Knowl Inf Syst (2017) 50:751–762
DOI 10.1007/s10115-016-0952-x

REGULAR PAPER

Context-aware query expansion method using Language Models and Latent Semantic Analyses

Btihal El Ghali[1] · **Abderrahim El Qadi**[2]

Received: 19 May 2015 / Revised: 15 February 2016 / Accepted: 19 April 2016 /
Published online: 30 April 2016

Abstract One of the key difficulties for users in information retrieval is to formulate appropriate queries to submit to the search engine. In this paper, we propose an approach to enrich the user's queries by additional context. We used the Language Model to build the query context, which is composed of the most similar queries to the query to expand and their top-ranked documents. Then, we applied a query expansion approach based on the query context and the Latent Semantic Analyses method. Using a web test collection, we tested our approach on short and long queries. We varied the number of recommended queries and the number of expansion terms to specify the appropriate parameters for the proposed approach. Experimental results show that the proposed approach improves the effectiveness of the information retrieval system by 19.23 % for short queries and 52.94 % for long queries according to the retrieval results using the original users' queries.

Keywords Contextual information retrieval · Language Models · Query recommendation · Latent Semantic Analyses · Query expansion

1 Introduction

Most of the existing search engines rely only on the keywords that co-occur between user queries and returned documents. However, a study was made on queries submitted on a search engine [24] and it was observed that users usually formulate very short queries. Thus, the keywords-based systems incapable to retrieve documents responding to the information needs of the search engines' users.

✉ Btihal El Ghali
btihal.elghali@gmail.com

[1] LRIT-CNRST (URAC No. 29), Faculty of Science, Mohammed V University in Rabat, Rabat, Morocco

[2] TIM Team, High School of Technology, Moulay Ismaïl University in Meknes, Meknes, Morocco

Due to the increasing volume of information in the World Wide Web, and the lack of organization in it, the retrieval of relevant documents using the initial user query is becoming an almost impossible task [4]. As queries get longer, there is more possibilities that some important terms co-occur in the query and the relevant documents to it.

Moreover, the query considered separately is insufficient to clearly identify what the user is looking for, because of the fact that the query is only a partial and often ambiguous expression of the user's needs of information [5]. Considering the context around the query, the ambiguity can be resolved to a certain degree and the partial information of the user's needs can be completed.

For a correct interpretation of the user's query, it has been demonstrated that it should be placed in its appropriate context [4]. But as known, the context is a large notion that includes the user context (his domains of interest, his preferences and his historic of research) and the query context, which means the environment of the query (its relevant documents and its terms…). The first context needs the research to be done using user's profiles, but a single profile can group a large variety of domains and interests, which are not always relevant for a particular query. Thus, the solution is to use the environment of the query as an appropriate context to improve the precision of the query.

In this paper, we propose an approach of query expansion based on the context around the query. This approach is divided into two phases:

- In the first phase, we build the query context using the Language Models [2,5,6,26] as a query recommendation method. The technique of recommendation proposed estimates the probability of generation of the user query by the LM of the past user queries.
- In the second phase, a query expansion method is to apply using: (1) the new query to expand; (2) the top-ranked documents of it; (3) the best-recommended queries to it; and (4) the top-ranked documents of each recommended query. We propose also to apply the LSA method [14,20] to search for the most similar terms to the new user query terms. Then, these similarities are merged together with the expression of the cohesion weight presented in Cui et al. [9] to order the candidate terms for expansion according to the whole query to expand.

The Latent Semantic Analyses (LSA) method has already been used for query expansion. The originality of this work appears in the fact of extracting the context of the query using the LMs. While, in the field of information retrieval [26], the concept of LMs was exploited in estimating the probability of generation of a new query by the LM of the document. The aim of this utilization is classifying documents according to a query. However, our contribution arises also in the fact of calculating the recommendation score based simultaneously on terms and documents vectors.

In the rest of this paper, we firstly provide a literature review in Sect. 2. Then, we describe in Sect. 3 the different processes of our approach including the query recommendation, and the query expansion approaches proposed. We expose our results in the fourth section. Finally, we conclude in Sect. 5 and discuss future works.

