Electronic Publishing '01 –
2001 in the Digital Publishing Odyssey
A. Hübler et al. (Eds.)
IOS Press, 2001

Improving Information Retrieval Performance by Experience Reuse

Lobna Jéribi

LISI (Laboratory of Information Science Engineering)
INSA Lyon France, 20 avenue A.Einstein, 69621 Villeurbanne
lobna.jeribi@lisi.insa-lyon.fr

Abstract: The goal to build a knowledge base, making "permanent" the user evaluations experiences on search results, constitutes the main motivation of this paper. In information retrieval systems, an experience is a search instance, represented by a search context and a set of evaluated document results. In this paper, we propose a search context model, characterized by the user's profile and query. Similarity functions between users' profiles and search contexts are defined, on the basis of the proposed model. The reuse process consists in expanding the initial user query by the documents terms contained in the similar instances. In contrast with the *Rocchio* method, these documents are those validated beforehand by users, having "similar" profiles as the current user, and being in a "similar" search situation. Our proposition enables to reduce system interactions with the user during his/her search session. These features were carried out in the COSYDOR (Cooperative System for Document Retrieval) project, based on *Intermedia* (Oracle 8i). Tests and evaluations are carried out using the test corpus of TREC (Text Retrieval Conference). The results show, for first search iterations, a significant improvement of performance compared to that of *Intermedia*. This work is carried out within the framework of a regional project, in which the sight deficient users constitute our application case. This research represents a considerable contribution for these users, considering their use difficulties of current information retrieval systems (too interactive systems, accessibility problems, etc).

0. Introduction

"Personalized" information retrieval systems, exploit "information about users" during retrieval, or filtering (*NewT* [MAE 94]), or query expansion (*smart*, [BUC 98]), or navigation process (*WebWatcher*, [ARM 95]). The user knowledge generally used in these systems, is often limited to some key words representing user interests, or some documents relevant to the user. However, using this knowledge, without defining its ***context*** (search situation, search goal, user features, etc.), reduces significantly its contribution to improve information retrieval results.

In a study carried out on intelligent system architectures and machine learning approaches for information retrieval, such as rings architectures [JER 98b] and multi-agents systems [JER 98a] [JER 2000a], we were directed towards the reuse techniques and instance based reasoning [JER 2001b]. This approach, based on the user reasoning cycle (intention, action, acquisition, evaluation), allows to carry out reasoning, to evaluate the document results automatically, and to improve them. This approach is relevant, because the user knowledge used is very "reliable" since "lived" and "evaluated" by this user. Moreover, the reuse approach enables to minimize the user interaction, which has a significant advantage for users having particular difficulties of access to information. Furthermore, the field of experience reuse, offers methods to specify and construct the

experience ***context***, so that the reuse is optimal [MIL 99].

We propose in this paper to build an instance memory of information retrieval instances. When the users search contexts are "similar", these instances are reused to improve user query formulation. The user profile is considered as a significant element of an instance search context. Thus we propose in this paper a formal representation of "user profile" model. We also define a representation of an information retrieval instance model, in order to evaluate the similarity between instances. The similarity evaluation enables to highlight candidate instances to be reused, from the instance memory.

Some related works of context definition and experience reuse were proposed in the early literature. RADIX project [COR 99] proposes the modeling of internet navigation sessions carried out by the user. These models are reused in order to suggest similar sessions to the user. CABRI-N [SMA 99] is a personalized image retrieval system. Smaïn proposes a modeling of user strategy during an information retrieval process. Retrieval sessions are memorized and reused to improve user strategy search.

In this paper, we present briefly a modeling study of the user during a search session, and a representation of a search instance or search situation. Then, we present our approach of instance reuse for query expansion. Afterwards, we expose our project context and our application case. Lastly, we present the results of our first tests and the prospects for evaluations.

1. Information retrieval instance Modeling

In this section, we firstly define the user profile model, representing a relevant feature of the retrieval instance. Next, a global information retrieval instance is proposed.

1.1 User profile Modeling

Intelligent information systems aim to automatically adapt to individual users. Hence, the development of appropriate user modeling techniques is of central importance. Algorithms for intelligent information agents typically draw on work from the information retrieval and machine learning communities. Both communities have previously explored the potential of established algorithms for user modeling purposes [BEL 97] [WEB 98]. However, "work in this field is still in its infancy" [BILL 99].

1.1.1 User knowledge

Intelligent systems for information access are typically aimed at assisting a user in his search for interesting or useful information. A large variety of agents that make use of machine learning techniques have been developed and presented in the literature (e.g. [PAZ 97]). Most of this work focuses on the acquisition of a precise model of the user information need. However, in order to build truly useful information retrieval systems we also need to be aware of the user's knowledge.

To define the specific user knowledge, that is exploited during a search session, we have exploited **cognitive approach** results [ALL 91]. We have classified the user knowledge into four ***knowledge categories***, ranked according to their *evolution* degree.

