Integrating Query Context and User Context in an Information Retrieval Model Based on Expanded Language Modeling

Rachid Aknouche, Ounas Asfari, Fadila Bentayeb, and Omar Boussaid

ERIC Laboratory (Equipe de Recherche en Ingnierie des Connaissances)
5 Avenue Pierre Mends France, 69676 Bron Cedex, France
{Rachid.Aknouche,Ounas.Asfari,Fadila.Bentayeb,
Omar.Boussaid}@univ-lyon2.fr
http://eric.univ-lyon2.fr/

Abstract. Access to relevant information adapted to the needs and the context of the user is a real challenge. The user context can be assimilated to all factors that can describe his intentions and perceptions of his surroundings. It is difficult to find a contextual information retrieval system that takes into account all contextual factors. In this paper, both types of context user context and query context are integrated in an Information Retrieval (IR) model based on language modeling. Here, the query context include the integration of linguistic and semantic knowledge about the user query in order to explore the most exact understanding of user's information needs. In addition, we consider one of the important factors of the user context, the user's domain of interest or the interesting topic. A thematic algorithm is proposed to describe the user context. We assume that each topic can be characterized by a set of documents from the experimented corpus. The documents of each topic are used to build a statistical language model, which is then integrated to expand the original query model and to re-rank the retrieved documents. Our experiments on the 20_Newsgroup corpus show that the proposed contextual approach improves significantly the retrieval effectiveness compared to the basic approach, which does not consider contextual factors.

1 Introduction

Most existing Information retrieval systems depend, in their retrieval decision, only on queries and documents collections; information about actual users and search context is largely ignored, and consequently great numbers of irrelevant results occur. Towards the optimal retrieval system, the system should exploit as much additional contextual information as possible to improve the retrieval accuracy, whenever this is available.

Context is a broad notion in many ways. For instance, [11] define context as any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves. The effective use of contextual information in computing applications still

G. Quirchmayr et al. (Eds.): CD-ARES 2012, LNCS 7465, pp. 244–258, 2012.

remains an open and challenging problem. Several researchers have tried over the years to apply the context notion in information retrieval area; this will lead to the so-called contextual information retrieval systems which combine a set of technologies and knowledge on the query and the user, in order to deliver the most appropriate answers according to the user's information needs.

As information needs are generally expressed via queries and the query is always formulated in a search context, contextual factors (such as the user's domain of interest, preferences, knowledge level, user task, etc.) have a strong influence on relevance judgments [16]. But it is difficult to find a contextual information retrieval system that takes into account all the available contextual factors at the same time. Thus the context can be defined as the combination of some contextual factors which may be more or less relevant according to the actual performed research. Indeed, in this paper, we try to consider two types of context, user context and query context. We think that these two contextual factors can improve the information retrieval model.

In this paper, the user context is defined by the user's domain of interest or the interesting topic. We propose a thematic algorithm to describe the predefined user's topics which are characterized by a set of documents. Considering the user's interested topics allows providing more relevant results. The second considered contextual factor is the query context, which includes a linguistic and a semantic knowledge about the user query in order to explore the most exact understanding of user's information needs. Thus, we extend the user query by related terms automatically by using the linguistic and semantic knowledge. Also, we propose a framework based on language modeling approach in order to integrate the two contextual factors.

For instance, if a user submits the query "*apple*" into a Web search engine, knowing that user queries are generally shorts and contain words with several meanings, there are different topics in the top 20 documents selected by the search engine. Some users may be interested in documents dealing with "*apple*" as "*fruit*", while other users may want documents related to Apple computers. In order to disambiguate this query, we can assign a set of topics with this query. For example, we can assign the topics "*cooking*", "*fruit*" or "*computer*" with the query "*apple*". This is the user's domain of interest. In addition, to extend the query "apple" with the so-called query context, we can add concepts to this query like: "*Computers*", "*Systems*", "*Macintosh*", etc.

The language models in information retrieval (IR) are used to compute the probability of generating query q given a document D (i.e. compute: $P(q|D)$); and the documents in the collection C are ranked in descending order of this probability. Several methods have been applied to compute this probability as [19]. In most approaches, the computation is conceptually decomposed into two distinct steps: Estimating the document model and computing the query likelihood using the estimated document model, as in [21].

In this paper, we propose an approach to build a statistical language model that extends the classic language modeling approach for information retrieval in order to integrate the two contextual factors. The extended model has been tested based on one of the common IR test collections; 20_Newsgroup corpus. The results show that our contextual approach improves significantly the retrieval effectiveness compared to the basic approach, which does not consider contextual factors. The rest of this paper is organized as follows: Section 2 introduces the state of the art and some related works; Section 3 introduces our contextual information retrieval model based on language modeling; Section 4 shows the experimental study and the evaluation of our approach. Finally, Section 5 gives the conclusion and future work.

