Latent Semantic Indexing versus Inverted Files Indexing Technique: An Evaluation of Effectiveness of the Retrieval Method

Roslan Sadjirin
Universiti Teknologi MARA Pahang, Malaysia
Email: roslan81@pahang.uitm.edu.my

Nurazzah Abd Rahman
Universiti Teknologi MARA, Malaysia

ABSTRACT

Information retrieval (IR) is part of computer science which studies the techniques of retrieval of information that aim in satisfying users' queries. Therefore, this paper presents the comparison of two different retrieval techniques in information retrieval which are Latent Semantic Indexing (LSI) and Inverted Files Indexing technique. A total of 210 of Malay translated hadith documents from hadith Sahih Muslim and Hadith Sahih Bukhari were used to study the comparison of these retrieval techniques. The result of comparison signifies that retrieval method used in LSI performed better compared to the retrieval method used in Inverted Files technique.

Keywords: *latent semantic indexing, retrieval method, searching method, inverted files*

Introduction

Searching or retrieving relevant document from the collection of stored digital document is tedious and difficult task if the collection stored and named with inappropriate file names which cannot describe its content. Therefore, prior to the searching, many indexing techniques have been

ISSN 0128-5599

introduced in order to make the retrieval task more convenient and effective.

To date, there are two types of textual indexing which are indexing by term and indexing by concept. Inverted Files, Signature Files and Suffix Array are the main indexing techniques categorised under indexing by term. However, according to Yates et al. (1999) and Trotman (2004), Inverted Files has outperformed the other techniques in its category. Meanwhile, the other indexing technique categorised under indexing by concept is called Latent Semantic Indexing (LSI) technique. Inverted Files Indexing technique and Latent Semantic Indexing technique employ distinct retrieval methods to discover and rank the relevant document against the query formulated by the users. Inverted Files Indexing technique uses exact term-matching technique, while LSI uses cosine similarity measurement to retrieve relevant documents in the collection.

Inverted Files Indexing technique is a well-known indexing technique which is widely used in information retrieval application and it is a fundamental technology underlying all internet search engines to speed up retrieval task (Navarro et al., 2004; Shaporenkov, 2005; Puglisi et al., 2006). Inverted Files technique consists of two main components which are Inverted List and Posting Files. Inverted List, which is also known as vocabulary table or dictionary contains an entry for every term in the collection. In the other words, Inverted List is a vector that stores the unique or distinct term in lexicographical order for each document in which the term is present. While Posting File, which is also known as lexicon, is a list of pointers to occurrences of term in collection (Czerski et al., 2007; Yates et al., 1999; Wensi, 2002; Navarro et al., 2004; Trotman, 2004; Vo & Moffat, 2004; Marin & Gil Costa, 2007). Furthermore, Inverted Files Indexing technique is classified as indexing by term because it uses word-oriented mechanism to index and retrieve text collection. Precisely, it uses hashing technique or exact term-matching technique to discover the relevance document against the query terms performed. Therefore, all documents in collection will be retrieved if the query term formulated by users is identical or exactly the same to the term in the documents collection (Yates & Neto, 1999).

Meanwhile, Latent Semantic Indexing (LSI) technique is classified under indexing by concept because it determines the rules set of concept between terms and documents, documents and document as well as terms and terms using mathematical representations. LSI does not use thesauri or expansion mechanism to expand a query to find the different ways the same thing has been represented. LSI is not a traditional natural language processing or artificial intelligence program, and it does not use humanly constructed dictionaries, knowledge bases, semantic networks, grammars, syntactic parsers or morphologies. However, LSI is a fully mathematical technique for extracting and inferring relations of expected contextual usage of words or terms in passages of discourse or set documents (Launder, 1998, Kowalski, 2008).

The main component in LSI technique is Singular Value Decomposition (SVD). SVD is a linear algebra method used to discover and map the associate relationship between terms and terms, terms and documents, and document and documents. SVD transformed a high-dimensional vector into a lower-dimensional semantic vector by projecting it into a semantic subspace called LSI space (Inien et al., 1996; Chunqiang et al., 2004).

In LSI technique, the effectiveness of the retrieval relies on the selection threshold value (ε) and the selection of the dimensionality k for SVD computation in LSI space. A good, and widely used heuristic is $\varepsilon = 0.7$ because this heuristic value is equal to a 45° angle between the corresponding vectors (Marcus et al., 2003). In SVD computation, if the corresponding vectors are orthogonal, or the two corresponding vectors are 45° angle to each other, they are consider similar. However, the optimal of dimensional k is currently determined by exhaustive evaluation, and how to calculate the optimal k directly from LSI space remains an open question (Ding, 1999; Marcus et al., 2003). Nevertheless, Furnas et al. (1988) stated that the value of k has to be large enough to fit all the real structure in the data but then it also should be small enough so that the sampling error or unimportant detail will not be fit in. Furthermore, the similarity of the documents to the set of query in LSI technique is typically measured by using a cosine measurement (Marcus et al., 2003; Gee, 2003). The highest cosine similarity of documents to the query in LSI space would be identified as the best matches to the given query (Hoffman, 1999; Dasgupta et al., 2005).

