Studying the effects of conflicting tokenization on LSA dimension reduction

Mahmoud Fahsi*
Computer Science Departement
EEDIS Laboratory, Djillali Liabes University,
Sidi Bel Abbes, Algeria
mfahci@univ-sba.dz

Sidi Mohamed Benslimane
Computer Science Departement
EEDIS Laboratory, Djillali Liabes University,
Sidi Bel Abbes, Algeria
benslimane@univ-sba.dz

Abstract—**with the growing needs of dimension reduction for term selection and recommendation and the up to date trends in natural language processing modules integrated in existing architectures and multiple semantic web system such as search engine. The existence of multiples tokenization techniques of the same text represents a persistent problem in current semantic search engine practice and create a non-trivial problem the query expansion and their efficiency in general. In this work we try to study the effect of the tokenization technique in context of query expansion terms selection within a statistical latent semantic indexing. Finally we talk about the results from a corpus-linguistic point of view.**

Keywords: Dimension Reduction, LSA, Tokenisation, Query Expansion.

I. INTRODUCTION

We can consider the Semantic information retrieval field as one of the oldest information retrieval research domain started with Lisowsky description of the core challenge and mission of automated text mining [1]. In this description translated detailed by [2], Lisowsky reports that the same concept can be expressed using different terms (synonymy) and on the contrary the apparently same term can have very different meanings in different contexts (polysemy). Starting with these two associations, many other semantic relations can be developed and used to relay terms of subject or concepts. To improve Information Retrieval effectiveness, search engines and text mining systems must manage the context to discover the diverse concept expressing ways and to extract and disambiguate terms. Those systems must also provide that data in a structured way to allow utilization and future extension.

The most popular way to represent documents is Bag of words; it gives an accounting for term's occurrences but ignore the order. From applied mathematics, Vector representation can be used as a bag of words high dimensionality complement to enhance the indexing process as well as dimension reduction [2], used to identify a sub-domain representation of a set of vectors that preserves important properties related to one concept or one topic. Dimension reduction is applied to find out the sub space of conceptual representation and its relations to the terms representation. The new representation in semantic space reveals the main structure of the corpus contents more clearly than the original bag of words representation.

We intend in this work to study the effects of varying tokenization methods on most successful dimension reduction technique like latent semantic indexing as one of many dimension reduction techniques that have been applied starting with the idea that we can improve the results of statistical application by improving inputted data quality reflecting corpus or document. In section two, we will see tokenization methods used to refine text contents and convert it into more significant bag of word considered to feed the LSI dimension reduction in our experiment. In section three, we will illustrate our system implementation and results, a human expert evaluation will be provided and discussed before a conclusion dedicated to explaining the whole assumptions and future perspectives.

II. BACK GROUND

A. Latent Semant Indexing Survey

LSI uses a singular vector decomposition method to find a latent semantic space. An initial terms and documents space is reduced to represent concepts as a substitute of terms. With these changes, LSI wants to remove the noise, extracting the underlying semantic concepts in a document collection [3].

A probabilistic framework for the dimension reduction present an enhancement of LSI model and includes probabilistic latent semantic indexing PLSI [4] and Latent Dirichlet Allocation LDA which is used for topic modelling [5].

Latent semantic indexing (*LSI*) as an automatic indexing method is one of most used information extraction technique in information retrieval systems that reduce documents and terms dimension into a lower space and give a best representation of the semantic concepts in the document.

The first LSI is simply an application of the Latent Semantic Analysis to a textual data for conceptual indexation at the same time as dimension reduction [6]. The only difference between these two applications is that the concept's origins must be from the corpus in dimension reduction but in

978-1-4799-3824-7/14/$31.00 ©2014 IEEE

conceptual indexation, the concept can be extracted from outside resources like anthologies or thesaurus. LSI applications cover many other tasks in the knowledge management field, including assigning submitted papers to reviewers by most representative keywords extraction for each article and matching rules with reviewers interesting domains also extracted from their published papers [7]. Another successful LSI application was cross-lingual retrieval with multilingual corpus dimension reduction [8] and/conceptual indexation [9]. Furthermore, the use of this numerical technique was benefitted in sentence retrieval context with the purpose of extraction of relevant sentences for a query from a set of documents for that query [10].

LSI implementation is based on text corpora as input, each entry xvd can be the occurrence of term v in document d but Zipf's law shows that usually, infrequent terms are likely to occur multiple times in a document if it occurs at all [11]. Therefore, the use of term frequency has a tendency to over stress the contribution of the term and data quality. That's why the global term-weight methods, such as TF/IDF (term frequency weighted with inverse document frequency) [12], provides a good information about the terms/document distribution. The spatial information about term proximity, terms order or frequency occurrence can't be provided by TF/IDF, that's why a multi-resolution matrix representation for documents was introduced by the BOW representations [13].

