A TAXONOMY-BASED FOCUSED RETRIEVAL METHOD FOR THE WEB SPACE

Said Mirza Pahlevi and Hiroyuki Kitagawa

University of Tsukuba
Tsukuba, Ibaraki 305-8573, Japan
{mirza, kitagawa}@kde.is.tsukuba.ac.jp

ABSTRACT

The problem of word ambiguity is fundamental to information retrieval in the web space. This problem originates from the use of very short queries which is common in web information retrieval [1]. One way to deal with this issue is to provide taxonomy to the user so that the user can express his/her query intent to the system by using it. This approach is taken by existing taxonomy (directory)-based search engines. In this paper we propose a novel method to increase the precision of the retrieval results by modifying the user query using a rule-based classifier constructed from a document collection provided by a taxonomy-based search engine. The modification process is dynamic – it depends on both the query given by the user and a selected category from taxonomy of the taxonomy-based search engine. We also describe an alternative static approach and conduct some experiments showing that the dynamic approach outperforms the static one.

1. INTRODUCTION

Crawler-based search engines utilize crawlers that scour the Internet in order to look for pages/documents[1] that will be indexed. Since all the process is done automatically with only a bit of human intervention, the search engines can cover a significant part of the web. However, they suffer from low precision of the search results because the use of very short queries that make the search engines hard to catch the user intent.

Besides providing a short query to a search engine, the user may inform the engine his/her search intent by selecting an appropriate category from taxonomy provided by the engine. The engine then can restrict its search to the category specified by the user. This technique is used by existing taxonomy (directory)-based search engines such as Open Directory Project/ODP (http://dmoz.org/) to improve the quality of the search results. However, since the classification of documents into the taxonomy is done manually, the engines can only cover a small fraction of the web.

There are many attempts to classify the web content automatically into a taxonomy [2][3]. They start with a small sample of corpus that is classified by hand to build a hierarchical classifier. At run time, each document retrieved will be classified automatically by the classifier into an appropriate category. However, this approach has the following disadvantages.

- It is very hard to build a good and large hierarchical classifier that can deal with a wide variety of topics like ODP.
- Most of the classifiers cannot deal with modification of category hierarchies, for instance deletion and addition of category nodes and their associated documents, which is important in the dynamic web environment.

Typically, crawler-based search engines retrieve many relevant documents to the user query, but they are clouded with many non-relevant documents. This happens because of term ambiguity problem originating from the use of the short queries mentioned above. We do not want to let the user lose many useful documents that may exist in the crawler-based search engines and want to provide the documents with as few as possible noise documents. The idea here is to first catch the user intent by using a taxonomy-based search engine, then “put” the intent into the user query, and finally send the modified query to the crawler-based search engines.

In this paper, we propose a dynamic information retrieval method for the web space that combines the taxonomy-based search engine and a machine learning technique in order to improve the quality of search results from the crawler-based search engines. More specifically, we modify the user query by using a rule-based classifier constructed from a document collection provided by a taxonomy-based search engine and send the modified query to the crawler-based search engines. The query is modified such that the results returned by the crawler-based search engines will almost contain documents that will be categorized into a selected category on the taxonomy of the taxonomy-based search engine. The

[1] In the remaining part, we use page and document interchangeably.

