Enhanced Latent Semantic Indexing Using Cosine Similarity Measures for Medical Application

Fawaz Al-Anzi[1] and Dia AbuZeina[2]
[1]Department of Computer Engineering, Kuwait University, Kuwait
[2]Computer Science Department, Palestine Polytechnic University, Palestine

Abstract: *The Vector Space Model (VSM) is widely used in data mining and Information Retrieval (IR) systems as a common document representation model. However, there are some challenges to this technique such as high dimensional space and semantic looseness of the representation. Consequently, the Latent Semantic Indexing (LSI) was suggested to reduce the feature dimensions and to generate semantic rich features that can represent conceptual term-document associations. In fact, LSI has been effectively employed in search engines and many other Natural Language Processing (NLP) applications. Researchers thereby promote endless effort seeking for better performance. In this paper, we propose an innovative method that can be used in search engines to find better matched contents of the retrieving documents. The proposed method introduces a new extension for the LSI technique based on the cosine similarity measures. The performance evaluation was carried out using an Arabic language data collection that contains 800 medical related documents, with more than 47,222 unique words. The proposed method was assessed using a small testing set that contains five medical keywords. The results show that the performance of the proposed method is superior when compared to the standard LSI.*

Keywords: *Arabic Text, Latent Semantic Indexing, Search Engine, Dimensionality Reduction, Text Classification.*

Received December 25, 2018; accepted January 28, 2020
https://doi.org/10.34028/iajit/17/5/7

1. Introduction

As a result of intense development in the presence of online data, search engines noticeably play a title role in Information Retrieval (IR) and web data mining applications. The web is primary source to a plethora of open data. Such a massive data resource certainly entails effective algorithms to retrieve and clean out textual or other kinds of data. Therefore, search engines are suited to become to be developed intelligently in order to obtain the desired content. Textual data can be broadly represented using the Vector Space Model (VSM), wherein each document is represented using a vector of attributes, many of which could be nil. However, VSM faces some challenges such as there are extremely long features and a tendency to observe vague semantic representation. Hence, Latent Semantic Indexing (LSI) method is suggested to ease such encounters and to optimistically enhance the presentation. LSI aims at converting the original textual vectors into conceptual vectors that are written off by two properties: reduced dimensions and semantic rich features. The determining feature of the LSI can be attributed to the semantic property, carried out by returning semantically close documents without the restriction to match the exact search keywords.

LSI is based on a proposition from linear algebra named Singular Valued Decomposition (SVD). SVD can transform the textual data, represented as a large term-by-document matrix, into a lesser semantic space characterized as three matrices. The product of the thus created matrices must be equivalent to the original term-by-document matrix. Therefore, the primary step of LSI is to decompose the term-by-document (A) matrix as follows: $A=USV^T$ where U is a matrix that gives the weights of terms, S makes available the eigenvalues for each principal component direction, and V^T is a matrix that offers the weights of documents. V^T matrix consists of the document feature vectors that are normally used in IR and text mining. Figure 1 shows the decomposition procedure that truncates a term-by-document matrix (A) into the three matrices.

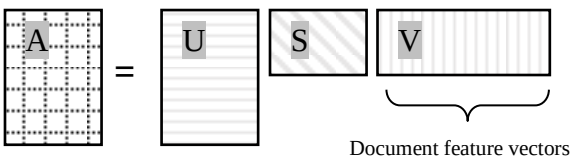


Figure 1. SVD decomposition process.

Accordingly, a standard process of creating LSI starts with using a term-by-document matrix to generate the required feature vectors necessary for classification. Nevertheless, a term-by-document matrix is generally formed utilizing different values such as Boolean flags, counts, or weights. This is employed to track the required term occurrence in documents. For classification, the produced LSI feature vectors are developed using measures similar to the

likes of Euclidian, Mahalanobis, Manhattan, cosine similarity, etc., the cosine similarity measure is known to be a popular distance measure in pattern recognition. In this research, we propose an extension of the LSI implementation by using cosine measures as a replacement to the standard word co-occurrence values generally used in text classification processes. Hence, the proposed method suggests creating a new term-by-document matrix using the cosine measures between documents before engaging the SVD process.

In the succeeding section presents the motivation, followed by the literature review in section 3. In section 4, we present the proposed method followed by the experimental results in section 5. We conclude in section 6.

