Assisting Information Retrieval with Domain Specific Topic Identification

R.Rajendra Prasath, Vijai Kumar and Sudeshna Sarkar
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur - 721 302, India
Email: {rajendra, sudeshna}@cse.iitkgp.ernet.in

Abstract:- ***This work attempts to report the effects behind assisting Information Retrieval with domain specific topic identification through query and document classification. Users search for specific information related to his / her topics of interest and hence the information retrieval systems need to focus particularly on domain specific retrieval of relevant content. To assist such retrievals with the most specific information to the given query, we have presented a method to identify the topic(s) of the given query; to retrieve the documents through filtering with the most matching query topic(s) and to return the ranked list of retrieved documents with respect to the specific domain. We have collected domain specific queries and crawled the tourism related web data for our experiments. The preliminary experimental results show that domain specific topic identification narrows down the retrieval process with more specific information related to the tourism domain.***

Keywords: *Domain Specific Information Retrieval, Topic Identification, Query and Document Classification, Retrieval Efficiency*

I. INTRODUCTION

Retrieving relevant documents at the rtop positions of the ranked list is always a challenging task in Information Retrieval. Different attempts have been made to explore the information needs of users behind the user query so as to assist the retrieval tasks [1], [2], [3], [4], [5], [6], [7], [8]. A major challenge in Information Retrieval (IR) systems is to deal with incomplete or under specified information in the form of queries issued by users. The IR systems receiving such queries need to fill in the gaps in the under specified query of the users so as to improve the retrieval efficiency[9]. The problems multiply in cross lingual domain where CLIR systems often utilize translation to cross the language barriers between a query and the documents. However the disambiguation of under specified queries make the retrieval and ranking of documents more challenging.

Query Structuring method for grouping query keywords has been proposed in[10]. This method suggests the use of query operators in such a way that more weight is being assigned to important or correct keywords than the other keywords. Also this query structuring captures the clue on the intention of users information need with English-Finnish news data. Here we are attempting towards such a mechanism to retrieve the documents by identifying the query type that gives a clue on the information need of users across the topics in a specific domain and then to rank these retrieved documents towards the better ordering.

II. MOTIVATIONS

Machine learning for IR is quite attractive in recent years. As the user search moves towards

the specific issues in deeper level, modern retrieval mechanisms struggle to identify the documents matching the expectations behind the information needs of users. To perform the information retrieval towards providing topic specific text information, it becomes essential to apply topic / class identification on text documents. Since the manual topic identification tasks are not feasible in a scalable IR system, Machine Learning techniques assist us to classify / cluster the documents with appropriate topic information. Representing text documents in the knowledgerich space of constructed features [predefined topics / training instances] leads to a greater categorization accuracy. Using the automated machine learning scheme, given the domain specific knowledge, document texts are examined and their representation is enriched in a completely mechanical way. Text categorization algorithms using ontologies and open source external knowledge repositories attract more attention due to the importance of its context oriented search and its close association with the retrieval of relevant documents in real time applications. Motivated by above considerations, we focus here on applying machine learning techniques for information retrieval with a substantially wider body of domain specific knowledge on Tourism domain.

III. THE OVERVIEW OF THE PROPOSED APPROACH

We briefly describe the overview of the proposed system. Initially, we have identified the domain specific, in our case tourism related, resources like query patterns and topics [class labels] and then built domain specific ontologies for tourism domain. Using these resources, the query classifier is built to identify the set of topics each with its matching score. Next we have applied the classification task on documents in the tourism domain and tagged each of them with its maximum matching topic. Then we have performed documents retrieval using the standard TFIDF (Term Frequency * Inverse Document Frequency) model with cosine similarity with and without topic identification.

A. Topic Identification

Topic identification helps to narrow down the search towards the focused information retrieval. This task is crucial in various search domains like domain specific web search, personalized IR, online advertisement and so on. Web query classification (QC) has been widely studied for this purpose. Most of the previous QC algorithms classify individual queries without considering their context information. However, as exemplified by the well known example on the query jaguar, many web queries and their real meanings are uncertain without the context information. Because these queries are short and ambiguous, interpreting the queries in terms of a set of target categories has become a major research issue. As a result, understanding the search intent behind the queries issued by web users has become an important research problem. Direct matchings between queries and target categories often produce insignificant results. In addition, the target categories can often change depending on new web contents as the web evolves, and as the intended services change as well. Thus Query Classification [or Query Categorization] (QC) has been studied for classifying user queries under a ranked list of predefined domain specific categories. Such category information can be used to trigger the most appropriate vertical searches corresponding to a query, to improve web page ranking [2], and to fetch the relevant documents.

