データベースシステム　125－42
（２００１．７．１８）

分類器に基づくテキストデータベースに対するフォーカスされた検索

サイド ミルザ パレビ[†]　北川 博之[‡]　石川 佳治[‡]
[†] 筑波大学工学研究科
[‡] 筑波大学電子・情報工学系

情報検索において、多くの場合、ユーザは簡単なキーワード等のみを問合せ条件を与える。このため、しばしば不要な文書を多数含むような検索結果が返されてしまう。この問題を解決するため、ユーザの問合せを拡張する自動的な問合せ拡張方式が一般に使用される。本論文では、キーワードのみでなくタキソノミーを使って検索結果の精度をあげるための方法を提案する。ユーザは問合せのキーワードを与えるだけでなく、その問合せのコンテクストノードをタキソノミーから選ぶ。システムは、問合せ結果をタキソノミーを用いて分類し、コンテクストノードに対応する文書とそれ以外の文書を分別する。これらの文書からルールベースの分類器を構築し、その結果として得られるルールを使ってユーザの問合せを拡張する。

キーワード　分類器に基づく情報検索、自動的な問合せ拡張方式、ルールベース分類器

Classifier-based Focused Retrieval for Text Databases

Said Mirza Pahlevi[†], Hiroyuki Kitagawa[‡], Yoshiharu Ishikawa[‡]
[†] Doctoral Program in Engineering, University of Tsukuba
[‡] Institute of Information Sciences and Electronics, University of Tsukuba

Most of the casual users always use very short queries in information retrieval, and cannot be expected to take the trouble to formulate longer and more elaborate queries that can further improve the accuracy of their retrieval results. The well-known method to deal with this issue is automatic query expansion. In this paper, we propose a method to increase the accuracy of the retrieval results by not only using keywords from the queries but also a taxonomy. A user gives keywords as a query to the system, and at the same time he/she selects from the taxonomy a context node of the query. After retrieving initial results, system then uses the taxonomy to classify the results. Documents classified to the context node and its descendants are considered relevant to the query, while the others are considered non-relevant. Based on the relevancy of the documents, the system constructs a rule-based classifier and used the rules from the classifier to expand the query.

key words　classifier-based information retrieval, automatic query expansion, rule-based classifier

1 Introduction

As the number of casual users who use information retrieval systems continues to grow at a high rate, the demand for higher accuracy information retrieval also continues to escalate.

Most of the casual users always use very short queries and cannot be expected to take the trouble to formulate longer and more elaborate queries that can further improve the accuracy of their retrieval results. Even though the advance users, they may face the *word mismatch* problem that may decrease the accuracy. The word mismatch problem is the mismatch between words in queries and the ones in documents resulting from the use of different words by query composers and document authors in describing the same concepts.

The well-known method to deal with these issues is automatic query expansion [1] [3] [7] [8] [9]. By adding more useful words to the queries constructed by the users, this method may significantly increase the accuracy of the retrieval results.

One of main techniques to expand a query is to consider the top ranked documents returned by the original query to be relevant to the query and common words from them are then extracted and added to the query. The disadvantage of this technique is obvious. Its effectiveness highly depends on the proportion of relevant documents in the top ranked documents. If the large fraction of documents that are assumed relevant is actually non-relevant, the most words added to the original query are likely to be unrelated to the topic required by the user. Thus, the performance of the expanded query would seem likely to be worse than that of the original query.

One approach to eliminate this problem is to refine the set of documents used in query expansion [1]. This can be achieved by filtering the non-relevant documents and use only the relevant ones to expand the original query. Another approach is to cluster the documents returned by the original query [2]. User chooses which cluster can best express his/her required topic. The original query is then expanded by using document vector digesting the cluster.

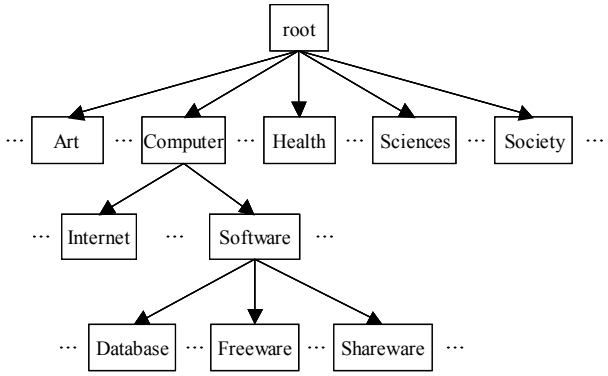

Figure 1: A portion of Yahoo!'s taxonomy

In this paper, we propose a new method to increase the accuracy of the retrieval results by not only using keywords from a query but also a taxonomy. After constructing a query, the user informs the system the semantics of his/her query by selecting a context node in the taxonomy. The system then sends the query to a search interface and gets the retrieval results. The results are then sent to the taxonomy to find their topic distribution. Documents classified to the context node and its descendants are considered relevant to the query, while the others are considered non-relevant. Based on the relevancy of the documents, the system constructs a rule-based classifier and uses the rules from the classifier to expand the query.