Other works similar to Dupert [14] studied how to determine the optimal number of latent factors who represent the ideal dimension of the latent semantic space.

B. Latent Semant Indexing Model & Application Level

To reduce dimension, LSI uses the term-document matrix X of a corpus with d documents created essentially on the singular value decomposition *SVD* model. This matrix is built using the term frequency and inverted term frequency metrics. Columns in this case characterize X_d vector of a document *d* in the corpus. The *v* rows of the matrix *X* represent a vector of a term *v* in the corpus. The core contribution within this paper relies on the variation of the encoded terms generally considered as tokens and their effect on the results of the LSI method. Several possibilities for the tokenization will be discussed in the next section.

The singular value decomposition $X = U \Sigma V^T$ of the term-document matrix *X* generate tow orthonormal matrices *U* and *V* and diagonal matrix *Σ(equation 1)*.

$$\Sigma = \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_{\min\{W,M\}} \end{bmatrix}$$

The singular values of the matrix *X* are $\sigma_1, \sigma_2 \ldots \sigma_{min\{W,M\}}$ of the diagonal vector. After supposing that the singular values are arranged in descending order, $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_{min\{W,M\}}$ To continue with the dimension reduction of the matrix *X*, an approximation $\hat{X}$ by a *rank-K* to *X* must be done. This stage can be done with a partial *SVD* using the singular vectors corresponding to the largest singular values *K* expressed in equation (2).

$$\hat{X} = \hat{U}\hat{\Sigma}\hat{V}^T$$

$$= [U_1 \ldots U_K] \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_K \end{bmatrix} \begin{bmatrix} V_1^T \\ \vdots \\ V_K^T \end{bmatrix} \quad (1)$$

The matrix $\hat{X}$ is generally not sparse and can be viewed as a smoothed version of *X*. We can see from Equation (2) that each document *d* is represented by a K-dimensional vector $\hat{X}_d$, which is a row of the matrix $\hat{V}$. Then, The relation between the representation of document *d* in term space X_d and the latent semantic space $\hat{X}$ d is given by:

$$\hat{X}_d = \hat{U}\hat{\Sigma}\hat{X}_d \quad (2)$$

In the same way, each term *v* can be represented by the K-dimensional vector $\hat{T}_v$ given by

$$T_v = \hat{V}\hat{\Sigma}\hat{T}_v \quad (3)$$

Once the equations (3) and (4) considered as a projection of both terms and documents into the K-dimensional latent semantic space. These projections are used for several tasks such Information retrieval To retrieve a need expressed by a query q which contains several keyword terms using an information retrieval system, this one must return all or maximum of the documents that describe the user's information need. To do that, information retrieval system requires to associate a query term's concept to the document's terms concepts. In this case, both the query and the corpus are considered as documents, but the query as a short one projected into the latent semantic space like that:

$$\hat{q} = \hat{\Sigma}^{-1}\hat{U}^T q \quad (4)$$

Using this equation, the similarity between the query and document can be measured.

Another task is Document similarity used for text document classification aims to calculate the similarity between two documents *d* and *d'*. This similarity can be considered by measuring distance between their LSI vector representations (columns) in the latent semantic space [15]. Vector distance can be a local distance if we compare only two documents or global distance if we compare each document with all other one.

All equations above meant for Term similarity to also ***c***omparable to the document similarity, term similarities are measured in the latent semantic space, in order to cluster or

classify terms with similar meanings. This is done to facilitate terms conceptualization, ontology creation and document summarization.

C. *Tokenization*

Before talking about term weighting, SVD implementation or latent semantic space, researchers should focus their efforts on text decomposition into tokens. Tokenization is the process of mapping sequences of characters into sequences of words [12]. However, as it has often been discussed in the literature, the definition of the term 'token' is not at all trivial: it designates units in text that approximate words. Word or sentence boundaries may be indicated by whitespaces or special characters, but neither does each such marker correspond to a boundary, nor is every boundary marked in such a way [16].

Many tokenization strategies (table 1) such as whitespace, TnT, PTB and lexeme tokenization can be used following application context like tagging, parsing or information extraction. This last one focuses on different cleaning levels such as stemming, stop-words suppression and word stemming. Starting with the fact that the initial tokens vary depending on used tokenization strategy, different results can evolve and many conclusions can be perceived.

TABLE 1: ILLUSTRATIVE EXAMPLES OF TOKENIZATION TECHNIQUES.