B. Topic Identification of Queries

Query classification is dramatically different from traditional text classification because of two issues: At first, web queries are usually very

short and consist of 2-3 words on an average [3]. Secondly, many queries are ambiguous [11], and it is common that a query may belong to multiple categories. For example, [5] manually labeled 800 randomly sampled queries from the public data set from ACM KDD Cup'051, and 682 queries have multiple category labels. To address the challenges in QC, different query classification approaches have been proposed in the literature. In general, these approaches can be divided into three categories: The first category tries to augment the queries with extra data, including the search results returned for a certain query, the information from an existing corpus, or an inter- mediate taxonomy [6], [5]. The second category leverages unlabeled data which help to improve the accuracy of supervised learning[3]. Finally, the third category of approaches expands the training data by automatically labeling some queries in some click-through data via a self-training-like approach [12].

Intuitively, use of search context information, such as the adjacent queries in the same session as well as the clicked URLs of these queries, can help better to understand users search intent and thus improve the classification accuracy. As shown in [4], [7], [8], adjacent queries raised by the same user are usually semantically related. Moreover, compared with search queries, which are often short and ambiguous, the URLs that are selectively clicked by a user after issuing the queries may better reveal the search intent of the user.

There are many ways in which a query can be written. Each user may use different set of words for searching the same information or writing the same query in different forms. So we have to collect a wide range of unbiased queries from regular web users and different other sources on the Internet like 'yahoo questions', 'travel blogs', www.raahi.com/travel-questions, www.oktatabyebye.com/travelquery.

Initially, 10 query topics were identified manually by analyzing the collected queries (refer to section. III-B1. These labeled queries are then used for training the query classifier to predict the topic of the new unlabeled query.

1) Identified Query Classes:

We have identified the following query topics in Tourism domain: Accommodation, Activities, Attractions, Climate, Food, Idealtime, Inf, info mation, Location, Tourpackage and Travel. For each identified query topic, we have derived a set of patterns that are used to identify the type of the user query. Here we have listed the pattern derived for the tourism specific topic : **Climate:**

ϴ Climate:

Example: *Climate of Goa in may, Weather and rainfall in Darjeeling.*

Possible attribute - value pairs:

(a) Place Name - like Goa, Darjeeling

(b) Time (month or season) - like May, weather, rainfall The following method is used to extract the attribute - value pairs related to Climate:

```
For each query term, q
    If q is place Name
        Then place = q
    If q is equal to in or during or of
        if next term is place name
            then place = next term
    if next term is a valid month or season
        then time = next term
```

C. Topic Identification of Queries

We list here three methods for query classification(QC).

1) Keyword Matching:

In this method, we have a list of resource terms for each predefined topics and the given query having k terms is compared with each of the resource term lists(having keywords). The topic of the maximum matching terms with the resource list is assigned as the identified topic to the given query. Basically this is a "Bag of words" approach. For each basic query type, we maintain a list of resource terms to define it. For example, if the basic query type is related to:

(a) Travel:

Keywords - route, reach, nearest, flights, airport, best, driving, road, bus, station, train, railways.

(b) Food:

Keywords - restaurants, eat, food, vegetarian, dining, dinner, lunch, breakfast.

After removing stopwords, each query term is matched with the list of resource terms. For each query given by user, after removing stopwords we will match each query term with the list of resource terms one by one. The score is calculated as follows: Score = No of match between query terms and the keywords / No of query terms We will find the score for every basic query type. Finally the input query is mapped to the basic query type with the maximum score.

2) Pattern Matching:

In this section, we present a method for query classification task using the identified domain specific patterns. The main object of this method is find the most matching type of the given user query related to the tourism domain using manually crafted tourism related patterns. We have listed below the patterns that are used to find the corresponding topic of the given query:

- **Climate**
  (climate | weather | temperature | rainfall)\\s+(of | in)\\s+[a-zA-Z]+
- **Ideal time**
  (time | season)\\s+(to | for)\\s+(visit | tourism | stay(| ing))\\s +in\\s+[a-zA-Z]+

