraction ontologies as an approach that may lead to semantic understanding. Li et al. [21] proposed a hierarchical learning approach for information extraction. Hwang [22] approach is on dynamic ontologies that are automatically constructed from text data. OntoX [23] detect out-of-date constructs in the ontology to suggest changes to the user. Vulcain [24] identifies domain-specific terms and concepts, using syntactic information and an existing domain ontology. Vargas-Vera et al. [25] proposed Semantic Annotation Tool for extraction of knowledge structures from web pages through the use of simple user-defined knowledge extraction patterns. KIM [26] provides a mature and semantically enabled infrastructure for scalable and customizable information extraction. IDocument [27]. These conceptual dictionary approaches and ontology approaches assume that a semantic is decided statically. In our method, we assume that a semantic is relatively decided by a context dynamically. Our method is the one of the weighting methods. Until now, there are many researches about weighting method. The reference [28] describes survey about the weighting methods such as binary [29], term frequency (TF) [29], augmented normalized term frequency [29][30], log [30], inverse document frequency (IDF) [29], probabilistic inverse [29][30], document length normalization [29].

However, our method differs in the purpose from other methods clearly. The purpose of current method of weighting methods is to weight the appropriate feature and to raise the precision of search results.

III. METHODOLOGY

In this section, we present the method of calculating weight of term on the basis of length of documents. For the input documents set we use Google search engine to aggregate the documents set. Our method consists of four different steps

A. *Data Filtration*

Before creating the document term index we filter out the text documents set by eliminating the stop words and other meaningless terms. We use following steps to filter the data

1) Convert text into lower case.
2) Remove stop word, number and punctuation character from text.
3) Calculate the term frequency for each documents.

B. Document Term Index creation:

For effectively retrieving relevant documents by information retrieval strategies, the documents are typically transformed into a suitable representation. Each retrieval strategy incorporates a specific model for its document representation. We use vector space model to create the document term index. In vector space model Documents and queries are represented as vectors.

$D_j = (w_{1j}, w_{2j}, \ldots . w_{tj})$
$Q = (w_{1q}, w_{2q}, \ldots . w_{nq})$

Each dimension corresponds to a separate term. If a term occurs in the document, its value in the vector is non-zero.

C. Weight calculation:

We use following steps to assign weight to the term.

1. Calculate the term frequency for a single document. If the term T appear in document D, x times then its term frequency will be x.
2. Normalize the term frequency by dividing the length of document. If the term frequency of term T is x and the length of document is y then normalized term frequency NTF will be NTF = x/y.
3. Calculate the inverse document frequency of each term. If the total number of documents in document set is S and p is the number of documents in which term T appear then inverse document frequency of term T will be $T_{idf} = S/p$
4. Assign the weight of term T as normalized term frequency (NTF) multiplied by inverse document frequency of (T_{idf}) of term T.

D. Query Evaluation:

To calculate the relevancy between query and document any type methods are acceptable such as dot product, distance, norm, cosine similarity, etc. In this paper, we use dot product. The dot product of two vectors A and B are defined as

$$A.B = \sum A_iB_i = A_1B_1+A_2B_2+\ldots\ldots\ldots+A_nB_n$$

Where $1 \leq i \leq n$

The precision of the document is evaluated as rank of the document in the retrieved ordered document set divide by total number of relevant document retrieved.

IV. Experimental System

We use Google Web API for aggregation of web documents. We aggregate 20 documents of different length to test our system.

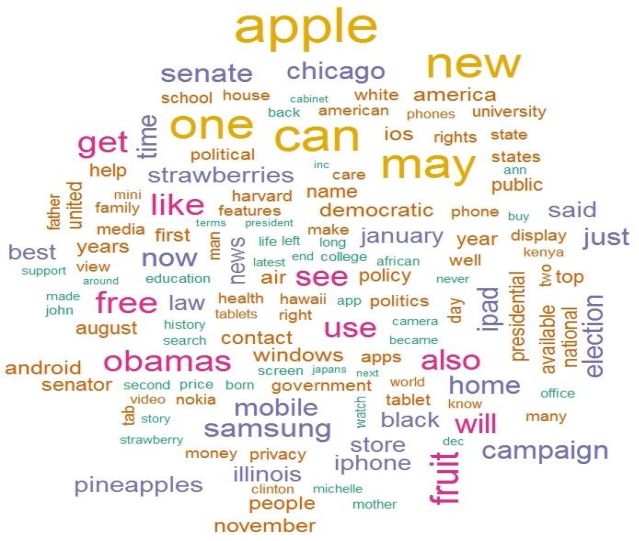

Figure 1: Lexeme Cloud

This experiment system is implemented by R including text mining (tm) package. Figure 1 shows the lexeme cloud of used documents set. The table I show the set of documents used in the experiment with the length of document. We have used eight documents related to apple fruit, seven documents related to electronic device phone and five documents related to computer system.