1. Cultural knowledge (features having *little* or *no evolution*)
2. Professional knowledge (features having *long term evolution*)
3. System knowledge (features having *mean term evolution*)

4. Search knowledge (features having *short term evolution*), related to the current search session. The Figure 1 shows knowledge user features.

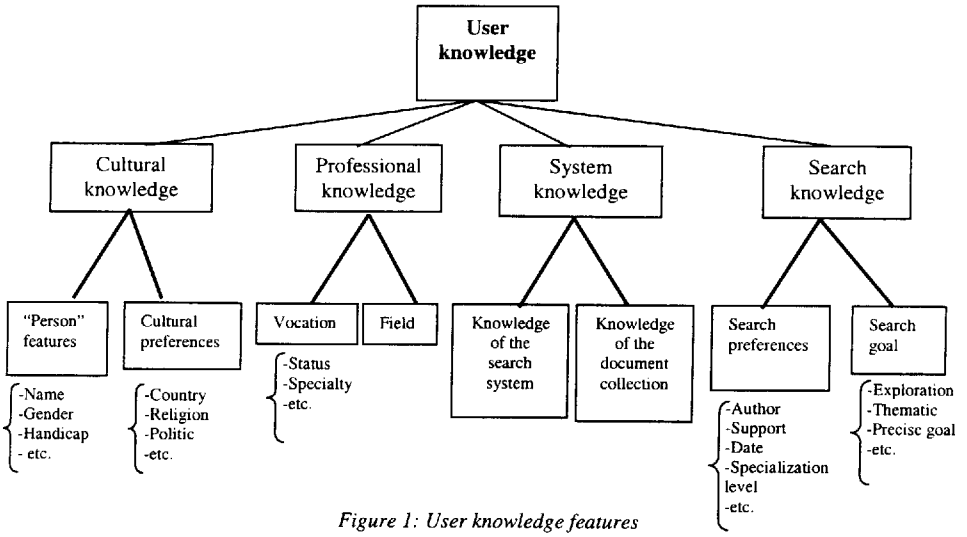

Figure 1: User knowledge features

Some features varies from an application case to another. These variations depend on the document, the search system and the user types. For example, in image retrieval systems, the search preferences include specific features as: plan, luminosity, contrast, colors, etc. Our specific model, concerning our use context, is presented in the next session (§4.1.2)

User knowledge features presented above, constitute a *generic model* of user profile, which *specific models* are related to the application cases used.

1.1.2 Representation formalism

The chosen representation formalism is the vector model. It is the formalism commonly used in both communities: information retrieval [SAL 86] and instance based reasoning [KOL 88]. The vector model presents several advantages in processing similarity between vectors.

Let $U=<U_1, U_2, U_3, U_4>$, be the vector representation of U, where U_i represents the i^{th} category of user knowledge U.

$\boldsymbol{U_i = \{a_{i,j}\}}$; $\forall j \in [1, n]$; a_{ij} represents the j^{th} attribute of the category U_i;

$\boldsymbol{a_{i,j} \in \{v_k\}}$; $\forall k \in [1, n]$; v_k represents the k^{th} possible instance of $a_{i,j}$

We define then the space Γ_i with dimension n, of vector U_i as follows:

$$\Gamma_i = < (v_k^j)^{j=1 \text{ to } n} >; \forall k \in [1, m]$$

1.1.3 Similarity function

We propose to memorize user retrieval experiences, in order to reuse them, when users have "similar" profiles. Thus, our first goal of formalizing the user model, is to define the "distance" between user profiles. The expression (1) shows the similarity function S_U, between two user profiles U^i and U^j

$$S_U(U^i, U^j) = \frac{s_1(U_1^i, U_1^j) + \nu\, s_2(U_2^i, U_2^j) + \kappa\, s_3(U_3^i, U_3^j) + \lambda\, s_4(U_4^i, U_4^j)}{\mu + \nu + \kappa + \lambda} \quad (1)$$

Where:

U_p^i is the vector representing the p^{th} category of U^j , which is the profile of the user j.

s_p: similarity function between vectors of the p^{th} ***category*** of U

$s_p \in [0, 1]$; $\mu, \nu, \kappa, \lambda \in [0, 1]$

$\mu, \nu, \kappa, \lambda$, represent the parameters enabling to "contextualize" the similarity.

Details about the expression (1) are presented in [JER 2001b].

1.2 Search instance modeling

1.2.1 Search instance representation

The results of various studies on search instance [JER 2001b], make highlight of following features of a search instance:

- The user profile represented by ***U***
- The user information need expressed by a query, represented by ***Q***
- The documents solutions represented by ***D***
- The evaluations ***E*** of relevancy of the documents ***D,*** given by the user U

Referring to the problem resolution field, the initial problem in information retrieval system is represented by the user profile ***U***, and his query ***Q***. Collected documents ***D*** represent the solution to the problem, and ***E*** the solution evaluation.