2 Related work

Query Suggestion [22] is the fact of proposing to the user queries that are similar to its submitted query, whereas Query Expansion [7,8] is the fact of adding the terms that are almost similar to the user query terms, and it can be considered as a method for improving retrieval performance by extracting the context around the user's query.

Several query suggestion methods were proposed in the literature. For example, the reference [25] proposes a method for recommending a list of queries that are related to the user's query, based on a clustering of similar past queries. The authors hypotheses' that semantically similar queries may not share query terms but they certainly share terms in the documents selected by users while submitting these queries. Thus, the query recommendation algorithm presented by this work contains three steps. First, past users' queries are represented by a term weighted vectors with the text of their clicked URL's, and are clustered. The second step is an online step, while the first is processes offline. This step is applied when a new query is submitted to the search engine, by finding the cluster to which it belongs and then measures a rank score for each query in this cluster according to the new user's query. Finally, the related queries are ordered according to their rank score and returned as an output of the algorithm. This paper has also proposed a new similarity measure called as "Tanimoto Coefficient" in order to rank the related queries. The experiments over a real query logs show the effectiveness of this algorithm.

On the other hand, the terms used for expansion can be selected from external sources or from the corpus itself. Some of the methods for selecting the terms from the corpus are based on global analysis, where the list of candidate terms is generated from the whole collection, while others are based on local analysis [13] in which the relevance feedback techniques [12] are used in the aim of expanding terms are selected from the top-ranked documents by users.

Global analysis methods [12] are computationally very expensive and their effectiveness is generally not better and sometimes worse than local analysis. The problem with local analysis is that user's feedback is required to provide information regarding top-ranked relevant documents. User's involvement makes it difficult to develop such automatic methods for query expansion. To avoid this problem, a pseudo-relevance feedback approach is preferred where documents are retrieved using an efficient matching function and the top-retrieved documents are assumed to be relevant [9,19].

However, these methods of expansion were limited in the extraction of expansion terms from a set of documents and have not used information about interactions between the users and the system; this is the case of the expansion based on the use of query logs [11,15,23].

Query logs are a mine of information, which gives an idea about the interaction between the users and the information retrieval system. As an example, the approach proposed in Fonseca et al. [11] is a method of concepts suggestion for expanding the original user query with additional context. The first part of the approach is a method of concepts generation from query logs. This part is composed of three steps: it starts by an offline step for determining query relations in the log, then two steps for building a query relation graph and identifying concepts in this graph are performed during the query processing time. The second part of the approach is a concept-based query expansion method, where the system proposes to the user a set of concepts related to his query. Once the user has selected one concept related to the query, this concept is added to the original user query and the expanded query is processed. Such approaches are simple, intuitive and effective according to the experiments done on it, but it uses only past users queries for expanding the new query, and do not minimize the gap between queries' terms and documents' terms.

In the last decades, the information retrieval field has also known an integration of the semantic aspect to query expansion, document ranking and clustering, question-answer systems and query recommendation. In the work presented in Meng et al. [15], a new model for measuring similarity between web queries was proposed. The model is taking into account both the word form and the semantic information of the two queries. It uses WordNet [16] as a thesaurus that focuses on word meaning instead of word forms, in order to obtain the semantic information. The approach described uses the bottom-up hierarchical clustering so

that it can cluster past users queries and discover the different topics gathered in the search engine's log. The clustering process is carried out based on the new similarity metric proposed. The experiments made on this new model show its good performance in improving the precision of query expansion by 9.2 % and its recall by 8.1 %.

Traditionally, the concept of LM [2,5,6,26] is exploited in the field of information retrieval in order to represent the relationship of relevance between a document and a submitted query, by estimating the probability of generation of the query by the LM of the document. In this work, we exploit the LM in order to classify the past user's queries, extracted from the query log of the search engine, according to their capacity to generate the new user query. Then, we propose to apply the LSA method [14,20] as a query expansion technique. As a result, the LSA method gives a matrix that linked the documents with their terms, so we can compute the similarity between a query and a document or between two terms.

3 Context-aware query expansion method

The approach proposed in this paper (Fig. 1) is a context-aware query expansion method called "Latent Semantic Analyses using Recommended Queries" (LSARQ), and it is composed of two phases:

(1) Query Recommendation: we used the LMs in order to find the most related past queries to the user query.
(2) Query Expansion: we used the LSA model to select the candidate terms in order to expand the user query.

