# A Semi-Supervised Incremental Algorithm to Automatically Formulate Topical Queries

Carlos M. Lorenzetti Ana G. Maguitman

*Grupo de Investigación en Recuperación de Información y Gestión del Conocimiento*
*LIDIA - Laboratorio de Investigación y Desarrollo en Inteligencia Artificial*

*Departamento de Ciencias e Ingeniería de la Computación*
*Universidad Nacional del Sur, Av. Alem 1253, (8000) Bahía Blanca, Argentina*
*phone: 54-291-4595135 fax: 54-291-4595136*

*Email:* {cml,agm}@cs.uns.edu.ar

## Abstract

The quality of the material collected by a context-based Web search systems is highly dependant on the vocabulary used to generate the search queries. This paper proposes to apply a semi-supervised algorithm to incrementally learn terms that can help bridge the terminology gap existing between the user's information needs and the relevant documents' vocabulary. The learning strategy uses an incrementally-retrieved, topic-dependent selection of Web documents for term-weight reinforcement reflecting the aptness of the terms in describing and discriminating the topic of the user context. The new algorithm learns new descriptors by searching for terms that tend to occur often in relevant documents, and learns good discriminators by identifying terms that tend to occur only in the context of the given topic. The enriched vocabulary allows the formulation of search queries that are more effective than those queries generated directly using terms from the initial topic description. An evaluation on a large collection of topics using a standard and two ad-hoc performance evaluation metrics suggests that the proposed technique is superior to a baseline and other existing query reformulation techniques.

*Key words:* Web search, context, topical queries, query formulation

## 1. Introduction

A user's information need is usually situated within a thematic context. For example, if the user is editing or reading a document on a specific topic, he may be willing to explore new material related to that topic. Context-based search is the process of seeking information related to a user's thematic context [7, 21, 17, 26]. Meaningful automatic context-based search can only be achieved if the semantics of the terms in the context under analysis is reflected in the search queries. From a pragmatic perspective, terms acquire meaning from the way they are used and from their co-occurrence with other terms. Therefore, mining large corpora (such as the World Wide Web) guided by the user's context can help uncover the meaning of a user's information request and to identify good terms to incrementally refine queries.

Attempting to find the best subsets of terms to create appropriate queries is a combinatorial problem. The situation worsens when we deal with an open search space (i.e., when other terms that are not part of the current context vocabulary can be part of the queries). The need to use terms that are not part of the current context is not an atypical situation when attempting to tune queries based on a small context description and a large external corpus. We can think of this query tuning process as a by-product of learning a better vocabulary to characterize the topic under analysis and the user's information needs.

### *1.1. Research Questions*

This paper presents general techniques for incrementally learning important terms associated with a thematic context. Specifically, we are studying three questions:

1. Can the user context be usefully exploited to access relevant material on the Web?
2. Can a set of context-specific terms be incrementally refined, based on the analysis of search results?
3. Are the context-specific terms learned by incremental methods better query terms than those identified by classical information retrieval (IR) techniques or classical query reformulation methods?

The contribution of this work is a semi-supervised algorithm that incrementally learns new vocabularies with the purpose of tuning queries. The goal for the queries is to reflect contextual information and to effectively retrieve semantically related material when posed to a search interface. In our work we use a standard and two ad-hoc performance evaluation measures to assess whether the queries generated by the proposed methods are better than those generated using other approaches.

### *1.2. Background and Related Work*

To access relevant information, appropriate queries must be formed. In text-based Web search, users' information needs and candidate text resources are typically characterized by terms. Substantial experimental evidence supports the effectiveness of using weights to reflect relative term importance for traditional IR [30, 29]. However, as has been discussed by a number of sources, issues arise when attempting to apply conventional IR schemes for measuring term importance to systems for searching Web data [16, 4]. One difficulty is that methods for automatic query formulation for Web search do not have access to a full predefined collection of documents, raising questions about the suitability of classical IR schemes for measuring term importance when searching the Web. In addition, the importance of a given term depends on the task at hand; the notion of term importance has different nuances depending on whether the term is needed for query construction, index generation, document summarization or similarity assessment. For example, a term which is a useful descriptor for the content of a document, and therefore useful in similarity judgments, may lack discriminating power, rendering it ineffective as a query term, due to low precision of search results, unless it is combined with other terms which can discriminate between good and bad results [9].

The IR community has investigated the roles of terms as descriptors and discriminators for several decades. Since Sparck Jones' seminal work on the statistical interpretation of term specificity [13], term discriminating power has often been interpreted statistically, as a function of term use. Similarly, the importance of terms as content descriptors has been traditionally estimated by measuring the frequency of a term in a document. The combination of descriptors and discriminators gives rise to schemes for measuring term relevance such as the familiar *term frequency inverse document frequency* (TFIDF) weighting model [30].

On the other hand, much work has addressed the problem of computing the informativeness of a term across a corpus (e.g., [1, 27, 8]). Once the informativeness of a collection of terms is computed, better queries can be formulated.

Query tuning is usually achieved by replacing or extending the terms of a query, or by adjusting the weights of a query vector. Relevance feedback is a query refinement mechanism used to tune queries based on the relevance assessments of the query's results. A driving hypothesis for relevance feedback methods is that it may be difficult to formulate a good query when the collection of documents is not known in advance, but it is easy to judge particular documents, and so it makes sense to engage in an iterative query refinement process. A typical relevance feedback scenario will involve the following steps:

**Step 1:** A query is formulated.

**Step 2:** The system returns an initial set of results.

**Step 3:** A relevance assessment on the returned results is issued (relevance feedback).

**Step 4:** The system computes a better representation of the information needs based on this feedback.

**Step 5:** The system returns a revised set of results.

Depending on the level of automation of step 3 we can distinguish three forms of feedback:

- **Supervised Feedback**: requires explicit feedback, which is typically obtained from users who indicate the relevance of each of the retrieved documents (e.g., [28, 34]).

- **Unsupervised Feedback**: it applies blind relevance feedback, and typically assumes that the top $k$ documents returned by a search process are relevant (e.g., [6]).

- **Semi-supervised Feedback**: the relevance of a document is inferred by the system. A common approach is to monitor the user behavior (e.g., documents selected for viewing or time spent viewing a document [32]). Provided that the information seeking process is performed within a thematic context, another automatic way to infer the relevance of a document is by computing the similarity of the document to the user's current context (e.g., [14]).

