Term Weighting Schemes Experiment Based on SVD for Malay Text Retrieval

Nordianah Ab Samat[†], Masrah Azrifah Azmi Murad[††], Muhamad Taufik Abdullah[††], Rodziah Atan[††]

Faculty of Computer Science and Information Technology
University Putra Malaysia, 43400 Serdang, Selangor, Malaysia

Summary
The goal in information retrieval is to locate relevant documents in response to a user's query at the same time retrieving as few as possible of the irrelevant documents. One possible approach to this problem is to use the Singular Value Decomposition (SVD) which models documents and queries as vectors in reduced space. The components of the vector are determined by the term weighting scheme, a function of the frequencies of the terms in the document or query. In this paper, we discuss term weighting schemes and the results from experiment on Malay text retrieval using a set of Malay document collection.

Keywords:
Singular Value Decomposition, term weighting, Malay document

1. Introduction

The volume of digital documents increases rapidly in recent years, an automatic information retrieval is needed imminently. The traditional vector space model (VSM) models documents and queries as vectors and computes similarity scores using an inner product. Although the VSM is simple and fast, there are a few drawbacks of using the VSM. It cannot reflect similarity of words and only counts the number of overlapping words and it ignores synonymy and polysemy.

Latent Semantic Indexing (LSI) [1] is an information retrieval technique that was designed to address the deficiencies of the classic VSM technique. LSI assumes that there is some underlying or latent structure in word usage that is partially obscured by variability in word choice. LSI extends the vector space model by modeling term-document relationships using a reduced approximation for the column and row space computed by the SVD of the term by document matrix.

For our work, we intend to use SVD for the vector representation instead of classic VSM in order to get the better retrieval results. However, SVD needs an additional term weighting algorithm before they can be implemented. Term weighting scheme is the function that determine the components of the vectors.

The paper is organized as follows. In the next section, we discuss about the SVD. Section 3 explains the term weighting, section 4 describes our experimental details and section 5 reports on results obtained. Finally, section 6 concludes the paper**.**

2. Singular Value Decomposition

Singular Value Decomposition (SVD) is a form of factor analysis, or more properly, the mathematical generalization of which factor analysis is a special case [2]. It constructs an *n* dimensional abstract semantic space in which each original term and each original document are presented as vectors.

In SVD a rectangular term-by-document matrix *A* is decomposed into the product of three other matrices *T*, *S*, and *D'* (refer to figure 1).

$$\{A\} = \{T\}\{S\}\{D\}' \quad (1)$$

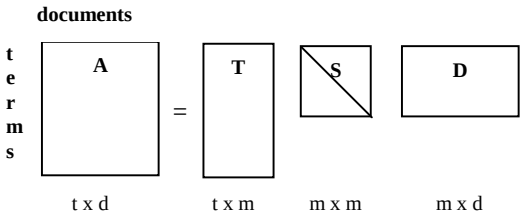

Fig. 1: SVD Description

T is an orthonormal matrix and its rows correspond to the rows of *A*, but it has *m* columns corresponding to new, specially derived variables such that there is no correlation between any two columns; i.e., each is linearly independent of the others. *D* is an orthonomal matrix and has columns corresponding to the original columns but *m* rows composed of derived singular vectors. The third

Manuscript received October 5, 2008
Manuscript revised October 20, 2008

matrix *S* is an *m* by *m* diagonal matrix with non-zero entries (called singular values) only along one central diagonal. The role of these singular values is to relate the scale of the factors in the other two matrices to each other such that when the three components are matrix multiplied, the original matrix is constructed.

Following the decomposition by SVD, the *k* most important dimensions (those with the highest singular values in *S*) are selected. All other factors are omitted, i.e., the other singular values in the diagonal matrix along with the corresponding singular vectors of the other two matrices are deleted. Ideally, *k* should be large enough to fit the real structure in the data, but small enough such that noises, sampling errors or unimportant details are not modeled [1].

The reduced dimensionality solution then generates a vector of k real values to represent each document. The reduced matrix ideally represents the important and reliable patterns underlying the data in A. It corresponds to a least-squares best approximation to the original matrix A.

3. Term Weighting

Term weighting is an important factor in the performance of information retrieval systems. Many weighting methods have been developed within text search, and their variety is astounding. Proper term weighting can greatly improve the performance of the SVD approach.

A collection of d documents described by t terms can be represented as a $t \times d$ matrix A, hereafter referred to as the term-document matrix. It is a sparse matrix whose rows correspond to documents and whose columns correspond to stemmed terms that appear in the documents. Each element w_{ij} of the term-document matrix represents the weight of term t_i in each of the document d_j. Generally speaking, the weight w_{ij} of term t_i in document d_j is given by the product of three different factors:

$$w_{ij} = L_{ij} G_i N_j$$

where L_{ij} is the local weight of term *i* in document *j*, G_i is the global weight of term i in the document collection, and N_j is the normalization factor for document *j*.