3.1 Phase 1: LM for the query's context extraction

The main objective of this work is to provide high-level suggestions for the original user query that we are using later for expanding the query with additional context. We used the LM to calculate the correlation between two queries. In this aim, we order past queries extracted from a query log according to their capacity to generate the new user query. The expression used in order to calculate the recommendation score of a past query $\boldsymbol{Q_p}$ according to the initial query $\boldsymbol{Q_n}$ is as follows:

$$RankScore\left(Q_n, Q_p\right) = \gamma Score_{LM\_T}\left(Q_n, Q_p\right) + (1-\gamma)\, Score_{LM\_D}\left(Q_n, Q_p\right) \quad (1)$$

With $\gamma \in [0, 1]$ is a parameter that we used for normalization.

The $Score_{LM\_T}$ is calculated using vectors that represent the presence or not of a term in a query. $Score_{LM\_D}$ is computed using vectors that represent the presence or not of a document between the clicked documents of a query.

The typical score function defined by *KL*-divergence in the language modeling framework [1,3] is used as a measuring function for $Score_{LM\_T}$ and $Score_{LM\_D}$. Its expression is as follows:

$$Score_{LM}\left(Q_n, Q_p\right) = \sum\nolimits_{t \in V} P\left(t|\theta_{Qn}\right) \log(P(t|\theta_{Qp}) \approx -KL(\theta_{Qn} \| \theta_{Qp}) \quad (2)$$

where θ_{Q_n} is the LM of the new query, θ_{Q_p} the LM created for a past query, and V the vocabulary of terms.

$P\left(t|\theta_Q\right)$ represents the probability of a term t in the LM of the query and is computed using the Maximum Likelihood Estimation (MLE), by the equation:

$$P\left(t|\theta_Q\right) = \frac{f\left(t\right)}{\sum_{t_i \in Q} f\left(t_i\right)} \tag{3}$$

With $f\left(t\right)$ is the frequency of t in the query.

Thus, Eq. (1) is representing the global ranking score of a past query Q_p according to the new query Q_n. Equation (2) is to be used to calculate $Score_{LM\_T}$ and $Score_{LM\_D}$ of Eq. (1). The $P\left(t|\theta_{Q_n}\right)$ and $P(t|\theta_{Q_p})$ in Eq. (2) are computed using the expression presented in Eq. (3).

However, the size of the training corpus cannot reach the size of a language. Thus, when a query contains a term which is absent from the training corpus, this term is estimated by a null probability. Therefore, a null probability is assigned to any sequence of words containing that word. This issue is known as the underrepresentation of data, and it represent the main problem that occurs for LMs.

The proposed solution to this problem is the "Smoothing" expressions whose aim is to assign a not null probability to the absent terms from the training corpus, by redistributing the probability mass observed.

Several smoothing methods have been developed in the literature [18]. The choice of the appropriate smoothing technique depends on the environment of experimentation according to Cao et al. [6]. One of the common smoothing methods used in information retrieval is the Jelinek–Mercer interpolation smoothing:

$$P\left(t|\theta'_{Q_p}\right) = (1-\lambda)\, P\left(t|\theta_{Q_p}\right) + \lambda P\left(t|\theta_C\right) \tag{4}$$

where λ is an interpolation parameter and θ_C the LM of the collection of queries extracted from the search engine's log.

For our approach, we use the Jelinek–Mercer interpolation smoothing only for past queries. The new query model θ_{Q_n} is estimated by the maximum likehood estimation without any smoothing.

3.2 Phase 2: query expansion using the LSA method

The LSA is a method that tries to overcome the problems of lexical matching by retrieving information on the basis of a conceptual meaning instead of individual words for retrieval.

LSA assumes that there is some latent (underlying) structure in word usage that is partially obscured by the variability in word choice. A particular mathematical technique called singular value decomposition (SVD) is applied to a word-document matrix in order to estimate the structure in word usage across documents [14].