2 Related Works

2.1 Context in Information Retrieval

Several contextual factors can be considered in Information Retrieval (IR) area in order to improve the information retrieval systems. In this section we review some of studies in IR concerning the user context and query context, as long as we take them into consideration to extend the language modeling approach for IR[3].

User Context. The user context can be assimilated to all factors that can describe his/her intentions and perceptions of his/her surroundings [22]. These factors may cover various aspects: physical, social, personal, professional, technical, task etc. Figure 1 shows these factors and examples for each one [18].

However, the problems to be addressed include how to represent the context, how to determine it at runtime, and how to use it to influence the activation of user preferences. It is very difficult to modeling all the contextual factors in one system, so the researchers often take into account some factors, as in [2], they defined the user context as the user's current task together with his/her profile. In the contextual information retrieval, user context has a wide meaning based on the user behavior; we can mention some of them in the following:

- Visited Web pages [26]: Here, the user context is defined as the information extracted by using the full browsing history of the user.
- Recently accessed documents [5]: In this case, the user context is defined as words which indicate a shift in context; these words are identified by information about the sequence of accessed documents. This is carried out by monitoring a user's document access, generating a representation of the user's task context, indexing the consulted resources, and by presenting recommendations for other resources that were consulted in similar prior contexts.
- Past queries and click-through data [25]: Several context-sensitive retrieval algorithms are proposed based on statistical language models to combine the preceding queries and clicked document summaries with the current query for better ranking of documents.
- Recent selected items or purchases on proactive information systems [6].

- Information that is previously processed or accessed by the user via various forms: email, web page, desktop document, etc. Stuff I've Seen SIS, [12].
- Implicit feedback: Implicit feedback techniques often rely on monitoring the user interaction with the retrieval system, and extract the apparently most representative information related to what the user is aiming at [17].

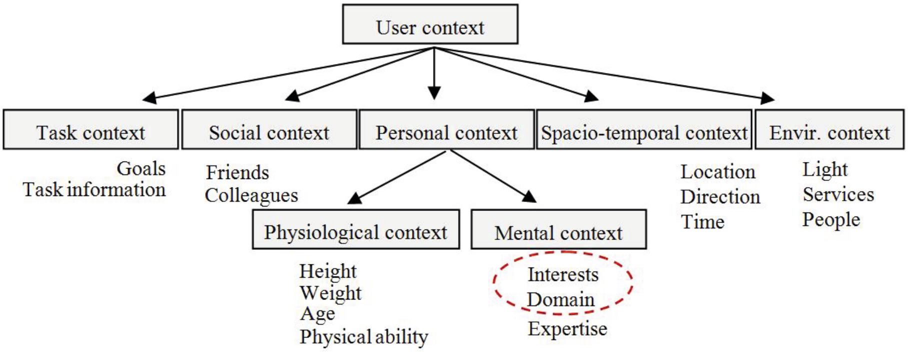


Fig. 1. Context Model

Query Context. The notion of query context has been widely mentioned in many studies of information retrieval like [4][9]. The objective is to use a variety of knowledge involving query to explore the most exact understanding of user's information needs. A query context will minimize the distance between the information need, I, and the query q, [1]. Distance (I to q) is minimized by minimizing:

- The lack of precision in the language used in the query terms. Lexicons which comprise the general vocabulary of a language can minimize this lack of precision in the language by identifying terms with minimal ambiguity.
- The use of the wrong concepts in the query to represent the information needs. Prior research suggests Ontology's for doing that.
- The lack of preferences in the query to constrain the concepts requested. This lack can be minimized by using user profiles.

The query context, in other studies, is defined as the elements that surround the query, such as:

- Text surrounding a query, Text highlighted by a user [13].
- Surrounding elements in an XML retrieval application [15][24].
- Broadcast news text for query-less systems [14].

[10] exploit the query context for predicting the user intent as being informational related to the content retrieval, navigational related to the web site retrieval or transactional related to the online service retrieval. They construct the query context by associating it with ontology concepts from the ODP (Open Directory Project) taxonomy.

2.2 Statistical Language Models

A statistical language model is a probability distribution over word sequences, using language models for information retrieval has been studied extensively, like in [19][21][27], because of its solid theoretical background as well as its good empirical performance. The main idea of language models in IR is to order each document D in the collection C according to their ability to generate the query q. Thus, it is the estimation of the generation probability $P(q|D)$; Probability of a query q given a document D. Several different methods have been applied to compute this conditional probability, such as the works of [21][8].