The best-known algorithm for relevance feedback has been proposed by Rocchio [28]. Given an initial query vector $\overrightarrow{q}$ a modified query $\overrightarrow{q_m}$ is computed as follows:

$$\overrightarrow{q_m} = \alpha \overrightarrow{q} + \beta \sum_{\overrightarrow{d_j} \in D_r} \overrightarrow{d_j} - \gamma \sum_{\overrightarrow{d_j} \in D_n} \overrightarrow{d_j}.$$

where $D_r$ and $D_n$ are the sets of relevant and non-relevant documents respectively and $\alpha$, $\beta$ and $\gamma$ are tuning parameters. A common strategy is to set $\alpha$ and $\beta$ to a value greater than 0 and $\gamma$ to 0, which yields a positive feedback strategy. When user relevance judgments are unavailable, the set $D_r$ is initialized with the top $k$ retrieved documents and $D_n$ is set to $\emptyset$. This yields an unsupervised relevance feedback method.

Several successors of the Rocchio's method have been proposed with varying success. One of them is selective query expansion [2], which monitors the evolution of the retrieved material and is disabled if query expansion appears to have a negative impact on the retrieval performance. Other successors of the Rocchio's method use an external collection different from the target collection to identify good terms for query expansion. The refined query is then used to retrieve the final set of documents from the target collection [18]. A successful generalization of the Rocchio's method is the Divergence from Randomness mechanism with Bose-Einstein statistics (Bo1-DFR) [1]. To apply this model, we first need to assign weights to terms based on their informativeness. This is estimated by the divergence of its distribution in the top-ranked documents from a random distribution as follows:

$$w(t) = tf_x.log_2 \frac{1 + P_n}{P_n} + log_2(1 + P_n)$$

where $tf_x$ is the frequency of the query term in the top-ranked documents and $P_n$ is the proportion of documents in the collection that contain $t$. Finally, the query is expanded by merging the most informative terms with the original query terms.

The main problem of the above query tuning methods is that their effectiveness is correlated with the quality of the top ranked documents returned by the first-pass retrieval. On the other hand, if a thematic context is available, the query refinement process can be guided by computing an estimation of the quality of the retrieved documents. This estimation can be used to predict which terms can help refine subsequent queries.

During the last years several techniques that formulate queries from the user context have been proposed [7, 17]. Other methods support the query expansion and refinement process through a query or browsing interface requiring explicit user intervention [31, 5]. Limited work, however, has been done on semi-supervised methods that simultaneously take advantage of the user context and results returned from a corpus to refine queries. Next section presents our proposal to tune topical queries based on the analysis of the terms found in the user context and in the incrementally retrieved results.

## 2. A Novel Framework for Query Term Selection

A central question addressed in our work is how to learn context-specific terms based on the user current task and an open collection of incrementally retrieved Web documents. In what follows, we will assume that a user task is represented as a set of cohesive terms summarizing the topic of the user context. Consider for example a topic involving the *Java Virtual Machine*, described by the following set of terms:

```
java        virtual      machine   programming   language
computers   netbeans     applets   ruby          code
sun         technology   source    jvm           jdk
```

Context-specific terms may play different roles. For example, the term *java* is a good descriptor of the topic for a general audience. On the other hand, terms such as *jvm* and *jdk*—which stand for "Java Virtual Machine" and "Java Development Kit"—may not be good descriptors of the topic for that audience, but are effective in bringing information similar to the topic when presented in a query. Therefore, *jvm* and *jdk* are good discriminators of that topic.

In [22] we proposed to study the descriptive and discriminating power of a term based on its distribution across the topics of pages returned by a search engine. In that proposal the search space is the full Web and the analysis of the descriptive or discriminating power of a term is limited to a small collection of documents—incremental retrievals—that is built up over time and changes dynamically. Unlike traditional information retrieval schemes, which analyze a predefined collection of documents and search that collection, our methods use limited information to assess the importance of terms and documents as well as to manage decisions about which terms to retain for further analysis, which ones to discard, and which additional queries to generate.

To distinguish between topic descriptors and discriminators we argue that *good topic descriptors* can be found by looking for terms that occur often in documents related to the given topic. On the other hand, *good topic discriminators* can be found by looking for terms that occur only in documents related to the given topic. Both topic descriptors and discriminators are important as query terms. Because topic descriptors occur often in relevant pages, using them as query terms alleviates the false-negative match problem. Similarly, good topic discriminators occur primarily in relevant pages, and therefore using them as query terms helps reduce the false-positive match problem.

### *2.1. Computing Descriptive and Discriminating Power*

As a first approximation to compute descriptive and discriminating power, we begin with a collection of $m$ documents and $n$ terms. As a starting point we build an $m \times n$ matrix $\mathbf{H}$, such that $\mathbf{H}[i,j] = k$, where $k$ is the number of occurrences of term $t_j$ in document $d_i$. In particular we can assume that one of the documents (e.g., $d_0$) corresponds to the initial user context. The following example illustrates this situation:

$$\mathbf{H} = \begin{array}{r} \\ \texttt{java} \\ \texttt{machine} \\ \texttt{virtual} \\ \texttt{language} \\ \texttt{programming} \\ \texttt{coffee} \\ \texttt{island} \\ \texttt{province} \\ \texttt{jvm} \\ \texttt{jdk} \end{array} \begin{array}{c} \begin{array}{ccccc} d_0 & d_1 & d_2 & d_3 & d_4 \end{array} \\ \left(\begin{array}{ccccc} 4 & 2 & 5 & 5 & 2 \\ 2 & 6 & 3 & 2 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 2 & 1 & 1 \\ 3 & 0 & 2 & 2 & 0 \\ 0 & 3 & 0 & 0 & 3 \\ 0 & 4 & 0 & 0 & 2 \\ 0 & 4 & 0 & 0 & 1 \\ 0 & 0 & 2 & 1 & 0 \\ 0 & 0 & 3 & 3 & 0 \end{array}\right) \end{array}$$

**Documents:**

| | |
|---|---|
| $d_0$: | user context |
| $d_1$: | espressotec.com |
| $d_2$: | netbeans.org |
| $d_3$: | sun.com |
| $d_4$: | wikitravel.org |