We propose a formal description of a search instance, carried out by a user, during an information retrieval session, (vector representation) as follows:

Instance = <U, Q, D, E>

- ***U*:** represents the user features during the search session: $U=<U_1, U_2, U_3, U_4>$
- ***Q***: As defined in information retrieval vector model [SAL 94]; giving the space Γ including all the corpus terms: $\Gamma = <t_1, t_2, t_3, ..., t_n>$, Q is the weighted term vector representing the user query, in the space E: $Q = <a_1, a_2, a_3, ..., a_n>$; a_i corresponds to the weight of the i^{th} term of the query
- $\boldsymbol{D} = \{d_i\}$; D is a set of documents d_i; $i \in [1, p]$; $\boldsymbol{p}$: number of documents evaluated by the user; $p = |D|$; $\boldsymbol{d_i} = <b_{i,1}, b_{i,2}, b_{i,3}, ..., b_{i,n}>$; $\boldsymbol{d_i}$ is the vector representation of a document in the defined space Γ_{corpus}; $\boldsymbol{b_{i,j}}$ represents the weight of the j^{th} term of the document d_i. $\boldsymbol{d_i}$ is the weighted term vector representing the document (or a part of the document) that the user have chosen and evaluated.
- ***E*** represents the evaluation given by the user U of the relevance of D according to Q

1.2.2 Instances similarity function

Giving the current instance represented by: $I_{current} = (U_{current}, Q_{current}, D_{current}, E_{current})$; We define the similarity function S_I between $I_{current}$ and another memorized instance I as shows the expression (2).

$$S_I(I_{current}, I) = S_U(U_{current}, U) * S_Q(Q_{current}, Q) \quad (2)$$

S_U was defined previously (§2.1.3)
Next we present the similarity function S_Q.

1.2.3 Similarity between queries and thesaurus use

The similarity of a query in a search instance, with the current user query, constitutes a discriminating criteria of choice of the instance to be reused. The question is how to quantify this similarity? As the queries are expressed in a vector space, to define their similarity, one is based on the functions of similarity suggested in the vector model in information retrieval field [SAL 94]. More precisely, the function of the cosine will constitute the basis of proposed similarity function.

The cosine function for similarity process consists in highlighting the joint terms of the two vectors, and making the sum of the products of the joint terms weights (expression (3)). The major disadvantage of this similarity function is that it is limited to the joint terms, and ignore "close" terms expressed in two vectors. This problem represents the major drawback in the Vector Space Model.

In order to mitigate the terms independence problem in the vector model, we propose to enrich the queries with their ***linguistic contexts***. The linguistic context enables to define the "semantically" related terms to the query terms. The linguistic context terms issue from a ***thesaurus*** (or linguistic knowledge base specialized in the application field), allowing to highlight semantic links between the terms. In the case of the thesaurus, it has three principal types of link: synonymy, hierarchy and neighborhood .
Giving the current user query Q_c, let us define
$Q_c = <a_{1c}, a_{2c}, a_{3c}, ..., a_{nc}> + <a_{(n+1)c}, a_{(n+2)c}, ..., a_{(n+m)c}>$ (Linguistic context)
$<a_{1c}, a_{2c}, a_{3c}, ..., a_{nc}>$ represent the weights of terms of the corpus space Γ_{corpus}
$<a_{(n+1)c}, a_{(n+2)c}, ..., a_{(n+m)c}>$ represent the weights of terms of the thesaurus space $\Gamma_{thesaurus}$
Supposing t $(t = n+m)$ is the dimension of the total space terms, the simplified expression of Q_c is: $Q_c = <a_{1c}, a_{2c}, a_{3c}, ..., a_{tc}>$; where $a_{ic} \in [0,1]$
$Sim(Q_r, Q_c)$ is the similarity function between the current query Q_c and another query Q_r; where $Q_r = <a_{1r}, a_{2r}, a_{3r}, ..., a_{tr}>$, represented in the same space terms.
The proposed similarity function, based on the Cosine product, is expressed as follows:

$$Sim(Q_r, Q_c) = \frac{\sum_{i,j=1}^{t} a_{ic} a_{jr}}{\sqrt{\sum_{i=1}^{t} a_{ic}^2 \sum_{i=1}^{t} a_{ir}^2}} \qquad (3)$$

$Sim(Q_r, Q_c) \in [0,1]$; $Sim(Q_r, Q_c) = 1$ when $Q_r = Q_c$

1.3 Confidence degree of a search instance

The search instance (judged similar to the current one) is reused with a confidence degree. The concept of "confidence" is inspired by the multi-agents approaches. It is taken into account during the communication and the co-operation inter-agents [SHE 93]. It depends on the agent "performance". The performance is calculated on the basis of the evaluation results of agent contribution [MOU 96]. This implies a system tractability, and a global evaluation of an iteration success or failure. The agents confidence degree also intervenes in the system maintenance, in order to eliminate non-active and non confident agents.

In our context, we enrich the instance case, by a confidence degree which reflects the relevance of the instance.

The confidence degree of the instance is a function φ which depends on:

- S_I: the distance between the current instance and the useful instance. φ decreases when the S_I distance increases.