Applied in a set of documents and a user query to expand, the LSA method build at first a word-document weighted matrix $\boldsymbol{A_{t\times d}}$, with t the number of terms and d the number of documents plus a column representing the query vector. Then, the SVD projection is computed by decomposing the term-document matrix $\boldsymbol{A_{t\times d}}$ into the product of three matrices $\boldsymbol{T_{t\times n}}$, $\boldsymbol{S_{n\times n}}$ and $\boldsymbol{D_{d\times n}}$:

$$A_{t\times d} = T_{t\times n} S_{n\times n} D'_{n\times d} \tag{5}$$

where $n = \min(t, d)$ is the number of dimensions for $\boldsymbol{A}$, also called the rank of $\boldsymbol{A}$ and $\boldsymbol{D'}$ is the transpose of $\boldsymbol{D}$.

The matrices $\boldsymbol{T}$ and $\boldsymbol{D}$ represent terms and documents in the new space and have orthonormal columns, i.e., $\mathbf{TT^T} = \mathbf{DD^T} = \mathbf{I}$. The matrix S is a diagonal matrix and contains

the singular values of A in descending order. The ith singular value indicates the amount of variation along the ith axis.

In the next step, the SVD matrices are truncated by reducing the rank n of the matrix $\boldsymbol{A}$. The objective of the method is to find the rank $k < n$ that gives a new matrix $\boldsymbol{A}'$ which is the best approximation of $\boldsymbol{A}$.

$\boldsymbol{A}'$ is constructed by restricting the matrices $\boldsymbol{T}$, $\boldsymbol{S}$ and $\boldsymbol{D}$ to their first k rows and multiplying them as follows:

$$A'_{t\times d} = T_{t\times k} S_{k\times k} D'_{k\times d} \tag{6}$$

The reduction of rank has to be done to the matrix $\boldsymbol{A}$ in the lower dimensional space, while minimizing the "distance" between the two matrices as measured by the 2-norm:

$$\Delta = \left|\left|A - A'\right|\right|_2 \tag{7}$$

The choice of the number of dimensions k for $\boldsymbol{A}'$ is an interesting problem. A reduction in n can remove much of the noise, by keeping too few dimensions important information may be lost. However, it is observed that the LSA method works well with a relatively small number of dimensions k. This observation shows the fact that these dimensions are capturing a major portion of the meaningful structure [20].

The truncated SVD captures most of the important underlying structure in the association of terms and at the same time removes the noise or variability in word usage. For example, terms that occur in similar queries or documents will be near each other in the k-dimensional space even if they never co-occur in the same query. In fact, some terms that never co-occur with the new query terms can be similar to them in the k-space.

Finally, the new user query terms vectors can be compared to all candidate terms for expansion, and they can be ranked by their similarity to each term of the query to expand using the common measure of similarity Cosine [21], whose expression is as follows:

$$Simc\left(\vec{t_i}, \vec{t_j}\right) = |\cos\left(\vec{t_i}, \vec{t_j}\right)| = \frac{|\vec{t_i} \times \vec{t_j}|}{\|\vec{t_i}\| \times \|\vec{t_j}\|}. \tag{8}$$

By combining the similarities of each candidate term t_j for all the new query terms t_i^Q, we can calculate the cohesion weight [9] of a candidate term, which represent the correlation (relationship) between this term and the whole query to expand.

The cohesion weight of a term t_j for a user query $\boldsymbol{Q}$ is measured by the expression [9]:

$$CoWeight\left(Q, t_j\right) = \ln\left(\prod_{t_i^Q \in Q}\left(Simc\left(t_j | t_i^Q\right) + 1\right)\right). \tag{9}$$

This method returns a list of weighted terms. The top-ranked terms can be selected as expansion terms for the new user query.

3.3 Integrating phase 1 and 2

The main idea of this work is to expand queries by enriching them with additional context based on the process illustrated in Fig. 1. In this aim, we are using a term vector and a document vector to represent each query. The term vector represents the presence or not of a term in a query, whereas the document vector is representing the presence or not of a document between the clicked documents of a query. The information about clicks are extracted from