- **Accommodation**
  1. [a-zA-Z]+\\s+to\\s+stay
  2. accomodation(| s)\\s+available\\s+in\\s+[a-zA-Z]+
  3. room(| s)\\s+for\\s+[09]+\\s+(person | day)(| s)
  4. (lodge | resort | dharamshala | hotel | accomodation)(| s)\\s+in \\s+[a-zAZ]+
  5. (to | for)\\s+stay(| ing)\\s+in\\s+[a-zA-Z]

Algorithm 1 The proposed overview of Query Classification Methods

Input:

Given Query having k terms $Q = \{t_1, t_2, t_3 \cdots, t_k\}$, $k > 0$
A set of resource terms OR identified patterns OR topic tagged queries

Description:

1: Load the resources (keywords, patterns, training data) for the respective methods
2: Using the resources (keywords, patterns, training data), find the matching topics [each with its corresponding score] of the given query.
3: Sort the matching topics with respect to its score in decreasing order
4: return the top most matching topic as the identified topic of the given query

Output: The identified topic of the given user query

3) Naive Bayes:

The list of manually tagged queries each with its specific topic related to the tourism domain is considered. These topic tagged queries are used as the training data and the classification model is built using Naive Bayes method. This classification model is used to predict the topic

of the given user query. The output will be the topic with highest matching score for the given user query.

D. Topic Identification of Documents

Identifying the topic of given documents lies in Machine Learning. The task is to assign a document to one or more categories, based on its contents. Automatic document classification is an important research area and has significant involvement in Information Retrieval. Text (in terms Document) classification is highly a matured research topic with lots of advances towards intelligently classifying document content with relatively high performance. However, most of them are basically built on a simple bag of words representation of documents [13], [14], [15], [16]. We list the type of classification algorithms in the sequel:

E. Rule-based Classifiers

These classifiers learn by inferring a set of rules (a disjunction of conjunctions of atomic tests like "this feature has that value") from pre-classified documents. A good example is the Ripper algorithm [17]. The decision rules may take the form of decision trees. Other algorithms come from work on Theorem Proving, and may be based on propositional logic or first and even second order logic.

F. Linear Classifiers

In algorithms for linear classifiers, for each topic a class profile is computed, a vector of weights, one for each feature, based on occurrence frequency and probabilistic reasoning. For each class and document, a score is obtained by taking an in-product of class profile and document profile. This class contains various adaptations in Information Retrieval.

G. Example-based Classifiers

These classifiers classify a new document by finding the k documents nearest to it in the train set and doing some form of majority voting on the classes of these nearest neighbours [18], [19]. Generally, document recognition score depends much on feature selection and classifier selection. The commonly used feature selection methods include Document Frequency, Mutual Information, Information Gain, Chi-Square, Term Strength. etc. [20]. Presently many algorithms on document classification such as Med, k−Nearest neighbour, Artificial Neural Networks, Hidden Markov Model [21], have been put forward. Fuzzy clustering has also been used for document categorization [22]. One of the major characteristics or difficulties of the document categorization is the high dimensionality of the feature space. The dimensionality problem, as questioned by Yang and Pedersen [19] continues to attract the attention of researchers nowadays.

Document classification tasks can be divided into two sorts: **Supervised document classification** where some external mechanism (such as human feedback) provides information on the Correct classification for documents:

Unsupervised document classification where the classification must be completely done without reference to external information.

Here, we have used the following supervised document classification algorithms to classify the text filtered from web documents using CMLifier:

- K-NN (k-nearest neighbor algorithm).
- Naive Bayes

IV. CONTENT FILTERING USING CMLIFIER

CMLifier1 is a Content Filtering tool used to extract focused content of the web document by

eliminating noisy contents. This tool provides a type of structured content in the form of attribute - value pairs and this similar is similar to XML. CMLifier takes the raw HTML content as the input and re-formulates the ill-tagged / semi structured html content into well formed html tree structure. Then based on the structure of the underlying markup language, noisy contents like advertisements, images, copyright information and so on are filtered out. CMLifier converts the hex representation to unicode representation and provides facility to customize with user specific fields. The extracted contents along with other fielded attribute - value pairs will be used as the input documents. Then we we present a method for finding the topic of CMLified content using Naive Bayes and k−NN methods.