Local weights are functions of how many times each term appears in a document. Local weighting formulas perform well if they work on the principle that the terms with higher within-document frequency are more pertinent to that document.

Global weights are functions of how many times each term appears in the entire collection. It tries to give a "discrimination value" to each term. Many schemes are based on the idea that the less frequently a term appears in the whole collection, the more discriminating it is [3].

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.

The document vectors and query vectors are weighted using separate schemes. The local weight is computed according to the terms in the given document or the query. The global weight, however, is based on the document collection regardless of whether we are weighting documents or queries. The normalization is done after the local and global weighting. Normalizing the query vectors is not necessary because it does not affect the relative order of the ranked document list.

4. Experiment Details

In order to construct datasets, we ran Malay queries on Google and selected a number of the top-ranked search results. For query matching in our experiment, we used a short query, 'perasaan' (feeling) and the relevant results are composed of about 90 documents.

There are several weighting schemes that can be used to construct document vectors [4]. We have experimented several weighting schemes, as follows. Let us call f_{ij} the frequency of term *i* in document *j*, F_i the global frequency of term *i* in the whole document collection, n_i the number of documents in which term *i* appears, *N* the total number of documents, and x_j is the maximum frequency of any term in document *j*. The following table summarizes the local weights, global weights and normalization weight that have been used in our experiments (see Table 1, Table 2 and Table 3).

Table 1: Established local weights used

Abbr.	***Name***	*Formula*
BNRY	Binary	1 if $f_{ij} > 0$ 0 if $f_{ij} = 0$
FREQ	within-document frequency	f_{ij}
LOGA	Log	$1 + \log f_{ij}$ if $f_{ij} > 0$ 0 if $f_{ij} = 0$
LOGN	Normalized Log	$\frac{1 + \log f_{ij}}{1 + \log a_j}$ if $f_{ij} > 0$ 0 if $f_{ij} = 0$
ATF1	Augmented normalized term frequency	$0.5 + 0.5 (\frac{f_{ij}}{x_j})$ if $f_{ij} > 0$ 0 if $f_{ij} = 0$
ATFC	Changed-coefficient ATF1	$0.2 + 0.8 (\frac{f_{ij}}{x_j})$ if $f_{ij} > 0$
ATFA	Augmented average term frequency	$0.9 + 0.1 (\frac{f_{ij}}{a_j})$ if $f_{ij} > 0$ 0 if $f_{ij} = 0$
LOGG	Augmented log	$0.2 + 0.8 \log (f_{ij} + 1)$ if $f_{ij} > 0$ 0 if $f_{ij} = 0$
SQRT	Square root	$\sqrt{f_{ij} - 0.5} + 1$ if $f_{ij} > 0$ 0 if $f_{ij} = 0$

Table 2: Established global weight formulas used

Abbr.	***Name***	*Formula*
IDFB	Inverse document frequency	$\log (\frac{N}{n_i})$
IDFP	Probabilistic inverse	$\log (\frac{N - n_i}{n_i})$
ENPY	Entropy	$1 + \sum_{j=1}^{N} \frac{\frac{f_{ij}}{F_i} \log \frac{f_{ij}}{F_i}}{\log N}$
IGFF	Global frequency IDF	$\frac{F_i}{n_i}$
NONE	No global weight	1
IGFL	Log-global frequency IDF	$\log (\frac{F_i}{n_i} + 1)$
IFGI	Incremental global frequency IDF	$\frac{F_i}{n_i} + 1$
IGFS	Square root global frequency IDF	$\sqrt{\frac{F_i}{n_i}} - 0.9$

Table 3: Normalization factors used

Abbr.	***Name***	*Formula*
COSN	Cosine normalization	$\frac{1}{\sqrt{\sum_{i=0}^{m} (G_i L_{ij})^2}}$
PUQN	Pivoted unique normalization	$\frac{1}{(1 - slope) + slopel_j}$
NONE	None	1

Furthermore, we also test these term weighting formulas on the traditional VSM to compare with SVD approach. The results can be found in section 5.

5. Results and Discussion

In this study, we did several experiments for term weighting methods on our data collection. To test these weighting formulas, we implemented LSI approach for the vector representation and then be computed by the SVD in order to reduce the column and row space.

For a given weighting scheme, we computed the similarity between the documents and each query in test collection and returned a list of documents ranked in order of their similarity scores. Table 4 shows various weighting schemes used in our weighting experiment and the average precision for retrievals in descending order.

