Latent Semantic Indexing Model for Boolean Query Formulation

DaeHo Baek **HeuiSeok Lim†** **HaeChang Rim**
Natural Language Processing Lab.
Dept. of Computer Science and Engineering, Korea University
1, 5-ga, Anam-dong, Sungbuk-gu, SEOUL, 136-701, KOREA
†Dept. of Information Communications, Chonan University
{daeho, rim}@nlp.korea.ac.kr, †limhs@infocom.chonan.ac.kr

Abstract

A new model named Boolean Latent Semantic Indexing model based on the Singular Value Decomposition and Boolean query formulation is introduced. While the Singular Value Decomposition alleviates the problems of lexical matching in the traditional information retrieval model, Boolean query formulation can help users to make precise representation of their information search needs. Retrieval experiments on a number of test collections seem to show that the proposed model achieves substantial performance gains over the Latent Semantic Indexing model.

1 Introduction

Most information retrieval methods depend on exact matches between words in users' queries and words in documents. Typically, documents containing one or more query words are returned to the user. However, lexical matching methods can be inaccurate when they are used to match a user's query. Since there are many ways to express a given concept (synonymy), the literal terms in a user's query can not match those of a relevant document. In addition, most words have multiple meanings (polysemy), so terms in a user's query may literally match terms in irrelevant documents[5]. The Latent Semantic Indexing (LSI) tries to overcome the problems of lexical matching by using statistically derived conceptual indices instead of individual words for retrieval. The LSI assumes that there are some underlying or latent semantic structures in word usage that is partially obscured by variability in word choice[5].

Nowadays, most of the commercial information retrieval systems use the extended Boolean retrieval model because trained users can make precise representation of their information search needs using structured Boolean operators[3]. Previous research also supports the argument by showing that the extended Boolean models usually outperform the vector models in information retrieval[2]. Since it is difficult for untrained users to generate an effective Boolean search request, several methods are introduced which reduce the role of the search intermediaries by making it possible to generate Boolean search formulations automatically from natural language statements provided by the system patrons[1, 3, 4]. The queries generated by these automatic methods exhibited similar performance to manually constructed Boolean queries by experts.

Unfortunately, both in the vector model and the LSI model, it is not possible to distinguish query phrases using *and* connectives from *or* connectives. In this paper, we propose a new information retrieval model named Boolean LSI model which makes the LSI model possible to process Boolean query formulations.

2 LSI Model

The main idea of the LSI model is to map each document and each query vector into a lower dimensional space associated with concepts. The specific form of this mapping is based on the Singular Value Decomposition (SVD) of the corresponding term/document matrix A. After a weighting scheme has been applied to each element of A, the SVD of the matrix A is computed by the following equation.

$$A = U\Sigma V^T$$

SIGIR 2000 7/00 Athens, Greece
© 2000 ACM 1-58113-226-3/00/0007...$5.00

In this equation, the $(m \times n)$ matrix U and $(n \times n)$ matrix V are orthogonal, i.e. $U^TU = V^TV = I_n$. And, the singular values of A are defined as the diagonal elements of Σ which are the non-negative square roots of the n eigenvalues of A^TA[6].

The first k columns of U and V and the first k diagonal elements of Σ are used to construct a rank-k approximation to A as defined in the following equation.

$$A_k = U_k \Sigma_k V_k^T$$

Using the rank-k model A_k, the associated vector space represents a semantic structure for the term and the document. Each term vector q_i in the vector space is in the ith row of U_k whose columns are scaled by the k singular values of Σ_k[7].

For the purpose of information retrieval, a user's query must be represented as a vector in k-dimensional space and the vector is compared to documents. The user query can be represented by

$$\hat{q} = q^T U_k \Sigma_k^{-1}$$

where q is simply the vector of words in the user query, and the right multiplication Σ_k^{-1} differentially weights the separate dimensions[7].

3 Boolean LSI Model

The Boolean LSI model allows one to combine Boolean query formulations with characteristics of the LSI model. We use the P-norm model to process Boolean query formulations and the LSI model to compute the weight w_{ij} of the term t_i in the document d_j.

In the LSI model, the term t_i and the document d_j can be represented as k-dimensional vectors by the Singular Value Decomposition. To compute the degree of similarity of $\vec{t_i}$ and $\vec{d_j}$, we should transform $\vec{t_i}$ into a *pseudo-document* vector. Therefore, $\vec{t_i}$ is scaled by Σ_k^{-1}. The degree of similarity of the term t_i and the document d_j can be quantified by the cosine of the angle between $\vec{t_i}\Sigma_k^{-1}$ and $\vec{d_j}$.

First, we define $\hat{w}_{ij}$ as

$$\hat{w}_{ij} = sim(\vec{t_i}\Sigma_k^{-1}, \vec{d_j}) = \frac{\vec{t_i}\Sigma_k^{-1} \cdot \vec{d_j}}{|\vec{t_i}\Sigma_k^{-1}| \times |\vec{d_j}|}$$

where $\vec{t_i}$ is the ith row of U_k and $\vec{d_j}$ is the jth row of V_k.