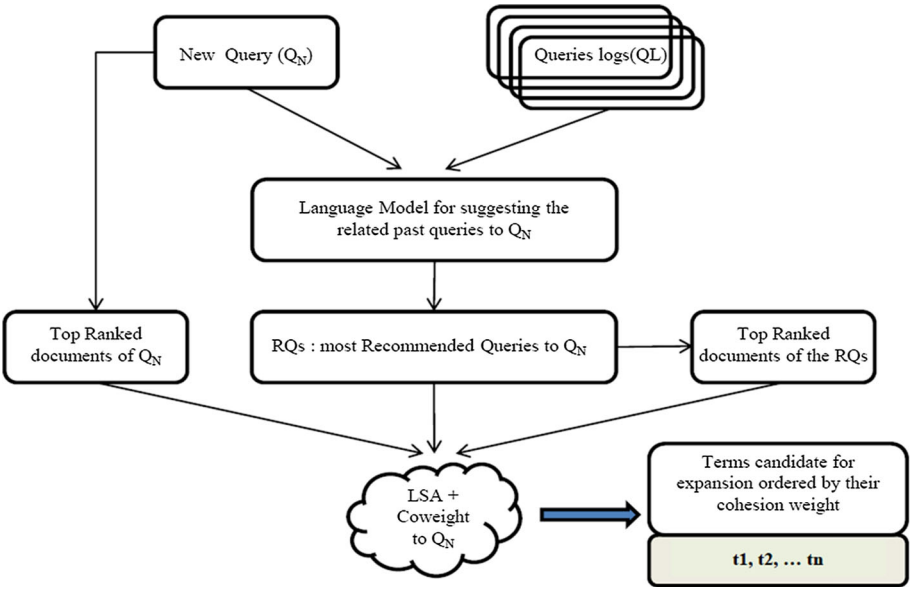

Fig. 1 Context-aware query expansion method

the "Query Logs," while the top-ranked documents of the new query to expand are considered instead of the clicked documents.

Therefore, we apply the query recommendation model presented as phase 1 to order the users' past queries (extracted from the Query Log) according to the new user query that we intend to expand in the end of this process. The output of this phase is a list of past queries ordered by their *RankScore* presented in Eq. (1).

The context of the new query Q_n is considered as the most recommended queries to Q_n. That means the top of the list resulted from phase 1.

As a *context-aware query expansion method*, phase 2 consist of applying the LSA method using the context extracted in phase 1. Thus, the LSA method is applied using:

- The new query to expand;
- The top-ranked documents to it;
- The best ranked recommended queries to it;
- The top-clicked documents to every recommended query used.

The output of LSA is a term-document truncated matrix of k dimensions (Sect. 3.2). This matrix is used to compute the similarities between the new query terms and all the terms contained in the matrix using Eq. (8). Finally, the cohesion weight (Eq. 9) measures the weight of the relationship between the whole query and a candidate term of expansion.

The result of the approach is a list of terms candidate for expansion ranked according to their correlation to the new query subject of expansion.

4 Experimental results

As a collection of test, we used the database CISI from the standard collection SRT. This collection provides 111 queries, 1460 documents and a matrix representing the relevance or non-relevance of each document to each query.

The first step of experimentation is done by applying the Query Recommendation method presented in Sect. 3.1. We used the average internal similarity (AIS) measure to identify the best value of the Jelinek–Mercer interpolation smoothing parameter λ (Eq. 4) for the LM based on terms vectors. We considered as a cluster each input query (query to expand) and its five best-recommended queries, and we calculated the AIS of each cluster. The AIS is computed using the expression [17]:

$$\text{AIS}\,(c) = \frac{\text{sum}\,(c)^2 - |c|}{|c|\,(|c| - 1)} \tag{10}$$

With c the cluster of queries, $|c|$ the number of vectors (queries) in the cluster and sum(c) is a vector, which represents the sum of all the vectors in the cluster c.

In Table 1, we represent the AIS values for short queries (contain less than 5 terms) and long queries (contain more than 5 terms).

Table 1 shows that for short queries the AIS increases from a low value for $\lambda = 0$ to its best value when smoothing with 0.2 and keeps it until $\lambda = 0.8$, while for long queries the best value of AIS is reached at 0.2 and then increases until having its lowest value at 0.8. Thus, we can conclude that using LMs based on terms for query recommendation with the parameter of smoothing equal to 0.2 is enough to reach the best values of AIS using our collection of test.

In order to test the relevance of this approach, we compared it with the Query Recommendation Algorithm (QRA) proposed in [25], which we improved and we used for context extraction in a previous work [10].

We present in Table 2 the AIS using the 5 best-recommended queries for each input query using our Language Model Recommendation (LMR) technique and Query Recommendation Algorithm (QRA), and considering the parameter of normalization $\lambda = 0.2$ for short queries and $\lambda = 0.4$ for long queries in Eq. 1.