2.3 Discussion

User query is an element that specifies an information need, but the majorities of these queries are short and ambiguous, and often fail to represent the information need. Many relevant terms can be absent from queries and terms included may be ambiguous, thus, queries must be processed to address more of the user's intended requirements [2]. Typical solution includes expanding the initial user query by adding relevant terms. In this study we will expand the query representation by the query context which is defined above.

As we mentioned previously, it is difficult to consider all the available contextual factors. Thus, in this study, our definition of the user context is the user's interesting topic. Consequently, when we talk about the user context we talk about the user's interesting topics and taking into consideration the query context.

The language models for information retrieval have some limitations to capture the underlying semantics in a document due to their inability to handle the long distance dependencies. Also queries are typically too short to provide enough contexts to precisely translate into a different language. Thus, many irrelevant documents will be returned by using the standard language model approach for IR without integrating contextual factors.

In this paper, we will integrate the above two types of context within one framework based on language modeling. Each component contextual factor will determines a different ranking score, and the final document ranking combines all of them. This will be described in Section 3.

3 Contextual Information Retrieval Model Based on Language Modeling

In this section, we present our approach to construct a statistical language model given user's interested topics, user context, and considering the query expansion by using the linguistic and semantic processing.

3.1 Language Models for IR

Let us consider a query $q = t_1 t_2 ... t_n$, the generation probability is estimated as follows:

$$P(q \mid D) = \prod_{t \in q} P(t \mid \theta_D)^{c(t;q)}$$

$$= P(t_1 \mid \theta_D) P(t_2 \mid \theta_D) ... P(t_n \mid \theta_D) \qquad (1)$$

where: $c(t;q)$ Frequency of term t in query q. θ_D is a language model created for a document D. $P(t \mid \theta_D)$: The probability of term t in the document model. In order to avoid zero probability by assigning a low probability to query terms ti which are not observed in the documents of corpus, smoothing on document model is needed. The smoothing in IR is generally done by combining documents with the corpus [27], thus:

$$P(t_i \mid \theta_D) = \lambda P(t_i \mid \theta_D) + (1 - \lambda) P(t_i \mid \theta_C) \qquad (2)$$

where: λ is an interpolation parameter and θ_C the collection model. In the language modeling framework the similarity between a document D and a query q (a typical score function) can be also defined by measuring the *Kullback-Leibler* (KL-divergence) [19] as follows:

$$Score(q, D) = -KL(\theta_q \parallel \theta_D) = \sum_{t \in V} P(t \mid \theta_q) \log \frac{P(t \mid \theta_D)}{P(t \mid \theta_q)}$$

$$= \sum_{t \in V} P(t \mid \theta_q) \log P(t \mid \theta_D) - \sum_{t \in V} P(t \mid \theta_q) \log P(t \mid \theta_q)$$

$$\propto \sum_{t \in V} P(t \mid \theta_q) \log P(t \mid \theta_D) \qquad (3)$$

Where: θ_q is a language model for the query q, generally estimated by relative frequency of keywords in the query, and V the vocabulary. In the basic language modeling approaches, the query model is estimated by *Maximum Likelihood Estimation* (MLE) without any smoothing [8].

$P(t \mid \theta_q)$: The probability of term t in the query model.

Note that the last simplification is done because $\sum P(t \mid \theta_q) \log P(t \mid \theta_D)$ depends only on the query, and does not affect the documents ranking.

3.2 General IR Model

The classic information retrieval systems (Non-context model) generate query directly based on similarity function or matching between the query and the documents, according to a few terms in the query. In fact, query is always formulated in a search context; contextual factors have a strong influence on relevance judgments. To improve retrieval effectiveness, it is important to create a more complete query model that represents better the information need. In particular, all the related and presumed terms should be included in the query model. In these cases, we construct the initial query model containing only the original terms, and a new contextual model containing the added terms. We generalize this approach and integrate more models for the query.

Let us use θ_q^0 to denote the original query model, θ_q^s to denote the query context model and θ_q^u to denote the user context model. θ_q^0 can be created by MLE (*Maximum Likelihood Estimation*), as in [7]. We will describe the details to construct θ_q^s and θ_q^u in the following sections.

Given these models, we create the following final query model by interpolation:

$$P(t \mid \theta_q) = \sum_{i \in X} a_i P(t \mid \theta_q^i) \tag{4}$$

where: $X = \{0, u, s\}$ is the set of all component models.

a_i (With $\sum_{i \in X} a_i = 1$) are their mixture weights.

Thus formula (3) becomes:

$$Score(q, D) = \sum_{t \in V} \sum_{i \in X} a_i P(t \mid \theta_q^i) \log P(t \mid \theta_D) = \sum_{i \in X} a_i Score_i(q, D) \tag{5}$$

Where the score according to each component model is:

$$Score_i(q, D) = \sum_{t \in V} P(t \mid \theta_q^i) \log P(t \mid \theta_D) \tag{6}$$

Like that, the query model is enhanced by contextual factors. Now we have to construct both query context model and user context model and combine all models. We will describe that in the following sections.