V. THE PROPOSED METHOD

In this section, we propose the algorithm that is assisted by the topic identification of queries and documents. The detailed algorithm is given in the sequel:

A. Assisting Retrieval with Topic Identification

At first, documents are classified with domain specific class labels as an offline process. Then user inputs the query to the retrieval system. Query classifier finds the topic of the given query and retrieves documents using TF-IDF along with matching topic information of query and documents using:

$$tf - idf_{t,d} = tf_{t,d} * \log \frac{N}{df_t} \quad (1)$$

where *tft,d* - denotes term frequency of the term *t* in document *dft* denotes the document frequency of the term t and tf − idft,d assigns to term t a weight in document d.

$$sim(q^c, d_i^c) = tf - idf_{t,d} * \alpha \log(\frac{w}{1-w}) \quad (2)$$

The similarity between q and di having the matching topic is computed using cosine angle between them.

where q^c - given query with the identified topic *c*; d^c_i - i^{th} document with the identified topic that matches with the topic of the query; α - the preference factor to support the documents with the same topic as the candidate towards meeting the users' information need based on the given query;

Algorithm 2 The Proposed Classification method

Input:

A set of n text documents D = {*d1, d2, d3* · · · , dn}

A set of predefined category labels C

Description:

1: Preprocess the raw HTML content and extract the clean text using CMLifier.

2: Split the data into 70% for training and 30% for testing

3: Build classifiers (Naive Bayes / k−NN) using WEKA APIs and generate the classification model on the training data having the clean text.

4: Now using 30% of the test documents, perform classification and compute the classification accuracy using the classification model build in the previous

1The Content Extraction tool developed under Cross Lingual Information Access (CLIA) project, Govt. of India. The main system can be accessed using: http://www.clia.iitb.ac.in:8080/clia-latest/locale.jsp?en

step. The classification model returns the topics [each with its corresponding score] for each of the given input documents.

5: Sort the topics with respect to its score

6: return the top most matching topic as the identified topic of the given document

Output: The topic label for each document

---

w - denotes the document boosting factor with respect to the identified matching topics of the query and document. If the retrieved documents are not enough, then we take the next matching topic of the query into consideration for retrieval. Then the retrieved documents are ranked and top k results are returned to the user.

Algorithm 3 Document Retrieval with Topic Identification

Input:

A set of queries: Q = {*q1, q2, q3* · · · , *qk*}

The Query Classifier

The Document Classifier

Description:

1: Users input the domain specific query to IR System

2: Topic of the input query is identified using one of the query classifiers.

3: Similarity between the query and document with the identified matching topic is computed using Eqn. 2.

4: Retrieved documents are sorted based on similarity score.

5: If retrieved documents are not enough for the identified most matching topic of the query, then the next most matching topic of the query is taken into consideration and repeat (steps 3 and 4) two steps are repeated with the next most matching topic.

6: return the list of top n ranked documents

Output: The ranked list of documents for each query.

VI. EXPERIMENTAL RESULTS

A. Dataset and Queries

First we present the experimental results obtained for query classification. Figure. 1 shows the number of queries manually crafted for each topic.

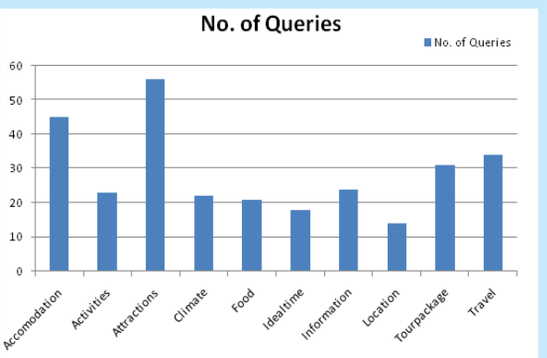

Fig. 1. Statistics on the number of Queries

	Keywords	Pattern Matching	Naive Bayes
Precision	0.847	0.9714	0.789
Recall	0.56	0.56	0.765
F-Measure	0.674	0.71	0.777
Accuracy	0.46	0.53	0.65

TABLE I
EFFECTS OF EVALUATION MEASURES ON QUERY CLASSIFICATION METHODS

At first, we have collected 288 queries through different web users on tourism domain. The collected queries are belonging to specific tourism related topics and additionally we have included most of the possible variations to each topic. In tourism domain, the distribution of the collected queries shows that the main interest of

the users lies in querying for the content related to the topics : attractions, accommodation and travel.