- The session date of the instance. Considering the fast evolution of the fields of user interest and his search profile, we consider that a "recent" case is more "useful" comparing to a "non recent" instance. Thus, φ increases when (current date - date session) decreases.
- The instance performance: To each instance, we assign an index of performance which will reflect its rate of contribution to improve the results of retrieval. After the reuse phase, the similar instance having contributed to the reasoning will see its performance to increase or decrease according to the quality of the documentary answers (principle of reward / penalty). In our case, since the relevance feedback is not obligatory for the user, it is difficult to evaluate the success or the failure of an iteration. In order to simplify this study, we propose to carry out an average of the evaluations E given by the user, to evaluate overall results of the search iteration.

2. Experience reuse for query expansion

Before presenting our proposition of query expansion based on experience reuse, we introduce in the next paragraph the classical approaches of query expansion based on relevance feedback.

2.1 Relevance feedback for query expansion

2.1.1 Relevance feedback

Query based information retrieval is a sequential process. It is rare that the initial search retrieves exactly what the user is looking for, therefore he or she can generally benefit from submitted a revised query based on what has been learned about the topic from the initial set of relevant documents. This process is known as relevance feedback [ROC 71]. One can modify the query by adding or deleting search terms and/or modifying term weights. Clearly, manual query construction and revision is a relatively arduous and complex task. Given a relevant document, it is difficult to determine what terms to add to the query and which terms are more or less important. Therefore, the most desirable approach is to develop an automatic relevance feedback process so that the system does the work of revising the query.

An automatic relevance feedback system can be designed like this: the system processes the initial query and returns a list of documents ranked in order of their predicted relevance to the user. The user examines a few of the highest ranking documents, determines whether or not they are relevant, and sends this information back the system. The system uses the analyzed documents to automatically construct a revised query and produces a new ranking of the remaining documents in the collection. The user can examine more documents and repeat the relevance feedback process as often as desired.

2.1.2 Feedback using the Vector Space Model: Rocchio

The VSM is ideally suited for automatic relevance feedback since it accepts free text input for the query. Therefore, we can simply incorporate some or all of the text from the relevant documents directly into the query. Non relevant documents can be utilized in a similar fashion. One of the most successful relevance feedback strategies was developed by *Rocchio* [ROC 71], and works as follows:

$$Q' = \alpha Q + \beta\left(\frac{1}{|D_r|}\sum_{d_i \in D_r} d_i\right) - \gamma\left(\frac{1}{|D_n|}\sum_{d_j \in D_n} d_j\right) \quad (4)$$

$\alpha, \beta, \gamma \in [0,1]$; d_i is weighted term vector representing a collected document
$|D_r|$ cardinality of relevant documents set
$|D_n|$ cardinality of non-relevant documents set
This strategy simply adds a weighted sum of the relevant document vectors and subtracts a weighted sum of the non relevant documents from the query. Relevance feedback using this strategy produces a very large improvement in retrieval performance (from 20-80% or more, depending on the collection) [SAL 86] [HAR 92] [AAL 92].

Adding the full document text directly into the query does have its drawbacks. The number of terms in the query grows rapidly with the addition of evaluated documents, causing searches to take longer and longer. A full search must be repeated for each iteration of relevance feedback. Since relevance feedback is supposed to be an interactive process, the system must be able to return feedback results in a relatively short period of time.

2.1.3 Drawbacks of Rocchio method based on relevance feedback

Although relevance feedback has proven to be an effective way to improve information retrieval performance, it is rarely used in practice [NIE 96]. This mechanism has been incorporated into several online search engines, but few users actually use it. Nie thinks that an important reason is the short term effect of a relevance feedback. A user has to make great efforts in evaluating documents, but the evaluation only has an effort on a single query. Once a new query is input, new evaluations have to be made. From the user's point of view, it is simply not worth the effort. However, we think users are ready to make the effort when the effect is permanent. Therefore, we will suggest a way to adjust the system's knowledge according to relevance feedback.

2.2 Knowledge base for relevance feedback

2.2.1 Relevance feedback and Instance reuse

Our proposal is based on the *Rocchio* method of query expansion. The principle approach of the proposed solution, consists on "completing" the documents used for the query expansion issued from the current relevance feedback, by the evaluated documents extracted from the previous search instances. On the basis of these documents -coming from both sources-, we apply the *Rocchio* approach of query expansion, as illustrated in Figure 2. Thus, terms added to the user query could result from the instance base documents, evaluated previously by the user or other users being in similar search contexts, and having similar profiles.

However, these two document sources (showed in Figure 2) are not independent, since the documents evaluated during the previous search iteration (Relevance feedback) represent also an instance contained in the memory of instances.