3.3 Constructing Query Context Model

We will use both a linguistic knowledge and a semantic knowledge to parse the user query. Because linguistic knowledge does not capture the semantic relationships between terms and semantic knowledge does not represent linguistic relationships of the terms [1]. We use *WordNet* as the base of a linguistic knowledge. For the semantic knowledge we depend on ODP[1](Open Directory Project) Taxonomy, it is one type of ontology.

The integration of linguistic and semantic knowledge about the user query into one repository will produce the query context which can help to understand the user query more accurately.

Thus the initial query $q = t_1 t_2 ... t_n$ is parsed using *WordNet* in order to identify the synonymous for each query term $\{t_{w1}, t_{w2}, ..., t_{wk}\}$.

The query and its synonyms q_w are queried against the ODP taxonomy in order to extract a set of concepts $\{c_1 c_2 ... c_n\ \}$ (with $m \geq n$) that reflect the semantic knowledge of the user query. The concepts of the terms set q_w and their sub-concepts produce the query-context $C_q = \{c_1 c_2 ... c_n\ \}$. Thus the elements of

[1] ODP: Open Directory Project: `www.dmoz.org`

C_q are the concepts extracted from the ODP taxonomy by querying the initial query and its synonyms against it. For each term t, we select the concepts of only first five categories issued from ODP taxonomy.

Among the concepts of query context C_q, We consider only the shared concepts between at least two query terms, that means, a concept $C_i \in C_q$ the context of the query q if one of the following holds:

- C_i and $t \in q$ are the same, *i.e.* a direct relation.
- C_i is a common concept between the concepts of at least two query terms or their synonymous.

For instance, if a user query is "Java Language", the query context C_q, in this case, will be: < *computers, programming, languages, Java, JavaScript* >. Thus, the corresponding query context model θ_q^s is then defined as follows:

$$P(t \mid \theta_q^s) = \sum_{C_i \in C_q} p(t \mid C_i) p(C_i \mid \theta_q^0) \tag{7}$$

Accordingly, the score according to the query context model is defined as follows:

$$Score_s(q, D) = \sum_{t \in V} P(t \mid \theta_q^s) \log P(t \mid \theta_D) \tag{8}$$

3.4 Constructing User Context Model

As we previously mentioned, in this paper, the user context is defined as the user's domain of interest or the interesting topic. We will depend on predefined topics, which are derived from the used corpus. To define these topics, we can use our thematic algorithm, which will be discussed in the following (see Fig.2).

We suppose that the user can select his own topic from these predefined topics by assigning one topic to his query.

We exploit a set of documents already classified in each topic. The documents of each topic can be identified using same thematic algorithm. Thus a language model, for each topic, can be extracted from the documents of this topic. To extract only the specific part of the topic and avoid the general terms in the topic language model, we apply our thematic algorithm as follows (see Fig.2).

A thematic unit refers to a fragment of a document D referring to one topic. The steps are based on a thesaurus that includes all lexical indicators which are considered important for this segmentation process. They are created, in this study, manually but to extend the thematic units coverage in a document, we added synonyms and lexical forms that may have these units in the corpus. The obtained text fragments are then grouped by the thematic unity. Seeking information related to the user context (term context) will be done from these fragments.

Next maximum likelihood estimation is used on the documents of each topic to extract the topic language model.

We suppose that we have the following predefined topics: $TC_1, TC_2, ..., TC_j$, and the user assign the topic TC_j to his query. We can extract the user context

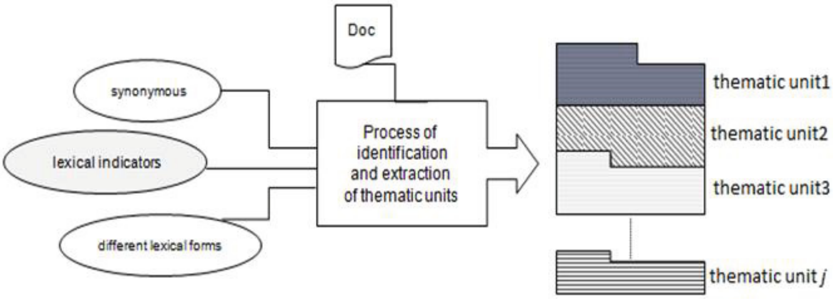


Fig. 2. A thematic algorithm

language model which is extracted from the documents of the topic TC_j, as follows (considering the smoothing):

$$\theta_u = \arg\max \prod_{D \in TC_j} \prod_{t \in D} \left[\mu P\left(t \mid \theta_{TC_j}\right) + (1-\mu)\, P\left(t \mid \theta_C\right)\right]^{c(t;D)} \tag{9}$$