The matrix $\mathbf{H}$ allows us to formalize the notions of good descriptors and good discriminators. We define *descriptive power of a term in a document* as a function $\lambda : \{d_0, \ldots, d_{m-1}\} \times \{t_0, \ldots, t_{n-1}\} \rightarrow [0, 1]$:

$$\lambda(d_i, t_j) = \frac{\mathbf{H}[i, j]}{\sqrt{\sum_{k=0}^{n-1} (\mathbf{H}[i, k])^2}}.$$

If we adopt $\mathrm{s}(k) = 1$ whenever $k > 0$ and $\mathrm{s}(k) = 0$ otherwise, we can define the *discriminating power of a term in a document* as a function $\delta : \{t_0, \ldots, t_{n-1}\} \times \{d_0, \ldots, d_{m-1}\} \rightarrow [0, 1]$:

$$\delta(t_i, d_j) = \frac{\mathrm{s}(\mathbf{H}[j, i])}{\sqrt{\sum_{k=0}^{m-1} \mathrm{s}(\mathbf{H}[k, i])}}.$$

Note that $\lambda$ and $\delta$ satisfy the conditions

$$\sum_j (\lambda(d_i, t_j))^2 = 1 \quad \text{and} \quad \sum_j (\delta(t_i, d_j))^2 = 1.$$

Given a term $t_i$ in a document $d_j$, the term $t_i$ will have a high descriptive power in $d_j$ if it occurs often in $d_j$, while it will have a high discriminating power if it tends to occur only in $d_j$ (i.e., it seldom occurs in other documents). The descriptive power and discriminating power values for the terms in the example given above are as follows:

$$\begin{array}{r}
\\
\texttt{java}\\
\texttt{machine}\\
\texttt{virtual}\\
\texttt{language}\\
\texttt{programming}\\
\texttt{coffee}\\
\texttt{island}\\
\texttt{province}\\
\texttt{jvm}\\
\texttt{jdk}
\end{array}
\begin{array}{c}
\lambda(d_0, t_j)^T\\
\left(\begin{array}{c}0.718\\0.359\\0.180\\0.180\\0.539\\0.000\\0.000\\0.000\\0.000\\0.000\end{array}\right)
\end{array}
\begin{array}{c}
\delta(t_i, d_0)\\
\left(\begin{array}{c}0.447\\0.500\\0.577\\0.500\\0.577\\0.000\\0.000\\0.000\\0.000\\0.000\end{array}\right)
\end{array}$$

The above weights reflect some of the limitations of this first approach. For instance, the weights associated with the terms *jvm* and *jdk* do not reflect their importance as discriminators of the topic under analysis. In the same way as the well-known TF and IDF measures [30], the functions $\lambda$ and $\delta$ allow to discover terms that are good descriptors and good discriminators of a document, as opposed to good descriptors and good discriminators of the *topic* of a document.

Our current goal is to formulate notions of topic descriptors and discriminators suitable for the Web scenario. Rather than extracting descriptors and discriminators directly from the user context, we want to extract them from *the topic* of the user context. This requires an incremental method to characterize the topic of the user context, which is done by identifying documents that are similar to the user current context. Assume the user context and the retrieved Web documents are represented as document vectors in term space. To determine how similar two documents $d_i$ and $d_j$ are we adopt the IR cosine similarity [3]. This measure is defined as follows:

$$\sigma(d_i, d_j) = \sum_{k=0}^{n-1} [\lambda(d_i, t_k) \cdot \lambda(d_j, t_k)].$$

The similarity values between the user context ($d_0$) and the other documents in our example are as follows:

$$\sigma(d_0, d_j) = \begin{array}{c} \begin{array}{cccc} \quad d_1 & \quad d_2 & \quad d_3 & \quad d_4 \end{array} \\ \begin{pmatrix} 0.399 & 0.840 & 0.857 & 0.371 \end{pmatrix} \end{array}$$

The notion of topic descriptors was informally defined earlier "as terms that occur *often* in the context of a topic." We define the *term descriptive power in the topic of a document* as a function $\Lambda : \{d_0, \ldots, d_{m-1}\} \times \{t_0, \ldots, t_{n-1}\} \rightarrow [0, 1]$. If $\sum_{\substack{k=0 \\ k \neq i}}^{m-1} \sigma(d_i, d_k) = 0$ then we set $\Lambda(d_i, t_j) = 0$. Otherwise we define $\Lambda(d_i, t_j)$ as follows:

$$\Lambda(d_i, t_j) = \frac{\sum_{\substack{k=0 \\ k \neq i}}^{m-1} [\sigma(d_i, d_k) \cdot [\lambda(d_k, t_j)]^2]}{\sum_{\substack{k=0 \\ k \neq i}}^{m-1} \sigma(d_i, d_k)}$$

Thus, the descriptive power of a term $t_j$ in the topic of a document $d_i$ is a measure of the quality of $t_j$ as a descriptor of documents similar to $d_i$. As we informally formulated earlier, a term is a good discriminator of a topic if it "tends to occur *only* in documents associated with that topic." We define the *discriminating power of a term in the topic of a document* as a function $\Delta : \{t_0, \ldots, t_{n-1}\} \times \{d_0, \ldots, d_{m-1}\} \rightarrow [0, 1]$ calculated as follows:

$$\Delta(t_i, d_j) = \sum_{\substack{k=0 \\ k \neq j}}^{m-1} [[\delta(t_i, d_k)]^2 \cdot \sigma(d_k, d_j)].$$

Thus the discriminating power of term $t_i$ in the topic of document $d_j$ is an average of the similarity of $d_j$ to other documents discriminated by $t_i$. The following are the topic descriptive and discriminating power for the terms in our example:

$$\begin{array}{r cc} & \Lambda(d_0, t_j)^T & \Delta(t_i, d_0) \\ \texttt{java} & 0.385 & 0.493 \\ \texttt{machine} & 0.158 & 0.524 \\ \texttt{virtual} & 0.014 & 0.566 \\ \texttt{language} & 0.040 & 0.517 \\ \texttt{programming} & 0.055 & 0.566 \\ \texttt{coffee} & 0.089 & 0.385 \\ \texttt{island} & 0.064 & 0.385 \\ \texttt{province} & 0.040 & 0.385 \\ \texttt{jvm} & 0.032 & 0.848 \\ \texttt{jdk} & 0.124 & 0.848 \end{array}$$

Guided by the notions of topic descriptors and discriminators, it is possible to learn novel context-specific terms and reinforce the weights of existing ones. This results in a better representation of the user search context, facilitating query refinement and context-based filtering.