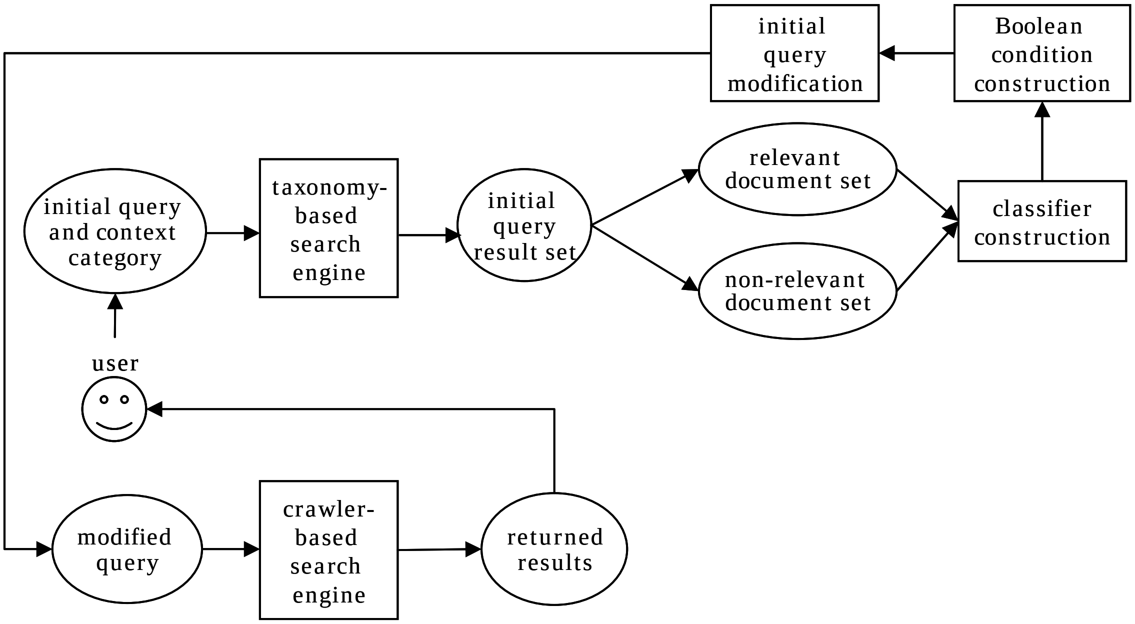
initial query modification
Boolean condition construction
relevant document set
initial query and context category
taxonomy-based search engine
initial query result set
classifier construction
non-relevant document set
user
modified query
crawler-based search engine
returned results

Figure 1. The flow of the proposed method

modification process is dynamic – the classifier used to modify the user query is different depending on both the selected category and the query itself. We show by experiments that modifying queries dynamically has better performance than modifying them statically (i.e., modifying them by using a fixed pre-built classifier).

The paper is organized as follows. Section 2 describes the proposed method. Section 3 gives an alternative approach, which will be compared with the proposed one in Section 4. Section 4 presents the experiments and results. Section 5 reviews related work. In the final section, we give our conclusions.

2. PROPOSED METHOD

Our purpose is to use the existing taxonomy-based search engines to facilitate searches in the web space. One way to do this is to learn/extract useful information from them based on a given user query and a selected category from their taxonomy. The extracted information then can be used to enrich the user query so that the query result quality from crawler-based search engines can be improved. Many of search engines available in the web space typically support Boolean query. Thus, in order to retrieve useful information from them, the enriched user query should be in a Boolean form too.

In short, the following questions should be answered in order to make use of the taxonomy-based search engines to facilitate searches in the web space.

- How to extract the useful information regarding to the user intent from the taxonomy-based search engines?
- How can we enrich the user query with the extracted information so that the resulting query is in a Boolean form?

In this paper, we assume that a crawler-based search engine and a taxonomy-based search engine are available and they can process queries in a Boolean form. We further assume that the taxonomy-based search engine allows search based on all categories existing in the taxonomy and provides additional information about the category of each matched document[2].

Figure 1 shows the flow of the proposed method. We employ the technique used by the taxonomy-based search engines to formulate a query. That is, besides providing search terms to the system, the user specifies a related category from the taxonomy. The system retrieves documents matching the user query, and extracts useful information in the form of relevant terms from them based on the selected category. The relevant terms are used to modify the user query and the modified query is sent to the crawler-based search engine. The following subsections explain details of the steps involved in the proposed method.

[2] Most of the major taxonomy-based search engines support this.

2.1 Query Formulation and Context Category Selection

As mentioned earlier, the query formulation process is same as the search process that is usually used in a taxonomy-based search engine. To find relevant information, first the user navigates the taxonomy provided by the taxonomy-based search engine. After the user has found a category related to the topic sought, he/she then constructs a keyword-based query[3] that will be sent to the engine. We call the category selected by the user as a *context category*. The user may choose the context category after browsing some documents under the category.

2.2 Separation of Relevant and Non-relevant Documents

The system sends the given query condition to the taxonomy-based search engine without specifying a specific category. After the system receives the query results from the engine, it separates the relevant and non-relevant documents based on the context category as follows. (Note that each returned document is associated with its category name.)

