Assisting Web Documents Retrieval with Topic Identification in Tourism Domain

R.Rajendra Prasath, Vijai Kumar and Sudeshna Sarkar
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur - 721 302, India
Email: frajendra, sudeshnag@cse.iitkgp.ernet.in; viz.kec2009@gmail.com

***Abstract:-** We present an Information Retrieval (IR) system for retrieving the documents having the specific types of information as that of the user query pertaining to the tourism domain. By user study, we observed that the user information needs in the tourism domain span over a few major types. Based on this observation, we have identified the major types (topics) in the tourism domain and built the tourism specific ontology. Then, we have developed (i) a document classifier to identify the topic of the web documents and (ii) a query classifier to identify the topic of the user query, both pertaining to the tourism domain. Finally the proposed system performs the documents retrieval by matching the type of user query with the matching type of documents. The experimental results show that the tourism specific topic identification of queries and documents actually improves the retrieval of documents having more specific information to satisfy user queries in the tourism domain.*

Keywords: *Topic Identification, Query Classifier, Document Classifier, Tourism Specific Retrieval, Retrieval Efficiency*

I. INTRODUCTION

Information Retrieval (IR) Systems accept a sequence of words as a query, perform the retrieval task on the specified collection of documents and retrieve a ranked list of documents as search results. Each retrieved document matches with the information needs specified by the user query only if the user perceives it to be relevant and useful. Since queries are the partial specifications of user information needs; often short having 2-3 keywords on an average [1] and ambiguous, IR systems have to identify and fill the gaps in such queries. This reduces to the effectiveness of the IR systems [2]. A system is considered to be effective, if a large proportion of top k (k > 0) retrieved documents are relevant. Because of underspecified or ambiguous queries, the retrieved top k documents may have information irrelevant to the user needs. Mere the presence of query terms in documents may not ensure that the retrieved top k documents carry the information pertaining to the user query. Additionally, ranking of retrieved documents with more specific information to the user query becomes really challenging in IR systems.

In this paper, we focus on the retrieval of documents with specific information needs related to the tourism domain. User queries in the tourism domain often contains the places of interest. For example, consider the query, “cheap comfortable way to Ooty”. The user intention in this query is to find: “how to reach Ooty (comfortably by bus / train with cheap or economical fares)?” type of information. Documents, that contain the actual information on different modes of transportation like “buses”, “trains” with the cheapest or economical fares to Ooty, may not contain the actual query terms. Hence the query type needs to be identified so as to retrieve documents having the terms “bus”, “train” pertaining to “how to reach” type of information specific to the place “Ooty”.

The ambiguous query: “bus services in Java” submitted to Google1 to know about the transportation, especially the information on bus facilities in Java Island, fetched only two documents (one is fully relevant and another is partially irrelevant) pertaining to the query topic: “how to reach” and the rest are specific to “JAVA programming language”. In this query, “bus services” pertaining to the topic “how to reach” needs to be identified and given more importance to “buses”, “travel”, “how to reach” during the documents retrieval.

Consider the query: “staying in alwar city”. This query seeks information related to hotels / guest houses / hostels in Alwar city and hence falls under the topic “accommodation”. Hence more importance has to be given to the pages having accommodation related terms “hotels”, “guest houses”, “hostels” of Alwar city rather than considering the importance of each of the direct query terms.

Let us consider a few more examples: the query, “kanchipuram varadharaja perumal temple”, seeks the details about a particular topic: “places to visit”, in turn “attractions”, in the specific city of “kanchipuram”. Hence the query terms “varadharaja perumal temple” pertaining to the specific topic, “attractions” in the city of “Kanchipuram” needs to be given more importance.

Lastly, consider the user query, “climbing nanda devi”. Documents that contain the query term, “devi”, may refer to the temple related information. But the query term “climbing” seeks the information specific to the topic: “how to reach” the place called “nanda devi”. So retrieving the documents having the matching type information on “how to reach” is more important than the details of a particular “attraction”.

In each of these queries, the specific focus of the user information needs has to be identified with the right query term(s) pertaining to a specific topic and then retrieval has to be performed with the documents having the matching topic of the query. Since user information needs focus on a specific aspect of tourism related information either like places or services or hospitality, the supplied query terms are alone not sufficient to capture the underlying information needs and hence additional information has to be identified and incorporated with user queries.

In this paper, we present a system to search for the tourism specific documents pertaining to the tourist places in India.