In summary, Inverted Files Indexing technique is a word-oriented mechanism that uses exact-term matching for its retrieval, and LSI technique is a mathematical computation that uses cosine similarity measurement to discover the association and relations between terms and terms, terms and documents and documents for its retrieval. In standard vector model, number of dimension is equal to the number of term, while in reduced LSI vector space model, number of dimension is lesser than number of terms. In standard vector model, the relevant document will only be retrieved if the given query terms is identically matching with the terms in the corresponding documents. While, in the reduced LSI vector space model, all terms and documents are mapped into the reduced LSI space whereby every terms and documents is potentially be semantically related to each other because the similarity and associatively of terms and documents depend on the cosine similarity measurement and threshold value selection.

Therefore, this paper aims to present the result of evaluation of the effectiveness (*E*) measurement for two different retrieval methods employed in Inverted Files Indexing technique and Latent Semantic Indexing technique.

Test Collections

In this study, the test collections used consisted of 210 Malay translated hadith documents from 95 hadith Sahih Bukhari and 115 hadith Sahih Muslim, 1196 number of extracted terms, 12 set of queries, and a list of relevant judgement provided by Hadith Muslim's and Hadith Bukhari book. These translated hadith comprise of 20 hadith about "ilmu", 20 hadith about "Wuduk", 20 hadith about "Solat", 20 hadith about "Zakat", 20 hadith about "Haji", 20 hadith about "Puasa", 10 hadith about "Makan", 10 hadith about "Minuman", 10 hadith about "Persiapan", 20 hadith about "Rasulullah", 20 hadith about "kiamat" and 20 hadith about "Iman". Furthermore, Ahmad (1995) said Malay stop words were used to stem vocabulary and remove the stopwords from the collection of documents. Table 1 illustrates the selected queries that were used for the evaluation and measurement of the retrieval effectiveness between Inverted Files Indexing technique and Latent Semantic Indexing technique. Meanwhile Table 2 illustrates the list of relevant judgment which was used in this study.

Evaluation and Measurement Technique of the Retrieval

The evaluation of retrieval for the Latent Semantic Indexing techniques and Inverted File Indexing technique on Malay translated hadith used well-known Information Retrieval metrics which are recall (*R*) and precision (*P*) and effectiveness (*E*). Recall was used to measure the relevant documents which were effectively retrieved. On the other hand,

Table 1: List of Queries

Query #	Query Words
1	Kewajipan menuntut ilmu di dalam Islam
2	Bersuci dari hadas sebelum mengerjakan solat
3	Dosa-dosa meninggalkan solat wajib
4	Jenis-jenis zakat dan kepentingannya
5	Rukun mengerjakan haji
6	Kelebihan berpuasa
7	Adab-adab makan mengikut sunnah
8	Perhiasan dan pakaian yang dibenarkan di dalam Islam
9	Kisah-kisah Rasulullah
10	Tanda-tanda kiamat
11	Ciri-ciri orang yang beriman
12	Minuman yang diharamkan di dalam Islam

Table 2: List of Relevant Judgment

#	Query Words	Relevant Judgment
1	Kewajipan menuntut ilmu di dalam Islam	H0049BJ1, H0050BJ1, H0051BJ1, H0052BJ1, H0053BJ1, H0054BJ1, H0055BJ1, H0056BJ1, H0057BJ1, H0058BJ1, H0059BJ1, H0060BJ1, H0061BJ1, H0062BJ1, H0063BJ1, H0064BJ1, H0065BJ1, H0066BJ1, H0067BJ1, H0068BJ1
2	Bersuci dari hadas sebelum mengerjakan solat	H0096BJ1, H0097BJ1, H0098BJ1, H0099BJ1, H0100BJ1, H0101BJ1, H0102BJ1, H0103BJ1, H0104BJ1, H0105BJ1, H0106BJ1, H0107BJ1, H0108BJ1, H0109BJ1, H0110BJ1, H0111BJ1, H0112BJ1, H0113BJ1, H0114BJ1, H0115BJ1
3	Dosa-dosa meninggalkan solat wajib	H0211BJ1, H0212BJ1, H0213BJ1, H0214BJ1, H0215BJ1, H0216BJ1, H0217BJ1, H0218BJ1, H0219BJ1, H0220BJ1, H0326MJ1, H0327MJ1, H0328MJ1, H0329MJ1, H0330MJ1, H0331MJ1, H0332MJ1, H0333MJ1, H0334MJ1, H0335MJ1
4	Jenis-jenis zakat dan kepentingannya	H0716BJ2, H0717BJ2, H0718BJ2, H0719BJ2, H0720BJ2, H0721BJ2, H0722BJ2, H0723BJ2, H0724BJ2, H0725BJ2, H0927MJ2, H0928MJ2, H0929MJ2, H0930MJ2, H0931MJ2, H0932MJ2, H0933MJ2, H0934MJ2, H0935MJ2, H0936MJ2
5	Rukun mengerjakan haji	H0785BJ2, H0786BJ2, H0787BJ2, H0788BJ2, H0789BJ2, H0790BJ2, H0791BJ2, H0792BJ2, H0793BJ2, H0794BJ2, H1146MJ2, H1147MJ2, H1148MJ2, H1149MJ2, H1150MJ2, H1151MJ2, H1152MJ2, H1153MJ2, H1154MJ2, H1155MJ2
6	Kelebihan berpuasa	H0928BJ2, H0929BJ2, H0930BJ2, H0931BJ2, H0932BJ2, H0933BJ2, H0934BJ2, H0935BJ2, H0936BJ2, H0937BJ2, H1039MJ2, H1040MJ2, H1041MJ2, H1042MJ2, H1043MJ2, H1044MJ2, H1045MJ2, H1046MJ2, H1047MJ2, H1048MJ2

continued

Table 2: (*continued*)