2. Motivations

Text mining systems are intuitively in need of efficient algorithms that intelligently understand the search engine's documents, as well as the query's keywords. Moreover, whilst considering huge online data it is required to bear in mind semantic relationships between both the documents and the words (also called co-occurrences). Unlike trivial searching methods that are based on traditional text matching, LSI is characterized by semantic rich values, enabling the system to return useful results without having exact match between the document and the query's keywords. For example, if we search for the word "coffee," it is expected that the system will return many documents related to this word, however, it might return other related documents that have no "coffee" word in it, but are semantically related to the word "coffee." Specifically, it is possible to obtain documents that belong to the topic, such as stimulant effects, caffeine, etc., Figure 2 shows an example of an article that contains the word "coffee." It also indicates that the document has other words such as "nervous system." Therefore, the search process for the word "coffee" might return documents related to the topic "nervous system" that might not have the word "coffee" in it. This is the strength of the LSI method and one reason for its popularity.

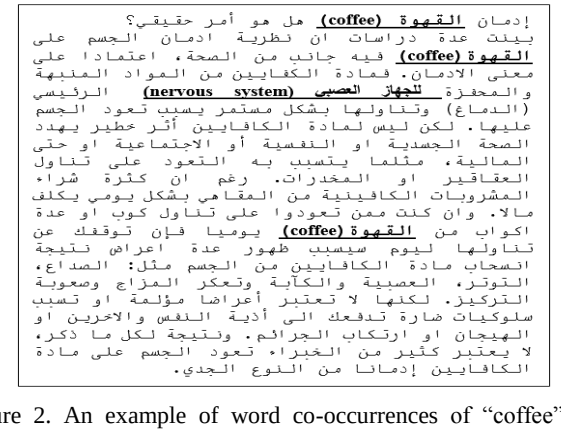
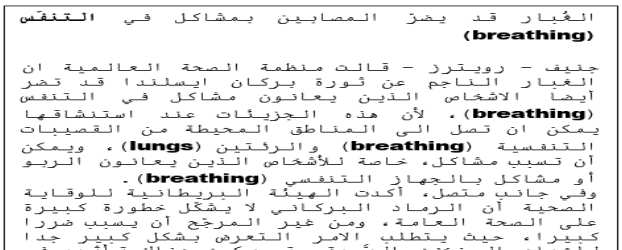

إدمان **القهوة (coffee)** هل هو أمر حقيقي؟
بينت عدة دراسات ان نظرية ادمان الجسم على **القهوة (coffee)** فيه جانب من الصحة، اعتمادا على معنى الادمان. فمادة الكفايين من المواد المنبهة والمحفزة **للجهاز العصبي (nervous system)** الرئيسي (الدماغ) وتناولها بشكل مستمر يسبب تعود الجسم عليها. لكن ليس لمادة الكافايين أثر خطير يهدد الصحة الجسدية او النفسية أو الاجتماعية او حتى المالية، مثلما يتسبب به التعود على تناول العقاقير او المخدرات. رغم ان كثرة شراء المشروبات الكافينية من المقاهي بشكل يومي يكلف مالا. وان كنت ممن تعودوا على تناول كوب او عدة اكواب من **القهوة (coffee)** يوميا فإن توقفك عن تناولها ليوم سيسبب ظهور عدة اعراض نتيجة انسحاب مادة الكافايين من الجسم مثل: الصداع، التوتر، العصبية والكآبة وتعكر المزاج وصعوبة التركيز. لكنها لا تعتبر أعراضا مؤلمة او تسبب سلوكيات ضارة تدفعك الى أذية النفس والاخرين او الهيجان او ارتكاب الجرائم. ونتيجة لكل ما ذكر، لا يعتبر كثير من الخبراء تعود الجسم على مادة الكافايين إدمانا من النوع الجدي.

Figure 2. An example of word co-occurrences of "coffee" and "nervous system".

Using the enhancing search process with thorough overwhelming digital data requires an endless research effort to satisfy the users' requests. In fact, text mining is a challenging task since documents usually have mixed contents that make it difficult to digitally understand the document's category. For an illustration, Figure 2 shows a document that has different words, such as "headaches": "صداع", "addiction": "ادمان", "drugs": "مخدرات", "tension": "التوتر", "frenzy": "الهيجان", and "crimes": "الجرائم". Such diverse words make it vague for IR algorithms to accurately search for the required data. By nature, medical documents require precise algorithms that can adequately find the proper results for the user.

The LSI has been proven to be a valuable tool that reveals the semantical relationship between data objects (i.e., the words in this research). Based on underlying semantic distinctions detected, LSI is able to bring out the relevant documents that may not contain the searched keyword. Figure 3 shows a medical article that is related to "التنفس": "breathing." If a user searches for the word "الاوكسجين": "oxygen," then semantic loss methods (i.e., plain keyword search) will fail since as there is no exact match between the search word and the document's words. However, LSI does support semantic search, and this might be the goal of the user looking for a particular topic.