Where: $c(t;D)$ is the occurrence of the term t in document $D.\mu$ is a smoothing parameter. μ is fixed to 0.5 in the experimentation, because, in our dataset, we have considered the middle point as a smoothing parameter. TC_j is the set of documents in the topic j. In the same method we can compute the set of user context models for all predefined topics. When the user assigns one topic to his query q manually, the related user context model θ_q^s has to be assigned to this query q, and the score depending on this user context model (represented by the related topic) will be:

$$Score_u(q,D) = \sum_{t \in V} P(t \mid \theta_q^u) \log P(t \mid \theta_D) \tag{10}$$

4 Experiments

To validate our approach, we will present an experimental study which is done by using a test collection, 20_Newsgroups[2] corpus. The objective of this experimental study is to compare the results provided by using an information retrieval model on the dataset without considering the contextual factors with those provided by a general information retrieval model considering the contextual factors.

Our approach (including steps) is implemented in Java by using Eclipse environment. The prototype use JWNL[3] (Java WordNet Library), which is an API that allows the access to the thesaurus *WordNet* to find synonyms of query

[2] http://people.csail.mit.edu/jrennie/20Newsgroups/

[3] http://jwordnet.sourceforge.net/handbook.html

terms. For the semantic knowledge we depend on ODP Taxonomy which is free and open, everybody can contribute or re-use the dataset, which is available in RDF (structure and content are available separately), i.e., it can be re-used in other directory services. Also we used the Oracle RDBMS database to host: (1) the thesaurus for terms synonyms, (2) the topics which are generated during the process of identification and extraction, and (3) the relevance scores of returned documents. In order to facilitate the evaluation, we developed an interface that helps users to enter their queries, to compute the evaluation criteria, and then to display the results which are ranked according to the degrees of relevance.

4.1 Newsgroup Data Sets

The 20_Newsgroup data set is a common benchmark collection of approximately 20,000 newsgroup documents, partitioned nearly evenly across 20 different newsgroups. This dataset was introduced by [20]. It has become a popular data set for experiments in text applications of machine learning techniques, such as text classification. Over a period of time, 1000 articles were taken from each of the newsgroups, which make this collection. The 20 topics are organized into broader categories: computers, recreation, religion, science, for-sale and politics. Some of the newsgroups are considered to have similar topics, such as the *rec.sport.baseball* and *rec.sport.hockey* which both contain messages about sports, while others are highly unrelated (e.g *misc.forsale/ soc.religion.christian*) newsgroups. Table 1 shows the list of the 20 newsgroups, partitioned (more or less) according to subject matter. This dataset is also used by [23].

Moreover, we preprocessed the data by removing stop words and all documents are stemmed using the *Porter algorithm.* The document-terms matrix is based on language models and each document is represented as a vector of occurrence numbers of the terms within it. The results of this preprocessing phase for the dataset before and after the classification topics are presented in Table2.

The query execution phase, in our approach, returns a ranked list of documents that match the query. The experimentation on the 20_Newsgroups collection doesn't provide the ability to compute the precision and recall metrics. In this way our experiments were conducted with the *Lemur Toolkit*[4] , which is a standard platform to conduct experiments in information retrieval. The toolkit has been used to carry out experiments on several different aspects of language modeling for ad-hoc retrieval. For example, it has been used to compare smoothing strategies for document models, and query expansion methods to estimate query models on standard TREC[5] collections. We used the language models for all our retrieval tasks. All the other parameters were set at their default values. We remove all *UseNet* headers from the Newsgroup articles and we used 20 queries, which are listed in Table 3, to retrieve results from this documents dataset. The queries vary from 1 term to 7 terms.