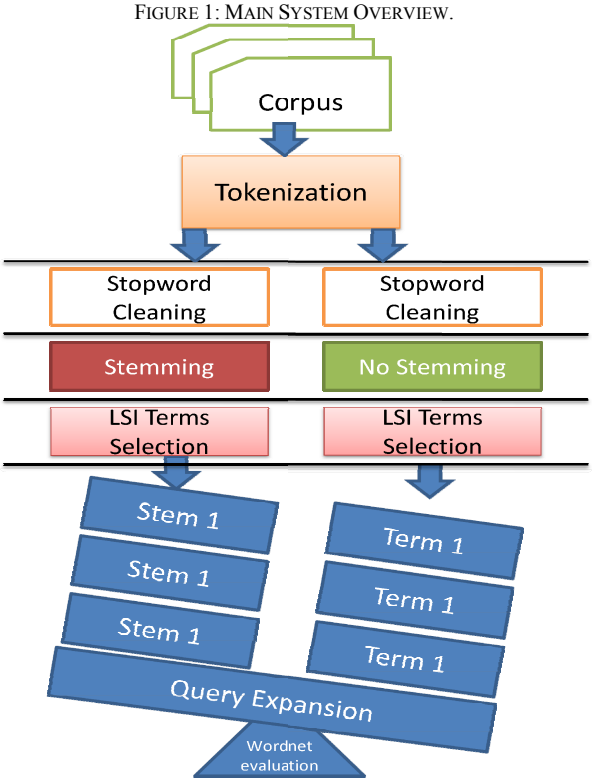
doesn't	
[does][n't]	(Marcus et al. 1993, PTB)
[doesn][']['t]	(Brants 2000, TnT)
the attorney general's office	
[attorney] [general's]	(Burnard 2007, whitespace tokenization)
[attorney] [general][']['s]	(Brants 2000, TnT)
[attorney] [general][']['s]	(Marcus et al. 1993, PTB)
[attorney general][']['s]	(Yamamoto et al. 2003, lexeme tokenization)

After text tokenization, document representation must be cleaned from stop-word terms [12]. This step aims to free the input terms from noise and improve the quality of the results. This cleaning must take in consideration the minimum size of a context to obtain interesting results with LSI [17]. After that all tokens must be Stemmed, that mean mapping words to their lemma by deleting term prefix and/or suffix. Stemming is not about semantic or linguistic purpose like stop words cleaning because stemmer user don't intend by stemming the improvement of text quality that's why stemming is only used to improve information retrieval performance [18]. Neither a prefix stemming such as Lancaster Stemmer can decrease precision when it take on the autonomy relation between token and lemma [19].

III. SYSTEM OVERVIEW

In this work we look forward to confirm that the individual term representation affects directly the results of any statistical technique by implementing LSI dimension reduction based on two points of view. We start by using original tokens resulting from simple text decomposition without cleaning text from stop-words or applying the stemming algorithms. We passed towards the partial application of stemming on stop-words cleaned text. The term frequency vector transformed to the inverted document frequency resulting from each phase will be supplied to the LSI core system responsible of reducing the term space reduction. A comparison between human evaluations of this phase will be discussed later.

FIGURE 1: MAIN SYSTEM OVERVIEW.



All experiments will be done on three (03) sub-collections extracted from the Reuter collection with three different subjects: network, cotton and jobs.

For the first corpus containing documents about "network" we have more than 1676 words. The second corpus talks about "cotton" and contain over 1166 terms. The last one is about "jobs" and contains more than 955 terms.

After the dimension reduction, the system will return a better representation of those documents represented by their ranked terms; those ones will then manually be associated with the query concept by a language domain expert. For example, with the first corpus "network", the user injected query was defined as "Networking Company". By filtering results the expert found that terms like: Cisco, router, business, trading and similar terms are pertinent but in other hand, other terms are totally not related. Then we perform several experiments to find out the effect of lexical representation on the results of LSI reduction dimension reduction.

IV. RESULTS AND DISCUSSION

As shown in figure 1, the principal implemented parts are: a differential tokenizer, a core LSI indexer and a result of terms ranking that should be manually evaluated by experts or combined with the WORDNET exploration result then re-evaluated again.

In this section, we define another task to be performed taking into account larger contexts. Thus, we will conduct one

experiments divided in two stages: classic latent semantic indexing for the terms as the hole token ranked by pertinence, and the second work is a study of terms variations effect on the n first selected. To do this, we will work on thematic sub-selection extracted from the RCV1 Reuter collection explicitly divided into different themes. We propose to work with the "Network Company", "Jobs" and "Cotton" training sub-collection. Each one of them can be considered as a pseudo relevance feedback.