الغُبار قد يضرّ المصابين بمشاكل في **التنفّس (breathing)**

جنيف – رويترز – قالت منظمة الصحة العالمية ان الغبار الناجم عن ثورة بركان ايسلندا قد تضر ايضا الاشخاص الذين يعانون مشاكل في التنفس **(breathing)**، لأن هذه الجزيئات عند استنشاقها يمكن ان تصل الى المناطق المحيطة من القصيبات التنفسية **(breathing)** والرئتين **(lungs)**، ويمكن ان تسبب مشاكل. خاصة للأشخاص الذين يعانون الربو أو مشاكل بالجهاز التنفسي **(breathing)**.
وفي جانب متصل، أكدت الهيئة البريطانية للوقاية الصحية أن الرماد البركاني لا يشكّل خطورة كبيرة على الصحة العامة، ومن غير المرجّح أن يسبب ضررا كبيرا، حيث يتطلب الامر التعرض بشكل كبير جدا للغبار المنخفض السُمية حتى يكون هناك تأثير في الناس.
وقال كين دونالدسون أستاذ علم السموم التنفسية **(breathing)** في جامعة أدنبره لـــ «رويترز»: «هناك تأثير ضعيف بشكل كبير في الغلاف الجوي، حيث يتشتت بفعل الرياح، ما يعني أن الكمية التي تصل إلى الأرض صغيرة للغاية». واتفق دونالدسون على ان الناس المصابين بأمراض بالرئة **(lung)** بالفعل يجب ان يبقوا في اماكن مغلقة إذا كان هناك تغيير ملموس في مستويات الجسيمات.

Figure 3. An example of "breathing" in a medical document.

3. Literature Review

In the literature, there are many studies that discuss the LSI technique. LSI is used for the text mining tasks, such as text classification, text summarization, text clustering, search engines, etc., LSI initially was presented by Deerwester in [12] as a standard dimension reduction technique in IR. Osinski and Weiss [23] presents an algorithm to enhance the results of search engines. The algorithm combines common phrase discovery and LSI techniques to separate search results into meaningful groups. Letsche and Berry [20] presents a new implementation of the standard LSI; aiming to provide efficient, extensible, portable, and maintainable LSI. Kontostathis and Pottenger [18] presents a theoretical model for understanding the

performance of LSI in retrieval applications. Dumais *et al*. [14] presents an LSI based method for fully automated cross-language document retrieval in which no query translation is required.

Bellegarda *et al*. [9] describes a word clustering approach that is based on LSI. Liu *et al*. [21] proposes a local LSI method called "local relevancy weighted LSI" to improve text classification by performing a separate SVD on the transformed local region of each class. Homayouni *et al*. [16] uses LSI to automatically identify the conceptual gene relationships from titles and abstracts in a database citation. Beebe and Clark [7] proposes and empirically tests the feasibility and utility of post-retrieval clustering of digital forensic text string search results-specifically by using Kohonen Self-Organizing Maps (SOM) as a self-organizing neural network approach. Inouye and Kalita [17] proposes a hybrid Term Frequency-Inverse Document Frequency (TF-IDF) that is based on algorithm and a clustering-based algorithm for obtaining multi-post summaries of Twitter posts along with the detailed analysis of Twitter post domain. Maletic and Valluri [22] uses LSI for automatic software clustering. LSI was used as the basis to cluster software components, source code, and its accompanying documentation. Yeh *et al*. [28] proposes two text summarization approaches: the Modified Corpus-Based Approach (MCBA) and the LSI-based approach.

LSI has been widely documented as a retrieval method that employs SVD for semantic rich reduced feature vectors. Nevertheless, utilizing LSI and SVD requires understanding which values in the reduced dimensional space contain the word relationships (latent semantic) information. Hence, many studies in the literature have discussed this important aspect. Bradford [10] presents an empirical study of the required dimensionality for large-scale LSI applications. Kontostathis [19] was developed as a model for understanding which values in the reduced dimensional space contain the term relationship (latent semantic) information.

Regarding cosine similarity, it is a well-known similarity measure that has been widely mentioned in the literature. Elberrichi *et al*. [15] indicates that cosine similarity dominants have similar measures in IR and text classification. This measure is based on the cosine of the angle between two vectors. Beil *et al*. [8] demonstrates that the similarity between two documents can be measured using the cosine of the angle between the two document feature vectors, which are represented by using VSM. Theodoridis and Koutroumbas [27] defines the cosine similarity measure as: $Scosine(x,y) = x^T y/\|x\|\|y\|$ where $\|x\|$ and $\|y\|$ are the lengths of the vectors x and y, respectively. Tata and Patel [26] proposes that the cosine similarity is a robust metric for scoring the similarity between two strings. Chattamvelli [11] demonstrates that the cosine similarity is used to find the vectors neighborhood. Dhillon and Modha [13] demonstrates that the cosine similarity is easy to interpret and simple to compute for sparse vectors, this indicates that it is widely used in text mining and IR. Sobh *et al*. [24] used the cosine similarity measure for the Arabic language text summarization. Takçı and Güngör [25] uses the cosine measure for the language identification problem.