TABLE I: Input Documents set

Serial no.	*Document Id*	*Document Length*
1	apple1.txt	112
2	apple2.txt	135
3	apple3.txt	175
4	apple4.txt	224
5	apple5.txt	826
6	apple6.txt	475
7	apple7.txt	372
8	apple8.txt	215
9	phone1.txt	736
10	phone2.txt	484
11	phone3.txt	221
12	phone4.txt	451
13	phone5.txt	148
14	phone6.txt	350
15	phone7.txt	478
16	computer1.txt	275
17	computer2.txt	173
18	computer3.txt	551
19	computer4.txt	436
20	computer5.txt	581

V. Experimental System

In our document set the document aplle4.txt is the most relevant document set according to the query apple. For the comparative study of our result we compare our result with the term frequency weighting scheme. The table II shows the result of ordered retrieved document set for query apple and term frequency weighting scheme.

TABLE II: Query=Apple and Weighting = Term frequency

Rank	*Document ID*	*Precision*
1	apple5.txt	0.166667
2	apple6.txt	0.333333
3	apple7.txt	0.5
4	phone1.txt	0.666667
5	apple4.txt	0.833333
6	phone7.txt	1

From the result it is clear that total six documents are retrieved for the query apple. The document apple4.txt appear on fifth position thus its precision is 0.83. The table III shows the result of ordered retrieved document set for query apple and normalized document length weighting scheme.

TABLE III: Query=Apple and Weighting = Normalized document length

Rank	Document ID	Precision
1	apple4	0.166667
2	apple7	0.333333
3	phone1	0.5
4	apple6	0.666667
5	phone7	0.833333
6	apple5	1

From the table III it is clear that document apple4.txt appear on first position thus its precision is 0.16. From the table II and III we can conclude that the normalized term weighting scheme improve the rank of relevant document from 0.83 precision to 0.16.

VI. CONCLUSION

This paper provides an improved term weighting scheme based on the detail analysis to the nature of vector space model. In our method, we take into account the following factors: document length, term frequency. The relevancy of the document can vary according to the length of document. Automatic information storage and retrieval system have to deal with documents of varying length in text collection. In this paper we are presenting a document term weighting scheme based on length of document. We have normalized the document term weight by length of document i.e. term frequency divide by number of tern exist in document. Our method increases the rank of relevant document in the retrieved ordered document set. From the result we have seen that our method increase the document rank from 0.83 precision to 0.16 precision**.** The experimental results show the success of the weighting scheme in terms of precision.

REFERENCES

[1] O. King, M. Kobayashi, "*Information Retrieval and Ranking on the Web: Benchmarking studies II*", IBM TRL Research Report :RT0298, Japan pp.1-38,1999.

[2] S. Michel, K. Nguyen, A. Rosenstein, L. Zhang, S. Floyd, V. Jacobson, "*Adaptive web caching: towards a new global caching architecture*", Computer Networks and ISDN systems, Vol.30, Issue.22, pp.2169-2177, 1998.

[3] G.E. Dupret, M. Kobayashi, "*Information Retrieval and Ranking on the Web: Benchmarking studies I*," *IBM TRL* Research Report, Japan, pp.1-138, 1999.

[4] M. Kobayashi, K. Takeda, "I*nformation Retrieval on the Web*", *IBM Research*, Japan, pp.1-64, 2000.

[5] G. Salton, A. Wong, C. S. Yang, "*A vector space model for automatic indexing*", Magazine Communications of the ACM CACM Homepage archive, Vol.18, Issue.11, pp.613-620, 1975.

[6] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, R. Harshman, "*Indexing by latent semantic analysis*", Journal of the American Society for Information Science, Vol.41, Issue.6, pp.391-407, 1990.

[7] T. Kitagawa, Y. Kiyoki, "*A mathematical model of meaning and its application to multidatabase systems*", In RIDE-IMS '93: Proceedings of the 3rd International Workshop on Research Issues in Data Engineering: Interoperability in Multidatabase Systems, Austria, pp.130-135, 1993.

[8] Y. Kiyoki, T. Kitagawa, T. Hayama, "*A metadatabase system for semantic image search by a mathematical model of meaning*", SIGMOD Record, Vol.23, Issue.4, pp.34-41, 1994.

[9] K. Takano, Y. Kiyoki, "*A superordinate and subordinate relationship computation method and its application to aerospace engineering information*", In ACST'07: Proceedings of the third conference on IASTED International Conference, Anaheim, CA, pp.510-516, 2007.

[10] G.A. Miller, R. Beckwith, C. Fellbaum, D. Gross, K.J. Miller. "*Introduction to LexemeNet: An on-line lexical database*", Journal of Lexicography, Vol.3, Issue.4, pp.235-244, 1990.