*2.2. An Incremental Mechanism to Tune Topical Queries*

Our proposal is to approximate the terms' descriptive and discriminating power for the thematic context under analysis with the purpose of generating good queries. Our approach adapts the typical relevance feedback mechanism to account for an evolving thematic context $C_i$. A schematic illustration of incremental method for tuning queries based on a thematic context is shown in figure 1 and summarized in algorithm 1.

**Algorithm 1** Main

```
$i \Leftarrow 0$
$C_i \Leftarrow InitialContext$
repeat
  $i \Leftarrow i + 1$
  Compute $C_i$
  Update $C_i$
until $(i > v) \wedge FinalConvergence$
```

**Algorithm 2** Compute $C_i$

```
$j \Leftarrow 0$
$\Lambda_j \Leftarrow \emptyset$
$\Delta_j \Leftarrow \emptyset$
repeat
  $j \Leftarrow j + 1$
  Create queries with $C_i$ and do Search
  Calculate $\Lambda'$ and $\Delta'$ based on search results
  $\{\Lambda_j|\Delta_j\} = \alpha\{\Lambda_{j-1}|\Delta_{j-1}\} + \beta\{\Lambda'|\Delta'\}$
  TestConvergence
until $(j > u) \wedge PhaseConvergence$
```

**Algorithm 3** TestConvergence

```
Require: $\mu > \nu$
  $PhaseConvergence \Leftarrow max(\sigma(Results, C_i)) < \mu$
  $FinalConvergence \Leftarrow max(\sigma(Results, C_i)) < \nu$
```

In order to learn better characterizations of the thematic context, the system undergoes a series of phases. At the end of each phase, the context characterization is updated with new learned material. Each phase evolves through a sequence of trials, where each trial consists in the formulation of a set of queries, the analysis of the retrieved results, the adjustment of the terms' weights, and the discovery of new potentially useful terms. For a given phase $\mathcal{P}_i$, the context is represented by a set of weighted terms. Let $w^{\mathcal{P}_i}(t, \mathcal{C})$ be an estimation of the importance of term $t$ in context $\mathcal{C}$ during phase $i$. If $t$ occurs in the initial context, then the value $w^{\mathcal{P}_0}(t, \mathcal{C})$ is initialized as the normalized frequency of term $t$ in $\mathcal{C}$, while the weight of those terms that are not part of $\mathcal{C}$ are assumed to be 0.

Let $w_{\Lambda}^{(i,j)}(t, \mathcal{C})$ and $w_{\Delta}^{(i,j)}(t, \mathcal{C})$ be an estimation of the descriptive and discriminating power of term $t$ for context $\mathcal{C}$ at trial $j$ of phase $i$. These values are incrementally computed as follows:

$$w_{\Lambda}^{(i,j+1)}(t, \mathcal{C}) = \alpha.w_{\Lambda}^{(i,j)}(t, \mathcal{C}) + \beta.\Lambda^{(i,j)}(t, \mathcal{C}).$$

$$w_{\Delta}^{(i,j+1)}(t, \mathcal{C}) = \alpha.w_{\Delta}^{(i,j)}(t, \mathcal{C}) + \beta.\Delta^{(i,j)}(t, \mathcal{C}).$$

We assume $w_{\Lambda}^{(i,0)}(t, \mathcal{C}) = w_{\Delta}^{(i,0)}(t, \mathcal{C}) = 0$ and use the results returned during each trial $j$ to compute $\Lambda^{(i,j)}(t, \mathcal{C})$ and $\Delta^{(i,j)}(t, \mathcal{C})$, the descriptive and discriminating power of term $t$ for the topic of $\mathcal{C}$. To form queries during phase $i$ we implemented a roulette selection mechanisms where the probability of choosing a particular term $t$ to form a query is proportional to $w^{\mathcal{P}_i}(t, \mathcal{C})$. Roulette selection is a technique typically used by Genetic Algorithms [12] to choose potentially useful solutions for recombination, where the fitness level is used to associate a probability of selection. This approach resulted in a non-deterministic exploration of term space that favored the fittest terms.

The system monitors the effectiveness achieved at each iteration. In our approach we use *novelty-driven similarity* introduced in section 3 as an estimation of the retrieval effectiveness. If after a window of $u$ trials the retrieval effectiveness has not crossed a given threshold $\mu$ (i.e., no significant improvements are observed after certain number of trials), the system forces a phase change to explore new potentially useful regions of the vocabulary landscape. A phase change can be regarded as a vocabulary leap, which can be thought of as a significant transformation (typically an improvement) of the context characterization. If a phase change takes effect during trial $j$, the value of $w_{\Lambda}^{\mathcal{P}_i}(t, \mathcal{C})$ is set to $w_{\Lambda}^{(i,j)}(t, \mathcal{C})$ and $w_{\Delta}^{\mathcal{P}_i}(t, \mathcal{C})$ is set to $w_{\Delta}^{(i,j)}(t, \mathcal{C})$. To reflect the phase change in the new characterization of the thematic context, the weight of each term $t$ is updated as follows:

$$w^{\mathcal{P}_{i+1}}(t, \mathcal{C}) = \gamma.w^{\mathcal{P}_i}(t, \mathcal{C}) + \zeta.w_{\Lambda}^{\mathcal{P}_i}(t, \mathcal{C}) + \xi.w_{\Delta}^{\mathcal{P}_i}(t, \mathcal{C}).$$

These weights are then used to generate new queries during the sequence of trials at phase $i + 1$. The final convergence of the algorithm is achieved after at least $v$ phase changes and once the retrieval effectiveness has not crossed a given threshold $\nu$ ($\nu < \mu$).

## 3. Evaluation

The goal of this section is to compare the proposed method against two other methods. The first is a baseline that submits queries directly from the thematic context and does not apply any refinement mechanism. The second method used for comparison is the Bo1-DFR described in section 1.2.