The literature shows many other applications that use LSI technique such as words clustering, software clustering, document summarization, documents clustering, information retrieval, sentiment analysis, indexing methods, and software comprehension. The literature also shows recent studies that use LSI for Arabic. For instance [4, 5, 6] a used the standard LSI technique for Arabic text classification. In this work, the classification process employs words as units for classification, however, recent studies indicate that two consecutive characters can be used in classification process, [2, 3]. AbuZeina and Al-Anzi [1] discusses Fisher discriminant analysis for Arabic text classification.

4. The Proposed Method

The standard LSI generally starts with the pre processing step, which is performed by declaring stop words and ignoring the characters' list. In addition, all small words less than two, three, and/or four characters in length were discarded. A normalization process was performed to change some Arabic characters such as (إ→ا) and (ٱ→ا). The proposed method is shown in Figure 4, as the term-by-document (A) matrix was created using the unique words from the used corpus. The term-by-document (A) matrix was weighted using TF-IDF. In order to compare the proposed method and the standard LSI, matrix A was decomposed into three matrices (U: Term by dimension; S: Singular values; and V^T: Document by dimension). The diagonal of matrix S contains singular values to enable one to choose the desired reduced dimensions. In general, not all singular values were considered. In this scenario, only the most important values were taken into consideration, starting from the first singular values up to the desired value (k).

An extension of the proposed standard LSI can be observed by creating a new matrix called the cosine similarity matrix. This new matrix uses the cosine similarities between all documents in the corpus instead of co-occurrences (i.e., instead of the frequency of a word in a document). Co-occurrences are usually used when creating term-by-document matrices. Hence, the enhanced method is summarized by using four main steps as follows:

1. Creating the standard term-by-document (A) matrix using word co-occurrences.
2. Weighing the matrix (A) using TF-IDF.

3. Forming the new matrix based on the standard term by-document (A) matrix, called the cosine similarity matrix containing cosine measures between each two vectors in the standard term-by-document matrix (A).
4. Using the SVD to truncate the cosine similarity matrix generating the enhanced feature vectors that are used in the search engine. Of course, different singular values (k) might be investigated to find the optimal performance.

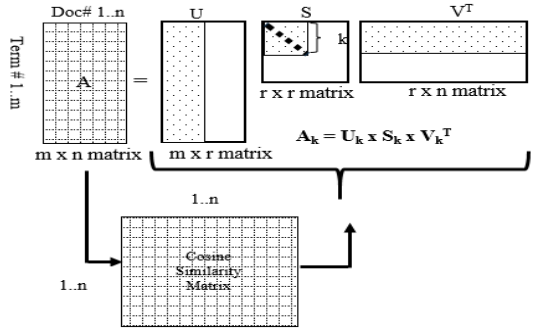


Figure 4. Forming cosine similarities matrix after SVD.

Figure 5 demonstrates how to create a cosine similarity matrix for three documents. The diagonal entries contain 1.0 as the cosine of angle zero, which is 1.0 (i.e., the document itself). Hence, as our corpus contains 800 documents, the cosine similarity matrix of the used corpus is of the size 800×800. Of course, the cosine similarity matrix is a symmetrical matrix.

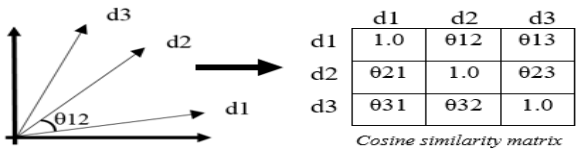


Figure 5. A matrix of cosine similarities for three vectors.

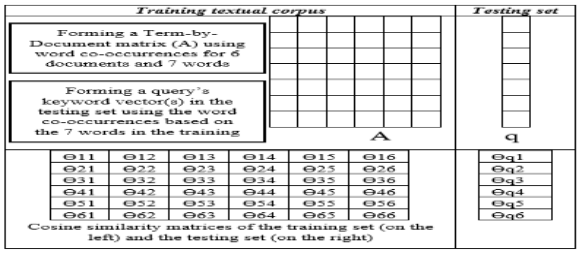


Figure 6. An example of creating cosine similarity matrices.