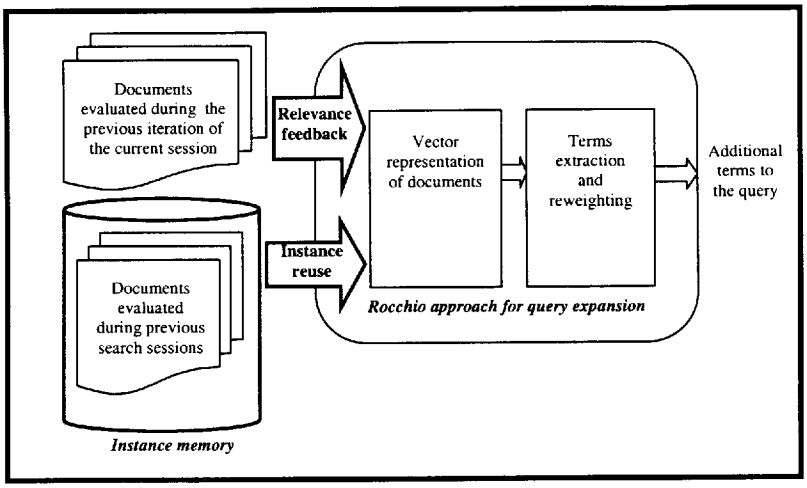

Figure 2: Instance reuse as complement to relevance feedback for query expansion

The interest of our proposal is primarily justified when no relevance feedback is made by the user during his search session. In this case, the reuse of the instance base constitutes an interesting alternative for the query expansion. Moreover, this enables to give a certain "freedom" to the user, because he doesn't have to evaluate documents relevancy to obtain system help to express his query.

Nevertheless, the instances reused cannot contribute in the same way for query expansion. For this, we propose to weight this contribution, according to the "confidence degree" of the reused instance. This concept will be detailed in the following paragraph.

2.2.2 Adapting Rocchio method for query expansion based on experience reuse

Giving the current instance represented by: $I_{current} = (U_{current}, Q_{current}, D_{current}, E_{current})$;

$\boldsymbol{Instance}_{similar} = (U_{similar}, Q_{similar}, D_{similar}, E_{similar})$: the most similar instance to the current one $(\boldsymbol{I}_{current})$, with the confidence degree $\varphi_{similar}$.

$D_{similar}$ is a set of documents d_i; $\boldsymbol{d}_i$ is the weighted term vector representing the document (or a part of the document) that the user have chosen to evaluate; $\boldsymbol{d}_i = <b_{i,1}, b_{i,2}..., b_{i,n}>$;

$E_{similar}$ represents the evaluation given by the user, on the relevancy of $D_{similar}$ according to $Q_{similar}$.

The proposed expression of query expansion of $Q_{current}$, consists on adapting the *Rocchio* approach, by reusing the evaluated documents $D_{similar}$ extracted from the instance $I_{similar}$.

$$Q'_{current} = \alpha \times Q_{similar} + \varphi_{similar} \times \beta \left[\frac{1}{|D_{similar-r}|} \sum_{d \in D_{similar-r}} d_i e_i \right] + \varphi_{similar} \times \gamma \left[\frac{1}{|D_{similar-n}|} \sum_{d \in D_{similar-n}} d_i e_i \right] \quad (4)$$

$d_i \in D_{similar}$; $e_i \in E_{similar}$; $e_i \in [-1, 1]$; e_i corresponds to the evaluation of d_i

$\alpha, \beta, \gamma \in [0,1]$; α, β, γ are coefficients defined in the *Rocchio* expression (3).

$\varphi_{similar} \in [0,1]$; $\varphi_{similar}$ *confidence degree of* $I_{similar}$

$D_{similar\text{-}r}$ represent relevant document set of the instance $I_{similar}$
$D_{similar\text{-}n}$ represent non relevant document set of the instance $I_{similar}$
The new expression for query expansion is based on documents extracted from the instance base. The added terms result from documents evaluated by users when they were in similar search situations. However, we give more importance to the contribution of the instance coming from a relevance feedback compared to those coming from the instance memory. Hence, we propose to decrease by φ *(confidence degree of the similar instance)* the contribution of this instance in the proposed expression (4).

Nevertheless, when the used instance corresponds to a relevance feedback (documents evaluated by the same user during the same session of search), the confidence degree of this instance is maximal ($\varphi = 1$). In this case, the expression enables to apply the classical *Rocchio* approach.

2.2.3 Combining learning to the Rocchio approach

As presented above, our system allows two types of learning:
Long term learning thanks to the instance memory. The reuse approach and instance based learning allows the user to benefit from the aid of the system without having to interact and evaluate during each session, the collected documents (contrary to traditional methods of query expansion based on relevance feedback).

However, our solution is optimal when the number of instances of the instance base is significantly important. In the opposite case, the system functions, as *Rocchio,* are based on relevance feedback. The effectiveness of the instance base is well exploited when the user population using the system have common interests and carry out more exploitable common searches then other users. This is a classical constraint in the co-operative systems.

- ***Short term learning*** thanks to the training by reward / penalty of the search instances. The system evolves according to the failure / success of the proposed solutions.