Table. I shows the effects of various evaluation measures on query classification methods. The Pattern Matching approach for the Query Classification shows high precision but low recall. This is evident from the fact that the retrieved results contain more number of true positives (retrieved results which are relevant) and very few false positives (retrieved but nonrelevant). However the pattern matching approach is not able to identify all possible patterns for the respective topics therefore providing low recall. On the other side, Naive Bayes performs better on an average as it has been applied to the domain specific tagged text collection which is treated as Bag of Words(BoW) model where the terms are assumed to be independent of each other. Figure. 2 shows topic wise query classification accuracy.

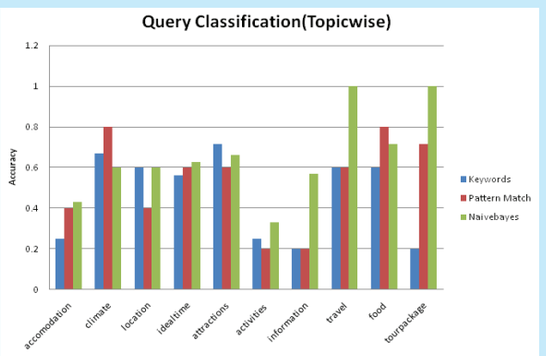

Fig. 2. Query Classification Accuracy (Topicwise)

1) Document Classification:

This section carries the simulation results of document classification using Naive Bayes and k−Nearest Neighbors.

We have used Weka APIs to build the classifier. WEKA is a practical machine learning tool whose APIs make it easy to “embed” it in machine learning related projects. We have built the classifier using Naive Bayes algorithm in Weka to classify CMLified document contents. The input to this classifier is a set of documents (training data) tagged with their respective class. The classification model is built using training data and then its performance is evaluated on test data which comes from the same distribution as that of the training data. This model is then used to predict the topic of an unlabeled document. Next we have built another classifier using WEKA APIs based on k-NN algorithm to classify CMLified documents.

The input to this classifier is the set of labeled documents. These document are stored as <Topic, Doc> pairs. Topic is the class tag to which the document belongs to and Doc denotes the specific document. In both classification experiments, we have used 10 fold cross validation and 70% documents for training and 30% for testing.

Figure. 3 shows the topic wise documents distribution in the corpus used for classification.

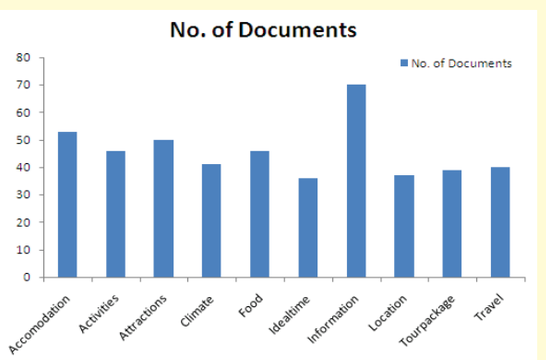

Fig. 3. Document distribution across various topics

Table. II shows the effects of various evaluation measures on. document classification methods.

Among these two methods, k−NN method with k = 5 performs better for the domain specific [tourism] data. Since the coverage of the tagged data is not sufficient, it is hardly possible to observe the best among Naive Bayes and k−NN methods.

	Naive Bayes	k−NN
Precision	0.624	0.66
Recall	0.598	0.655
F-Measure	0.611	0.657

TABLE II
EFFECTS OF EVALUATION MEASURES ON DOCUMENT CLASSIFICATION METHODS

Figure. 4 shows the classification accuracy [topic wise] obtained with Naive Bayes amd k−Nearest neighbors on 458 tourism related web documents.

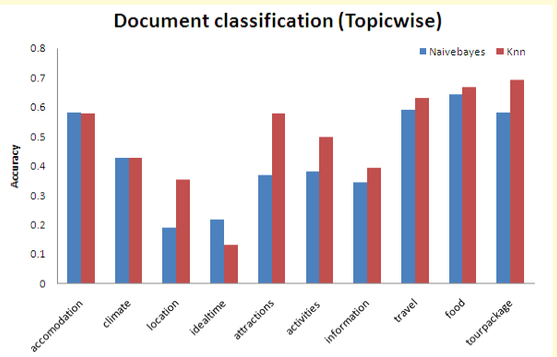

Fig. 4. Classification accuracy [topicwise] with Naive Bayes and k− NN methods

2) Retrieval with Topic Information:

In order to evaluate the proposed document retrieval method, we have used 1200 tourism specific contents extracted from web documents and 10 Tourism related English queries from pre FIRE2 2008 query set. Usually, FIRE queries are set in the standard TREC format with three fields: TITLE, DESC and NARR. Among these three fields, we have used TITLE (= actual query)and DESC (= expanded query) fields for document retrieval.