In both cases, the standard LSI or the proposed method, the query's keywords must transfer to the LSI space. For the standard LSI, the query's feature vectors transform into the new reduced space called "foldingin." This is done by using the following formula: $V^{'T}=AUS^{-1}$. Hence, $V^{'T}$ contains the reduced query's feature vectors that are used along with V^T in the classification process. For the proposed method, the query's feature vectors have two transformation steps. The first pertains to the cosine measures against all feature vectors of training documents before using the folding-in technique as a second step (i.e., like standard LSI, but for the cosine measures instead of word co-occurrences). Figure 6 shows how to generate the query's vector in terms of cosine similarity. Hence, the cosine similarity matrices of the training and the testing set are generated for the new SVD implementation.

5. The Experimental Results

The proposed method was evaluated using an Arabic textual corpus containing 800 documents, 353,888 words, and 47,222 unique words. The data collection refers to medical stories obtained from Alqabas newspaper from Kuwait. A testing set containing five medical keywords were used as queries for the developed search engine. Hence, the testing set arbitrarily contained {"الزهايمر": "Alzheimer", "فيروس": "virus", "الاوكسجين": "oxygen", "القهوة": "coffee", "اشعة": "rays"}. However, a query may have more than one word associated (i.e., a sentence of many words or an article). Table 1 shows more information regarding the testing set and its appearance in the training corpus. Table 1 shows that the word "القهوة": "coffee"appeared 143 times in 36 different documents.

Table 1. Testing set information.

Query word	Total appearance	Total documents
"الزهايمر" : "Alzheimer"	32	14
"فيروس" : "virus"	204	66
"الاوكسجين" : "oxygen"	55	36
"القهوة" : "coffee"	143	36
"اشعة": "rays"	324	103

Since the number of singular values is important in LSI applications, we considered a wide range of singular values to measure the performance for both techniques (i.e., standard LSI and the proposed method). As a result, the search engine was evaluated using the different number of feature vector dimensions (k). That is, a series of experiments were performed using the following k: {k=10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 150, 200, 250, 300, 350, 400, 500}. At each singular value, we analyzed the top-20 retrieved documents to investigate the query's keyword occurrences.

Table 2 shows the performance of the word "الزهايمر": "alzheimer." In the table, the first row indicates the medical query keyword first in the testing set. The first column indicates the singular values k that starts, as previously indicated, at 10 and ends at 500. At k=10, the word "الزهايمر": "Alzheimer" is found using the standard LSI; zero times in the first document as opposed to three times in the top-20 retrieved document. On the other hand, it was found one time in the first document and four times in the top-20 retrieved documents using the proposed method. The results show that the retrieved document using the proposed method is of a high quality compared to the standard LSI, even when confronted with lower dimensions. For instance, at k=80, the

standard LSI retrieved a document that contained 1 occurrence of the searched word with 11 occurrences in the top-20 documents, while the proposed method returned a document that contained 6 occurrences with 20 occurrences in the top-20 documents. Table 2 also shows that the maximum occurrences of the word "الزهايمر": "alzheimer" is 27 times using standard LSI, however, it scored 29 occurrences via the proposed method. The results presented in Table 2 did not require the exact match cases as we considered the word "الزهايمر": "alzheimer" to be the same as saying "the alzheimer" "زهايمر" and "بالزهايمر", etc. Hence, different variations of the same word were counted.

Table 2. Searching results with different k of "alzheimer" word.

"الزهايمر" : "Alzheimer"				
	The Standard LSI		The Proposed Method	
k	First document	Top-20 documents	First document	Top-20 documents
10	0	3	1	4
20	0	6	6	13
30	0	7	6	13
40	0	10	6	18
50	1	3	6	18
60	1	7	6	18
70	1	11	6	18
80	1	11	6	20
90	1	7	6	16
100	1	11	6	16
150	1	19	6	21
200	1	21	6	23
250	1	18	6	23
300	1	18	6	25
350	3	24	6	25
400	3	27 (max)	6	26
500	3	25	6	29 (max)

Table 3 shows the performance of the word "فيروس": "virus." Using k=40, the proposed method had 114 occurrences of the searched word while it returned only 106 occurrences at k=150. Hence, with lower dimensions, the proposed method demonstrated better results. The proposed method also gave improved results for the first retrieved document as it had 19 occurrences of the searched word, while standard LSI was not able to retrieve any occurrences.

Table 3. Searching results with different k of "virus".