- Documents that are classified into the context category (and subcategories under the context category)[4] are considered to be relevant to the user query. This conforms to the method used by the taxonomy-based search engines to catch the user intent.
- Otherwise they are considered to be non-relevant to the query.

Based on this procedure, a relevant document is a document that matches the user query condition and is classified into the context category.

2.3 Query Modification and Execution

After the relevant and non-relevant documents have been found, next the system modifies the user query and sends it to the crawler-based search engines. In this work, we use a rule-based classifier to modify a Boolean query.

First, we construct a classifier for two new categories: *relevant* and *non-relevant categories*. The relevant category is a category for the relevant documents while the non-relevant category is for the non-relevant documents. The classifier is constructed by setting the relevant and non-relevant documents as positive and negative examples, respectively. The resulting classifier is a set of rules in the form of $T \rightarrow c$, where T is a conjunction of terms and c is *Relevant* or *Non-relevant*.

Construction of such rule-based classifiers has been intensively studied in the area of machine learning [4][5][6][7]. In our experiment explained in Section 4, we use RIPPER [5] for constructing the classifier. RIPPER is a rule learning system that constructs a set of rules to distinguish positive examples from negative ones. It repeatedly adds rules to an empty rule set until all positive examples are covered. Rules are formed by greedily adding feature terms to the antecedent of a rule until no negative examples are covered. In short, this learning system can be regarded as learning a disjunction of "contexts", where each context is defined by a conjunction of simple terms.

Next we modify the initial/user query q with the rule set for the relevant category as follows.

1. Let $R = \{r_1, \ldots, r_n\}$ be the rule set for the relevant category, where $r_i = T_i \rightarrow$ *Relevant*. Note that T_i is a conjunction of terms.
2. Let q' be T_1 OR…OR T_n.
3. Finally, we modify q by AND-ing it with q', that is, q AND q' is the query condition of the modified query.

The modified query will be sent to the crawler-based search engine and the returned results will be presented to the user.

As a concrete example, let the initial query be "ATM AND company" with context category "/Computers/ Data_Communications/" selected from taxonomy of the taxonomy-based search engine. After the system sends the query to the taxonomy-based search engine, the system gets documents that have information about the asynchronous transfer mode as relevant documents and the others that have unrelated information (e.g., automated teller machines) as non-relevant documents. The system then constructs a classifier for the relevant category (which corresponds to the asynchronous transfer mode related topic) and non-relevant category (which corresponds to the other topics). Let the resulting rule set for the relevant category be {"networks AND internet → *Relevant*", "switch AND asynchronous → *Relevant*"}. Applying the transformation steps shown above to the initial query, the resulting modified query becomes "ATM AND company AND ((networks AND internet) OR (switch AND asynchronous))".

The classifier is used to tell whether a document that matches the initial query condition will be classified to the context category. Hence, by "sending" the classifier with the query to the crawler-based search engines (i.e., transforming it to a Boolean condition and modifying the

[3] Most of search engines treat the given terms as a term conjunction, and thus we assume this is a Boolean query.

[4] In the remaining part, we refer to the context category and its descendant subcategories just as the "context category".