A. LSI model

The result obtained by applying the LSI method on the first them corpus that have as subject: "network" are visible in Table 2 and want to learn out the best terms representing the subject for query "networking company".

As first evaluation, we manually estimate the relativity of the first n returned keywords to our queries with n=10 and n=20.

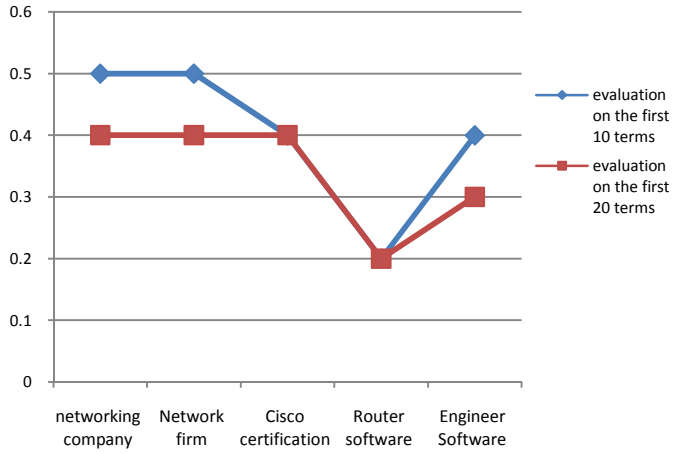


FIGURE 2: EFFECTIVENESS OF RETURNED TERMS IN SPITE OF THE INITIAL QUERIES WITH THE SUBJECT NETWORK COMPANY.

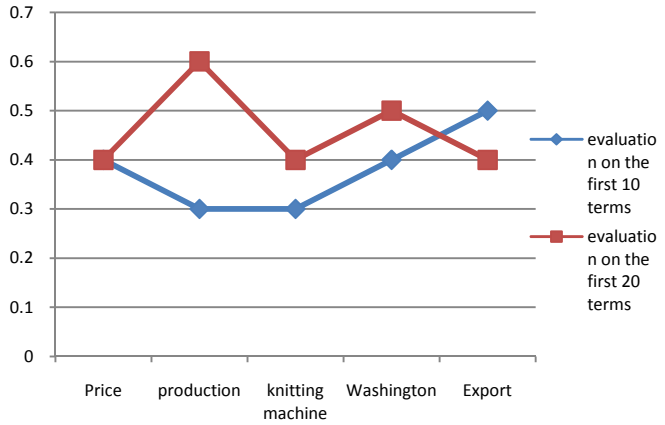


FIGURE 3: EFFECTIVENESS OF RETURNED TERMS IN SPITE OF THE INITIAL QUERIES WITH SUBJECT COTTON.

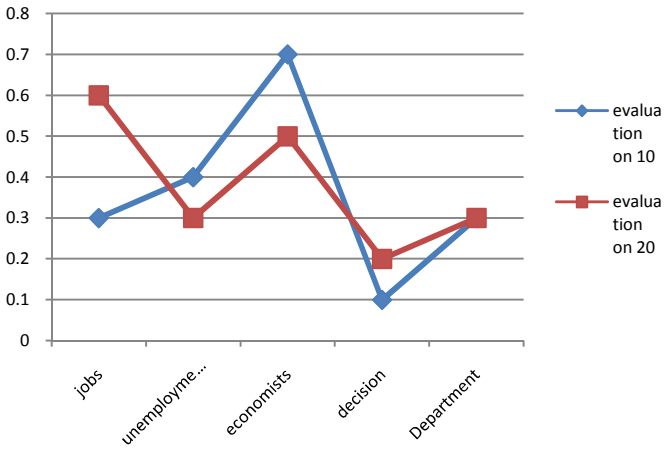


FIGURE 4: EFFECTIVENESS OF RETURNED TERMS IN SPITE OF THE INITIAL QUERIES WITH SUBJECT JOB.

In figures 2, 3 and 4 we clearly conclude that there is some terms that can be a better representation of the main document subject. Otherwise, terms like "decision" and "router soft" are not suited for those collections and will be poor in information because of their irrelevance context even if they occur too match within the text.

B. Stemming variation

Once we get our first results, we applied different stemming techniques starting with separate post and pre-stemming. Unlike the first results, those two methods provide different result expressed in table 2.

TABLE 2: LSI TERM SELECTION PRECISION AND RECALL RESULTS COMPARISON.