Table 2 shows that for short and long queries the highest value of AIS is reached by our LMR approach. With these results, once again the LMs show their performance in the information retrieval field.

As second step of experimentation, we applied the approach LSARQ proposed in Sect. 3. We used the Un-interpolated Average Precision (UAP) measure to evaluate the performance of the IR system. We varied the number of recommended queries, and we searched for relevant documents until the 20th retrieved document. In Fig. 2, we present the results of the UAP for short and long queries.

We notice in Fig. 2 that for short queries the value of UAP reached its highest value while using 2 recommended queries, while for long queries the highest value of UAP is given using

Table 1 Average internal similarity for short and long queries while varying the smoothing parameter

Queries\λ	0	0.2	0.4	0.5	0.6	0.8
Short queries	2.63	5.10	5.12	5.12	5.12	5.12
Long queries	3.63	5.96	5.43	5.43	5.43	4.53

Table 2 AIS for short and long queries using the LMR and QRA approaches for query recommendation

AIS	QRA	LMR
Short queries	19.05	19.87
Long queries	18.60	19.04

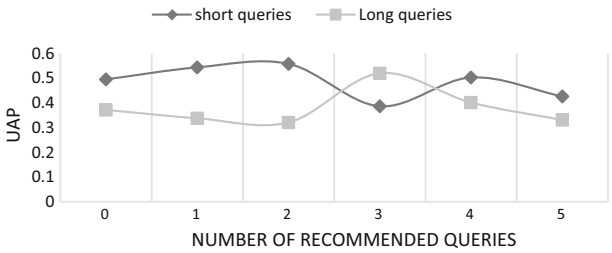

Fig. 2 Un-interpolated Average Precision for short and long queries using the LSARQ approach according to the number of recommended queries

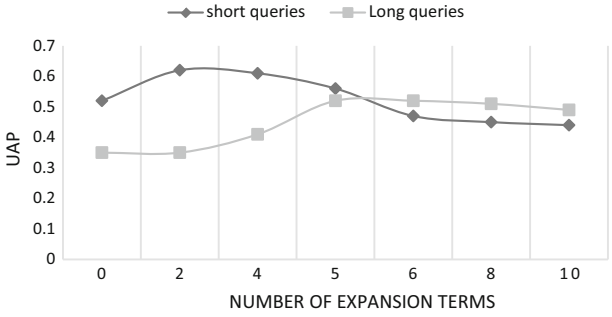

Fig. 3 Variation of the number of expansion terms used in the LSARQ approach for short and long queries

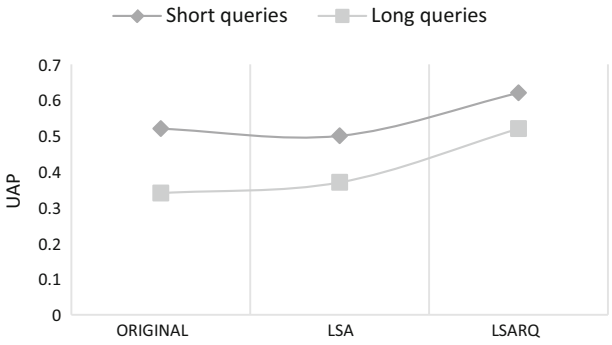

Fig. 4 Comparison of two query expansion methods for short and long queries

3 recommended queries. In what follows we used these results when varying the number of terms used to expand the initial queries. The results are presented in Fig. 3.

In Fig. 3, we notice that when expanding the short queries using 2 terms the UAP's value increases from 0.52 to 0.62 and then decreases more and more when adding more terms of expansion. Long queries keep the value of UAP 0.35 when adding only 2 terms using the LSARQ method. Then, this value increases until 5 and 6 terms of expansion where it reaches the highest value of UAP which is 0.52 to decrease after that when adding more than 6 terms.

In order to evaluate our approach, we compared it in Fig. 4 with the baseline (original query), and the LSA Model (query expansion using LSA without recommended queries).