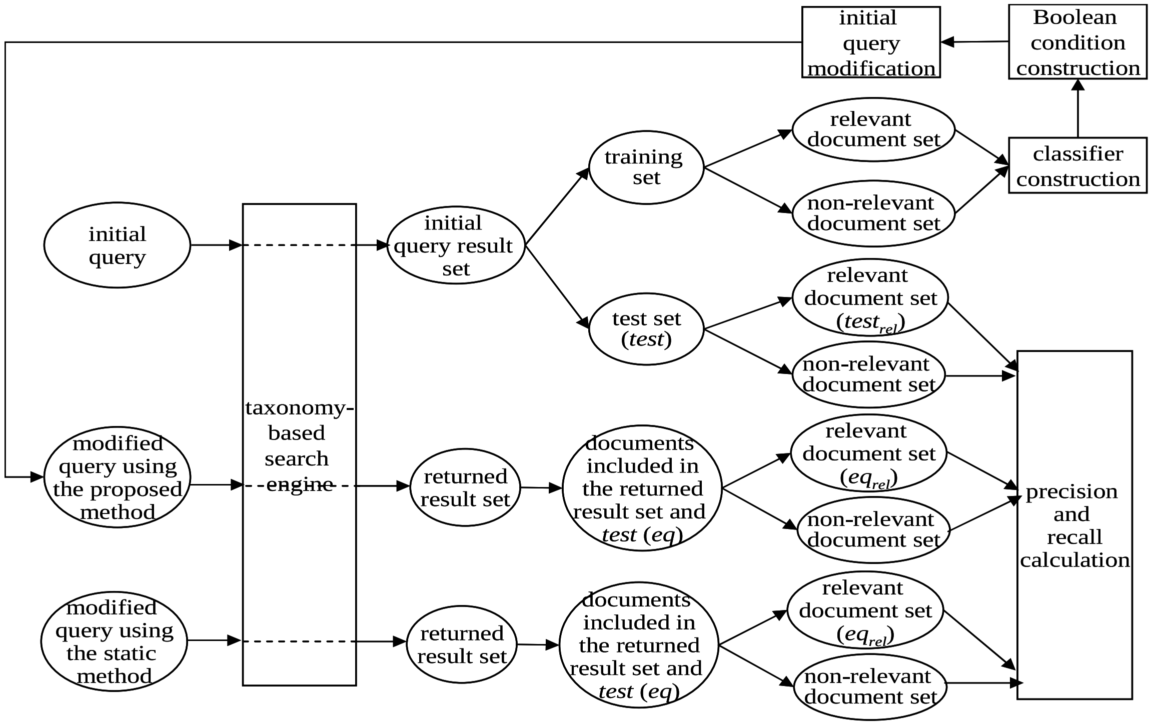
initial query modification
Boolean condition construction
relevant document set
training set
classifier construction
non-relevant document set
initial query
initial query result set
relevant document set ($test_{rel}$)
test set ($test$)
non-relevant document set
taxonomy-based search engine
modified query using the proposed method
returned result set
documents included in the returned result set and $test$ (eq)
relevant document set (eq_{rel})
non-relevant document set
precision and recall calculation
modified query using the static method
returned result set
documents included in the returned result set and $test$ (eq)
relevant document set (eq_{rel})
non-relevant document set

Figure 2. Experiment method

initial query), it seems that the returned results from the search engines will almost contain documents that are related to the user intent.

3. AN ALTERNATIVE APPROACH

As an alternative approach, we can create a classifier for each category in the taxonomy by treating documents in the category and other categories as positive and negative examples, respectively. The initial query then can be modified using the rule set for the selected context category by employing the procedure shown in Subsection 2.3.

We denote this approach as a *static* one because the classifier used to modify the initial query in each category is always fixed (i.e. they are built prior to the query processing time and do not depend on the given user query). Since the classifier is static and "forced" to cover many topics that may exist below the category, it seems that it cannot fit better to the user query, resulting in degradation of the retrieval performance. This point is validated in the next section.

4. EXPERIMENTS AND RESULTS

4.1 Experiment Method

To evaluate the effectiveness of the proposed method, we compare the precision and recall of the modified query based on the proposed method with those of the modified query based on the static method and those of the initial query. In order to calculate the precision and recall of a query, we have to know the "true answer" of the query with respect to a selected context category. One way to do this is to check whether each document in the results returned by the crawler-based search engines is relevant or not to the query. However, this approach requires too much effort since the returned result size is usually very large.

To make relevance judgment easy, we simulate the crawler-based search engine with a taxonomy-based one. This can be done by having the search carry out against documents in all categories of the taxonomy-based search engine. That is, the search is not done against a particular category as usual. The "true" answer of a query from the simulated crawler-based search engine is the subset of documents that match the query condition and that are classified into the context category. (Note again that the returned documents are associated with their categories.)

The detail of the experiment is shown in Figure 2. The taxonomy-based search engine has two functions: it is used to catch the user intent by the proposed method and used as the simulated crawler-based search engine.