3. Application case: sight deficient users

3.1 Project context

This research is carried out within the framework of a project of the Rhone-Alpes department. The project is entitled: "pedagogic information access systems for sightless users: use of speech and sound". From existing virtual libraries specialized in engineering sciences (scientific documents), the goal of this project is to produce an "intelligent" tool of information retrieval, adapted to the visually defective users. At this level, we are interested on the one hand in the design of an information retrieval system allowing to help the user to better formulate his information need on the one hand, and to offer to him a better access to the documents.

These specific users have particular information access difficulties, which is added to the traditional constraints of information retrieval (query formulation difficulties, irrelevant answers, etc). These accessibility difficulties are accentuated in the context of scientific documents (Braille transcribing and/or voice synthesis). This work concerning the accessibility aspects (Design of accessible IHM, transcribing Braille and/or vocal), was carried out in collaboration with the other partners of the project, and is detailed in [JER 2000b]. A work in close collaboration with "sight deficient" scientists enabled us to note that it becomes crucial for these users to have dedicated systems, taking in account their use profile. It becomes tiresome to these handicapped users to exploit the current information

retrieval. This is due not only to the accessibility problems, but also to the imposed use logic of these systems (too interactive system, visual criteria, navigational logic, etc).

3.2 Sight deficient user profile

In this work, we applied the user model (§ 2) to the specific case of the sight deficient users, under a university context using scientific documents. In a study on the behavior of these users [BER 98], we highlighted the following features:

- The visual user group *(U_1)*: Sighted, Sight deficient, blind user
- The user category or function *(U_2)*: Student, Teacher, Researcher, Engineer
- The field of interest *(U_2)*: Mathematical, Natural science, Biology, etc.
- The knowledge about the system *(U_3)*
- The documentary preferences *($U_{4.1}$)*: restitution preferences (Braille, vocal), abstract extraction, document support, etc.
- The finality or goal of the search *($U_{4.2}$)*: bibliographical synthesis, technological survey, project study, etc.

These features are taken into account to evaluate the similarity of user profiles, by our prototype COSYDOR.

4. Experiments and evaluations using test corpus

4.1 COSYDOR prototype presentation

Our prototype COSYDOR (Cooperative SYstem for DOcument Retrieval) is based on *Intermedia* de Oracle 8i. We enriched *Intermedia* by an intelligent layer (developed in java) enabling the users query expansion and the management of the instance base (Figure 3).

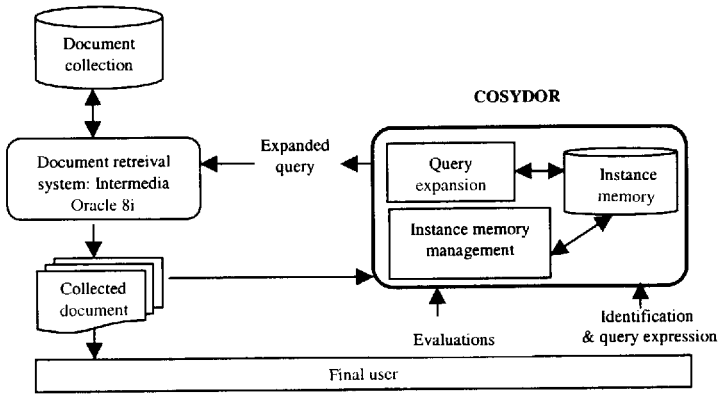

Figure 3: COSYDOR prototype

Intermedia is a textual DBMS, using linguistic tools (thesaurus, lexicon, etc.) for documents and queries representation. The choice of this tool results from a comparative survey on several information retrieval systems [JER 2001a]. *Intermedia* proved to be most relevant in our context. One of its advantages, is to offer paragraph extraction functionality,

enabling to present document "views" during the document restitution to the user. This makes the user evaluation more precise on the one hand, and makes easier the access to the long documents for the sight deficient users on the other hand.

4.2 Test and evaluations

4.2.1 TREC test corpus

In order to test and to evaluate the contribution of our system, we have used a TREC corpus of test. TREC[1] (Text Retrieval Conference) is an American organization which provides a corpus of tests and common procedures of analysis of performance. Among this base of tests, we used and indexed a whole of 7000 documents, in format HTML, relatively long (approximately 600 mots/document) and specialized in the biomedical field. We reused the examples of search provided by TREC, and calculated the various indices of performance of our system: the rate of precision (a number of collected relevant documents / a total number of collected documents) and the rate of recall (a number of collected relevant documents / a total number of relevant documents of the documentary corpus).
In this base of tests, we used the expansion possibilities of our system to note our work contribution and improvement, compared to the *Intermedia* results. Our procedure of tests consists on iterations of retrieval, where query expansion are evaluated by "initiated" users. For each iteration, the rate of recall and the precision of the answers are processed.