II. REVIEW OF LITERATURE

Since web queries are very short, consisting of 2-3 words on an average [3] and ambiguous [4], a query may belong to multiple topics. Goker [5] described a machine learning approach to infer the actual context behind user queries in an incremental way. This approach tried to learn the “problem situations” that represent the context of the query and the context learner helps to ‘learn’ from one query to another incrementally. Similar mechanisms were exploited the interest and context of users in various topics and have the potential to improve Web retrieval systems [6], [7]. In 2003, Kang and Kim [8] showed that category information can be used to trigger the most appropriate vertical searches corresponding to a query, to improve topic relevance tasks (informational) and home page finding tasks(navigational). They used ‘and’ and ‘sum’ operators for matching query terms. In the case, ‘and operator means that the result document has all query terms in it. ‘sum operator means that a result document has at least one query term in it. Ozmutlu and Cavdur [9] investigated the properties of a specific topic identification methodology with Excite Web Search Engine data logs. In this method, the parameters (term weights and a threshold) for the topic identification algorithm are determined using topic shift and continuation probabilities. Carmel et al. [10] presented a model that captures the main components of atopic and the relationship between those components and topic difficulty. Again in 2008, Ozmutlu et al. [11] proposed a topic identification algorithm without considering the context of queries, but rather by using the statistical characteristics of the transaction log queries. Web Query Classification(QC) has been studied for its wide usage in domain specific web search, personalized IR, online advertisement and so on and these methods follow the bag of words approach [12], [13], [14]. Lin and Chao [15] presented an algorithm that retrieves tourism related opinions which are then used to determine tourist attractions, that is, given an opinionated sentence, the algorithm determines whether it is tourism-related or not, and then decides which tourist attraction is the focus of the given opinion.

Hull [16] proposed a query structuring method. The basic idea of query structuring is to group query keys and to use query operators in such a way that more weight is being assigned to important or correct keys than the other keys. In this method, query input format consists of a series of attributevalue pairs, each is considered as a concept and all terms, entered in that specific attribute-value pair, are combined using OR operator. The user may designate the importance of each concept and these concepts are combined using a weighted AND operator. Pirkola [17] showed that applying the query structuring on cross lingual information retrieval with translation equivalents of query terms with n-grams captures the clue on the intention of users's information need. Recently, D0hondt et al. [18] proposed a technique to automatically identify the topics present in a document, based on the presence of lexical chains. Xiang et al. [19] conducted a study focusing on understanding the representation of tourism related information through current search technologies on the Internet by analysing 1) the size and visibility of the tourism specific information provided by Google and, 2) the representation of tourism specific unique websites on different -pages of search results.

III. OBJECTIVES

Our objective is to develop a domain specific IR system for tourism domain especially to retrieve documents pertaining to the tourist places in India. With the rapid expansion of the web, the volume of documents in the tourism domain grows as well. Since all these documents in the tourism domain may not have the useful information as requested by the users unless each of them is manually read and verified. Based on the study of user information needs in the tourism domain, we observed that the users search for the most specific information centered around the places of their interest. With such growing information demands of the users in the tourism domain, identifying the types of information in the web documents and categorizing them under major tourism specific topics may assist the retrieval of documents with useful information pertaining to the user query. Hence our goal is to develop an IR system that first identifies the type of user queries; the types of information in the documents and then performs retrieval of documents having topic specific information pertaining to the type of the user query.

IV. THE OVERVIEW OF THE PROPOSED IR SYSTEM

In this section, we describe the overview of the proposed tourism specific document retrieval system. The proposed system consists of the following components: the tourism ontology, document topic classifier, query topic classifier and the Proposed IR system. The proposed IR system, in turn, consists of the components: Content extraction with topic identification, Indexer and Searcher. The work flow of the proposed system, represented in Figure. 1, is divided into the following phases:

1) Initial Phase: During this phase, tourism ontology is built by identifying the set of query topics and the set of document topics, both pertaining to the tourism domain. Using this tourism ontology and tourism queries collected from the web users, query topic classifier is built. This query classifier can be used to identify the topic of the user query in the tourism domain. Similarly, using the text documents tagged with tourism related topics, the document topic classifier model is built using Naive Bayes approach. This model can be used to identify the topic(s) of a text document pertaining to the tourism domain.