The flow of the experiment is as follows. First, we define an initial query and select an appropriate context category for the query (i.e. select a context category that matches the query intent) from the taxonomy. After the query is

Table 1. Queries and their meanings at broad context categories

Query	Broad context category		Meaning
	Notation	Context category	
q1: ATM	c1.1	/Computers/	Find pages that mention ATM networks.
	c1.2	/Business/Financial_ Services/	Find pages that mention ATM of banks.
q2:salsa	c2.1	/Arts/	Find pages that mention Salsa dance and music.
	c2.2	/Shopping/	Find pages selling Salsa sauce (but they may also sell other products).
q3:apple	c3.1	/Computers/	Find pages related to Apple computers (companies, hardware, software, etc.).
	c3.2	/Home/Cooking/	Find pages about apple cooking but not pages selling apple food products.
q4:oil AND product	c4.1	/Business/Industries/	Find pages about fabrication of oil finished products including food related oil products and pages about oil and gas industries.
	c4.2	/Shopping/	Find pages selling oil products for health including beauty oil product, aromatherapy (essential oils), acne oil, etc.

submitted to the taxonomy-based search engine without specifying a specific category, we get the initial query result set. The result set is then divided into training set and test set (*test*), which in turn are divided into relevant and non-relevant document sets based on the selected context category. The relevant and non-relevant documents in the training set are used to construct the classifier, which in turn is used to modify the initial query. The resulting modified query is then sent to the simulated crawler-based search engine (in this case the taxonomy-based search engine itself) and the precision and recall of the returned results are calculated based on *test*. (Thus the role of the test set is for evaluation purpose.)

As mentioned earlier, we also do a comparison with the static approach. Thus we also send the modified query from the static approach to the simulated crawler-based search engine and calculate the precision and recall of the returned results.

The precision and recall of both modified queries are calculated as follows. Let *eq* be the set of documents that are included both in the result set of the modified queries and in *test*. Let eq_{rel} be the set of relevant documents in *eq*, namely, documents that meet the initial query condition and are classified into the context category. Similarly, let $test_{rel}$ be the set of relevant documents in *test*. In this experiment, $test_{rel}$ is the “true” answer of the initial query because it is a relevant document set[5] and it is not involved in constructing classifier of the proposed method. We calculate the precision and recall of the modified queries using the following equation.

$$precision = \frac{|eq_{rel}|}{|eq|} \qquad (1)$$

$$recall = \frac{|eq_{rel}|}{|test_{rel}|} \qquad (2)$$

Note that recall of the initial query is always 1, while the precision is calculated based on the following equation.

$$precision = \frac{|test_{rel}|}{|test|} \qquad (3)$$

We conduct the evaluation process with 3-fold cross validation. The initial query result set is randomly partitioned into 3 mutually exclusive subsets, s_1, s_2 and s_3, each of approximately equal size. The recall and precision calculation is performed 3 times, where at the ith iteration, the subset s_i is used as the test set and the remaining subsets are collectively used as the training set. The recall and precision values shown later in the experiment results are the average of the 3 times evaluation results.

We use Open Directory Project/ODP as the taxonomy-based search engine. ODP has about 440,000 categories and over 3 million recorded sites as of December 2001. When doing a search, ODP looks for matches with web site titles, comments, and URLs. Therefore, the search results are lists of site entries, each of which consists of a title, description, address and category name. In the experiments, each site entry is regarded as a document.

[5] That is, it is a set of documents that match the user query and are classified into the selected context category.

Table 2. Queries and their meanings at narrow context categories

Query	Narrow context category		Meaning
	Notation	Context category	
q1:ATM	c1.1'	/Computers/Data_Communications/	Find pages related to ATM networks in data communication.
	c1.2'	/Business/Financial_Services/Banking/ Services/	Find pages related to ATM of banks especially in banking services.
q2:salsa	c2.1'	/Arts/Performing_Arts/Dance/	Find pages mainly related to Salsa dance.
	c2.2'	/Shopping/Food/Condiments/	Find pages that mainly sell Salsa sauce.
q3:apple	c3.1'	/Computers/Systems/	Find pages specially related to Apple computer systems.
	c3.2'	/Home/Cooking/Fruits_and_Vegetables/	Find pages describing various recipes using apples.
q4:oil AND product	c4.1'	/Business/Industries/Energy/	Find pages about oil and gas industries.
	c4.2'	/Shopping/Health/Beauty/	Find pages that mainly sell beauty oil products.