4.2.2 Evaluations

In the carried out tests, we note that the rates of precision/recall are improved comparing to iteration 0, which corresponds to *Intermedia* performance. Thus, our solution enables to optimize *Intermedia* results. The experimental average of precision/recall rates for information retrieval is generally around 0,3 [NIE 96]. These rates are often antagonistic, which justifies the shape of the obtained curves (Figure 3). We noticed that this is related to the type of expansion carried out. Indeed, we noted that the positive query expansion (terms extracted from relevant documents) causes a significant increase in precision rate (iteration 2 and 4). However, negative expansions were less effective.

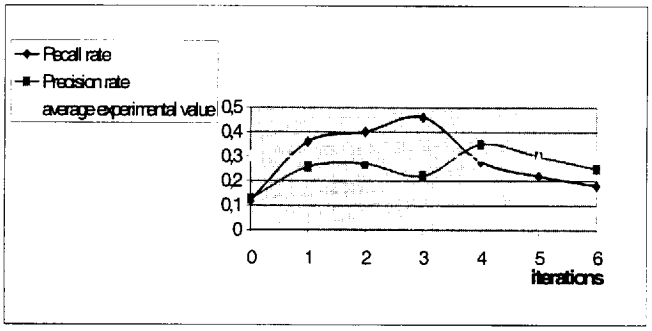

Figure 4: Performance evaluation of COSYDOR

Beyond the fourth iteration, we observed that the rates of recall / precision decreased. This is due to the length (a number of terms) of the query. Indeed, beyond a maximum number of added terms (5 terms in our context), the performance of expansion falls.

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

Moreover, the quantitative tests results enabled us to note the significant contribution to the system performance, of the weights assigned to the terms of the initial query.

The first evaluation results, although very encouraging, must be moderated by the limited number of tests (ten queries). Moreover, the users who contributed to the tests have very close profiles, and were previously initiated to the system use. It would be interesting to carry out these tests on a broader sample of users, having different profiles and having a visual handicap.

The second part of our evaluation consists on the comparison of COSYDOR performances versus other information retrieval systems using manual and automatic query expansion. The results of these systems were provided to us by TREC. First experimentation of these comparative tests are currently carried out in our laboratory.

5. Conclusion

The goal to build a knowledge base making "permanent" the user evaluations on search results, represents our first work motivation.

The search instances memory offers several improvements, compared to the *Rocchio* method of query expansion based exclusively on the relevance feedback.

The first advantage of the memorization consists to present an alternative solution to the *Rocchio* method, so that the user can be offered the system aid, without having to interact and to evaluate the collected documents, during each retrieval session. In fact, traditional methods of query expansion based on relevance feedback, are applied only after user interaction and after an evaluation of the retrieved documents relevancy. This approach "ignores" all knowledge chunks of the former search situations made by the user or by other users having "similar" profiles, and being in a close search situation.

The second advantage of the instance memory, is to enable the system to learn progressively, with experience acquisition. Indeed, the query expansion based on relevance feedback methods, constrain the system to make the same training during each session of search, from a search iteration to another, in order to arrive to an optimal query. Whereas, it would be interesting to avoid these repetitive stages, by memorizing the search instances and their corresponding evaluations.

The third advantage is related to user psychology. It will be much easier to get user to invest in the effort of evaluations, if they know that these will be permanent and reusable. This "reluctance" constitutes a real obstacle in the use of systems based on relevance feedback [NIE 96].

However, memorizing and reusing evaluations is a difficult task. It requires the ability to reproduce all the context of documents evaluation carried out by users, so that the memorized knowledge is representative of the moment, and the reuse is suitable. This justifies all the effort of contextual representation and modeling of search instance, presented in the first part of this paper.

These features were carried out in the COSYDOR system, implemented in Java, based on *Intermedia* (Oracle 8i). Tests and evaluations are carried out using the test corpus of TREC (7000 documents in the biomedical field) and their common procedures of performance analysis. The results show, for first search iterations, a significant improvement of performance compared to that of *Intermedia*. However, our sample of users is not sufficiently representative. Thus, an obvious direction for further research is to widen the sample of users in the experimental tests.

This work is carried out within the framework of a regional project, in which the sight deficient users constitute our application case. This research represents a considerable

contribution for these users, considering their use difficulties of the current information retrieval systems (too interactive systems, accessibility problems, etc).

6. Acknowledgments

We thank Robert Laurini, Beatrice Rumpler, Abraham Alvarez for their reviewing and remarks.

7. References

[ALL 91] Allen N. Cognitive Research in information science: implication for design. *Annual review of information science and technology,* 1991, vol. 26, pp. 3-37

[ARM 95] Armstrong, R.; Freitag, T.; Joachims, T.; and Mitchell, T. WebWatcher: A learning apprentice for the World Wide Web, *In Proceedings 1995 AAAI Spring Symposium on Information Gathering from Heterogeneous distributed Environments*, AAAI Press.

[BEL 97] Belkin, N. Kay, J., Tasso, C. Special issue on User Modelling and Information Filtering . *User modelling and User adapted Interaction*, 1997, vol 7(3), pp. 313-331.