Table 4: The results of average precision for each weighting schemes

Scheme	*Local weight*	*Global weight*	*Normalization factor*	*Average precision*
1	LOGA	NONE	COSN	0.980909
2	SQRT	IGFL	COSN	0.941103
3	LOGA	ENPY	COSN	0.936318
4	LOGN	NONE	PUQN	0.933039
5	FREQ	NONE	COSN	0.930513
6	FREQ	IDFB	COSN	0.912285
7	ATFC	IGFL	COSN	0.893284
8	LOGG	IGFS	COSN	0.888878
9	LOGG	IGFI	COSN	0.882813
10	ATFC	IGFS	COSN	0.879741
11	SQRT	IGFI	COSN	0.875952
12	ATFC	IGFI	COSN	0.871577
13	ATFA	IGFS	COSN	0.864485
14	SQRT	IGFS	COSN	0.862921
15	SQRT	IGFF	COSN	0.856039
16	LOGG	IGFF	COSN	0.850548
17	LOGA	IGFF	COSN	0.840387
18	ATF1	NONE	NONE	0.741047
19	FREQ	NONE	NONE	0.419147

The combination of local and global weights used does make a difference. A particular local weight when combined with one global weight may perform well but when combined with a different global weight may perform poorly. The combination of document weighting and query weighting also makes a difference in performance.

Our results show that the combinations of local weight LOGA, global weight NONE and normalization factor COSN works well in documents. Furthermore, they appear at the top-performing weighting schemes. The combination local weight FREQ without global weight and normalization factor produces lowest precision. For query weighting, we used only local weight BNRY and global weight IDFB because each term in a query appears only once or twice.

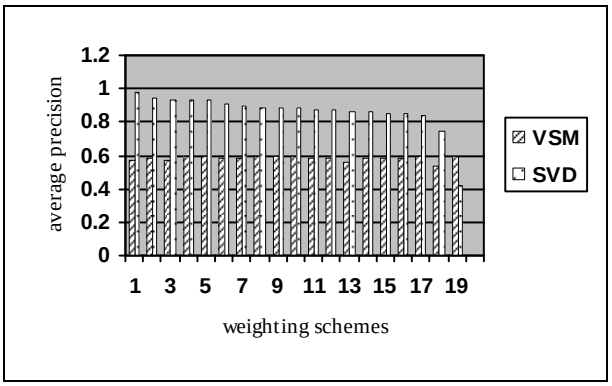

Figure 2: Average precision of SVD compared to VSM for each weighting schemes

Figure 2 visualize the differences between SVD and traditional VSM for each weighting scheme. It is proven that by using SVD as vector representation, the precision of retrievals increased. The graph shows VSM yields lower precision at all weighting schemes. SVD produces better result. If we take average precision to compare overall performance of the similarity measures, SVD is superior to VSM +47.82%. As the results show, SVD method offers improvement over the popular VSM method.

6. Conclusion and Future Work

We have experimented several of term weighting schemes on our Malay text collection. The weighting schemes with the highest average precision are a document weight of LOGA – NONE – COSN combined with a query weight of BNRY – IDFB.

The simple weighting formulas work best for the query weighting because each term in a query appears only once or twice. However, for the documents, more complex term weight formulas are necessary. This is possibly because documents contain more terms, these terms occurs with greater frequency, and length discrepancies are more noticeable in the documents than in the queries.

Our results also show that the vector representation implemented by the SVD produced better retrieval results compared to traditional VSM, when both combined with the weighting schemes.

In future, we plan to cluster documents into this space using the resulted term weighting. Document clustering is

a procedure to separate documents according different topics.

References

[1] Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., & Harshman, R., "Indexing by Latent Semantic Analysis," Journal of the American Society For Information Science, 41, 391-407, 1990.

[2] Berry, M. W., Dumais, S.T., & O'Brien, G.W., "Using linear algebra for intelligent information retrieval," SIAM Reviews, 37, 73-595, 1995.

[3] G. Salton & C. Buckley, "Term weighting approaches in automatic text retrieval," Information Processing and Management, 24(5):513-523, 1988.

[4] Erica, C. & Tamara, G.K., "New term weighting formulas for the vector space method in information retrieval," Technical Memorandum O RNL-13756, Oak Ridge National Laboratory, Oak Ridge, Tennessee, 1988.

[5] Van Rijsbergen, C.J., Information Retrieval, 2nd edition, Butterwoth 1979.

[6] Baeza-Yates, R., & Ribeiro-Neto, B. Modern Information Retrieval, ACM Press, 1999.

Nordianah Ab Samat received her Bachelor of Computer Science in Software Engineering from University Putra Malaysia in 2006. She is currently a second-year-research master student at Faculty of Computer Science and Information Technology, University Putra Malaysia. Her research interest includes information retrieval and text mining techniques.