"فيروس" : "virus"				
	The Standard LSI		The Proposed Method	
k	First document	Top-20 documents	First document	Top-20 documents
10	0	36	19	74
20	14	77	19	81
30	14	57	19	99
40	19	68	15	114 (max)
50	19	74	19	92
60	19	79	15	87
70	19	82	19	91
80	19	94	19	94
90	19	91	19	96
100	19	78	19	96
150	19	106 (max)	19	96
200	19	99	15	95
250	19	91	15	91
300	19	91	15	100
350	19	87	15	101
400	19	89	15	92
500	15	94	15	83

Table 4 shows the performance of the word "الاوكسجين": "oxygen." This word did not appear in the first retrieved document for both the standard LSI and the proposed method. However, for the top-20 list, the proposed method had 29 occurrences of this word while just 17 occurrences were displayed by means of standard LSI.

Table 5 shows that the first document had 60 occurrences (it is a relatively long document) of the word "القهوة": "coffee," while no query words were returned in the first document using standard LSI. It is worthwhile to observe that the standard LSI retrieved this long document at k=90 whereas it was retrieved at k=10 using the proposed method.

Table 4. Searching results with different k of "oxygen".

"الاوكسجين" : "oxygen"				
	The Standard LSI		The Proposed Method	
k	First document	Top-20 documents	First document	Top-20 documents
10	0	2	0	7
20	0	7	0	8
30	2	10	0	11
40	2	11	0	10
50	2	5	6	11
60	2	6	6	13
70	2	10	6	13
80	2	10	2	13
90	2	8	2	13
100	2	9	2	13
150	2	14	2	18
200	2	8	2	14
250	2	9	2	17
300	2	16	2	16
350	2	16	2	19
400	2	17 (max)	2	19
500	2	17	2	26 (max)

Table 5. Searching results with different k of "coffee".

"القهوة" : "coffee"				
	The Standard LSI		The Proposed Method	
k	First document	Top-20 documents	First document	Top-20 documents
10	0	68	60	60
20	8	80	60	88
30	8	84	60	93
40	60	84	60	94
50	8	85	60	105
60	8	85	60	105
70	8	95	60	111
80	8	95	60	110
90	60	105	60	115
100	60	107	60	116
150	60	105	60	119
200	60	111	60	121
250	60	114	60	122 (max)
300	60	118	60	122 (max)
350	60	117	60	122 (max)
400	60	116	60	122 (max)
500	60	120 (max)	60	119

Table 6 shows that the first document returned three occurrences of the word "اشعة": "rays" using both methods. Table 6 also shows that the performance started decreasing after k=200. Therefore, each LSI based application had a range of singular values (k) where it gave the optimal performance.

Table 6. Searching results with different k of "rays".

"اشعة" : "rays"				
	The Standard LSI		The Proposed Method	
k	First document	Top-20 documents	First document	Top-20 documents
10	3	64	3	40
20	3	39	4	108
30	2	34	13	104
40	3	56	4	104
50	10	101	5	105
60	8	104	8	113
70	8	93	8	112
80	8	93	8	112
90	8	100	8	112
100	8	100	8	114
150	8	112 (max)	22	116 (max)
200	8	108	22	111
250	22	110	22	108
300	10	105	22	106
350	10	104	22	102
400	10	105	22	108
500	2	100	22	97

In addition, performance was evaluated by measuring the percentage of the matched words among all occurrences in the training set. For example, the word "اشعة": "rays" appears 324 times in the corpus. The standard LSI showed this word 112 times in the top-20 list as indicated in Table 6. However, the proposed method listed it 116 times. Hence, the percentage for the standard LSI is 112/324=0.346. For the proposed method, the percentage is 116/324=0.358. These percentages are shown in table 7 for the investigated keywords (i.e., the testing set). The table also shows that the average percentages for standard LSI is 0.571 and for the proposed method is 0.629. This means that the proposed method outperforms the standard LSI by 5.83% with regards to the top-20 retrieved documents.

In fact, additional evaluation parameters would also be required. For instance, in this case only the match word is compared, while the semantic quality of the retrieved documents needs to be looked into as well. Figure 7 is the graphical representation of the performance differences between the standard LSI and the proposed method. The graph's information is based on the percentages calculated in Table 7.

Table 7. The percentage of the retrieved searching words in top 20.