The paper is organized as follows. Section 2 describes the taxonomy used in this paper. Section 3 introduces a new technique to separate the relevant and non-relevant documents in the initial results and a new technique to expand a query by using a rule-based classifier. Section 4 describes related work. In the final section, we give our conclusions and suggest future work.

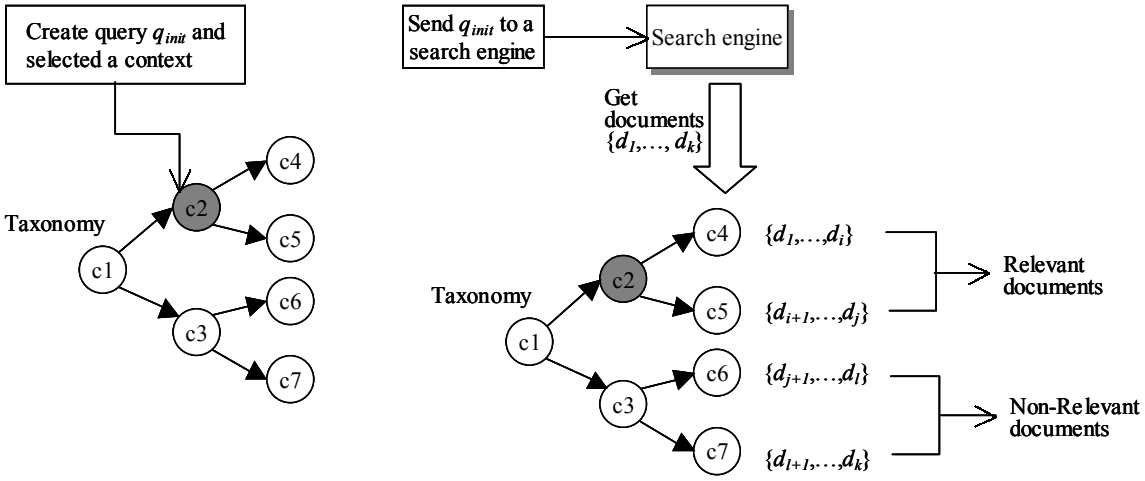

Figure 2: Separating relevant and non-relevant documents in the initial retrieval results

2 Taxonomy

It is common to manage a large and complex database by using a hierarchy, also called *taxonomy*. Examples of such taxonomies include Yahoo! (Figure 1), IBM's patent database and ACM digital library. Another use of the taxonomy is to relieve the user from the burden of sifting specific information from the large and low-quality response of most popular search engines [7]. Formally, we can define a taxonomy as follows:

Definition 1: A taxonomy is a rooted directed tree whose nodes correspond to categories and whose edges denote specialization. An edge from category *c* to category *c'* indicates that *c'* is a subcategory of *c*. □

We use the taxonomy to separate relevant and non-relevant documents in the initial retrieval results. For doing this, we associate each internal node (including root node) of the taxonomy with a classifier. We construct (rule-based) classifiers over an entire taxonomy as follows:

1. We associate each leaf node in the taxonomy with a set of preclassified documents.
2. For each internal node c_i and its children nodes $c_{i,1},\ldots, c_{i,k}$, we construct a rule-based classifier from the set of preclassified documents belonging to the children nodes.
3. The resulting classifier is a set of logical rules in the form of $w_{i,j} \rightarrow c_{i,j}$, $j = 1,\ldots, k$, where $w_{i,j}$ is conjunction of words and $c_{i,j}$ is a child of node c_i. We say a document d matches a rule $w_{i,j} \rightarrow c_{i,j}$, if d contains all words in $w_{i,j}$.

3 The Proposed Method

In this paper, we assume that the search interface processes a query in a boolean manner. Conforming to this, the initial user query and the expanded query are also boolean types.

3.1 Separating Relevant and Non-relevant Documents

In order to relieve the user from the burden of deciding which documents are relevant and non-relevant to his/her query, we utilize the taxonomy and the associated rule-based classifiers. A user is only required to select a context node (topic) in the taxonomy, where his/her query will be focused by the system. The system separates relevant and non-relevant documents based on the