[4] http://www.lemurproject.com

[5] http://trec.nist.gov/

Table 1. A list of the 20 Topics

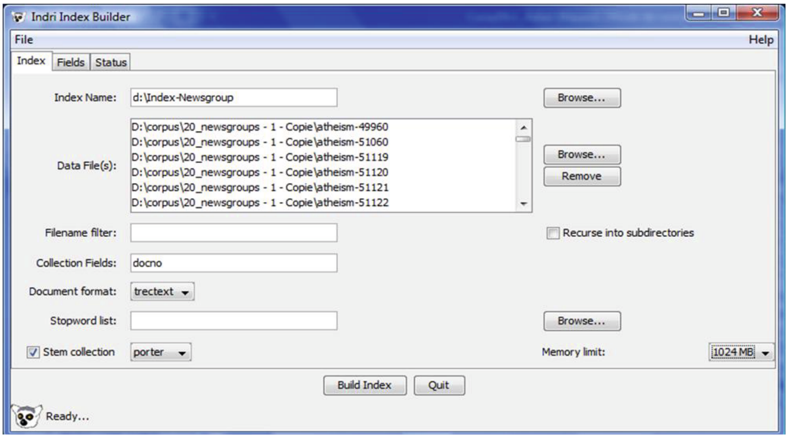
20_Newsgroups dataset			
comp.graphics	rec.autos	talk.politics.misc	soc.religion.christia
comp.os.ms-windows.misc	rec.motorcycles	talk.politics.guns	sci.crypt
comp.sys.ibm.pc.hardware	rec.sport.baseball	talk.politics.mideast	sci.electronics
comp.sys.mac.hardware	rec.sport.hockey	talk.religion.misc	sci.med
comp.windows.x	misc.forsale	alt.atheism	sci.space

Table 2. Pre-processing phase applied on the dataset

Corpus	docs	stems	Corpus	docs	stems
20 news group	20017	192375	ec.sport.baseball	1001	14000
alt.atheism	1001	15618	rec.sport.hockey	1001	15610
ccomp.graphics	1001	17731	sci.crypt	1001	17436
comp.os.ms-windows.misc	1001	54511	sci.electronics	1001	15622
comp.sys.ibm.pc.hardware	1001	16575	sci.med	1001	19963
comp.sys.mac.hardware	1001	15011	sci.space	1001	18432
comp.windows.x	1001	24915	soc.religion.christian	1001	13915
misc.forsale	1001	17518	talk.politics.guns	1001	20258
rec.autos	1001	15415	talk.politics.mideast	1001	20546
rec.motorcycles	1001	15108	talk.politics.misc	1001	17782

4.2 Baseline

Classic IR Baseline. As a baseline for comparison, for each dataset, we created an index of all the documents using *Lemur's indexer*. Figure 3 shows the indexation interface. Also we used *Lemur's retrieval engine* to return a list of relevant documents using the queries which are described above. This is the standard information retrieval setting. For instance, Figure 4 shows the document's ranking for the query "*athletics play*".



Index Name: d:\Index-Newsgroup

Data File(s):
D:\corpus\20_newsgroups - 1 - Copie\atheism-49960
D:\corpus\20_newsgroups - 1 - Copie\atheism-51060
D:\corpus\20_newsgroups - 1 - Copie\atheism-51119
D:\corpus\20_newsgroups - 1 - Copie\atheism-51120
D:\corpus\20_newsgroups - 1 - Copie\atheism-51121
D:\corpus\20_newsgroups - 1 - Copie\atheism-51122

Filename filter:

☐ Recurse into subdirectories

Collection Fields: docno

Document format: trectext

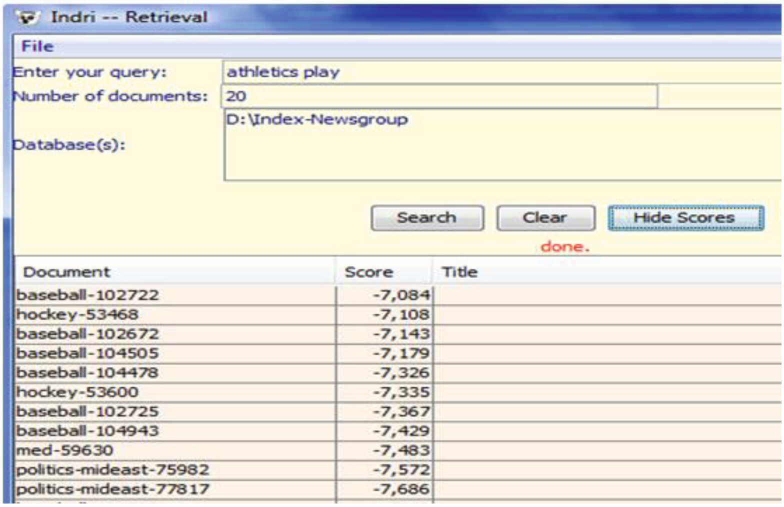
Stopword list:

☑ Stem collection: porter

Memory limit: 1024 MB

Ready...

Fig. 3. Lemur's indexer



Enter your query: athletics play
Number of documents: 20
Database(s): D:\Index-Newsgroup

Document	Score	Title
baseball-102722	-7,084	
hockey-53468	-7,108	
baseball-102672	-7,143	
baseball-104505	-7,179	
baseball-104478	-7,326	
hockey-53600	-7,335	
baseball-102725	-7,367	
baseball-104943	-7,429	
med-59630	-7,483	
politics-mideast-75982	-7,572	
politics-mideast-77817	-7,686	

Fig. 4. Lemur's retrieval

Contextual Information Retrieval Baseline. We expanded the query with both user context and query context. In the 20_Newsgroup dataset, the topics names are considered as a user context. The documents are indexed with *Lemur's indexer*. *Lemur's retrieval* engine was used to perform information retrieval using the expanded query.