#	Word	The Standard LSI	The Proposed Method
1	"الزهايمر" : "Alzheimer"	0.844	0.906
2	"فيروس" : "virus"	0.520	0.559
3	"الاوكسجين" : "oxygen"	0.309	0.473
4	"القهوة" : "coffee"	0.839	0.853
5	"اشعة" : "rays"	0.346	0.358
	Average	**0.571**	**0.629**

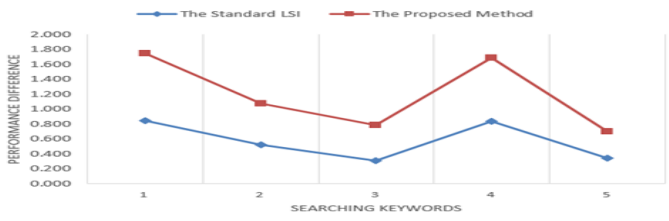


Figure 7. The performance enhancement using the proposed method.

Finally, the proposed method is suitable for relatively small data collections. However, it might not be very efficient for very large corpora that contains millions of documents. Creating the cosine similarity matrix is very complex and requires an extensive amount of time. It costs $O(n^2)$, where n is the total number of documents in the corpus. Nevertheless, this method shows a possible enhancement, especially precise results are necessary for highly mixed contents in medical documents.

The proposed method is indicative that some traditional methods can be mathematically enhanced after carefully investigating its operations. This method serves as an eye opener from an innovative perspective to enhance LSI based applications such as search engines. Of course, the proposed method can be used in other domains like sentiment analysis and automated essay scoring.

Alternatively, a limitation to the proposed is that it requires generating a cosine similarity measures matrix. This step consumes a lot of time and resource; however, with today's rate of developing faster machines and algorithms, it is expected that the proposed method will be implemented in the global search engines that intuitively employ High Performance Computing (HPC).

6. Conclusions

This paper presents a new variant of the LSI technique for search engines. A comprehensive experimental evaluation shows the feasibility of the LSI technique as well as the enhancements of the new method over the standard LSI technique. The results showed that using cosine similarities instead of just word co-occurrences enhances the performance of search engines. The proposed method's top-20 retrieved documents are of a higher quality than the ones retrieved using the standard LSI. In future, the proposed method can be developed for larger data collections. Furthermore, the time and space complexities of the proposed method can also be deeply investigated. Moreover, the evaluation should include semantic quality and not just matched words. As a final statement, the output of this paper promotes further research to find and employ intelligent methods in order to attain added satisfaction of search engines' users.

Acknowledgments

This work is supported by Kuwait University Research Grant Number EO03/18.

References

[1] AbuZeina D. and Al-Anzi F., "Employing Fisher Discriminant Analysis for Arabic Text

Classification," *Computers and Electrical Engineering*, vol. 66, pp. 474-486, 2018.

[2] Abuzeina D., "Exploring Bigram Character Features for Arabic Text Clustering," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 27, no. 4, pp. 3165-3179, 2019.

[3] Al-Anzi F. and AbuZeina D., "A Micro-Word Based Approach for Arabic Sentiment Analysis," *in Proceedings of IEEE/ACS 14th International Conference on Computer Systems and Applications*, Hammamet, pp. 910-914. 2017.

[4] Al-Anzi F. and AbuZeina D., "Big Data Categorization for Arabic Text Using Latent Semantic Indexing and Clustering," *in Proceedings of International Conference on Engineering Technologies and Big Data Analytics*, Bangkok, pp. 1-4, 2016.

[5] Al-Anzi F. and AbuZeina D., "Toward an Enhanced arabic Text Classification Using Cosine Similarity and Latent Semantic Indexing," *Journal of King Saud University Computer and Information Sciences*, vol. 29, no. 2, pp. 189-195, 2017.

[6] Al-Anzi F., AbuZeina D., and Hasan S., "Utilizing Standard Deviation in Text Classification Weighting Schemes," *The International Journal of Innovative Computing, Information and Control*, vol. 13, no. 4, pp. 1349-4198, 2017.

[7] Beebe N. and Clark J., "Digital Forensic Text String Searching: Improving Information Retrieval Effectiveness by Thematically Clustering Search Results," *Digital Investigation*, vol. 4, pp. 49-54, 2007.

[8] Beil F., Ester M., and Xu X., "Frequent Term-Based Text Clustering," *in Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Edmonton, pp. 436-442, 2002.

[9] Bellegarda J., Butzberger J., Chow Y., Coccaro N., and Naik D., "A Novel Word Clustering Algorithm Based on Latent Semantic Analysis," *in Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, Atlanta, pp. 172-175, 1996.

[10] Bradford R., "An Empirical Study of Required Dimensionality for Large-Scale Latent Semantic Indexing Applications," *in Proceedings of the 17th ACM conference on Information and Knowledge Management*, Napa Valley, pp. 153-162, 2008.

[11] Chattamvelli R., *Data Mining Algorithms*, Alpha Science International Ltd, 2011.