Since $\hat{w}_{ij}$ is the cosine similarity, it varies from -1 to 1. As in the P-norm model, the weight is laid between 0 and 1. So we define w_{ij} as follows.

$$w_{ij} = \begin{cases} \hat{w}_{ij} & \text{if } \hat{w}_{ij} > 0 \\ 0 & \text{otherwise} \end{cases}$$

Consider, as an example, a document D with assigned terms A and B and let w_A and w_B represent the weights or importance of the two terms in the document, $0 \leq w_A, w_B \leq 1$. A term weight of 0 indicates that the corresponding term is not assigned to an item; a weight of 1 represent a fully weighted term, and weights between 0 and 1 are partial term assignments. Given queries (A *and* B) and (A *or* B), it is possible to define the following query-document similarity functions between these queries and the document $D = (w_A, w_B)$.

$$sim(Q_{(A \text{ and } B)}, D) = 1 - \sqrt{\frac{(1-w_A)^2 + (1-w_B)^2}{2}}$$

$$sim(Q_{(A \text{ or } B)}, D) = \sqrt{\frac{w_A^2 + w_B^2}{2}}$$

4 Experimental Results

The performance of the Boolean LSI model has been systematically compared with the vector model and the P-norm model, as well as the LSI model. We have utilized the following two standard document collections: (i) MED (1033 document abstracts in biomedicine received from the National Library of Medicine) and (ii) CISI (1460 document abstracts in library science and related areas extracted from Social Science Citation Index by Institute for Scientific Information).

We removed stopwords from document collections and stemmed terms using the Porter stemmer. Words occurring in more than one document were selected for both the LSI and the Boolean LSI indexing. The $tf \cdot idf$ weighting scheme is used for the vector model, the LSI and the Boolean LSI model. Since the weight of the P-norm Model should be laid between 0 to 1, the P-norm model uses $log(tf+1)\frac{idf}{max\ idf}$ for its weighting scheme. In the LSI and the Boolean LSI model, $tf \cdot idf$ weighting scheme has been applied to each element of the original term/document matrix, and a reduced dimensional SVD of it is calculated.

Figure 1 represents average precision versus recall curves for four distinct retrieval models. The condensed results in terms of average precision recall are summarized in Table 1. In the MED and CISI collections, the performance of the P-norm model is better than that of the vector model and the performance of the Boolean LSI model is better than that of the LSI

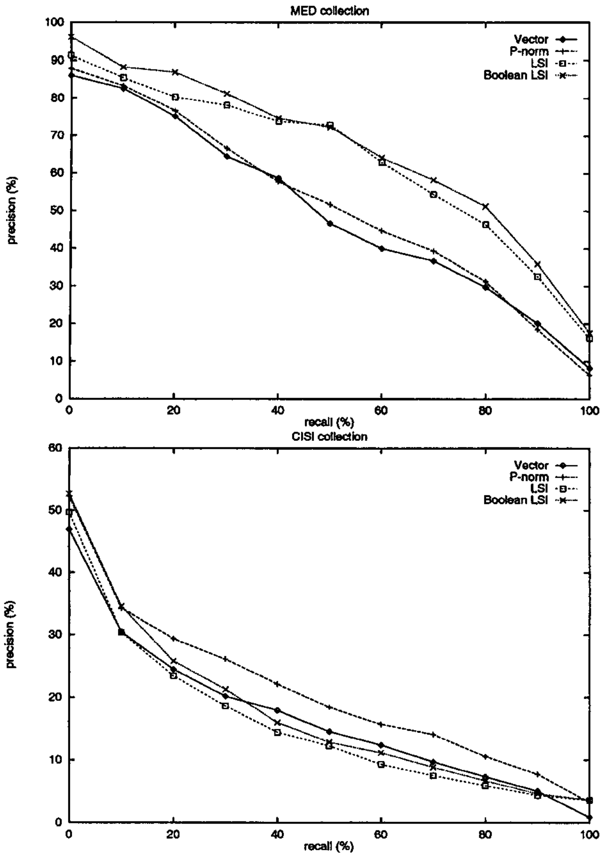

Figure 1: Precision at 11 standard recall levels for 4 models

model. However, the performance of the LSI model is not always better than that of the vector model. It is noticeable that the performance improvement of the Boolean LSI model over the LSI model is similar to that of the P-norm model over the vector model.

5 Conclusions

We have proposed a new model called Boolean LSI model for information retrieval. This new model allows one to combine Boolean query formulation with characteristics of the LSI model. It can take advantages of the LSI model and the P-norm model. First, it can alleviate the problems of lexical matching in the traditional information retrieval models. Second, it can also utilize the representation power of Boolean query formulation. The experimental results showed that the performance improvement of the Boolean LSI model over the LSI model is similar to that of the P-norm model over the vector model.

	MED		CISI	
Vector	0.498	-	0.172	-
P-norm	0.512	+2.8%	0.212	+23.2%
LSI	0.631	-	0.163	-
Boolean LSI	0.660	+4.6%	0.180	+10.3%

Table 1: Average precision for 4 models and relative improvement

References

[1] G. Salton, C. Buckley, and E.A. Fox, Automatic Query Formulations in Information Retrieval, *Journal of the American society for information science*, 34(4): 262-280, 1983.

[2] G. Salton, E.A. Fox, and H. Wu, Extended Boolean Information Retrieval, *Communications of ACM*, 26(12): 1022-1036, 1983.

[3] G.B. Lee, M.H. Park, and H.S. Won, Using syntactic information in handling natural language quries for extended Boolean retrieval model, *Proceedings of the 4th international workshop on information retrieval with Asian languages*, Academia Sinica, Taipei, 1999.

[4] M.E. Smith, Aspects of the P-norm Model of Information Retrieval: Syntactic Query Generation, Efficiency, and Theoretical Properties, *Phd Thesis*, Computer Science, Cornell University, 1990.

[5] M.W. Berry, S.T. Dumais, and T.A. Letsche, Computational Methods for Intelligent Information Access, *Proceedings of Supercomputing '95*, San Diego, CA, December 1995.

[6] M.W. Berry, S.T. Dumais, and G. O'Brien, Using Linear Algebra for Intelligent Information Retrieval, *SIAM Review*, 37: 573-595, 1995.

[7] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, and R. Harshman, Indexing by Latent Semantic Analysis, *Journal of the American society for information science*, 41(6): 391-407, 1990.