Table 3. The experimented queries list

N	Query	N	Query
1	Sport	11	Logitech Bus Mouse adapter
2	athletics play	12	Division Champions
3	Stanley Cup Champion: Vancouver Canucks	13	baseball fan
4	ordinary ISA card	14	HD drive
5	East Timor	15	System requirement
6	High speed analog-digital pc-board	16	memory controller pc
7	Chicago Blackhawks	17	league teams
8	Kevin Dineen play for the Miami Colons	18	macintosh apple hardware
9	National league pitchers	19	science cryptography
10	good defensive catcher	20	society religion

4.3 Results

To evaluate the performance of our approach we use the TREC evaluation software to calculate the metrics. We use *ireval.pl Perl script* which comes with the *Lemur toolkit* distribution for interpreting the results of the program *trec_eval.* Figure 5 illustrates the precision/recall curves of the IR classic and of our contextual retrieval model on the 20_Newsgroup dataset. The results of our approach, presented by the curves, show significantly improvement in measure of Precision/Recall compared to the classical model.

The improvement is precisely in the accuracy rate. It is obtained by using the contextual model which expands the original query model and re-rank the returned documents. Search engines involve query expansion to increase the quality of user search results. It is assumed that users do not always formulate their queries using the best terms. Using the general model which expands the user query with the contextual factors will increase the recall at the expense of the precision. This explains our motivation to consider the contextual factors and topic units in our approach.

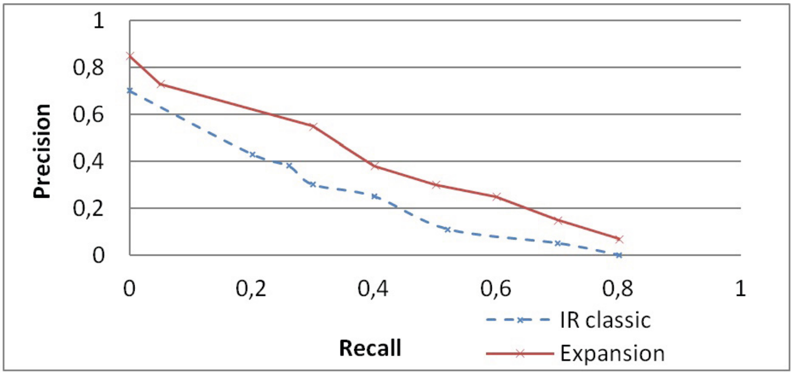


Fig. 5. Precision/recall curves for 20_Newsgroup corpus

5 Conclusion

In order to improve the information retrieval, we considered, in this paper, two types of context, user context and query context. We proposed an approach based on query expansion to provide a more efficient retrieval process. The initial query, the user context and the query context are integrated to create a new query. A thematic algorithm is proposed to describe the user context and both linguistic knowledge (WordNet) and semantic knowledge (ODP taxonomy) are used to represent the query context. The two contextual factors are integrated in one framework based on language modeling. We proposed a new language modeling approach for information retrieval that extends the existing language modeling approach to build a language model in which various terms relationships can be integrated. The integrated model has been tested on the 20_Newsgroup corpus. The results show that our contextual approach improves the retrieval effectiveness compared to the basic approach, which does not consider contextual factors. In the future, we plan to consider more contextual factors. The future objective is to combine our approach with the On-line Analytical Processing (OLAP) systems.