#	Query Words	Relevant Judgment
7	Adab-adab makan mengikut sunnah	H1630BJ4, H1631BJ4, H1632BJ4, H1633BJ4, H1634BJ4, H1635BJ4, H1636BJ4, H1637BJ4, H1638BJ4, H1639BJ4
8	Perhiasan dan pakaian yang dibenarkan di dalam Islam	H1674BJ4, H1675BJ4, H1676BJ4, H1677BJ4, H1678BJ4, H1679BJ4, H1680BJ4, H1681BJ4, H1682BJ4, H1683BJ4
9	Kisah-kisah Rasulullah	H1267BJ3, H1268BJ3, H1269BJ3, H1270BJ3, H1271BJ3, H1272BJ3, H1273BJ3, H1274BJ3, H1275BJ3, H1276BJ3, H2120MJ4, H2121MJ4, H2122MJ4, H2123MJ4, H2124MJ4, H2125MJ4, H2126MJ4, H2127MJ4, H2128MJ4, H2129MJ4
10	Tanda-tanda kiamat	H1880BJ4, H1881BJ4, H1882BJ4, H1883BJ4, H1884BJ4, H2385MJ4, H2386MJ4, H2387MJ4, H2388MJ4, H2389MJ4, H2390MJ4, H2391MJ4, H2392MJ4, H2393MJ4, H2394MJ4, H2395MJ4, H2396MJ4, H2397MJ4, H2398MJ4, H2399MJ4
11	Ciri-ciri orang yang beriman	H0001MJ1, H0002MJ1, H0003MJ1, H0004MJ1, H0005MJ1, H0006MJ1, H0007MJ1, H0008MJ1, H0009MJ1, H0010MJ1, H0011MJ1, H0012MJ1, H0013MJ1, H0014MJ1, H0015MJ1, H0016MJ1, H0017MJ1, H0018MJ1, H0019MJ1, H0020MJ1
12	Minuman yang diharamkan di dalam Islam	H1920MJ4, H1921MJ4, H1922MJ4, H1923MJ4, H1924MJ4, H1925MJ4, H1926MJ4, H1927MJ4, H1928MJ4, H1929MJ4

precision w used to measure the retrieved documents which were known to be relevant. As stated by Marcus, et al. (2003), if recall is 1, it means that all the relevant documents are retrieved, though there could be retrieved document that are not relevant. If the precision is 1, it means that the entire retrieved documents are relevant, though there could be relevant documents that were not retrieved. The evaluation of the recall and precision of the retrieval was done by applying Eqn 1. Eqn 2 is the formula of effectiveness measurement. The effectiveness measurement, *E*, is a weighted combination of *R* and *P* (van Rijsbergen, 1976). The value of $\beta > 0$ indicates how many times more important *R* is compared to *P*. if $\beta = 1.0$, *R* and *P* are equally important, while if $\beta > 2.0$, *R* is twice as important as *P*. On the other word, *R* is twice more important than *P*. In this study, *R* and *P* are equally important, therefore, β is set to $\beta = 1.0$.

$$\text{Recall (\%)} = \frac{\text{Total Retrieve \& Relevant}}{\text{Total Relevant}}$$

$$\text{Precision (\%)} = \frac{\text{Total Retrieve \& Relevant}}{\text{Total Retrieve}} \qquad \text{Eqn 1}$$

$$\text{Effectiveness (\%)} = 1 - \frac{(1+\beta^2)PR}{\beta^2 P + R} \qquad \text{Eqn 2}$$

Cosine Similarity Measurement

The similarity measurement of retrieval method in Inverted Files uses exact term-matching technique whereby the stored document that contains identical term with the formulated users' query are supposedly retrieved. However, the similarity measurement of retrieval method in LSI technique employs the cosine similarity computation (Gee, 2003). If the cosine is 1, it signifies that the two vectors are considered to exactly similar and a cosine of -1 means that they theoretically completely dissimilar. Figure 4 shows the formula of cosine similarity to measure the similarity between term and document, term and term as well as document and document in LSI space. To illustrate the example, let $A = (x_i, y_i)$ is the vector of the query (q_i) or term (t_i) or document (d_i) and $B = (x_j, y_j)$ is another vector of term (t_j) or document (d_j) in LSI space. Hence, the cosine similarity can be calculated as shown in Eqn 3 below.

$$\begin{aligned} \text{Sim}(A, B) &= \text{Cosine } \theta \\ &= A \,.\, B \,/\, |A||B| \qquad \text{Eqn 3} \end{aligned}$$

Experiment on Retrieval of Latent Semantic Indexing Technique Using Cosine Similarity Measurement

In Latent Semantic Indexing retrieval experiment, four values of threshold (ε), and five values of dimensional (k) were selected. The threshold values selected were $\varepsilon = [0.5, 0.6, 0.7, 0.8]$, and dimensionality values selected were $k = [2, 3, 4, 5, 6]$. Previous research suggested that, the good and widely used value of heuristic is 0.7 (Marcus et al., 2003) as it is equal to 45° angle between the corresponding vectors. In addition, $\varepsilon = [0.5, 0.6, 0.8]$ were also selected because these values are closely to 45°. However, the values of k were randomly picked because it has been proven that there is no single optimized value for k-dimensionality. Nevertheless, Chunqiang et al. (2004) explained that, the number of k-dimensionality in LSI space cannot be higher than $l = O(\log n)$ where n is the number of node in the system. Node is the number of terms multiplies by the number of documents that are scattered in LSI space. In this experiment, $k = 1$ was not chosen because in LSI space concept, the possible minimum value of k-dimensionality should be $k = 2$ which is equivalent to x-axes and y-axes.