4.2 Experiment Results

Tables 1 and 2 show queries and their context categories used in the experiment. There are 4 queries with 16 different meanings. The meaning of each query depends on its context category and is derived from category description of the context category provided by ODP. We select two context categories for each query such that the meaning of the query at each context category is different. For example, the meaning of query “apple” at context category “/Computers/” is completely different from the same query at different context category “/Home/ Cooking/”. Queries shown in Tables 1 and 2 are same, but the meanings of the queries in Table 1 are broader than those of Table 2. Hence, we call context categories in Tables 1 and 2 as *broad* and *narrow context categories*, respectively.

Figures 3 through 8 show the experiment results. I, S and P denote the initial query, query modified by the static method and query modified by the proposed method, respectively. The recall of the initial query is omitted, because it is always 1. As shown in Figures 3 and 6, at broad and narrow context categories the two modification methods can significantly increase the precision of the initial queries. However, it is clear that the precision of the proposed method (especially at the narrow context categories) is better than that of the static method. This indicates that the proposed method is more suitable for the search in a huge database collection like the web where the precision is more important than the recall.

At broad context categories, the recall of the proposed method is generally better than that of the static method (Figure 4), while at the narrow context categories it is the reverse (Figure 7). However, as shown in Figures 5 and 8, the F1-measure of the proposed method is generally better than that of the static method.

5. RELATED WORK

The most closely related to our work is the Inquirus 2 [8] developed at NEC research Institute. They proposed an automated method for learning query modifications to locate pages within specified categories using web search engines. In the first step, for a specific category a classifier is trained to classify pages by membership in the desired category. In the second step, query modifications are constructed by calculating expected entropy loss for each feature term extracted from document collection of the category. A query modification is a combination of terms to expand the user query. Finally, because all search engines will not give the same response to the query modifications, they use the classifier constructed in the first step to produce a ranking of search engine and query modification pairs.

Our work is difference from theirs in that we use an existing taxonomy and dynamically constructed classifiers to catch the user intent. By using the existing taxonomy, we can make best of it as a useful information source. On the other hand, they use flat categories that they have to construct and provide to users. In addition, their query modification is static, while ours changes depending on the query provided by the user.

Another related work is WebSifter II, a semantic taxonomy-based personalizable meta-search engine agent system [9]. In their system, first the user creates personalized search taxonomies expressing his/her query intent via the proposed Weighted Semantic-Taxonomy Tree. Next, the node/category labels in the tree are further refined by consulting a web taxonomy agent such as