B. Baseline Method - TFIDF

We have taken the standard Term Frequency * Inverse Document Frequency (TFIDF) as the base line (without any information on the topic) for our comparisons. Given a query, we compute the matching score using cosine similarity between query and documents and then rank them by decreasing order of their similarity score. Top matching documents are obtained and evaluated based on the guidelines used in the evaluations of Text REtrieval Conference (TREC) with the help of relevant judgments.

C. Retrieval with Classification Information

In this experiment, we have taken the standard Term Frequency * Inverse Document Frequency (TFIDF) with topic related information of query and document and results were compared against the standard TFIDF results. During the experiments, we have initially considered the most matching topic of the query to retrieve document of the same topic. As some of the queries belong to more than one topic, we have made variation to the retrieval method by including the topic of the next most matching query topic for associated documents retrieval. Then all the retrieved documents were taken into account and based on their cosine similarity score, they were ranked and the ranked list of documents is obtained. These top matching documents are evaluated based on the guidelines used in the evaluations of Text REtrieval Conference (TREC) with the help of relevant judgments.

2 Forum for Information Retrieval and Evaluation

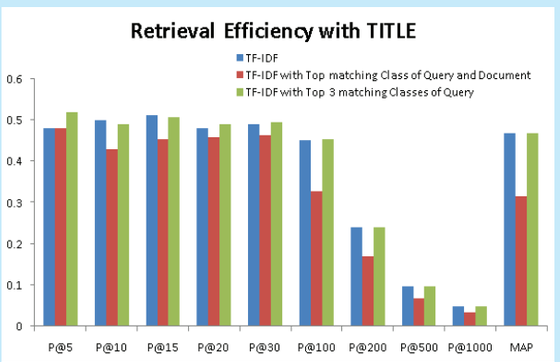

Fig. 5. Retrieval efficiency with TITLE field of the query

Figure.5 shows the retrieval efficiency of the documents with TITLE field of the query. We have shown the Precision @ top d (= 5, 10, 15, 20, 30, 100, 200, 500, 1000) documents and its corresponding Mean Average Precision(MAP). From the observed results, it is clear that when we have considered the most matching topic of

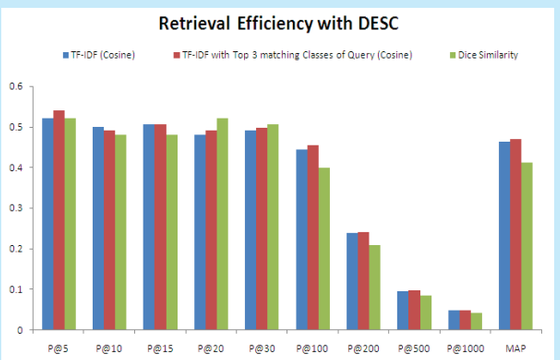

Fig. 6. Retrieval efficiency with DESC field of the query

the query to retrieve the documents of the same topic the precision is just below the standard TFIDF measure but when we also considered second most matching class for the input query, the precision is improved a little overthe standard TFIDF but the improvement is not significant. Figure. 6 shows the retrieval efficiency of the documents with DESC field of the query. We have shown the Precision @ top d (= 5, 10, 15, 20, 30, 100, 200, 500, 1000) documents and its corresponding Mean Average Precision(MAP).

VII. CONCLUSION

The problem of document retrieval with special focus on tourism domain is considered. At first, we have proposed three methods: keyword based, pattern matching based and Naive Bayes based approaches for query classification. Using one of these approaches, the topic of the given query is identified. Then we have used Naive Bayes and k−NN methods for building the document classifier. The content of web documents are filtered and their corresponding topic is identified. Finally we have proposed methods for document retrieval using standard TFIDF with and without topic identification tasks. The accuracies of the is not sufficient. In future, we plan to apply other similarity measures to document retrieval with class information.