2) Offline Process: The crawler obtains web documents; Each web document in this

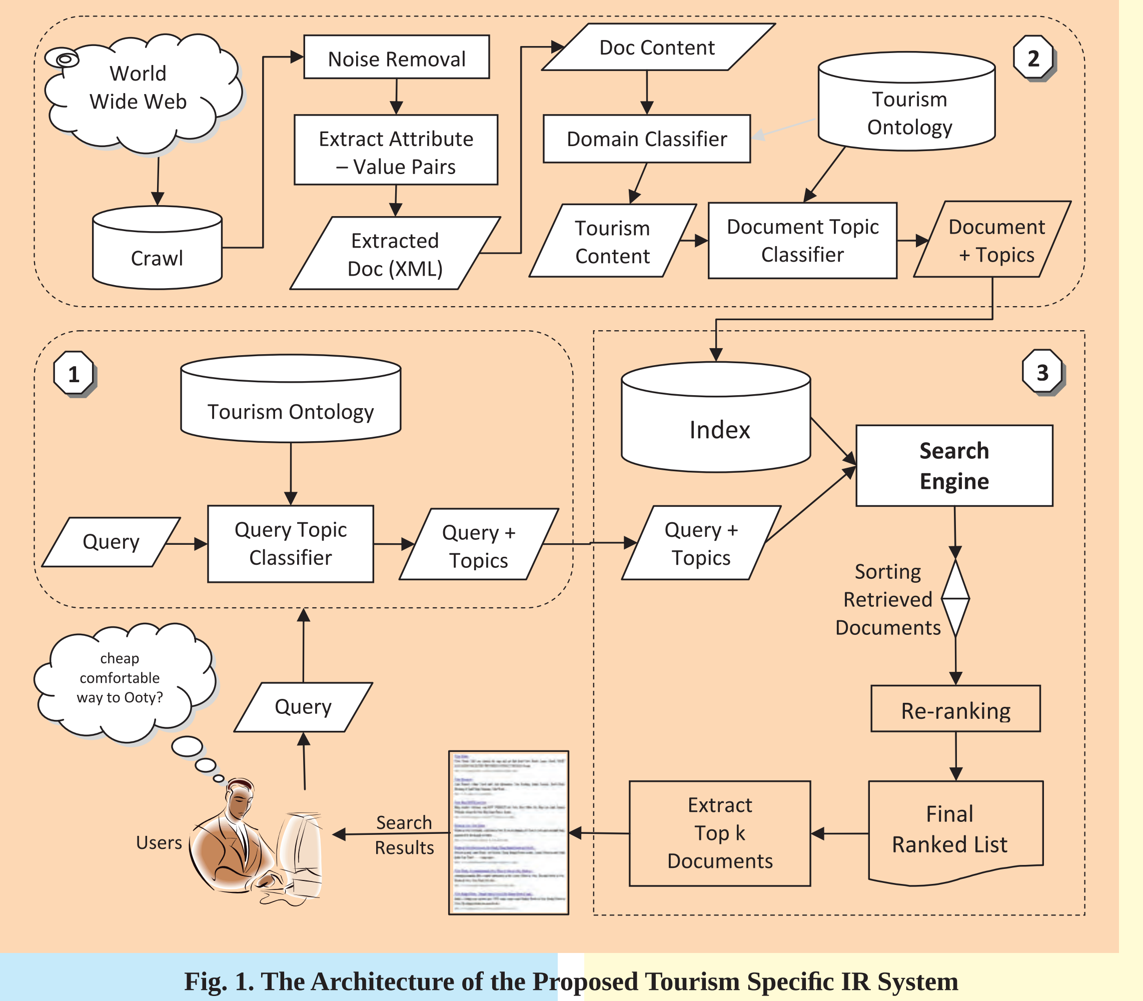

Fig. 1. The Architecture of the Proposed Tourism Specific IR System

crawled data may have noisy contents like advertisements, banners, forms and so on. Noise removal heurisitc is applied on these documents and the values of various fields like title, meta tags, description, byperlinks, the extracted text content, etc are filtered out in the form of attribute-value pairs. Additionally, tourism specific domain classifier checks the extracted text content; filters out the non-tourism text contents and allows the tourism related text content in the pipeline. This text content is splitted into coherent segments. Then document topic classifier is applied on each of these text segments and the topic of the each segment is identified. The identified topic information along with other attribute-value pairs is stored in an XML format.

Then the proposed IR system suitably performs the indexing of the documents represented in XML formats and stores the index into the disk.

3) Online Processes: When the proposed IR system receives the user query, it invokes the query topic classifier which in turn identifies and returns the topic(s) of the query. This topic information along with the supplied query terms is used by the searcher to retrive the subset of documents (from the index) pertaining to the user query topic. Then re-ranking is applied on the retrieved set of documents with the matching topics associated boost factors. This finally produces the ranked list of documents from which top k documents pertaining to the query topic(s) are returned to the user as search results.