Experiment on Retrieval of Inverted Files Indexing Technique Using Exact Term-Matching Retrieval Method

In Inverted Files indexing technique, the result of the retrieval does not rely on the selection of the certain variable as such in LSI technique. The retrieval of the relevant document for this technique completely depends on the exact pattern of the terms in the documents collection with the query terms formulated by the user. Therefore, the retrieval method in Inverted Files Indexing technique is called exact term-matching technique. The same test collections as such in LSI's experiment were used in order to carry out the experiment on this technique.

Result and Discussions

Result and Discussions for Experiment on Retrieval of Latent Semantic Indexing Technique

Table 3 illustrates the percentage of recall, precision and effectiveness which were using the 12 formulated queries as shown in Table 1. Furthermore, Table 3 also presents the average for each metrics which were using different dimensionality, $k = [2, 3, 4, 5, 6]$ and threshold values $\varepsilon = [0.5, 0.6, 0.7, 0.8]$.

Meanwhile, Tables 4, 5 and 6 show the average of recall (R), precision (P) and Effectiveness (E) respectively. Table 4 shows that the highest percentage of recall, which is 97.1 percent, can be obtained in LSI if the threshold ($\mathring{a}$) value chosen is 0.5 and k-dimensional value is set to 2. On the other hand, Table 5 shows that highest percentage of precision which 21.5 percent can be obtained if threshold (ε) value is 0.8 and the k-dimensional is set to 6. However, for effective retrieval in LSI technique, Table 6 suggests that the threshold value (ε) chosen should be 0.8 and the k-dimensional is set to 3. Therefore, it is suggested that to balance the percentage between recall and precision, thus, optimize the retrieval, the value of threshold should not be more than 0.8 and should not be less