[BER 98] Bergère, T., Portalier, S. (1998). Modélisation du comportement de l'utilisateur déficient visuel. Workshop NTI SPI & santé. Chassey le Camp (France).

[BIL 99] Billsus, D., Pazzani, M. A hybrid User Model for New Story Classification. *Proceedings of the Seventh International Conference on User Modelling (UM'99),* Banff, Canada, 20-24 Juin 1999.

[BUC 98] Buckley, C., Mitra, M., Walz, J., Cardie, C. Using clustering and superconcepts within SMART: TREC 6. In E. M. Voorhees and D. K. Harman, editors, *The Sixth Text REtrieval Conference (TREC-6).* NIST Special Publication 500-240, 1998.

[COR 99] Corvaisier, F., Mille, A., Pinon, J.M. – Recherche assistée de documents indexés sur l'expérience (RADIX): Mesures de similarité des épisodes de recherche sur le WEB. *IC'99 Ingénierie des connaissances.*

[JER 2001a] Jéribi, L., Rumpler, B., Pinon, J.M. Recherche et traitement documentaire: Etude comparative d'outils existants. To apprear in : Techniques et Sciences Informatiques review, Edition Hermès.

[JER 2001b] Jéribi, L., Rumpler, B., Pinon J.M. Système d'aide à la recherche et l'interrogation de bases documentaires, basé sur la réutilisation d'expériences., *InforSID 7-9 juin 2001, actes des conférences, XIXème congrès,* 2001.

[JER 2000a] Jéribi, L., Rumpler, B., Pinon J.M. Personalised information retrieval in specialised virtual libraries. *New Library Worl review,* MCB press, VOL 101 N° 1153, 2000, p21-27.

[JER 2000b] Jéribi, L., Rumpler, B., Pinon J.M. Intelligent System for document retrieval and access to scientific documents for visually deficient users. 12-14 Avril 2000. *Conférence internationale de Recherche d'Information Assistée par Ordinateur, RIAO'2000: Accès à l'information multimédia par le contenu*, Paris (France). ISBN 2-905450-07-X, p 870-884.

[JER 98a] Jéribi, L., Rumpler, B., Pinon J.M. Intelligent retrieval in virtual libraries for education and training. 20-22Avril 1998. *ICCC/IFIP International Conference on Electronic Publishing: Towards the information rich society.* Central European University, Budapest (Hungry). p 144 – 157.

[JER 98b] Jéribi, L., Rumpler, B., Caelen J., Pinon J.M. COSYDOR: CO-operative SYstem for DOcument Retrieval. Application Case: Access System to Scientific Document for Sightless Users. 08-10 Décembre 1998. *MCSEAI'98: 5th Maghrebian Conference on Software Engineering and Artificial Intelligence*, Tunis (Tunisia). ISBN 9973-831-00-4, pp. 77 – 91.

[KOL 88] Kolodner J. L. (édité par). – *Workshop on case-based Reasoning, DARPA 88.* Clearwater, Florida, Morgan-Kaufmann, San Mateo, 1988.

[MAE 94] Maes, P. Agents that reduce work and information overload; *CACM*, 37 (7): 31-40,146-147

[MIL 99] Mille, A., Fuchs, B., Chiron, B. Raisonnement fondé sur l'expérience en supervision industrielle. *Revue d'Intelligence Artificielle* 13, 1999, p. 97-128.

[MOU 96] Moukas, A. G. Amalthaea: information discovery and filtering using a multi-agent evolving ecosystem. *In Proceedings of the conference on Practical Application of Intelligent Agents and Multi-agent Technology.*

[NIE 96] Nie, J. Y., Brisebois, M., Lepage, F., (1996). Information retrieval as counterfactual, *The computer journal*, 38(8): 643-657.

[PAZ 97] Pazzani, M., Billsus, D. (1997). Learning and revising user profiles: The identification of interesting web sites. *Machine learning* 27: 313-331.

[ROC 71] Rocchio, J.J. Relevance feedback in formation retrieval. In Gerard Salton editor, *The SMART retrieval system: Experiments in Automatic Document Proceedings,* pages 313-323. Prentice Hall, 1971.

[SAL 86] Salton, G. Text-retrieval systems, *Communication of the ACM*. July 1986, n° 7, p. 648-655.

[SAL 94] Salton, G., Allan, J., Buckley, C. Automatic structuring and retrieval of large text files, *Communication of the ACM*. February 1994, vol 37, n° 37, p. 97-109.

[SHE 93] Sheth, B., Maes, P. Evolving Agents for Personalised Information Filtering. In Proceedings of the Ninth IEEE Conference on Artificial Intelligence Applications.

[SMA 99] Smaïl, M. Recherche de régularités dans une mémoire de sessions de recherche d'information documentaire, *InforSID 2-4 juin 1999, actes des conférences, XVIIème congrès*, 1999.

[WEB 98] Webb, G. Special issue on Machine Learning for User Modelling. *User Modelling and user Adapted Interaction*, vol8 (1-2), Kluwer Academic Publishers.