Next we describe each of these components in details:

A. Tourism Ontology

Based on the analysis of tourism related web pages, we have identified a list of topics in the tourism domain. Frequently used topics (- with their types of information) in the majority of the tourism specific webpages are:

1) Attractions - monuments and places to see in and around the specific tourism spot
2) Activities - events and festivals related information
3) Accommodations - “where to stay” and details about the hotels / guest houses / hostels / dormitories /etc
4) Climate - weather related information pertaining to the specific place
5) Food - where to eat / restaurants, cafeteria, motels related information and specific food items that region
6) Ideal Time - the best time to [or not to] visit the specific place
7) History - importance of the place in ancient days, the details about the rulers of that region
8) Travel - “how to reach” and “local transportation”related information
9) Tour Packages - tour operators details, guide services and sightseeing plan related information and
10) Others - travel guides / maps and other informations not pertaining to any of the above topics.

These tourism specific topics were used in the following subsections to identify the topics of the documents and topics of the query.

B. Document Topic Classifier

The Document topic classifier is a subsystem that takes the extracted content of the web document pertaining to the tourism domain; identifies the tourism specific topic(s) of the extracted tourism specific web document content and then tags the extracted content with one or more identified topic(s). The document topic classifier pertaining to the tourism domain can be built using either a simple bag of words approach [20], [21], [13], [14] or decision rules [22] or probability estimations on the manually tagged tourism documents collection.

We have considered a collection of 458 tourism related web documents. The topic of each document in this collection is manually identified and labelled that document with its topic. Using this as the training data, we have built a document topic classification model using Naive Bayes method. This model, given the tourism related text content, identifies the most matching topic of that content with its matching score - score(d, t) - the score assigned by the classifier to the document d pertaining to the topic. This identified topic with *score (d, t)* is stored in the index during the offline process. Later the searcher uses this information to retrieve the matching type of documents pertaining to the user query topic.

The offline processing module associated with the document topic classifier is pictorially represented by the component labelled as (2) in Figure. 1.

C. Query Topic Classifier

In this section, we describe the query topic classifier and its role in identifying the topics of user queries in the tourism domain. The query topic classifier is a subsystem that will take a sequence of terms, describing the user information needs in the tourism domain, as an input; identify one or more topics associated information pertaining to the user information needs and then enrich the query with the identified topic(s). The tourism specific query topic classifier can be built using the domain knowledge specific to the tourism domain. In the tourism domain, the query terms are information centric around either popular attractions or a cheap hotel or a famous monument or a voyage or a similar entity, pertaining to that specific place. The query topic classifier, on receiving such user queries, uses the tourism ontology to identify the matching topic(s) of interest and expands the given query with the identified topic associated information. The online processing module associated with the query topic classifier is pictorially represented by the component labelled as (1) in Figure. 1.

The query topic classifier identifies the topic(s) of the tourism specific query either by matching:

- lists of topic related keywords with query terms, or
- user query patterns with the sequence of query terms, or
- using the tourism specific classification model developed by machine learning approaches.

Now let us describe each of these ways in detail.

1) Matching Query Terms with Topic Related Keywords: We have identified the lists of keywords pertaining to each of the tourism specific topics. The user query having k terms is compared with each of the lists having the keywords specific to the one topic in the tourism domain. The topic of the list having the maximum terms matching with the query terms is assigned as the identified topic to the given query. Basically this is a “Bag of words” approach. For example, if a query is related to the transportation seeking “how to reach a place by air / train / bus?”, then the list having the keywords - route, reach, nearest, flights, airport, best, driving, road, bus, station, train, railways, from, to, till, upto - would be identified as the list with most matching keywords and hence the topic of this list: “travel” is assigned to the query as its topic. Similarly, for a food related query, the list having the keywords - restaurants, motels, mess, eat, food, vegetarian, non-vegetarian, dining, dinner, lunch, breakfast, taste, snacks, coffee, tea, cakes, juices - would be identified as the primary list with more matching keywords with the query terms. So “food” is assigned to that query as its topic. The matching ascore, between query terms and topic assocated keywords, is calculated as follows:

$$\text{Score} = \frac{\text{\#of match between query terms and keywords}}{\text{\#of query terms}}$$

Using this, we will find the score for each query topic. Finally the topic of the query is determined by the topic of the list having the maximum number of matching keywords.