Values ε = [0.5, 0.6, 0.7, 0.8] for Latent Semantic Indexing

Query #			1	2	3	4	5	6	7	8	9	10	11	12	Average
k = 2	ε = 0.5	*Recall (%)*	95.0	100.0	100.0	95.0	100.0	95.0	100.0	90.0	95.0	95.0	100.0	100.0	97.1
		Precision (%)	10.2	9.5	9.5	9.5	9.9	9.2	4.8	4.7	9.2	10.0	10.0	4.9	8.5
		Effectiveness (%)	81.6	82.6	82.6	82.7	82.0	83.2	90.8	91.1	83.2	81.9	81.8	90.7	**84.5**
	ε = 0.6	*Recall (%)*	95.0	100.0	100.0	90.0	100.0	95.0	100.0	90.0	95.0	90.0	100.0	100.0	96.3
		Precision (%)	11.1	9.6	9.5	9.5	10.1	9.2	4.8	4.7	9.3	10.2	10.6	5.6	8.7
		Effectiveness (%)	80.1	82.5	82.6	82.8	81.7	83.2	90.8	91.1	83.1	81.7	80.8	89.4	**84.2**
	ε = 0.7	*Recall (%)*	90.0	95.0	95.0	90.0	95.0	90.0	100.0	90.0	95.0	80.0	100.0	100.0	93.3
		Precision (%)	11.5	9.4	9.6	10.4	10.1	8.9	4.9	4.8	9.5	9.8	11.6	5.4	8.8
		Effectiveness (%)	79.6	82.9	82.6	81.4	81.7	83.8	90.7	90.9	82.7	82.5	79.2	89.8	**84.0**
	ε = 0.8	*Recall (%)*	70.0	85.0	80.0	85.0	90.0	80.0	90.0	60.0	80.0	60.0	100.0	100.0	81.7
		Precision (%)	10.7	9.1	9.1	11.0	10.7	8.4	4.9	3.4	8.6	8.4	13.0	5.6	8.6
		Effectiveness (%)	81.4	83.6	83.7	80.5	80.9	84.8	90.7	93.6	84.5	85.3	77.0	89.4	**84.6**
k = 3	ε = 0.5	*Recall (%)*	60.0	95.0	95.0	90.0	95.0	95.0	90.0	90.0	95.0	45.0	35.0	90.0	81.3
		Precision (%)	12.6	9.6	9.6	9.2	10.1	9.4	4.6	4.7	9.3	33.3	20.0	4.5	11.4
		Effectiveness (%)	79.2	82.6	82.6	83.3	81.7	82.9	91.2	91.1	83.1	61.7	74.5	91.4	**82.1**
	ε = 0.6	*Recall (%)*	40.0	95.0	90.0	85.0	90.0	19.5	90.0	80.0	85.0	25.0	20.0	90.0	67.5
		Precision (%)	11.6	9.8	9.2	9.7	10.0	10.2	4.8	4.7	8.7	31.3	17.4	4.7	11.0
		Effectiveness (%)	82.0	82.2	83.3	82.6	82.0	86.6	90.9	91.1	84.2	72.2	81.4	91.1	**84.1**
	ε = 0.7	*Recall (%)*	30.0	95.0	85.0	85.0	80.0	75.0	90.0	50.0	85.0	15.0	20.0	80.0	65.8
		Precision (%)	14.0	10.3	8.2	11.3	12.2	9.6	4.9	3.3	10.3	27.3	28.6	4.7	12.1
		Effectiveness (%)	80.9	81.4	85.0	79.7	78.8	83.0	90.7	93.8	81.6	80.6	76.5	91.1	**83.6**
	ε = 0.8	*Recall (%)*	10.0	95.0	85.0	35.0	45.0	75.0	80.0	10.0	50.0	10.0	15.0	60.0	49.2
		Precision (%)	6.7	11.0	9.7	10.3	10.1	4.6	5.0	0.9	8.3	50.0	37.5	4.2	13.2
		Effectiveness (%)	**92.0**	**80.3**	**82.4**	**82.4**	**83.5**	**91.3**	**90.4**	**95.3**	**85.8**	**83.3**	**78.4**	**92.1**	**86.3**
k = 4	ε = 0.5	*Recall (%)*	45.0	95.0	90.0	90.0	90.0	90.0	90.0	90.0	70.0	25.0	15.0	80.0	72.5
		Precision (%)	18.0	11.0	9.4	19.8	10.1	11.9	5.7	5.6	8.0	4.8	16.7	5.1	10.5
		Effectiveness (%)	74.3	80.3	83.0	67.5	81.8	79.0	89.3	89.5	85.6	91.9	84.2	90.4	**83.1**
	ε = 0.6	*Recall (%)*	20.0	90.0	80.0	75.0	75.0	65.0	90.0	90.0	40.0	25.0	15.0	70.0	61.3
		Precision (%)	16.0	11.4	9.7	18.8	11.9	11.4	6.1	6.7	5.3	4.9	20.0	4.8	10.6
		Effectiveness (%)	82.2	79.8	82.7	69.9	79.5	80.6	88.6	87.5	90.6	91.8	82.9	91.0	**83.9**
	ε = 0.7	*Recall (%)*	5.0	90.0	30.0	65.0	40.0	30.0	60.0	80.0	35.0	10.0	15.0	70.0	43.8
		Precision (%)	12.5	12.2	6.5	18.8	10.7	7.2	4.5	7.6	5.6	2.1	50.0	6.0	12.0
		Effectiveness (%)	92.9	78.5	88.5	70.8	83.1	88.4	91.6	86.1	90.3	96.5	76.9	88.9	**86.1**
	ε = 0.8	*Recall (%)*	5.0	80.0	20.0	35.0	20.0	20.0	60.0	20.0	15.0	10.0	15.0	70.0	32.5
		Precision (%)	12.5	12.5	3.7	84.6	11.8	8.2	4.8	3.7	4.1	2.1	60.0	8.5	18.0
		Effectiveness (%)	92.9	78.4	93.8	33.3	85.2	88.4	91.1	93.8	93.6	96.5	76.0	84.8	**84.0**
k = 5	ε = 0.5	*Recall (%)*	20.0	95.0	75.0	75.0	90.0	85.0	60.0	80.0	45.0	25.0	15.0	80.0	62.1
		Precision (%)	18.2	11.4	8.8	18.1	11.2	13.0	3.9	7.1	6.5	41.7	25.0	5.8	14.1
		Effectiveness (%)	80.9	79.6	84.2	70.8	80.1	77.4	92.7	87.0	88.6	68.7	81.8	89.2	**81.8**
	ε = 0.6	*Recall (%)*	10.0	90.0	65.0	70.0	50.0	25.0	60.0	50.0	45.0	0.0	15.0	80.0	46.7
		Precision (%)	9.1	11.9	9.3	17.7	10.2	3.9	4.4	6.5	7.3	0.0	37.5	7.0	10.6
		Effectiveness (%)	90.5	79.0	83.7	71.7	83.1	90.5	91.8	88.5	87.4	0.0	78.6	87.1	77.7
	ε = 0.7	*Recall (%)*	5.0	85.0	20.0	35.0	20.0	20.0	50.0	30.0	30.0	0.0	15.0	70.0	34.2
		Precision (%)	25.0	13.0	4.9	16.9	9.1	6.7	4.6	5.8	6.5	0.0	60.0	7.7	13.3
		Effectiveness (%)	91.7	77.4	92.1	74.1	87.5	90.0	91.6	90.3	89.3	0.0	76.0	86.1	**78.8**
	ε = 0.8	*Recall (%)*	5.0	70.0	5.0	55.0	15.0	5.0	40.0	0.0	4.2	0.0	15.0	70.0	23.7
		Precision (%)	25.0	16.1	2.6	20.8	16.7	4.2	6.3	0.0	6.4	0.0	75.0	10.6	15.3
		Effectiveness (%)	91.7	73.8	96.6	69.8	84.2	95.4	89.1	0.0	94.9	0.0	75.0	81.6	**71.0**
K = 6	ε = 0.5	*Recall (%)*	15.0	95.0	70.0	70.0	90.0	75.0	70.0	80.0	50.0	25.0	15.0	80.0	61.3
		Precision (%)	15.8	12.3	8.8	17.9	12.0	11.3	46.7	9.0	7.3	43.5	37.5	6.1	19.2
		Effectiveness (%)	84.6	78.2	84.4	71.5	78.8	80.1	44.0	83.8	87.3	67.7	78.6	88.7	**77.3**
	ε = 0.6	*Recall (%)*	10.0	85.0	35.0	0.7	45.0	25.0	70.0	50.0	30.0	15.0	15.0	80.0	40.1
		Precision (%)	20.0	12.5	8.8	18.7	10.0	5.6	33.8	7.9	7.6	42.9	42.9	7.5	19.9
		Effectiveness (%)	86.7	78.2	84.8	98.7	83.6	90.8	39.2	86.4	87.9	77.8	77.8	86.3	**81.5**
	ε = 0.7	*Recall (%)*	5.0	80.0	15.0	50.0	20.0	15.0	70.0	50.0	30.0	5.0	15.0	80.0	36.3
		Precision (%)	25.0	14.5	4.1	16.9	9.5	3.1	70.0	7.9	6.8	25.0	60.0	9.3	21.2
		Effectiveness (%)	91.7	75.4	93.6	74.7	87.1	92.4	30.0	86.4	88.9	91.7	76.0	83.0	**80.9**
	ε = 0.8	*Recall (%)*	0.0	50.0	5.0	30.0	15.0	5.0	40.0	30.0	20.0	5.0	15.0	60.0	22.9
		Precision (%)	0.0	14.9	2.9	14.0	21.4	3.5	66.7	8.8	6.8	33.3	75.0	10.3	21.3
		Effectiveness (%)	0.0	77.0	96.3	80.9	82.4	95.9	50.0	86.4	89.9	91.3	75.0	82.4	**75.6**