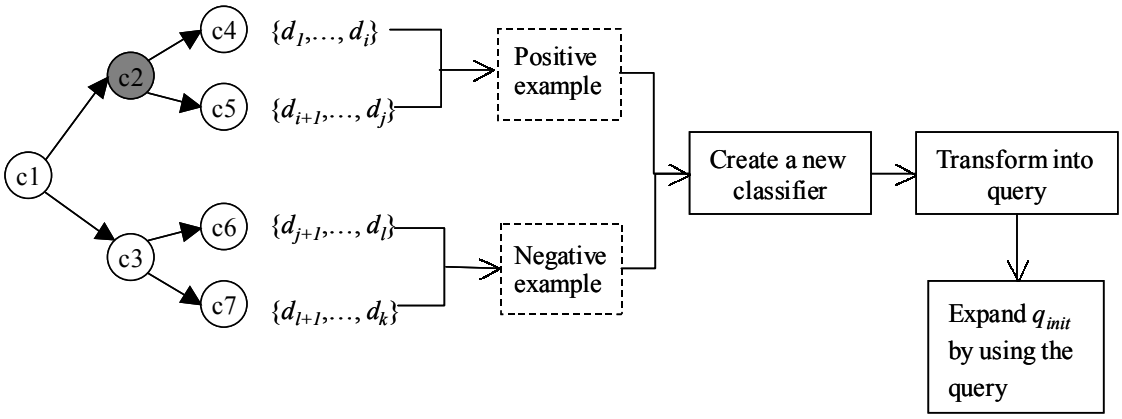

Figure 3: Expanding the initial query by using a newly created classifier

context node.

More specifically, the separation process is as follows (Figure 2).

1. After creating a query, a user selects a context of his/her query on the taxonomy. The context can be a leaf node or an internal node of the taxonomy that best describes the semantics of his/her query. By using the taxonomy, the user can now restrict his/her query not only by keywords that are usually very few in number but also by the context in the taxonomy.
2. System then sends the query to a specific search interface and gets documents from the retrieval results.
3. The documents are sent to the taxonomy to get their topic distribution.
4. Relevant documents are those that are classified into the context node and its descendants, while the others are considered to be non-relevant.

3.2 Focusing the Initial Query to the Required Topic

After the relevant and non-relevant documents are separated, next we have to expand the user query by selecting words from the relevant documents. We utilize a rule-based classifier to select "good" words to expand the query. The classifier does not take a word individually from a relevant document, rather it takes a word based on the co-occurrence with other words in the document.

Specifically, let q_{init} be the initial user query and D_{init} be documents from the query results. Let $D_{rel} \subset D_{init}$ and $D_{non\text{-}rel} \subset D_{init}$ be the sets of the relevant and non-relevant documents, respectively. We focus q_{init} to the topic required by the user by expanding it using words from D_{rel} as follows (Figure 3):

1. Create a rule-based classifier by setting D_{rel} and $D_{non\text{-}rel}$ as positive and negative examples, respectively.
2. Transform the new created classifier into a boolean query as follows:
   (a) Let the set of the classifier's rules be R.
   (b) For each rule $r_j \in R$, where $r_j = w_j \rightarrow c$, construct query $q_j = w_j$.
   (c) The resulting query is $Q_{classifier} = q_1 \vee \ldots \vee q_n$, where $n = |R|$.
3. Expand q_{init} by AND-ing it with $Q_{classifier}$, that is $Q_{expanded} = q_{init} \wedge Q_{classifier}$.