Figure 4 shows that our proposed approach improves the results returned by the information retrieval (IR) system in a significant way for short and long queries. Table 3 and

Table 3 Comparing the performance of the query expanded by the proposed approach LSARQ to the baseline, and The LSA Method

Queries		LSARQ (%)
Short queries	Baseline	19.23
	LSA	24
Long queries	Baseline	52.94
	LSA	40.54

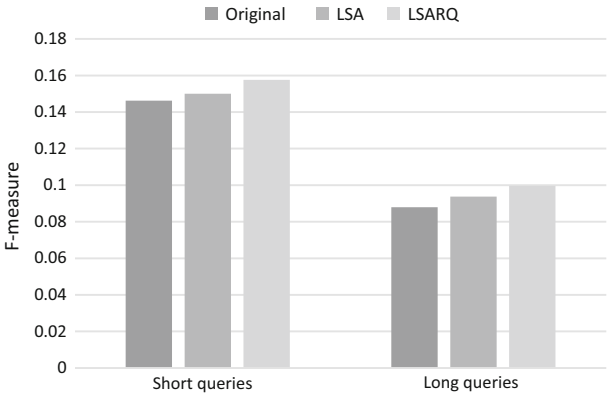

Fig. 5 Performance comparison by the F-measure

Fig. 5 proves these results, showing the performance of the LSARQ approach according to the baseline and the LSA model.

Table 3 shows that our context-aware query expansion approach "LSARQ" improves the precision of the IR system by 24 % according to the expansion using the LSA method only for short queries and by 40.54 % for long queries.

In Fig. 5, we evaluated the LSARQ using the F-measure. The results show an improvement of 7.76 % according to the search by the original query and 5.04 % according to the expansion by LSA for short queries. While for long queries, we can see that the F-measure's value increases by 13.31 % according to the original queries and 6.23 % according to the LSA method, which indicate the good performance of our approach.

5 Conclusion

In this paper, we proposed an approach for query expansion, which is based on the LSA method and the LMs. We used the LM in order to enrich the new user query with additional context extracted from the past user's queries Log. Using the text database CISI from the standard collection SMART, we have shown that our approach "LSARQ" improves the effectiveness of the information retrieval system with 24 % for short queries and 40.54 % for long queries according to the expansion using the LSA technique only, and with 19.23 % for short queries and 52.94 % for long queries according to the original users' queries. We conclude that the proposed combination gives better results than each individual method.

In future work, we intend to do more experimentations by investigating other combinations of query expansion methods by proposing a new method for the context extraction.