Table 4: Average of Recall for the Retrieval of Latent Semantic Indexing Technique

ε	Average of Recall, R (%)							
	k-dimensional (Factor)					Max	Min	Average
	2	3	4	5	6			
0.5	**97.1**	81.3	72.5	62.1	61.3	**97.1**	61.3	74.9
0.6	96.3	67.5	61.3	46.7	40.1	96.3	40.1	62.4
0.7	93.3	65.8	45.8	34.2	36.3	93.3	34.2	55.1
0.8	81.7	49.2	32.5	23.7	22.9	81.7	22.9	42.0
Max	**97.1**	81.3	72.5	62.1	61.3			
Min	81.7	49.2	32.5	23.7	22.9			
Average	92.1	66.0	53.0	41.7	40.2			

Table 5: Average of Precision for the Retrieval of Latent Semantic Indexing Technique

ε	Average of Recall, R (%)							
	k-dimensional (Factor)					Max	Min	Average
	2	3	4	5	6			
0.5	8.5	11.4	10.5	14.1	19.2	19.2	10.5	12.7
0.6	8.5	11.1	10.6	10.6	19.9	19.9	10.6	12.2
0.7	8.8	12.1	12.0	13.3	21.2	21.2	12.0	13.5
0.8	8.6	13.2	18.0	15.3	**21.5**	**21.5**	13.2	15.3
Max	8.8	13.2	18.0	15.3	**21.5**			
Min	8.5	11.1	10.5	10.6	19.2			
Average	8.7	12.0	12.8	13.3	20.5			

Table 6: Average of Effectiveness for the Retrieval of Latent Semantic Indexing Technique

ε	Average of Recall, R (%)							
	k-dimensional (Factor)					Max	Min	Average
	2	3	4	5	6			
0.5	84.5	84.6	83.1	81.8	77.3	84.6	77.3	82.3
0.6	84.2	82.1	83.9	77.7	81.5	84.2	77.7	81.9
0.7	84.0	84.1	86.1	78.8	80.9	86.1	78.8	82.8
0.8	84.6	**86.8**	84.0	71.0	75.3	**86.8**	71.0	80.3
Max	84.6	**86.8**	86.1	81.8	81.5			
Min	84.0	82.1	83.1	71.0	75.3			
Average	84.3	84.4	84.3	77.3	78.8			

than 0.2, and the value of *k*-dimensional should not be more than 6 and should not be less than 2.

Result and Discussion for Experiment on Retrieval of Inverted Files Indexing Technique

From the experiment conducted, the result of the average percentage for recall, precision and effectiveness measurement for retrieval of Inverted Files indexing techniques were 54.6 percent, 30.8 percent and 64.8 percent respectively as shown in Table 7 below.

Table 7: Recall, Precision and Effectiveness Measurement for Exact Term-Matching Technique

Query #	Recall, R (%)	Precision, P (%)	Effectiveness Measure, E (%), $\beta = 1$
1	30.0	54.5	61.3
2	40.0	12.7	80.7
3	50.0	14.3	77.8
4	80.0	45.7	41.8
5	55.0	39.3	54.2
6	55.0	34.3	57.7
7	80.0	34.8	51.5
8	90.0	23.7	62.5
9	55.0	7.2	87.3
10	35.0	33.3	65.9
11	25.0	50.0	66.7
12	60.0	20.0	70.0
Average	54.6	30.8	64.8

Comparison and Analysis of the Retrieval of Latent Semantic Indexing and Inverted Files Indexing Technique