### *3.1. Data Collection and Experimental Setup*

To perform our tests we used 448 topics from the Open Directory Project (ODP)[1]. The topics were selected from the third level of the ODP hierarchy. A number of

[1] http://dmoz.org

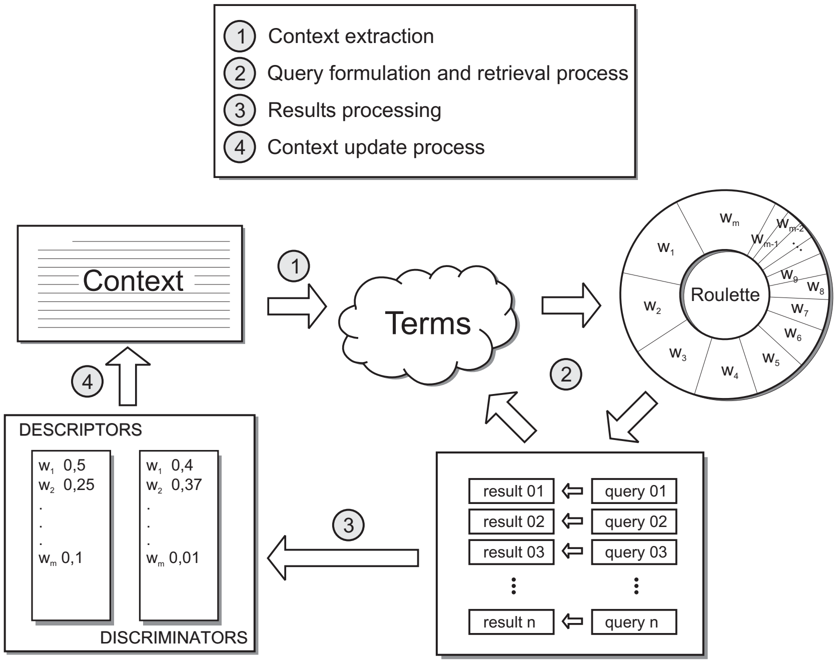

Figure 1: A schematic illustration of the incremental method for tuning topical queries.

constraints were imposed on this selection with the purpose of ensuring the quality of our test set. The minimum size for each selected topic was 100 URLs and the language was restricted to English. For each topic we collected all of its URLs as well as those in its subtopics. The total number of collected pages was more than 350K. The Terrier framework [25] was used to index these pages and to run our experiments.

In our tests we used the ODP description of each selected topic to create an initial context description $\mathcal{C}$. The proposed algorithm was run for each topic for at least $v = 10$ iterations, with 10 queries per iteration and retrieving 10 results per queries. The descriptor and discriminator lists at each iteration were limited to up to 100 terms each. The other parameters in our method were set as follows: $u = 10$, $\alpha$=0.5, $\beta$=0.5, $\gamma$=0.33, $\zeta$=0.33, $\xi$=0.33, $\mu$=0.2 and $\nu$=0.1. In addition, we used the stopword list provided by Terrier, Porter stemming was performed on all terms and none of the query expansion methods offered by Terrier was applied.

### *3.2. Performance Metrics*

In order to evaluate the proposed algorithm we used three measures to assess query performance: novelty-driven similarity, precision and semantic precision. These metrics are described below.

### *3.2.1. Novelty-driven similarity*

This ad-hoc measure of similarity is based on $\sigma$, the classical cosine similarity measure discussed in section 2.1. However, this new measure of similarity disregards the terms that form the query, overcoming the bias introduced by those terms and favoring the exploration of new material. Given a query $q$ and documents $d_i$ and $d_j$, the novelty-driven similarity measure is defined as $\sigma^N(\mathbf{q}, d_i, d_j) = \sigma(d_i - \mathbf{q}, d_j - \mathbf{q})$. The notation $d_i - \mathbf{q}$ stands for the representation of the document $d_i$ with all the values corresponding to the terms from query $\mathbf{q}$ set to zero. The same applies to $d_j - \mathbf{q}$.

### *3.2.2. Precision*

This well-known performance evaluation metric is computed as the fraction of retrieved documents which are known to be relevant, i.e., Precision$= |A \cap R|/|A|$, where $R$ and $A$ are the relevant and answer set respectively. The relevant set for each analyzed topic was set as the collection of its URLs as well as those in its subtopics.

### *3.2.3. Semantic Precision*

Other topics in the ontology could be semantically similar (and therefore partially relevant) to the topic of the given context. Therefore, we propose a measure of semantic precision defined as Precision$^S = \sum_{p \in A} \sigma^S(t(\mathcal{C}), t(p))/|A|$, where $t(\mathcal{C})$ is the ODP topic associated with the description used as the initial context, $t(p)$ is the topic of page $p$ and $\sigma^S(t(\mathcal{C}), t(p))$ is the semantic similarity between these two topics. To compute $\sigma^S$ we used a semantic similarity measure for generalized ontologies proposed by Maguitman et al. [23].

## 4. Results

We computed the novelty-driven similarity measure $\sigma^N$ between the initial context (topic descriptions) and the retrieved results. Figure 2 shows the evolution of the novelty-driven similarity averaged across all the tested topics. In addition, the graphic shows the error-bars every 10 iterations (which are usually coincidental with phase changes). The significant improvements observed, especially during the first phase changes, provide evidence that the proposed algorithm can help enrich the topic vocabulary.

The charts in figures 3, 4, 5 compare the performance of queries for each tested method using novelty-driven similarity, precision and semantic precision. Each of the 448 topics corresponds to a trial and is represented by a point. The point's vertical coordinate (z) corresponds to the performance of the incremental method, while the point's other two coordinates (x and y) correspond to the baseline and the Bo1-DFR methods. In addition we can observe the projection of each point on the x-y, x-z and y-z planes. For the x-z plane, the points above the diagonal correspond to cases in which the incremental method is superior to the baseline. Similarly, for the y-z plane, the points above the diagonal correspond to cases in which the incremental method is superior to Bo1-DFR. The x-y plane compares the performance of the baseline against Bo1-DFR. Note that different markers are used to illustrate the cases in which each of the methods performs better than the other two.