2) Matching User Queries with Query Patterns: Since user queries in the tourism domain follow specific patterns centered around place names, regular expressions can be used to identify patterns pertaining to each of the tourism specific topics. For example, consider the query patterns: “how to reach from ?”, “attractive (aroundjnear) ”, “best to visit ”, “cheap to stay in ” and so on. These query patterns have the frames to fill in up with query terms pertaining to the places of interest. Hence we rearrange the query terms in a meaningful way so as to find the matching query pattern pertaining to the topic of the user query. The topic of thepattern, whose frames are well filled in by the query terms, is assigned to the user query as its topic. However a query may be expressed in many ways all leading to same information needs. Additionally, each user may also use different set of query terms for searching the same information. So the query patterns used here may not be able to include all possible variations of the user query patterns. However, the matching score, between query patterns associated with the topics and the user query terms, is calculated as follows:

$$\text{Score} = \frac{\text{\#of frames filled}-\text{in with matching query terms}}{\text{\#of query terms}}$$

We list below a few regular expressions for the topics: *Climate*, *Ideal time* and *Accommodation*

⊘ **Climate**

a) (climate|weather|rainfall)* (of|in) (<place>)
b) (climate | weather | temperature) (of|in)* (<place>) (during|in)* (autumn|spring | summer | winter)
c) [cloudy|monsoon] (seasons|time)* (of|in) (<place>)
d) [average](temparature)* [var(ies|ying) | hover(s|ing)]* [from] (<degree_celsius>) [to] (<degree_celsius>) (during)* (autumn | spring | summer | winter)
e) (<place>)(weather|monsoon)(forecast|ing)* [during|in] <month|season>

⊘ **Ideal time**

a) (time | season)*(to | for)* (visit | (tour | stay(| ing))) [in] (<place>)
b) [Best | good] (season) to (see | visit | tour) (< place>) (from) (<Month | dates>) (to)* (<Month|dates>)
c) (ideal | pleasant)* (time | season)* for (sight seeing | outings | tourist activities) [in | around] (<place>)
d) ([perfect | best] time)*) to (feel | enjoy)* (chilly | warm | rainy)* [climate] for (honeymooners | trekkers | family tours)* in (<place>)
e) which is (|not)* the (best|peak|festival)* (time | season) to (view | see | visit) (<place>) from (<place>)

⊘ **Accommodation**

a) [luxury | cheap | best] (< Hotel(|s) | Guest houses | hostel(| s)>) to (stay | lodge) in (<place(|s)>)
b) (accomodation(| s) | hotel) available near (<place>) during (<month | season>)
c) (room(|s) | dormitory) for (<count>) (person|day)(|s)
d) (lodge | resort | dharamshala | hotel | accomodation)(|s) near (<place>)
e) [star | budget | cheap] (hotel(|s))(to | for) stay(|ing) in (<place>) near (<place>)

These regular expressions are used to find the pattern of the underlying query terms whose topic is decided by the most matching query pattern. For example, the query - **weather** in **goa** during **may** - belongs to the topic: “Climate”; another query - mighty forts of chhatrapati **shivaji** in **maharastra** - belongs to the topics: “Attractions” and / or “History”.

3) Naive Bayes: We have manually identified the list of web queries, tagged each of them with its specific topic related to the tourism domain. This tagged set of queries is used as the training data and the the query topic classifier model is built using Naive Bayes method with 10-fold cross validation. This classification model is then used to predict the topic of the new user queries. The output will be the query tagged with the tourism specific topic(s) having the highest matching score(s).

Using any one of the above ways, the query topic classifier is built. Then this classifier is added in the proposed architecture as a subsystem and used to identify the topic(s) of the new user queries during an online process.

D. The Proposed IR system

The proposed IR system consists of the components: Content extraction with topic identification, Indexer and Searcher. From the web documents, textual descriptions are extracted by eliminating noisy contents. Then this textual content is split into segments and topic of each segment is identified. Indexer is used to store the identified topics in the index. Then the searcher, on receiving a user query, performs the retrieval of documents by matching the query and document topics. Now let us look into the details of eachr component:

1) Content extraction with topic identification:

Web documents contains noisy contents like advertisements, banners, forms, etc amidst of the textual descriptions. Many web documents use different layouts with HTML markups and their own style sheets. Most of the web pages are semistructured or ill-structured. So at first, we create a tree like document structure using the existing marksup of a web document. By traversing through the node levels in the well formed document structure, advertisements, images, banners, forms, etc are identified by their tags and filtered out. Among rest of the nodes, some may contain only navigational links and we filter out all such nodes.

Next the document structure is suitably split into different segments using div class tags and table division tags as the nodes. Then each node content is validated with the heurisitc: link-to-text ratio, which is defined as the ratio between the size of the text tagged with and without hyperlinks. The text segments which exceed this threshold will be extracted by starting at div or table level nodes. The filtered content is converted into the unicode, if not already. The filtered text segments are then sent to the document topic classifier.

The filtered text segments may have one or more overlapping topics pertaining to the tourism ontology. The document topic classifier built in section IV-B is invoked on each extracted segment and its topic is identified. The identified topics of the extracted web content is stored in the XML structure and indexed subsequently as on offline process using the indexing system powered by Lucene[2].