[12] Deerwester S., Dumais S., Furnas G., Landauer T., and Harshman R., "Indexing by Latent Semantic Analysis," *Journal of the American Society for Information Science*, vol. 41, no. 6, pp. 391-407, 1990.

[13] Dhillon I. and Modha D., "Concept Decompositions for Large Sparse Text Data Using Clustering," *Machine Learning*, vol. 42, no. 1-2, pp. 143-175, 2001.

[14] Dumais S., Letsche T., Littman M., and Landauer T., "Automatic Cross-Language Retrieval Using Latent Semantic Indexing," *in AAAI Spring Symposium on Cross-Language Text and Speech Retrieval*, vol. 15, pp. 15-21, 1997.

[15] Elberrichi Z., Rahmoun A., and Bentaalah M., "Using WordNet for Text Categorization," *The International Arab Journal of Information Technology*, vol. 5, no. 1, pp. 16-24, 2008.

[16] Homayouni R., Heinrich K., Wei L., and Berry M., "Gene Clustering by Latent Semantic Indexing of MEDLINE Abstracts," *Bioinformatics*, vol. 21, no. 1, pp. 104-115, 2005.

[17] Inouye D. and Kalita J., "Comparing Twitter Summarization Algorithms for Multiple Post Summaries," *in Proceedings of IEEE 3th International Conference on Social Computing*, Boston, pp. 298-306, 2011.

[18] Kontostathis A. and Pottenger M., "A framework for understanding Latent Semantic Indexing (LSI) performance," *Information Processing and Management*, vol. 42, no. 1, pp. 56-73, 2006.

[19] Kontostathis A., "Essential Dimensions of Latent Semantic Indexing (LSI)," *in Proceedings of 40th Annual Hawaii International Conference on System Sciences*, Waikoloa, pp. 73-73, 2007.

[20] Letsche T. and Berry M., "Large-Scale Information Retrieval with Latent Semantic Indexing," *Information Sciences*, vol. 100, no. 1, pp. 105-137, 1997.

[21] Liu T., Chen Z., Zhang B., Ma W., and Wu G., "Improving Text Classification Using Local Latent Semantic Indexing," *in Proceedings of IEEE International Conference on Data Mining*, Brighton, pp. 162-169, 2004.

[22] Maletic J. and Valluri N., "Automatic Software Clustering Via Latent Semantic Analysis," *in Proceedings of 14th IEEE International Conference on Automated Software Engineering*, Cocoa Beach, pp. 251-254, 1999.

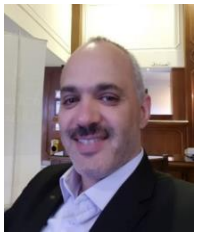
[23] Osinski S. and Weiss D., "A Concept-Driven Algorithm for Clustering Search Results," *IEEE Intelligent Systems*, vol. 20, no. 3, pp. 48-54, 2005.

[24] Sobh I., Darwish N., and Fayek M., "A Trainable Arabic Bayesian Extractive Generic Text Summarizer," *in Proceedings of the 6th Conference on Language Engineering ESLEC*, Egypt, pp. 49-154, 2006.

[25] Takçı H. and Güngör T., "A High Performance Centroid-Based Classification Approach for

Language Identification," *Pattern Recognition Letters*, vol. 33, no. 16, pp. 2077-2084, 2012.
[26] Tata S. and Patel J., "Estimating the Selectivity of Tf-Idf Based Cosine Similarity Predicates," *ACM Sigmod Record*, vol. 36, no. 2, pp. 7-12, 2007.
[27] Theodoridis S. and Koutroumbas K., *Pattern Recognition*, Academic Press, 2008.
[28] Yeh J., Ke H., Yang W., and Meng I., "Text Summarization Using A Trainable Summarizer and Latent Semantic Analysis," *Information Processing and Management*, vol. 41, no. 1, pp. 75-95, 2005.

Fawaz Al-Anzi Professor Al-Anzi received his Ph.D. & M.Sc. in Computer Science from Rensselaer Polytechnic Institute, New York, USA in 1995. He earned his B.Sc. with honors in EE from Kuwait University in 1987. He received the National Research Production Award and Kuwait University Award. He is the founding dean of College of Computing Sciences and Engineering. His research interest includes data science and engineering, text classification and speech recognition.



Dia AbuZeina received his Ph.D. in Computer Science and Engineering from King Fahd University of Petroleum and Minerals, Saudi Arabia, 2011. He received his M.Sc. in information technology from Southern New Hampshire University, Manchester, USA, 2005. He received his B.Sc. in computer system engineering from Palestine Polytechnic University, 2001. His research interest includes speech recognition and text classification for modern standard Arabic (MSA)