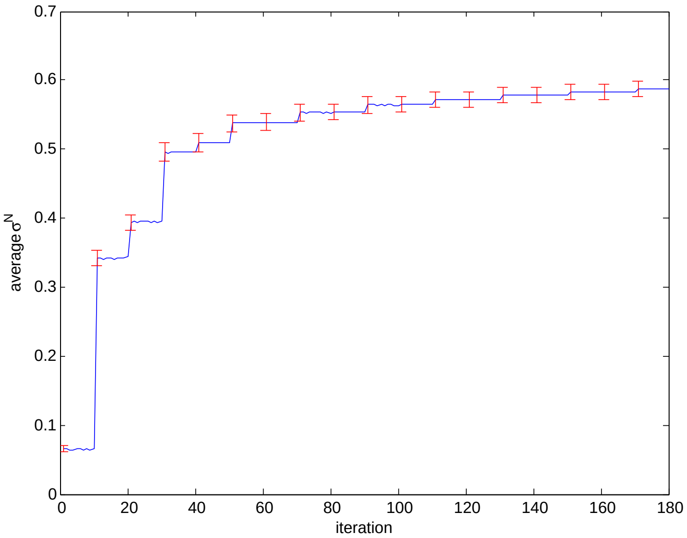

Figure 2: The evolution of the maximum novelty-driven similarity

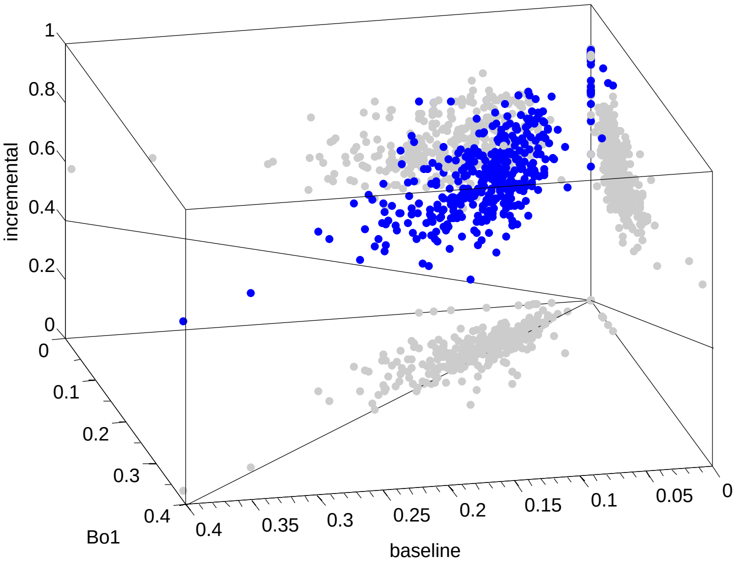

Figure 3: A comparison of the three tested methods based on novelty-driven similarity.

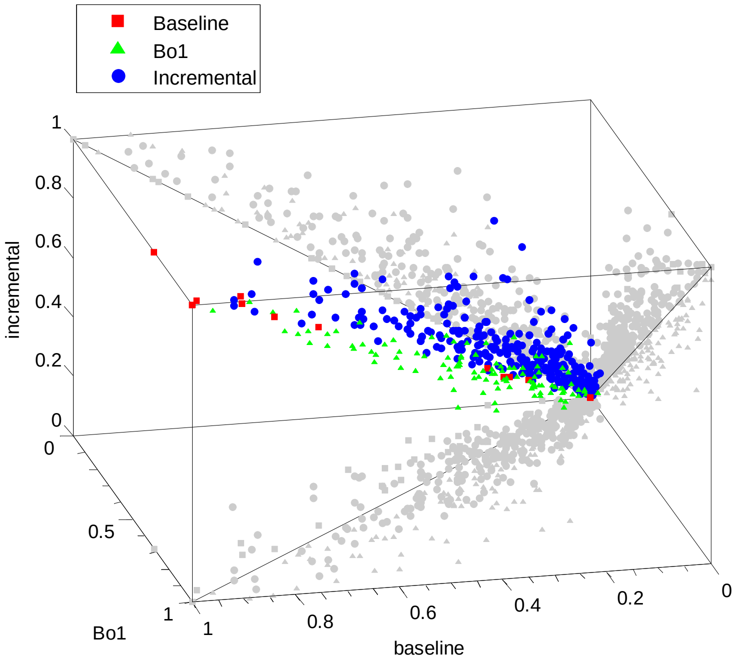

Figure 4: A comparison of the three tested methods based on precision.

It is interesting to note that for all the tested cases the incremental method was superior to the baseline and the Bo1-DFM method in terms of novelty-driven similarity. This highlights the usefulness of evolving the context vocabularies to discover good query terms. For the precision metric, the incremental method was strictly superior to the other two methods for 66.96% of the evaluated topics. Bo1-DFR was the best method for 24.33% of the topics and the baseline performed as well as one of the other two other methods for 8.70% of the topics. Finally, for the semantic precision metric the incremental method was strictly superior to the other methods for 65.18% of the topics, Bo1-DFR was superior for 27.90% of the topics and the baseline performed as well as one of the other two methods for 6.92% of the topics.

Figure 6 presents the means and confidence intervals of the methods' performance based on $\sigma^N$, Precision and Precision$^S$. This comparison table shows that the improvements achieved by the incremental method with respect to the other methods are statistically significant.

## 5. Findings and Implications

The automatic formulation of queries from a thematic context requires techniques that have the ability to associate the given context with relevant sources. The false-

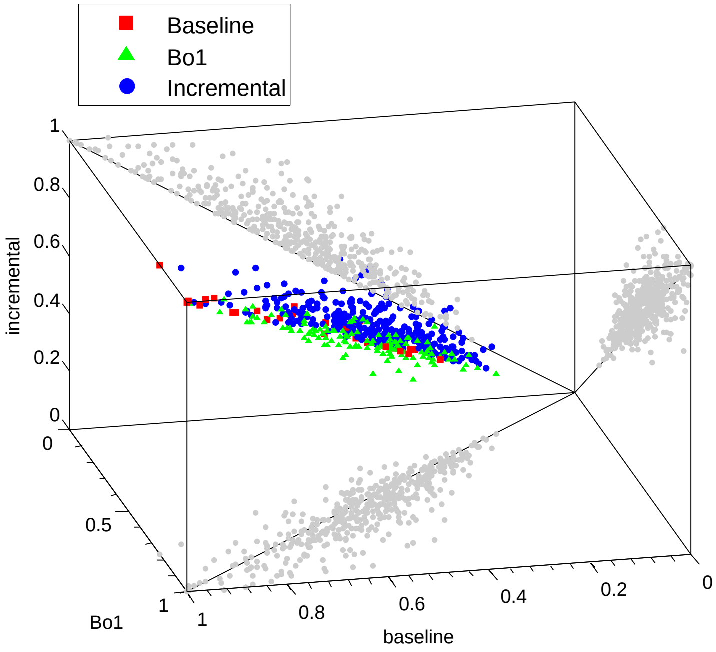

Figure 5: A comparison of the three tested methods based on semantic precision.

negative match problem is a common situation that arises when textual content with similar topics but different term vocabulary cannot be associated. A complementary problem, the false-positive match problem, is the result of associating textual content with similar term vocabulary but different topics. These two problematic situations have long been recognized as main challenges by the information retrieval community and many proposals have tried to overcome these issues with varying degree of success.