REFERENCES

[1] Murata, M., Ma, Q., Uchimoto, K., Ozaku, H., Utiyama, M., Isahara, H.: Japanese probabilistic information retrieval using location and category information. In: IRAL '00: Proceedings of the fifth international workshop on on Information retrieval with Asian languages, New York, NY, USA, ACM (2000) 81–88

[2] Kang, I.H., Kim, G.: Query type classification for web document retrieval. In: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval. SIGIR '03, New York, NY, USA, ACM (2003) 64–71

[3] Beitzel, S.M., Jensen, E.C., Frieder, O., Grossman, D., Lewis, D.D., Chowdhury, A., Kolcz, A.: Automatic web query

classification using labeled and unlabeled training data. In: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval. SIGIR '05, New York, NY, USA, ACM (2005) 581–582

[4] Fonseca, B.M., Golgher, P., Pˆossas, B., Ribeiro-Neto, B., Ziviani, N.: Concept-based interactive query expansion. In: Proceedings of the 14th ACM international conference on Information and knowledge management. CIKM '05, New York, NY, USA, ACM (2005) 696–703

[5] Shen, D., Sun, J.T., Yang, Q., Chen, Z.: Building bridges for web query classification. In: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval. SIGIR '06, New York, NY, USA, ACM (2006) 131–138

[6] Broder, A.Z., Fontoura, M., Gabrilovich, E., Joshi, A., Josifovski, V., Zhang, T.: Robust classification of rare queries using web knowledge. In: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval. SIGIR '07, New York, NY, USA, ACM (2007) 231–238

[7] Cao, H., Jiang, D., Pei, J., He, Q., Liao, Z., Chen, E., Li, H.: Contextaware query suggestion by mining click-through and session data. In: Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. KDD '08, New York, NY, USA, ACM (2008) 875–883

[8] Cao, H., Jiang, D., Pei, J., Chen, E., Li, H.: Towards context-aware search by learning a very large variable length hidden markov model from search logs. In: Proceedings of the 18th international conference on World wide web. WWW '09, New York, NY, USA, ACM (2009) 191–200

[9] He, D., Wu, D.: Enhancing query translation with relevance feedback in translingual information retrieval. Inf. Process. Manage. 47 (January 2011) 1–17

[10] Pirkola, A., Puolam¨aki, D., J¨arvelin, K.: Applying query structuring in cross-language retrieval. Inf. Process. Manage. 39 (May 2003) 391–402

[11] Cui, H., Wen, J.R., Nie, J.Y., Ma, W.Y.: Probabilistic query expansion using query logs. In: Proceedings of the 11th international conference on World Wide Web. WWW '02, New York, NY, USA, ACM (2002) 325–332

[12] Li, X., Wang, Y.Y., Acero, A.: Learning query intent from regularized click graphs. In: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval. SIGIR '08, New York, NY, USA, ACM (2008) 339–346

[13] Salton, G., Wong, A., Yang, A.C.S.: A vector space model for automatic indexing. Communications of the ACM 18 (1975) 229–237

[14] Sebastiani: Machine learning in automated text categorization. ACM Computing Surveys 34 (2002) 1–47

[15] Leopold, E., Kindermann, J.: Text categorization with support vector machines. how to represent texts in input space? Mach. Learn. 46 (March 2002) 423–444

[16] Zhang, L., Zhang, D., Simoff, S.J., Debenham, J.: Weighted kernel model for text categorization. In: Proceedings of the fifth Australasian conference on Data mining and analystics - Volume 61. AusDM '06, Darlinghurst, Australia, Australia, Australian Computer Society, Inc. (2006) 111–114

[17] Cohen, W.W., Singer, Y.: Context-sensitive learning methods for text categorization. In: Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval. SIGIR '96, New York, NY, USA, ACM (1996) 307–315

[18] Joachims, T.: Learning to Classify Text using Support Vector Machines. Kluwer (2002)

[19] Yang, Y., Pedersen, J.O.: A comparative study on feature selection in text categorization. In: Proceedings of the Fourteenth International Conference on Machine Learning. ICML '97, San Francisco, CA, USA, Morgan Kaufmann Publishers Inc. (1997) 412–420

[20] Ng, H.T., Goh, W.B., Low, K.L.: Feature selection, perceptron learning, and a usability case study for text categorization. In: Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval. SIGIR '97, New York, NY, USA ACM (1997) 67–73

[21] Yang, Y., Liu, X.: A re-examination of text categorization methods. In: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval. SIGIR '99, New York, NY, USA, ACM (1999) 42–49

[22] Saraccouglu, R., Tutuncu, K., Allahverdi, N.: A new approach on search for similar documents with multiple categories using fuzzy clustering. Expert Syst. Appl. 34 (May 2008) 2545–2554