References

1. Asfari, O., Doan, B.-L., Bourda, Y., Sansonnet, J.-P.: A Context-Based Model for Web Query Reformulation. In: Proceedings of the International Conference on Knowledge Discovery and Information Retrieval, KDIR 2010, Spain, Valencia (2010)
2. Asfari, O., Doan, B.-L., Bourda, Y., Sansonnet, J.-P.: Personalized Access to Contextual Information by using an Assistant for Query Reformulation. IARIA Journal, IntSys 2011 4(34) (2011)
3. Asfari, O.: Personnalisation et Adaptation de L'accès à L'information Contextuelle en utilisant un Assistant Intelligent. PhD thesis, Université Paris Sud - Paris XI, tel-00650115 - version 1 (September 19, 2011)
4. Allan, J.: Challenges in information retrieval and language modeling. In: Workshop Held at the Center for Intelligent Information Retrieval, University of Massachusetts, Amherst, SIGIR Forum vol. 37(1), pp. 31–47 (2003)
5. Bauer, T., Leake, D.: Real time user context modeling for information retrieval agents. In: CIKM 2001: Proceedings of the Tenth International Conference on Information and Knowledge Management, pp. 568–570. ACM, Atlante (2001)
6. Billsus, D., Hilbert, D., Maynes-Aminzade, D.: Improving proactive information systems. In: IUI 2005: Proceedings of the 10th International Conference on Intelligent User Interfaces, pp. 159–166. ACM, San Diego (2005)
7. Bai, J., Nie, J., Bouchard, H., Cao, H.: Using Query Contexts in Information Retrieval. In: SIGIR 2007, Amsterdam, Netherlands, July 23-27 (2007)
8. Bouchard, H., Nie, J.: Modèles de langue appliqués à la recherche d'information contextuelle. In: Proceedings of CORIA 2006 Conf. en Recherche d'Information et Applications, Lyon, pp. 213–224 (2006)
9. Conesa, J., Storey, V.C., Sugumaran, V.: Using Semantic Knowledge to Improve Web Query Processing. In: Kop, C., Fliedl, G., Mayr, H.C., Métais, E. (eds.) NLDB 2006. LNCS, vol. 3999, pp. 106–117. Springer, Heidelberg (2006)
10. Daoud, M., Tamine, L., Duy, D., Boughanem, M.: Contextual Query Classification For Personalizing Informational Search. In: Web Information Systems Engineering, Kerkennah Island, Sfax, Tunisia. ACM (Juin 2009)
11. Dey, A.K., Abowd, G.D.: Toward a better understanding of context and context-awareness. In: Workshop on the What, Who, Where, When, and How of Context-Awareness (1999)
12. Dumais, S., Cutrell, E., Cadiz, J.J., Jancke, G., Sarin, R., Robbins, D.C.: (Stuff I've Seen): A system for personal information retrieval and re-use. In: Proceedings of 26th ACM SIGIR 2003, Toronto, pp. 72–79 (July 2003)
13. Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., Ruppin, E.: Placing search in context: the concept revisited. In: WWW, Hong Kong (2001)
14. Henzinger, M., Chang, B.-W., Milch, B., Brin, S.: Query-free news search. In: The 12th International Conference on World Wide Web, Hungary (2003)
15. Hlaoua, L., Boughanem, M.: Towards Contextual and Structural Relevance Feedback in XML Retrieval. In: Beigbeder, M., Yee, W.G. (eds.) Workshop on Open Source Web Information Retrieval, Compiègne, pp. 35–38 (2005)
16. Ingwersen, P., Jäverlin, K.: Information retrieval in context. IRiX, ACM SIGIR Forum 39(2), 31–39 (2005)
17. Kelly, D., Teevan, J.: Implicit Feedback for Inferring User Preference: A Bibliography. SIGIR Forum 32(2), 18–28 (2003)

18. Kofod-Petersen, A., Cassens, J.: Using Activity Theory to Model Context Awareness. In: American Association for Artificial Intelligence, Berlin (2006)
19. Lafferty, J., Zhai, C.: Language models, query models, and risk minimization for information retrieval. In: SIGIR 2001, The 24th ACM International Conference on Research and Development in Information Retrieval, New York, pp. 111–119 (2001)
20. Lang, K.: NewsWeeder: learning to filter net news. In: The 12th International Conference on Machine Learning, San Mateo, USA, pp. 331–339 (1995)
21. Liu, X., Croft, W.B.: Statistical language modeling for information retrieval. In: Cronin, B. (ed.) Annual Review of Information Science and Technology, ch. 1, vol. 39 (2006)
22. Mylonas, P., Vallet, D., Castells, P., Fernandez, M., Avrithis, Y.: Personalized information retrieval based on context and ontological knowledge. Knowledge Engineering Review 23, 73–100 (2008)
23. Ratinov, L., Roth, D., Srikumar, V.: Conceptual search and text categorization. Technical Report UIUCDCS-R-2008-2932, UIUC, CS Dept. (2008)
24. Sauvagnat, K., Boughanem, M., Chrisment, C.: Answering content and structure-based queries on XML documents using relevance propagation. In: Information Systems, Numéro Spécial Special Issue, SPIRE 2004, vol. 31, pp. 621–635. Elsevier (2006)
25. Shen, X., Tan, B., Zhai, C.: Context-sensitive information retrieval using implicit feedback. In: SIGIR 2005: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 43–50. ACM, Brazil (2005)
26. Sugiyama, K., Hatano, K., Yoshikawa, M.: Adaptive Web Search Based on User Profile Constructed without Any Effort from Users. In: WWW, New York, USA, pp. 17–22 (2004)
27. Zhai, C., Lafferty, J.: Model-based feedback in the language modeling approach to information retrieval. In: Proceedings of the CIKM 2001 Conference, pp. 403–410 (2001)