We have proposed a method that makes an advance towards overcoming the above mentioned term-match problems by learning new vocabularies. We have proposed to use topic descriptors to identify those terms that occur more often in documents associated with the given topic. These terms are not necessarily part of the specification of the user's information needs. However, they can be iteratively found by analyzing sets of documents that are incrementally retrieved from the Web. On the other hand we have proposed to use topic discriminators to identify those terms that tend to occur only in the given topic and rarely in documents that do not have that topic. Our evaluations suggest that by combining terms with high descriptive and discriminating power its possible to implement methods with high retrieval performance.

Developing methods to evolve high-quality queries and collect context-relevant resources can have important implications on the way users interact with the Web. These methods can help build systems for a range of information services:

| $\sigma^N$ | | | |
|---|---|---|---|
| method | N | mean | 95% C.I. |
| Baseline | 448 | 0.087 | [0.0822;0.0924] |
| Bo1-DFR | 448 | 0.075 | [0.0710;0.0803] |
| Incremental | 448 | 0.597 | [0.5866;0.6073] |

| Precision | | | |
|---|---|---|---|
| method | N | mean | 95% C.I. |
| Baseline | 448 | 0.266 | [0.2461;0.2863] |
| Bo1-DFR | 448 | 0.307 | [0.2859;0.3298] |
| Incremental | 448 | 0.354 | [0.3325;0.3764] |

| Precision$^S$ | | | |
|---|---|---|---|
| method | N | mean | 95% C.I. |
| Baseline | 448 | 0.553 | [0.5383;0.5679] |
| Bo1-DFR | 448 | 0.590 | [0.5750;0.6066] |
| Incremental | 448 | 0.622 | [0.6068;0.6372] |

Figure 6: Means and confidence intervals for the performance of the three methods based on novelty-driven similarity, precision and semantic precision.

- **Task-Based Search.** Task-based search systems exploit user interaction with computer applications to determine the user's current task and contextualize information needs [20, 7]. Basic keyword searches could very easily miss task-relevant pages. By evolving high-quality queries, a task-based search system can automatically generate suggestions that are richly contextualized within the user's task.

- **Resource Harvest for Topical Web Portals.** Topical Web portals have the purpose of gathering resources on specific subjects. The collected material is used to build specialized search and directory sites. Typically, focused crawlers are in charge of mining the Web to harvest topical content and populate the indices of these portals [10, 24]. As and alternative to focused crawlers, this process can be supported by formulating topical queries to a search engine and selecting from the answer set those resources that are related to the topic at hand.

- **Deep Web Search.** Most of the Web's information can be found in the form of dynamically generated pages and constitutes what is known as the deep Web [15]. The pages that constitute the deep Web do not exist until they are created dynamically as the result of a query presented to search forms available in specific sites (e.g., pubmedcentral.nih.gov, amazon.com). Therefore, the formulation of high-quality queries is of utmost importance at the moment of accessing deep Web sources. For that reason, searching the deep Web in context is an important area of application for the proposed techniques.

- **Support for Knowledge Management.** Effective knowledge management may require going beyond initial knowledge capture, to support decisions about how to extend previously-captured knowledge [19, 21]. The Web provides a rich source of information on potential new material to include in a knowledge model. Thus material can be accessed by means of contextualized queries presented to a conventional search engine, where the context is given by the knowledge model under construction. Using the Web as a huge repository of collective memory and starting from an in-progress knowledge model, the techniques discussed here can facilitate the process of capturing knowledge to help extend organizational memories.

## 6. Conclusions

In this paper we propose a solution to the semantic sensitivity problem, that is the limitation that arises when semantically related documents with different term vocabulary won't be associated, resulting in a false-negative match. In addition, by identifying good topic discriminators our proposal helps alleviate the false-positive match problem, that occurs when the same term (e.g., java) occurs in different topics. Our method operates by incrementally learning better vocabularies from a large external corpus such as the Web.

Other corpus-based approaches have been proposed to address the semantic sensitivity problem. For example, latent semantic analysis [11] applies singular value decomposition to reduce the dimensions of the term-document space, harvesting the latent relations existing between documents and between terms in large text corpora. Another corpus-based technique that has been applied to estimate semantic similarity is PMI-IR [33]. This information retrieval method is based on pointwise mutual information, which measures the strength of association between two elements (e.g., terms) by contrasting their observed frequency against their expected frequency. Differently from our approaches, these techniques are not based on an incrementally refined query submission process. Instead, they use a predefined collection of document to identify latent semantic relations. In addition, these techniques do not distinguish between the notions of topic descriptors and topic discriminators. The techniques for query term selection proposed in this paper share insights and motivations with other methods for query expansion and refinement [31, 5]. However, systems applying these methods differ from our framework in that they support this process through query or browsing interfaces requiring explicit user intervention, rather than formulating queries automatically.

In this paper we have shown that by implementing an incremental context refinement method we can perform better than a baseline method, which submits queries directly from the initial context, and to the Bo1-DFR method, which does not refine queries based on context. This points to the usefulness of simultaneously taking advantage of the terms in the current thematic context and an external corpus to learn better vocabularies and to automatically tune queries.

Learning better vocabularies is a way to increase the awareness and accessibility of useful material. We have proposed a promising method to identify the need behind the query, which is one of the main goals for many current and next generation Web services and tools.

We are currently working on applying the proposed method for learning better vocabularies to other IR tasks, such as text classification. We are also analyzing different strategies for helping the system keep its focus on the initial context after several incremental steps have taken place. As part of our future work we expect to use standard large-scale collections (such as the TREC Web collection) to further evaluate our techniques. Finally, we expect to investigate different parameter settings for the proposed algorithm and to develop methods that automatically learn and adjust these parameters.

### Acknowledgement

This research work is supported by Agencia Nacional de Promoción Científica y Tecnológica (PICT 2005 Nro. 32373), Universidad Nacional del Sur (PGI 24/ZN13) and Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET).

### References

[1] Giambattista Amanti. *Probabilistics Models for Information Retrieval based on Divergence from Randomness*. PhD thesis, Department of Computing Science, University of Glasgow, UK, 2003.