Table 8 shows the comparison of retrieval result between exact term-matching and cosine similarity technique. Exact term-matching technique retrieved 64.8 percent in average of its retrieval effectiveness, and Latent

Table 8: Comparison of Retrieval Result between Exact-Matching and Cosine Similarity

#	Exact Term-Matching Technique (Inverted Files) %			Cosine Similarity (LSI) %		
	Recall	Precision	Effectiveness	Recall	Precision	Effectiveness
1	30.0	54.5	61.3	10.0	6.7	92.0
2	40.0	12.7	80.7	95.0	11.0	80.3
3	50.0	14.3	77.8	85.0	9.7	82.6
4	80.0	45.7	41.8	55.0	10.3	82.6
5	55.0	39.3	54.2	45.0	10.1	83.5
6	55.0	34.3	57.7	75.0	4.6	91.3
7	80.0	34.8	51.5	80.0	5.0	90.6
8	90.0	23.7	62.5	10.0	0.9	98.3
9	55.0	7.2	87.3	50.0	8.3	85.8
10	35.0	33.3	65.9	10.0	50.0	83.3
11	25.0	50.0	66.7	15.0	37.5	78.6
12	60.0	20.0	70.0	60.0	4.2	92.1
Average	54.6	30.8	64.8	49.2	13.2	86.8

Semantic Indexing technique that uses cosine similarity had retrieved 86.8 percent in average of the retrieval effectiveness. This result indicates that LSI performed better compared to exact term-matching technique which is used in Inverted Files Indexing technique. Hence, for overall retrieval effectiveness, it signifies that the performance of retrieval method used in LSI technique is about 34 percent better compared to retrieval method used in Inverted Files technique. Therefore, it shows that LSI technique has outperformed the Inverted Files technique in terms of the retrieval effectiveness.

Analysis of the Similarity in Latent Semantic Indexing Technique

In LSI technique, the relevance documents are searched based on the distance of the term-to-term, term-to-document, and document-to-document. If the term is near to the document based on the LSI space context calculated by the Singular Value Decomposition (SVD) computation, the document is considered relevant to the term. For

Table 9: Relevant Document in Collection

Document's Name	Common Word in the Content
H0096BJ1	wuduk, hadas
H0097BJ1	Wuduk
H0099BJ1	Wuduk
H0100BJ1	Wuduk
H0109BJ1	Suci
H0110BJ1	Suci
H0111BJ1	Suci
H0112BJ1	Suci
H0113BJ1	Suci
H0115BJ1	Suci

example, if the cosine similarity of term-to-document is more than 0.79, they are considered relevant although no similarity of term found on that document.

For example query #2 ("*Bersuci dari hadas sebelum mengerjakan solat*") had retrieved not only documents that contain the term identical to the query but also the relevant document in which there are no identical term can be found in those documents. Table 9 illustrates the name of relevant documents retrieved and the common terms that can be found in those documents.

In exact term-matching technique, the relevant documents that are supposedly to be retrieved are only documents that contain terms are that identical to the query #2 which are H0096BJ1, H0109BJ1, H0110BJ1, H0111BJ1, H0112BJ1, H0113BJ1 and H0115BJ1. However, in LSI technique, the relevant documents retrieved include the documents that are conceptually similar or related to the query #2 such as H0097BJ1, H0099BJ1 and H0100BJ1. These relevant documents were retrieved although no identical terms were found because the co-occurrences of the term "wuduk" can be found in document H0096BJ1.

Should the relationship of term co-occurrences between "*wuduk*" and "*hadas*" be represented using theoretical set, Figure 1 illustrates the associations of the terms and documents by operator 'OR' or 'UNION'. Meanwhile, Figure 2 illustrates the representation of terms and documents in LSI space. For ease of illustration, the representation of the vector has been reduced to 2-dimensional space, whereby in this study the

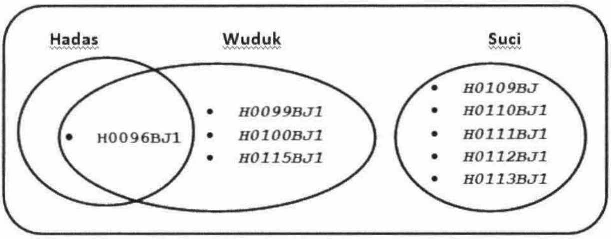

Figure 1: Representation of Terms Co-occurrences from Separated Document

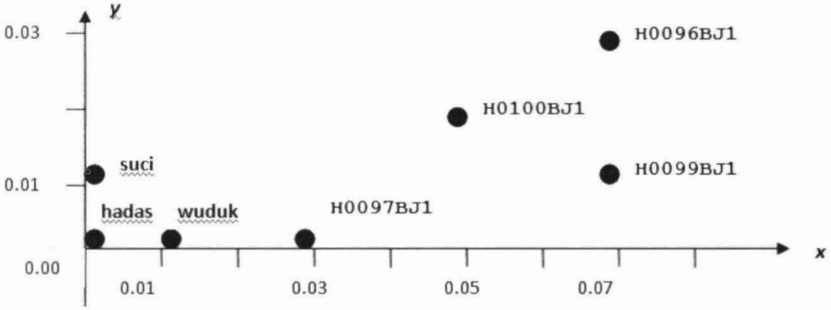

Figure 2: Representation of Term and Document in LSI
Vector Space (2-dimensional space)

number of *k*-dimensional used for effective retrieval is set to k=3 and the nearest of cosine similarity selected is $\varepsilon = 0.8$.