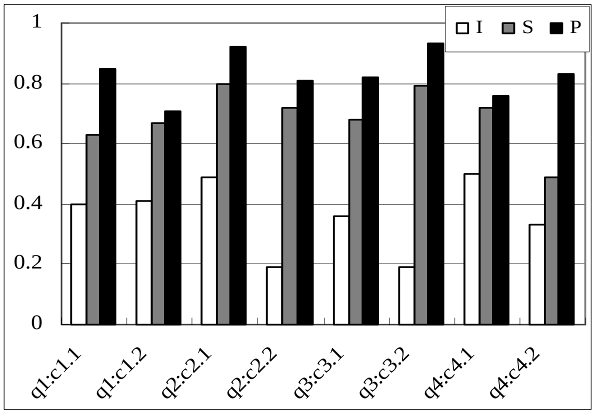

Figure 3 Precision at broad context categories

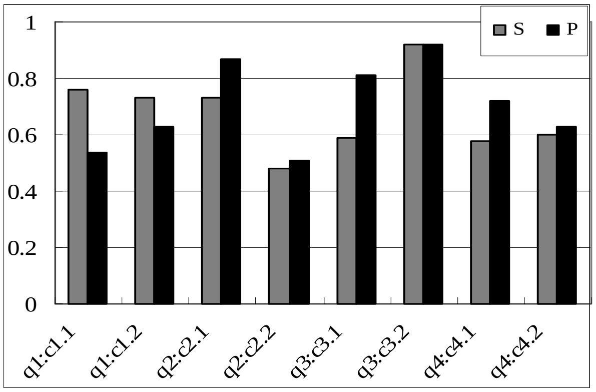

Figure 4 Recall at broad context categories

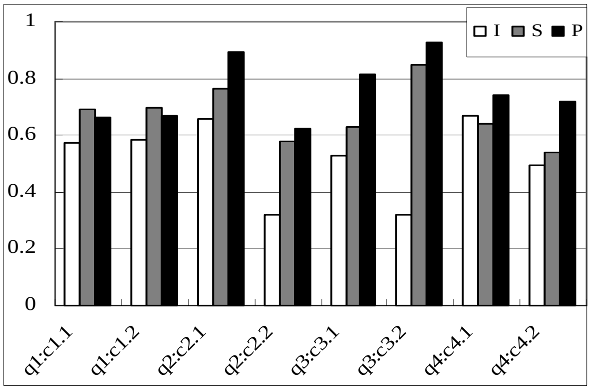

Figure 5 F1-measure at broad context categories

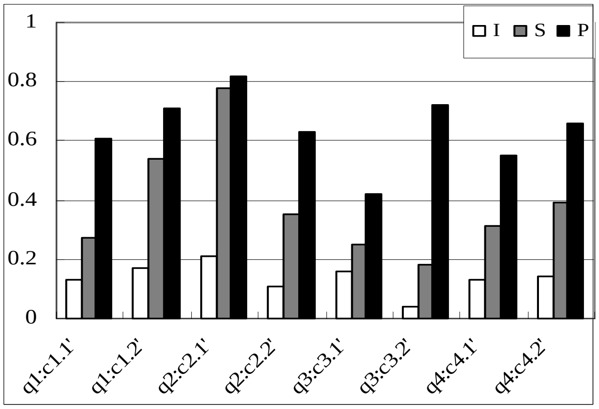

Figure 6 Precision at narrow context categories

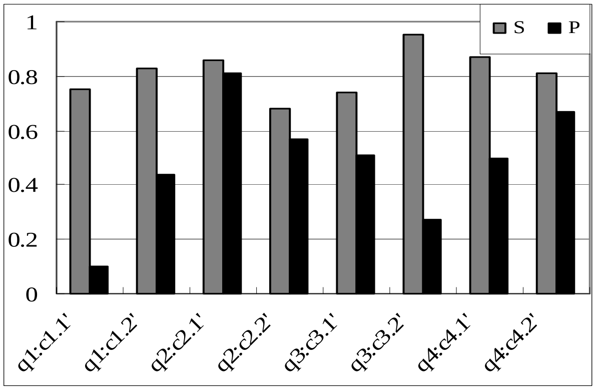

Figure 7 Recall at narrow context categories

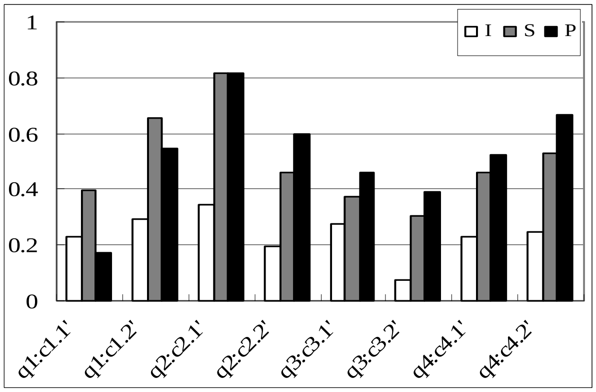

Figure 8 F1-measure at narrow context categories

Wordnet to eliminate the term ambiguity problem. Finally, the concepts represented in the tree are transformed into Boolean queries processed by existing search engines. Although the system uses taxonomies, it does not employ classifiers. In addition, the system needs a new taxonomy for each query intent.

Ref. [10] studied the automatic classification of web documents into pre-specified categories, with the objective of increasing the precision of web search. They start by building a classifier for a set of categories using pre-classified training set of pages. In the query formulation step, the user specifies not only the query terms, but also one or more categories in which he/she is interested. The system retrieves documents matching the query, then filters them by comparing their categories given by the classifier. This method only classifies the query results and does not modify the user query.