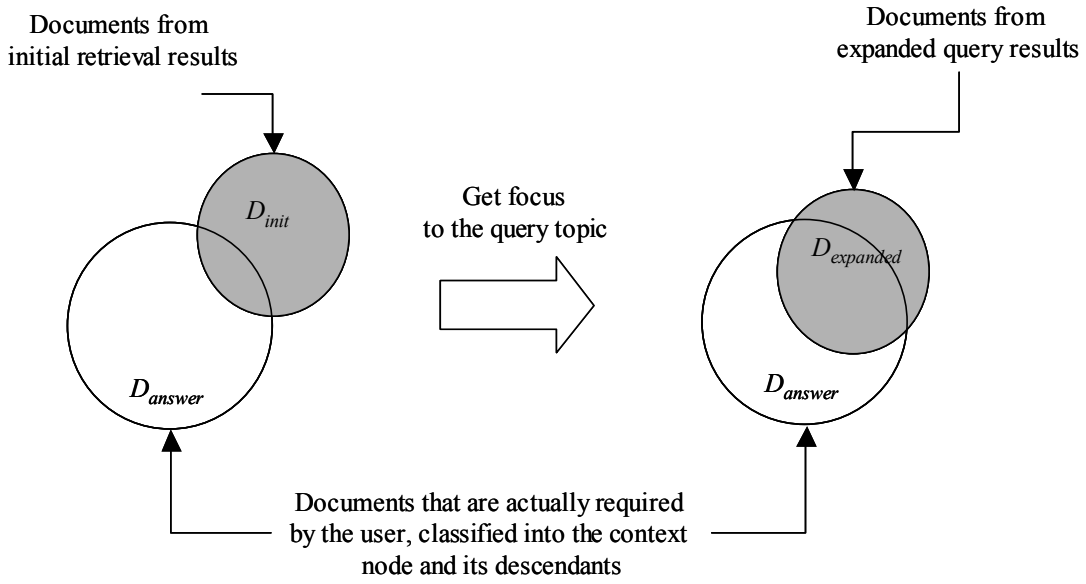

Figure 4: Focusing the initial query to the required topic

Let documents actually required by the user, denoted as D_{answer}, are a subset of documents that can be classified into the context node or its descendants. Documents retrieved by $Q_{expanded}$, denoted as $D_{expanded}$, *tends* to get "closer" to D_{answer} because the classifier constructed at step 1 above is the one that "separates" the relevant documents from the non-relevant ones. In other words, by "sending" the classifier to the search interface (i.e. transform it to a boolean query and send it to the search interface), it seems that we can get more documents actually required by the user while leaving the non-relevant ones (Figure 4).

4 Related Work

The most closely related to our work is the *interactive query learning system* proposed in [5]. This system was proposed to keep resource directories up-to-date. Resource directories are documents that collect together links to all known documents on a specific topic. A user via an augmented web browser specifies the positive and negative example incrementally for the current topic. The user can invoke the system to create new rules by using positive and negative examples collected so far. The resulting rules are then transformed into a query for web search interfaces in order to detect any new instances that may be added in the specific resource directories.

Another approach to separate relevant and non-relevant documents is to use clustering technique [2]. Documents retrieved by the initial query are organized into conceptual groups (clusters), such that the user could get a quick overview of what the query actually retrieves. After relevant and non-relevant clusters have been identified by the user, each cluster is digested as a document vector and the initial query is modified by the Rocchio's algorithm.

Our work is difference from them in that the structure of the taxonomy that we are used here is static (i.e. it does not depend on the initial results and specific parameters), while the structure of their clusters are dynamic (i.e. it depends on the query results and some clustering parameters). By using a static hierarchical classification (taxonomy), a user can freely specify the broadness of topic/concept of

his/her query, while it is difficult to do it in a dynamic, flat clustering approach.

5 Conclusions and Future Work

We have proposed a new method to increase the accuracy of the initial retrieval results by combining keywords from a query and a taxonomy. To start with, a user selects a context node in the taxonomy in order to inform the system the semantics of his/her query.

Next, the system sends the query to a search interface and retrieves documents from the query results. The relevant and non-relevant documents are separated by using the taxonomy based on the selected context node. Documents classified to the context node and its descendants are considered relevant, while the others are considered non-relevant.

Finally, the system constructs a classifier by using the relevant and non-relevant documents as positive and negative example, respectively. The newly constructed classifier is then transformed into a boolean query by OR-ing all of the antecedents of the classifier's rules. The initial query is expanded by AND-ing it with the OR-ed conjunctive queries.

In future, we would like to evaluate the effectiveness of our proposed method. More precisely, we plan to evaluate the following points:

- The accuracy of $D_{expanded}$.
- The relationship between the size of D_{init} and the accuracy of $D_{expanded}$. Since the size of D_{init} may become so large, we have to find an "optimal" size of D_{init} in order to obtain a maximum accuracy of $D_{expanded}$.
- The relationship between negative example indication and the accuracy of $D_{expanded}$. For example, we may specify $D_{non\text{-}rel}$ as before but excluding documents classified into the ancestors of the context node.

References

[1] Mandar Mitra, Amit Singhal, and Chris Buckley, "Improving Automatic Query Expansion," in SIGIR, 1998.

[2] Chia-Hui Chang and Ching-Chi Hsu, "Integrating query expansion and conceptual relevance feedback for personalized Web information retrieval," 7th international WWW Conference, 1998.

[3] Jinxi Xu and W. Bruce Croft, "Query Expansion Using Local and Global Document Analysis," in SIGIR, 1996.

[4] William W. Cohen, "Fast Effective Rule Induction," Proceedings of the 12th International Conference, 1995.

[5] William W. Cohen and Yoram Singer, "Learning to Query the Web," in AAAI, 1996.

[6] Soumen Chakrabarti, Bryon Dom, Rakesh Agrawal, and Prabhakar Raghavan, "Scalable feature selection, classification and signature generation for organizing large text databases into hierarchical topic taxonomies," The VLDB Journal, 1998

[7] Chris Buckley, Amit Singhal, Mandar Mitra, Gerard Salton, "New Retrival Approaches Using SMART : TREC 4.", In Harman D., editor, Proceedings of the TREC 4 Conference, 1996.

[8] Chris Buckley, Gerard Salton, James Allan, "The Effect of Adding Relevance Information in a Relevance Feedback Environment.", in SIGIR, 1994.

[9] Ellen M. Voorhees, "Query Expansion Using Lexical-Semantic Relations.", in SIGIR 1994.