Conclusion and Recommendation

This study has demonstrated that the retrieval method of Latent Semantic Indexing technique performed better compared to the retrieval method that used in Inverted Files technique. Furthermore, this study has given significance to the other researchers for selecting the appropriate technique in developing the information retrieval application and search engine in order to satisfy user information need rather than to satisfy the given query. However the computation of Singular Value Decomposition (SVD) for LSI incurred much time of processing and required high exceptional performance of processor and great capacity of computer's memory. Therefore, to enhance the result of the study, a sufficient performance of computer's processor and memory should be employed and the number of test collection such as formulated user's query and the number of document in test collection should be large enough.

References

Ahmad, F. (1995). *A Malay Language Document Retrieval System and Experimental Approach and Analysis*. PhD Thesis. Universiti Kebangsaan Malaysia.

Chunqiang Tang, Sandhya Dwarkadas & Zhichen Xu, S. (2004). *On scaling latent semantic indexing for large peer-to-peer systems*. Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, 112-121. Retrieved from ACM database.

Czerski, D., Ciesielski, K., Dramiñski, M., K³opotek, M. & Czerski, W.S. (2007). *Inverted Lists Compression Using Contextual Information*. Advances in Information Processing and Protection, 1: 55-66. Retrieved from Springer, USA.

Dasgupta, A., Kumar, R., Raghavan, P.M. & Tomkins, A. (2005). *Variable latent semantic indexing. Conference on Knowledge Discovery in Data*. Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining, 13-21. Retrieved from ACM database.

Ding, Q. H. (1999). *A similarity-based probability model for latent semantic indexing*. Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval. Retrieved from ACM database.

Furnas, W. G., Deerwester, S., Dumais, S. T., Landauer, T. K., Harshman, R. A., Streeter, L. A. & Lochbaum, K. E. (1988). *Information retrieval using a singular value decomposition model of latent semantic structure*. Proceedings of the 11th annual international ACM SIGIR conference on Research and development in information retrieval, 465-480. Retrieved from ACM database.

Gee, R. K. (2003). *Using latent semantic indexing to filter spam.* Symposium on Applied Computing. Proceedings of the 2003 ACM symposium on Applied computing, 460-464. Retrieved from ACM database.

Hofmann, T. (1999). *Probabilistic latent semantic indexing.* Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, 50-57. Retrieved from ACM database.

Inien, S., Lang, D.S. & Deo, N. (1996). *Incorporating latent semantic indexing into a neural network model for information retrieval.* Proceedings of the fifth international conference on Information and Knowledge Management, 145-153. Retrieved from ACM database.

Kowalski, G. (2008). *Cataloging and Indexing.* The Information Retrieval Series, Book of Information Retrieval Systems, 8: 51-69. Retrieved from Springer database.

Launder, T.K., Folts, P.W. & Laham, D. (1998). *An Introduction to Latent Semantic Analysis.* Retrieved November 13, 2008. http://Isa.colorado.edu/papers/dp1.LSAintro.pdf.

Marcus, A., Jonathan, I. & Maletic, I. J. (2003). *Recovering documentation-to-source-code traceability links using latent semantic indexing.* Proceedings of the 25th International Conference on Software Engineering, 125-135. Retrieved from IEEE Computer Society database.

Marin, M. & Gil-Costa, V. (2007). *High-Performance Distributed Inverted Files.* Proceedings of the sixteenth ACM conference on Conference on information and knowledge management, Lisbon, Portugal, 935-938. Retrieved from ACM database.

Navarro, G., de Moura, S.E., Neubert, M., Ziviani, N. & Yates, B. R (2004). *Adding Compression to Block Addressing Inverted Indexes.* Information Retrieval, 3(1): 49-77. Retrieved from Springer database.

Puglisi, J.S., Smyth, F.W. & Turpin, A. (2006). *Inverted Files Versus Suffix Arrays for Locating Patterns in Primary Memory*. Lecture Notes in Computer Science, 4209: 122-133. Retrieved from Springer database.

Shaporenkov, D. (2005). *Efficient Main-Memory Algorithms for Set Containment Join Using Inverted Lists*. Lecture Notes in Computer Science, 3631: 139-152. Retrieved from Springer database.

Trotman, A. (2004). *Compressing Inverted Files*. Information Retrieval, 6(1): 5-19. Retrieved from Springer database.

Vo, N. A. & Moffat, A. (2004). *Inverted Index Compression Using Word-Aligned Binary Codes*. Information Retrieval, 8(1): 151-166. Retrieved from Springer database.

Wensi, X., Sornil, O., Ming, L. & Fox, A. E. (2002). *Hybrid Partition Inverted Files Experimental Validation*. Lecture Notes in Computer Science, 2458: 101-116. Retrieved from Springer database.

Yates, B.R. & Neto, R. B. (1999). *Modern Information Retieval*. Lecture Notes in Computer Science, 1. Retrieved from ACM database.

Zainuddin, H., Fachruddin, Nasaruddin, T, Johar, A. & Abdul Rahman, Z. (2002). *Terjemahan Hadith Shahih Bukhari, Jilid I, II, III, IV*. Darul Fajar Publishing House, Singapore.