2) Indexer:

The parsed contents of the web documents are taken from XML structure which stores information in the form of attribute-value pairs. These attribute-value pairs are then indexed for faster retrieval. This process is done in offline.

3) Searcher:

The searcher is the core part of the proposed document retrieval system. It uses the topic information of the query and documents to assist the retrieval task in the tourism domain. At first, on receiving the user query, we invoke the query topic classifier and get the reformulated query - the query with the identified topic(s). Then using this reformulated query, the initial

[2] Lucene available at: http://www.apache.org/dist//lucene/java/

list of documents is retrieved by computing doc-score(q;d) - the cosine similarity score between q and d; t-he *doc-score* of the retrieved document, whose topic matches with the topic of the query, is updated with the document weighting factor w; and then the final list is generated by the updated *doc-score*. The document weighting factor w, is computed as the product of the preference factor and *score(d; t)* over the top k topics taken into account, where the preference factor is the parameter to support the documents with more / less scores independent of the topic information (here we assume = 0:85); score (d; t) is the classifier score for the topic t of the retrieved document and score(d; t) 2 [0; 1]. Thus the system retrieves the initial list of documents whose similarity scores are further enhanced with the topic(s) associated document weighting factor. Then the final ranked list of top k documents pertaining to the topic of the user query is presented as the search results.

The online processing module associated with the searcher part is pictorially represented by the component labelled as (3) in Figure. 1. The proposed document retrieval procedure is given in Algorithm IV-D3.

```
Algorithm 1 The Searcher of the Proposed IR system
Input:
A query having $k$ terms: $Q = \{t_1, t_2, t_3, \cdots, t_k\}$, $k > 0$
A query topic classifier
Index - Documents indexed with their topics
Description:
On receiving a user query, invoke the query topic classifier and identify the most matching topic(s) (with its matching score) of that query
Reformulate the user query with the identified topic(s) and feed to the search engine
Search engine retrieves the initial set of documents each with its $doc\_score$
Apply re-ranking of the retrieved documents as follows: consider each document $d$ in the retrieved list; If the document topic(s) matches with the query topic(s), then obtain the document weighting factor $w$ using:
    $w = \sum_t \alpha * score(d, t)$
    where $\alpha$ is the weighting parameter; $score(d, t)$ - score of the document $d$ belonging to the topic $t$. Then combine $w$ with the document score to get the updated $doc\_score$:
    $doc\_score(q, d) = doc\_score(q, d) + w$
Generate the final list of documents based on the updated $doc\_score$ scores sorted in the decreasing order
return top $k$ documents as search results
Output: The ranked list of top $k$ retrieved documents
```

The effectiveness of the proposed topic assisted document retrieval method is given in the next section.

V. EXPERIMENTAL RESULTS

A. Dataset and Queries

We have collected 288 queries in the tourism domain through different sources like Yahoo! questions, travel blogs, web forums and web users. The collected queries belong to specific tourism related topics and additionally we have included most of the possible variations to each topic. Using these queries, we have built a query topic classifier. Then we have collected 458 tourism related text documents each tagged with one or more identified topics which were listed in IV-A. This collection has been used to build the document topic classifier. Finally for the evaluation of the proposed document retrieval system, we have, at present, used a corpus of 20,482 documents related to tourism domain and used the stemmed queries for evaluation. Since the document classification at the extracted segments level introduces additional constraints with time and space, we have limited our experiment with this documents collection.

[3] Weka APLs available at: http://www.waikato.ac.nz/ml/weka/

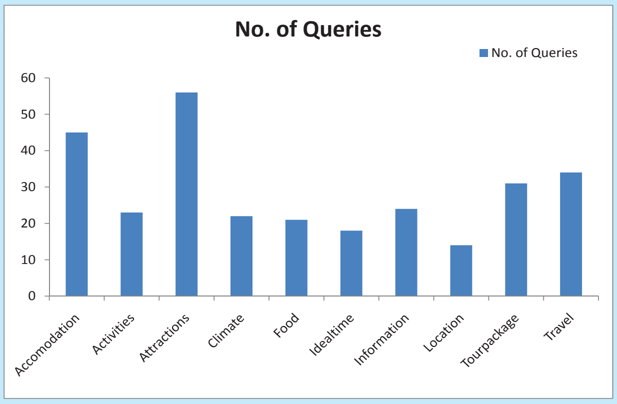

Fig. 2. Statistics on the number of Queries

First we present the experimental results obtained for query classification. Figure. 2 shows the number of queries manually collected for 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 (how to reach).