LSI aspect	**Terms recall**				**Term Precision**			
	Without stemmer		***With post-stemming***		***Without stemmer***		***With post-stemming***	
representative term number in corpus1	134	07%	260	17%	109	81%	179	68%
	1889		1507		134		260	
representative term number in corpus2	305	23%	296	26%	188	61%	167	56%
	1296		1102		305		296	
representative term number in corpus3	205	19%	203	22%	146	71%	134	66%
	1068		909		205		203	

We distinguish in table 2 that the stemmed terms recall is better than the non-stemmed one because many terms or tokens are changed into one lemma in all collections with the application of the post stemming technique. Otherwise, terms precision decrease by post-stemming that because many unrelated terms took a bigger place in the returned terms even if the number of returned good terms is almost equal.

V. CONCLUSION

The conducted experiments had shown that Latent semantic indexing model is a very effective mechanism for dimension reduction work. It also prove a valuable term selection results within information retrieval query expansion context. But it can be impressively improved by doing a special text preparation before it to adapt the inputted text to the statistical context of LSI such as text cleaning or text expansion. We are persuaded that in query expansion field, we should omit stemming step from text preparation. This will not reduce the possibility that relevant term could be selected by an LSA dimension reduction technique. In another hand, we will take advantage of specialized selected term that represent better the user needs.

In the future, we hope to study the effectiveness of other tokenization techniques to the input terms selection for LSA. For the evaluation of our results we tend to test the application of tokenization variation in context of pseudo relevance feedback to select terms for query expansion and compare results with previous works [20]. In this case, formal evaluation will be done with different information retrieval test collections to situate this work.

VI. REFERENCES

[1] G. Lisowsky and L. Rost. Concordance for hebraischen Old Testament Deutsche Bibelgesellschaft, 1958.

[2] Steven P, Ke Zhou, Shuang-Hong Yang, Hongyuan Zha. Dimensionality reduction and topic modeling: from latent semantic indexing to latent dirichlet allocation and beyond, Mining Text Data.129-161. 2012.

[3] R. Kubota Ando and L. Lee. Iterative residual rescaling: An analysis and generalization of LSI. In ACM-SIGIR, pages 154–162, 2001.

[4] Thomas Hofmann. Probabilistic Latent Semantic. In ACM-SIGIR, 1999.

[5] David M. Blei, Andrew Y. Ng, Michael I. Jordan, John Lafferty. Latent Dirichlet Allocation. JMLR.pages 993-1022, 2003.

[6] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman. Indexing by latent semantic analysis. Journal of the American Society for Information Science, 41(6):391–407, September 1990.

[7] S. Dumais and J. Nielsen. Automating the assignment of submitted manuscripts to reviewers. In SIGIR, pages 233–244, 1992.

[8] F. Jiang, m. L. Littman. Approximate Dimension Reduction. In NTCIR workshop, 2000.

[9] P.l Moravec, M. Kolovrat. LSI vs. Wordnet Ontology in Dimension Reduction for Information Retrieval. In DATESO. pages 67-77. 2004.

[10] David Parapar and Àlvaro Barreiro. Sentence Retrieval with LSI and Topic Identification. ECIR 2006, LNCS 3936, pp. 119–130, 2006.

[11] K. Church and W. Gale. Poisson mixtures. Natural Language Engineering, 1:163–190, 1995.

[12] C. Manning, P. Raghavan, and H. Schutze. Introduction to Information Retrieval. Cambridge University Press, New York, NY, USA, 2008.

[13] S. Yang and H. Zha. Language pyramid and multi-scale text analysis. In CIKM, pages 639–648, 2010.

[14] G. Dupret. Latent concepts and the number orthogonal factors in latent semantic analysis. SIGIR, pages 221–226, 2003.

[15] Q. Wang, J. Xu, and H. Li. Regularized latent semantic indexing. In SIGIR, 2011.

[16] C. Chiarcos, J. Ritz, M. Stede, Merging conflicting tokenizations, Lang Resources & Evaluation. pages 53–742012.

[17] Rehder, B., Schreiner, M. E., Wolfe, B. W., Laham, D., Landauer, T. K., & Kintsch, W. Using Latent Semantic Analysis to assess knowledge: Some technical considerations. Discourse Processes 25, 337-354, 1998.

[18] M.F. Porter. An Algorithm for suffix stripping. Readings in information retrieval. pages 313-316. 1997

[19] Lancaster University. What is Stemming? retrieved February 2013. from www.comp.lancs.ac.uk/computing/research/stemming/

[20] M. Fahsi, A. Lehirech, M. Malki. Using Google Web Service In Genetic Algorithms Query Reformulation. RIST journal. Vol. 18 - N° 1. 2010.