Ref. [11] proposed an interactive query learning system to keep resource directories up-to-date. Resource directories are “bookmarks” that collect together links to all known documents on a specific topic. The user via an augmented web browser specifies positive and negative examples incrementally for the current topic. Then he/she can invoke the system to create new rules 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. Their main focus is query generation rather than query modification.

6. CONCLUSIONS

We have proposed a dynamic information retrieval method combining a taxonomy-based search engine and a machine learning technique in order to improve the quality of search results from crawler-based search engines. The method utilizes taxonomy provided by an existing taxonomy-based search engine and can dynamically modify the user query based on the selected context category so that the returned results from the query may contain many documents matching the query intent. The user can freely shift the broadness of his/her intent topics just by selecting an appropriate category from the taxonomy.

Our method is dynamic in that the classifier constructed to modify the query is different depending on both the selected category and the query given by the user. In other words, our method dynamically constructs a small classifier corresponding to a small part of the taxonomy that is related to the current user query. As shown in the experiment results, the performance of the proposed (dynamic) method can outperform the static one. In particular, the proposed method can increase the precision much better than the static one. This indicates that the proposed method is more suitable for the search in a huge database collection like the web where the precision is more important than the recall.

Moreover, since the modified query is still in a Boolean form, our method will be applicable to any text databases supporting Boolean search. This characteristic is very important because we can get more relevant information not only from the crawler-based search engines but also from many legacy databases or hidden web sites (i.e. web sites whose pages cannot be indexed by crawlers) that actually occupy a large portion of the web.

7. ACKNOWLEDGEMENT

This research has been supported in part by the Grant-in-Aid for Scientific Research from Japan Agency for the Promotion of Science.

8. REFERENCES

[1] B. J. Jansen, A. Spink, J. Bateman and T. Saracevic. "Real Life Information Retrieval: A Study of User Queries on the Web". *SIGIR Forum*, 32(1), 1998, pages 5-18.

[2] S. Chakrabarti, B. Dom, R. Agrawal, and P. Raghavan. "Scalable Feature Selection, Classification and Signature Generation for Organizing Large Text Databases into Hierarchical Topic Taxonomies". *The VLDB Journal*, vol. 7, no. 3, 1998, pages 163-178.

[3] S. Dumais and H. Chen. "Hierarchical Classification of Web Content". *In Proceedings of the Annual Int. ACM SIGIR Conference on Research and Development in Information Retrieval*, 2000, pages 256-263.

[4] W. W. Cohen. "Fast Effective Rule Induction". *International Conference on Machine Learning*, 1995, pages 115-123.

[5] C. Apte, F. Daerau, and S. Weiss. "Towards Language Independent Automated Learning of Text Categorization Models". *In Proceedings of the Annual Int. ACM SIGIR Conference on Research and Development in Information Retrieval*, 1994, pages 23-30.

[6] Timothy A. H. Bell and A. Moffat. "The design of a high performance information filtering system". *In Proceedings of the Annual Int. ACM SIGIR Conference on Research and Development in Information Retrieval*, 1996, pages 12-20.

[7] I. Moulinier, G. Raskinis, and J. G. Ganascia. "Text categorization: a symbolic approach". *In Proceedings of SDAIR-96, 5th Annual Symposium on Document Analysis and Information Retrieval*, 1996, pages 87-99.

[8] E. J. Glover, G. W. Flake, S. Lawrence, W. P. Birmingham, A. Kruger, C. L. Giles, and D. M. Pennock. "Improving Category Specific Web Search by Learning Query Modifications". *Symposium on Applications and the Internet (SAINT)*, 2001, pages 23-31.

[9] L Kerschberg, W. Kim and A. Scime. "A Semantic Taxonomy-Based Personalizable Meta-Search Agent". *Proceedings of the second International Conference on Web Information Systems Engineering (WISE)*, 2001, pages 53-62.

[10] C. Chekuri and M. H. Goldwasser. "Web Search Using Automatic Classification". *Poster at the Sixth International WWW Conference (WWW6)*, 1997.

[11] W. W. Cohen and Y. Singer. "Learning to Query the Web". *In Proc. Workshop Internet- based Information Systems 13th Nat. Conf. Artificial Intelligence*, 1996, pages 16-25.