	Topic Keywords	Pattern Matching	Naive Bayes
Accuracy	0.46	0.53	0.65

TABLE I
EFFECTS OF QUERY TOPIC CLASSIFICATION PROCEDURES

Table. I shows the classification accuracy of various query topic classifiers. Naive Bayes classifier performs better on an average as it has been applied to the text documents collection tagged with tourism specific topics. Naives Bayes method assumes the documents as the Bag of Words(BoW) in which the terms are assumed to be independent of each other. Figure. 3 shows topic wise query classification accuracy.

1) Document Classification:

This section carries the experimental results of document classification using Naive Bayes Approach. We have used Weka[3] APIs (Application Programming Interfaces) to build the Naive Bayes classifier on the tagged collection of 458 tourism related web documents. The document topic classification model is built once with this training data and subsequently used to classify the newly extracted web contents with their respective topics. We list the distribution of documents in the tourism domain (# of documents in each topic): Accomodation (53), Activities (46), Attractions (50), Climate (41), Food (46), Idealtime (36), History (70), Location (37), Tour Package (39), Travel (40). Figure. 4 shows the accuracy of the documents classifier with Naive Bayes on tourism data.

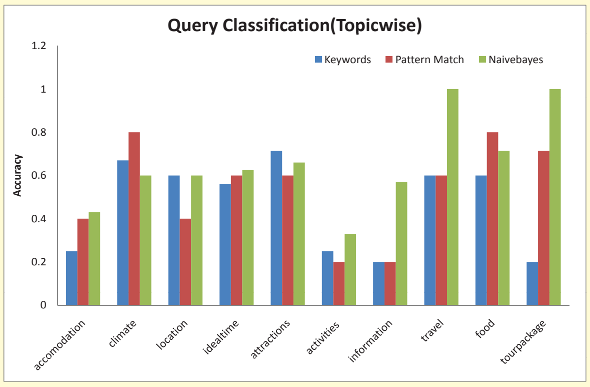

Fig. 3. Query Classification Accuracy (Topicwise)

In order to evaluate the proposed document retrieval method, we have conducted the experiments in a similar way like TREC4 evaluations. We have considered the actual user queries having 10 types of user information needs in the tourism domain and experimented the proposed document retrieval method with these 10 queries.

We have used the Vector Space Model [23] supported in Lucene as the base line (without

[3] Weka APIs available at: *http://www.cs.waikato.ac.nz/ml/weka/*

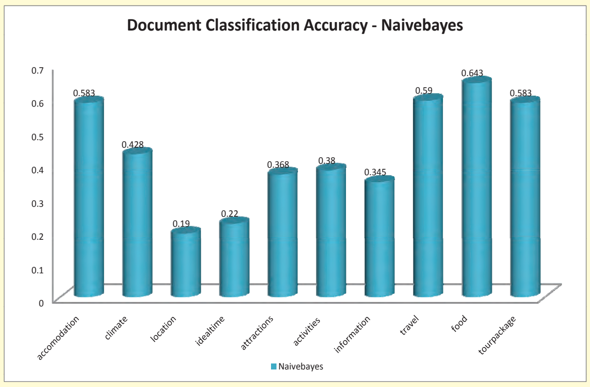

Fig. 4. Classification accuracy [topicwise] with Naive Bayes Algorithm

any information on the topic). In this base line experiment, 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. Documents with highest similarity scores are obtained and evaluated based on how many relevant documents are present among the retrieved top d documents.

B. Documents Retrieval Assisted by Topics

We have taken the VSM with information on topic(s) of query and documents and results were compared against the standard VSM results. We have initially considered the most matching topic of the query to retrieve documents pertaining to 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 topics of the next most matching query topic for assisting the documents retrieval. Then all retrieved documents were taken into account and based on their cosine similarity score, the ranked list of documents is presented to the user. The top 5, 10, 15 and 20 documents were evaluated manually for each of 10 queries and computed the retrieval scores in terms of precision @ top d documents.

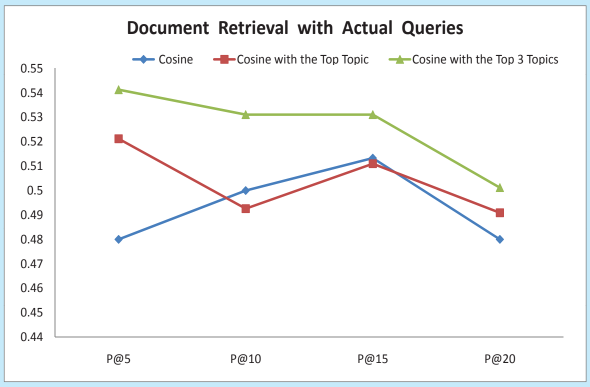

Fig. 5. Retrieval Scores with 10 queries

Figure.5 shows the retrieval scores of the documents with 10 queries pertaining to the tourism domain. We have shown the Precision @ top d (= 5, 10, 15, 20) documents.