[11] R. Rada, H. Mili, E. Bicknell, M. Blettner, "*Development and application of a metric on semantic nets*", IEEE Transactions on Systems, Man and Cybernetics, Vol.19, Issue.1, pp.17-30, 1989.

[12] Y. Kim, J. Kim, "*A model of knowledge based information retrieval with hierarchical concept graph*", Journal of Documentation, Vol.46, Issue.2, pp.113-136, 1990.

[13] Y. Li, K. Bontcheva, "*Hierarchical, perceptron-like learning for ontology-based information extraction*", In Proceedings of the 16th international conference on World Wide Web (WWW '07), NY, pp.777-786, 2007.

[14] C. Hwang, "*Incompletely and imprecisely speaking: Using dynamic ontologies for representing and retrieving information*", In Proceedings of the 6th international workshop on ontology-based information extraction system, Germany, pp.14-20, 1999.

[15] B. Yildiz, S. Miksch "*ontoX - A Method for Ontology-Driven Information Extraction*", Lecture Notes in Computer Science. Vol.4707, pp. 660-673, 2007.

[16] A. Todirascu, L. Romary, D. Bekhouche, "*Vulcain — An Ontology- Based Information Extraction System*", Lecture Notes in Computer Science, Vol.2553, pp.64-75, 2002.

[17] M. Vargas-Vera, E. Motta, J. Domingu, S. Shum, M. Lanzoni, "*Knowledge extraction by using an ontology-based annotation tool*", In Proceedings of the workshop on knowledge markup and semantic annotation, NY, pp.1-8, 2001.

[18] B. Popov, A. Kiryakov, D. Ognyanoff, D. Monov, A. Kirilov, "*KIM – a semantic platform for information extraction and retrieval",* Natural Language Engineering, Vol.10, Issue.3, pp. 375-392, 2004.

[19] B. Adrian, J. Hees, L. Elst, A. Dengel, "*iDocument: Using Ontologies for Extracting and Annotating Information from Unstructured Text*", Lecture Notes in Computer Science, VOl.5803, pp.249-256, 2009.

[20] T.G. Kolda, D.P. O'Leary, "*A semidiscrete matrix decomposition for latent semantic indexing information retrieval",* Journal ACM Transactions on Information Systems (TOIS) TOIS Homepage archive, Vol.16, Issue.4, pp. 322-346, 1998.

[21] G.Salton, C. Buckley, "Lexeme weighting approaches in automatic text retrieval," Journal Information Processing and Management, Vol.24, Issue.5, pp. 513–523, 1988.

[22] D. Harman, *"Ranking algorithms*In Information Retrieval: Data Structures and Algorithms," Prentice Hall, Englewood Cliffs, pp.363–392, 1992.
[23] B. Yildiz, S. Miksch "*ontoX - A Method for Ontology-Driven Information Extraction*", Lecture Notes in Computer Science, Vol.4707, pp.660-673, 2007.
[24] A. Todirascu, L. Romary, D. Bekhouche, "*Vulcain — An Ontology- Based Information Extraction System*," Lecture Notes in Computer Science, Vol. 2553, pp.64-75, 2002.
[25] M. Vargas-Vera, E. Motta, J. Domingu, S. Shum, M. Lanzoni, "*Knowledge extraction by using an ontology-based annotation tool",* In Proceedings of the workshop on knowledge markup and semantic annotation, NY, pp.1-13, 2001.
[26] B. Popov, A. Kiryakov, D. Ognyanoff, D. Monov, A. Kirilov, "*KIM – a semantic platform for information extraction and retrieval",* Natural Language Engineering, Vol.10, Issue3, pp. 375-392,2004.
[27] B. Adrian, J. Hees, L. Elst, A. Dengel, "*iDocument: Using Ontologies for Extracting and Annotating Information from Unstructured Text*", Lecture Notes in Computer Science, Vol.5803, pp.249-256, 2009.
[28] T.G. Kolda, D.P. O'Leary, "*A semidiscrete matrix decomposition for latent semantic indexing information retrieval",* Journal ACM Transactions on Information Systems, Vol.16, Issue.4, pp.322-346, 1998.
[29] G. Salton, C. Buckley, *"Term weighting approaches in automatic text retrieval",* Journal Information Processing and Management, Vol.24, Issue.5, pp.513–523, 1988.
[30] D. Harman, "*Ranking algorithms. In Information Retrieval: Data Structures and Algorithms*", Prentice Hall, Englewood Cliffs, pp.363–392, 1992.

Authors Profile

Mr. Dharmendra Sharma studying as research scholar in computer science engineering department of Mewar University, Cittorgarh, Rajasthan, India.His reaserch area is Information Storage and Retrieval System.

Dr Harish Nagar working as research director Mewar University. He has completed his Ph, D. Program form kota university, kola India, His subject of interest are Fractional Calculus, Integral Transform, Special Functions, Applied Mathematics.