References

1. Asfari O, Doan BL, Bourda Y et al (2010) Context-based hybrid method for user query expansion. In: Proceedings of the fourth international conference on advances in semantic processing, SEMAPRO 2010. Florence, Italy, pp 69–74
2. Bai J, Nie JY (2004) Using language models for text classification. In: Proceedings of the Asia information retrieval symposium (AIRS)
3. Bai J, Nie JY, Bouchard H, Cao G (2007) Using query contexts in information retrieval. In: Proceedings of the 30th annual international ACM SIGIR conference on research and development in information retrieval, New York, USA, pp 15–22
4. Baziz M (2005) Indexation conceptuelle guide par ontologie pour la recherche d'information, Ph.D. dissertation, Institut de Recherche en Informatique de Toulouse, Université Paul Sabatier de Toulouse
5. Bouchard H, Nie JY (2006) Modèles de langue appliqués à la recherche d'information contextuelle. Inf Interact Intell 6:51–75
6. Cao G, Nie JY, Bai J (2005) Integrating word relationships into language models. In: Proceedings of ACM-SIGIR'05, Salvador, Brazil, pp 298–305
7. Carpineto C, Romano G (2012) A survey of automatic query expansion in information retrieval. ACM Comput Surv 44(1):1–50
8. Chartree J, Cankaya EC, Phithakkitnukoon S (2013) Query expansion using association matrix for improved information retrieval performance. In: Proceedings of international conference on information and knowledge engineering, Las Vegas, NV, USA
9. Cui H, Wen JR, Nie JY et al (2002) Probabilistic query expansion using query logs. In: Proceedings of the 11th international conference on world wide web. Honolulu, Hawaii, USA, pp 325–332
10. El Ghali B, El Qadi A, El Midaoui O et al (2015) Query recommendation based terms and relevant documents using language models. WSEAS Trans Inf Sci Appl 12:112–119
11. Fonseca BM, Golgher P, Pôssas B, et al (2005) Concept-based interactive query expansion. In: Proceedings of the 14th ACM international conference on information and knowledge management, New York, USA, pp 696–703
12. Gupta Y, Saini A, Saxena AK (2013) A review on important aspects of information retrieval. Int J Comput Control Quantum Information Eng 7(12):990–998
13. Lin SM, Huang CM (2006) Personalized optimal search in local query expansion. In: Proceedings of the 18th conference on computational linguistics and speech processing. Hsinchu, Taiwan, pp 221–236
14. Manning CD, Raghavan P, Schätze H (2009) An introduction to information retrieval. Online edition (c) Cambridge University Press, Cambridge, pp 403–419
15. Meng L, Huang R, Gu J (2013) A new model for measuring similarity of web queries and its application in query expansion. Int J Grid Distrib Comput 6(4):51–62
16. Miller GA (1995) WordNet: a lexical database for English. Commun ACM 38(11):39–41
17. O'Connor B (2003) Clustering political words: senses and connotations. Final Project. CS224N/Ling 237
18. Puurula A (2013) Cumulative progress in language models for information retrieval. In: Proceedings of Australasian language technology association workshop. Brisbane, QLD, pp 96–100
19. Réquier AS, Dupont G, Adam S et al (2010) Évaluation d'outils de reformulation interactive de requêtes. In: Proceedings of Conférence en Recherche d'Infomations et Applications—CORIA 2010, 7th French Information Retrieval Conference. Sousse, Tunisia, pp 223–238
20. Rosario B (2001) Latent semantic indexing: an overview. INFOSYS 240
21. Slimani T, Yaghlane BB, Mellouli K (2007) Une extension de mesure de similarité entre les concepts d'une ontologie. In: Proceedings of SETIT 2007, 4th International Conference: Sciences of Electronic, Technologies of Information and Telecommunications. Hammamet, Tunisia
22. Song W, Liang JZ, Cao XL et al (2014) An effective query recommendation approach using semantic strategies for intelligent information retrieval. Expert Syst Appl Int J Arch 41(2):366–372
23. Wen J, Lao N, Ma W (2004) Probabilistic model for contextual retrieval. In: Proceedings of the 27th annual international ACM SIGIR conference on research and development in information retrieval. ACM, Sheffield, pp 57–63
24. Wen J, Nie J, Zhang H (2001) Clustering user queries of a search engine. In: Proceedings of the 10th international conference on world wide web. ACM, Hong Kong, pp 162–168
25. Zahera HM, El Hady GF, Abd El-Wahed WF (2011) Query recommendation for improving search engine results. Int J Inf Retr Res 1(1):45–52
26. Zhai C (2008) Statistical language models for information retrieval: a critical review. Found Trends Inf Retr 2(3):137–215

Btihal El Ghali received the French high school diploma in Mathematics in 2006. Then, the Bachelor degree in Mathematics and Computer Science in 2009 from the Faculty of Science of Rabat (Morocco) and finally had her Master degree in applied computer science in 2011 from the same Faculty. Currently she is preparing his Ph.D. in the Laboratory of Research in Informatics and Telecommunications (LRIT). Since 2012, she is employed as a temporary professor in Computer Science at the University Mohammed V of Rabat Morocco. Since 2013, she took the responsibility of the courses "Data Warehouse, Data Mining and Interactif Systems of Decision Support" and "Fuzzy Logic" in the engineer School "INSEA" in Rabat. Until now, she has participated to many international conferences, published two papers in the field of "Information Retrieval", a chapter in Scientific Books, and a Lecture Note in Computer Science.

Abderrahim El Qadi is currently Professor in Computer Science Department, High School of Technology, Moulay Ismail University—Meknes Morocco. He leads the research team: Information Technology and Multimedia. He received his Ph.D. from the faculty of science, Mohammed V University—Rabat Morocco, in July 2002. In June 2010, he has received his HDR from the same Faculty, in the subject: "Information Retrieval and Query optimization in Data warehouse". His research interests include: data mining, text mining, web usage mining, information retrieval, query expansion, query recommendation, semantic web, data integration, SQL query Optimization and Big Data. Pr. El Qadi has published several publications in international journals and conferences. He has also organized and participated as scientific committee member in several conferences.