Consider the query: “cheap comfort way to ooty”. The query topic classifier identifies the topics (the topics are given in the decreasing order of their topic scores): Accommodation; tour packages; travel; food, attractions . The terms “cheap” and “comfort” are mostly used to describe “budget hotels” / “cheap / economical hotels” / “cheap guest houses” in tourism related web pages. Since the query topic classifier identifies the topic “how to reach” with low topic score, the documents having travel related information are not boosted sufficiently. With one topic, more documents pertaining to the topic accommodation are boosted to bring them in top 5 places. The precision goes down below the base line method. This may be due to two facts: the query topic classifier accuracy which directly influences the boosting of topic associated information and the topic score which is multiplied with the document score and directly proportional to the final documents scores. Instead of using one topic, we have considered top 3 topics and analysed the retrieved documents. Each of the topic score is used to boost the document score and then combined to get the final document score.

[4] Text REtrieval Conference @ http://trec.nist.gov/

The top 10 retrieved results were analyzed based on this. Among these 10 results, 5 documents contain how to reach related information, 1 document contains tour packages related information and rest of them fall in accommodations and attractions. In overall observations, we have noticed that p@5 for the actual queries gives 14.1458% improvement over the base line retrieval. However for p@10, recall is achieved at the loss of precision which has come down to 9.62% improvement over the standard baseline. Since the accuracy of the query topic identification is marginal, we plan work on improving the retrieval of documents by improving the performance of the query topic classifier in future. Additionally, we would like to perform the scalable experiments with larger dataset of TREC and FIRE.

VI. CONCLUSION

We presented a system for documents retrieval pertaining to the tourist places in India. We make use of tourism ontology to build the proposed system. The proposed system consists of the document topic classifier which identifies the topics of the given text segments and the query topic classifier to identify the topic of the user query, both pertaining to the tourism domain. Then the tourism specific retrieval engine performs the documents retrieval by matching the type of user query with the matching type of documents in the index. The experimental results show that the topic identification helps the retrieval of relevant documents at the top of the ranked list. Subsequently, we plan to measure the effectiveness of the proposed method on a larger collection of multilingual web documents pertaining to the tourism domain.

REFERENCES

[1] Singhal, A., Kaszkiel, M.: A case study in web search using trec algorithms. In: Proceedings of the 10th international conference on World Wide Web. WWW ’01, New York, NY, USA, ACM (2001) 708– 716

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

[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] 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

[5] Goker, A.: Context learning in okapi. Journal of Documentation 53 (1997) 80–83

[6] Talja, S., Keso, H., Pietilainen, T.: The production of context in information seeking research: a metatheoretical view. Inf. Process. Manage. 35 (November 1999) 751–763

[7] Spink, A., Wolfram, D., Jansen, M.B.J., Saracevic, T.: Searching the web: the public and their queries. J. Am. Soc. Inf. Sci. Technol. 52 (February 2001) 226–234

[8] 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

[9] Ozmutlu, H.C., C¸ avdur, F.: Application of automatic topic identification on excite web search engine data logs. Inf. Process. Manage. 41 (September 2005) 1243–1262

[10] Carmel, D., Yom-Tov, E., Darlow, A., Pelleg, D.: What makes a query difficult? 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) 390–397

[11] Ozmutlu, H.C., Cavdur, F., Ozmutlu, S.: Cross-validation of neural network applications for automatic new topic identification. J. Am. Soc. Inf. Sci. Technol. 59 (February 2008) 339–362

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

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

[14] 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

[15] Lin, C.J., Chao, P.H. In: Tourism-Related Opinion Mining. (2010) 3–16 [16] Hull, D.A.: Using structured queries for disambiguation in crosslanguage information retrieval. in Working notes of AAAI spring symposium on cross-language text and speech retrieval (1997) 84–98

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

[18] D’hondt, J., Verhaegen, P.A., Vertommen, J., Cattrysse, D., Duflou, J.R.: Topic identification based on document coherence and spectral analysis. Information Sciences 181(18) (2011) 3783–3797

[19] Xiang, Z., Wober, K., Fesenmaier, D.R.: Representation of the Online Tourism Domain in Search Engines. Journal of Travel Research 47(2) (November 2008) 137–150

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

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

[22] 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

[23] Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Commun. ACM 18 (November 1975) 613–620