[2] Giambattista Amati, Claudio Carpineto, and Giovanni Romano. Query difficulty, robustness and selective application of query expansion. In *Advances in Information Retrieval, 26th European Conference on IR research*, pages 127–137. Springer Berlin / Heidelberg, 2004.

[3] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley, 1999.

[4] Nicolas J. Belkin. Helping people find what they don't know. *Commun. ACM*, 43(8):58–61, 2000.

[5] Bodo Billerbeck, Falk Scholer, Hugh E. Williams, and Justin Zobel. Query expansion using associated queries. In *Proceedings of the twelfth international conference on Information and knowledge management*, pages 2–9. ACM Press, 2003.

[6] Chris Buckley, Amit Singhal, and Mandar Mitra. New retrieval approaches using smart. In *TREC 4*, 1995.

[7] Jay Budzik, Kristian J. Hammond, and Larry Birnbaum. Information access in context. *Knowledge based systems*, 14(1–2):37–53, 2001.

[8] D. Cai and C. J. van Rijsbergen. Learning semantic relatedness from term discrimination information. *Expert Syst. Appl.*, 36(2):1860–1875, 2009.

[9] Guihong Cao, Jian-Yun Nie, Jianfeng Gao, and Stephen Robertson. Selecting good expansion terms for pseudo-relevance feedback. In *SIGIR '08: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 243–250, New York, NY, USA, 2008. ACM.

[10] Soumen Chakrabarti, Martin van den Berg, and Byron Dom. Focused crawling: a new approach to topic-specific Web resource discovery. *Computer Networks (Amsterdam, Netherlands: 1999)*, 31(11–16):1623–1640, 1999. 1999a.

[11] Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.

[12] John H. Holland. *Adaptation in natural and artificial systems*. Ann Arbor: The University of Michigan Press, 1975.

[13] Sparck K. Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28:11–21, 1972.

[14] Chris Jordan and Carolyn R. Watters. Extending the rocchio relevance feedback algorithm to provide contextual retrieval. In *AWIC*, pages 135–144, 2004.

[15] Henry Kautz, Bart Selman, and Mehul Shah. The hidden Web. *AI Magazine*, 18(2):27–36, 1997.

[16] Mei Kobayashi and Koichi Takeda. Information retrieval on the Web. *ACM Comput. Surv.*, 32(2):144–173, 2000.

[17] Reiner Kraft, Chi Chao Chang, Farzin Maghoul, and Ravi Kumar. Searching with context. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 477–486, New York, NY, USA, 2006. ACM.

[18] K. L. Kwok and M. Chan. Improving two-stage ad-hoc retrieval for short queries. In *SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 250–256, New York, NY, USA, 1998. ACM.

[19] David Leake, Ana Maguitman, Thomas Reichherzer, Alberto Cañas, Marco Carvalho, Marco Arguedas, Sofia Brenes, and Tom Eskridge. Aiding knowledge capture by searching for extensions of knowledge models. In *Proceedings of KCAP-2003*. ACM Press, 2003.

[20] David B. Leake, Travis Bauer, Ana Maguitman, and David C. Wilson. Capture, storage and reuse of lessons about information resources: Supporting task-based information search. In *Proceedings of the AAAI-00 Workshop on Intelligent Lessons Learned Systems. Austin, Texas*, pages 33–37. AAAI Press, 2000.

[21] Ana Maguitman, David Leake, and Thomas Reichherzer. Suggesting novel but related topics: towards context-based support for knowledge model extension. In *IUI '05: Proceedings of the 10th international conference on Intelligent user interfaces*, pages 207–214, New York, NY, USA, 2005. ACM Press.

[22] Ana Maguitman, David Leake, Thomas Reichherzer, and Filippo Menczer. Dynamic extraction of topic descriptors and discriminators: Towards automatic context-based topic search. In *Proceedings of the Thirteenth Conference on Information and Knowledge Management (CIKM)*, Washington, DC, November 2004. ACM Press.

[23] Ana G. Maguitman, Filippo Menczer, Heather Roinestad, and Alessandro Vespignani. Algorithmic detection of semantic similarity. In *WWW '05: Proceedings of the 14th international conference on World Wide Web*, pages 107–116, New York, NY, USA, 2005. ACM.

[24] Filippo Menczer, Gautam Pant, and Padmini Srinivasan. Topical web crawlers: Evaluating adaptive algorithms. *ACM Trans. Inter. Tech.*, 4(4):378–419, 2004.

[25] Iadh Ounis, Christina Lioma, Craig Macdonald, and Vassilis Plachouras. Research directions in Terrier: a search engine for advanced retrieval on the web. *Novatica/UPGRADE Special Issue on Web Information Access, Ricardo Baeza-Yates et al. (Eds), Invited Paper*, VIII(1):49–56, February 2007.

[26] Eduardo H. Ramirez and Ramon F. Brena. Semantic contexts in the internet. In *LA-WEB '06: Proceedings of the Fourth Latin American Web Congress*, pages 74–81, Washington, DC, USA, 2006. IEEE Computer Society.

[27] Jason D. M. Rennie and Tommi Jaakkola. Using term informativeness for named entity detection. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 353–360, New York, NY, USA, 2005. ACM.

[28] J. J. Rocchio. Relevance feedback in information retrieval. In G. Salton, editor, *The Smart retrieval system - experiments in automatic document processing*, pages 313–323. Englewood Cliffs, NJ: Prentice-Hall, 1971.

[29] G. Salton and C. Buckley. Term weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

[30] G. Salton and C. Yang. On the specification of term values in automatic indexing. *Journal of Documentation*, 29:351–372, 1973.

[31] Falk Scholer and Hugh E. Williams. Query association for effective retrieval. In *Proceedings of the eleventh international conference on Information and knowledge management*, pages 324–331. ACM Press, 2002.

[32] M.M. Sufyan Beg and Nesar Ahmad. Web search enhancement by mining user actions. *Information Sciences*, 177(23):5203–5218, 2007.

[33] Peter D. Turney. Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In *EMCL '01: Proceedings of the 12th European Conference on Machine Learning*, pages 491–502, London, UK, 2001. Springer-Verlag.

[34] Zuobing Xu and Ram Akella. Active relevance feedback for difficult queries. In *CIKM '08: Proceeding of the 17th ACM conference on Information and knowledge management*, pages 459–468, New York, NY, USA, 2008. ACM.