Text based Semantic information predictions using user behavior

Sonali Pawar
Department of Computer Science and Engineering, YCCE College, Nagpur

Abstract
For Searching and managing online growth of information is becoming a difficult task. The major challenge is to improve users search experience. The current technique that is involved in Content description and query processing in Information Retrieval (IR) are based on keywords. I am therefore trying to improve the quality of search results. In this paper I am trying to optimize the search engines results. Mostly used search engines are Google, Yahoo and Bing. Thus the query q is provided as an input to search engine followed by retrieving relevant d-documents/links to user. Depending upon the user behavior the documents are retrieved to user. For this we will firstly create a login section where user will provide interests, hobbies and designation in it, to make searching more useful.

Keywords – Content *description, keywords, Information Retrieval, Search engine, Query processing,*

I. INTRODUCTION

In the last decade many impressive enhancements has been done with respect to searching technology. Introducing new features Google, Yahoo, Bing are trying to improve the searching results. These search engines are basically based upon the keyword matching technique. Firstly, they don't retrieve results according to the user behavior. Secondly, it sometimes doesn't satisfy the user. Thirdly, many keywords have different meaning s associated with it. Fourthly, the retrieved documents must contain the query related word or keyword as much as possible.

As there are spammers which try to pollute the data of the documents using tricks like repetition, weaving and dumping.

It is to be known that the results retrieved by the Google, Yahoo, or Bing depend upon the visits on that particular page, but not on the quality of what the user is searching for. And hence this is becoming more and more challenging.

I.I.RELATED WORK

Semantic similarity method is applied to improve the re-ranking algorithm and to improve the searching quality. Top N results are returned to the user by search engine, and by using semantic similarities between the candidate and the query the re-ranking is done [1].

Aim of document [2] is to retrieve the list of ranked documents based on similarity document concept. Firstly it computes similarity scores between the documents based on a score function and the query. Thereafter, similarity score is generated and accordingly the documents are ranked. In the literature, Text Tiling algorithm is used here in three stages: tokenization into sentence-sized units, detecting the boundaries of the subtopic and score calculation. In this paper, evaluation of two different approaches for documents ranking is used. Firstly, Documents are ranked based on standard score calculation i.e. using the tf-idf concept. Secondly, Documents are ranked based on Textiling approach.

The novel ranking model presented in this paper [3] based on the concepts and relationship. Concepts and relationships exist both in the user query and document. This method is applied to improve the retrieval of relevant documents in the result-set produced by the search engine.

In this paper [4] Semantic evaluation of results are returned by search engines. The approach used here is not specific to a particular type of research tool; it is rather generic because the ontology that used here is not specific to a particular domain. The structuring of the architecture is set into modules that any one functioning module does not affect another module.

The authors have proposed a methodology [5] downloading relevant documents by using migrant technology that helps to reduce load on the servers. It means that an average amount of time spent on a web page is calculated and its history is maintained based on user profile and past knowledge. The authors suggested that their results based on limited web crawl and small user study, proposed algorithm can improve the quality of results.

I.II. PROPOSED METHOD

I.II.I. Information about data and resource

Our result heavily relies on search engine, and our importance is based on search engine, which directly affects our search result. We know the Google search has its own importance and is very good. So, in our experiment we desired to choose Google as our search engine. Further where the results retrieved by the Google search engine will be sorted. Our main experiment is based on ambiguity. Our purpose is to resolve the ambiguity. For example in our search engine if we fire a query in a search engine as apple, what results you think search engine will provide. The result retrieved either would be an apple fruit or Steve Jobs related apple products, or else consider another query as jaguar. Jaguar is an animal also, and a car also.

Here our main experiment is. Based on user's behavior we are trying to retrieve the result. We know Google search engine has a very large amount of data and it is not possible to take into consideration that huge amount of data. When a query is fired we retrieve few documents or say links as output. Therefore, we are proposing an approach based on considering only few links, and experimentations are performed on those links.

Firstly, we will create a login section for the user. If the user is new then the user has to to registration. In the registration section the user has to fill the entries like the name, address, email-id, hobbies, designation, etc. Our main focus is on user's designation. What kind of the person is user the searching will be performed. After successfully logging, the query is to be fired by the user. With the help of Google API the links and snippets are shown on the output. The links are retrieved to the user with the help of goggle web API, which lets incorporate Google web search into the web pages and applications that are developed, so the users can search all or part of the Web directly from the application. The Google Webs API is a web service that uses SOAP and WSDL standards. We will maintain a file of the links that are retrieved to us, and also we will maintain a file of every links paragraph. Thereafter, removing stop words of every links paragraph we will obtain some words. Thereby, we will maintain a file of it also. Then we will apply term frequency to every links paragraph separately.

Term Frequency: The number of times a word appears in the individual document itself is called Term Frequency. Suppose if multiple documents contain the same word many times then you run into a problem. That's why TF-IDF also offsets the approximation value by the frequency of the term in the entire collection, a value called Inverse Document Frequency (IDF).

Term Frequency (TF) = this is proportional to the number of word in the document.

TF= (Number of time t appearing in a document) / (Total number of terms in the document)

Inverse Document Frequency (IDF) = log_e(Total no. of documents / No. of documents with term 't' in it).

Approximation value = TF * IDF

TF-IDF is computed for each term in each document/links.

Term frequency is calculated by how any times the word is repeated in the same paragraph. If the word is repeated 3 times then it will consider it only 1 time. Same process is repeated with all the links. Thereafter term frequency is calculated we will store every links contained data separately and maintain a word count file of it.

Already seen that user has filled a registration form and has mentioned about designation. Designation contained files are already stored in our database, and we go on comparing it with the word count file. Each and every word of the word count file is checked with designation file and the sorting of the links is done. Whose words are matched more with designation file the links are thereby sorted accordingly.

I.II.II. PRESENTATION OF THE PROPOSED APPROACH

Figure 1 shows the overall approach for re-ranking mechanism, and how the links are will get retrieved to the user. Our overall aim to optimize the searches especially on ambiguous query and retrieve results to the user. Whenever the search is made (for suppose Google search, yahoo search ,etc) according to user behavior the search is not retrieved. What user behavior exactly means is why not artificially the devices that we use predict about human behavior. What the user wants in search results. Therefore according to user behavior we are going to retrieve the search results.

In the above figure 1 the registration form is being filled by the user. After logging, the user will fire a query (or say a keyword) and results will be retrieved as per the query. The links are retrieved to the user with the help of goggle web API, which lets incorporate Google web search into the web pages and applications we develop, so our users can search all or part of the Web directly from the application.

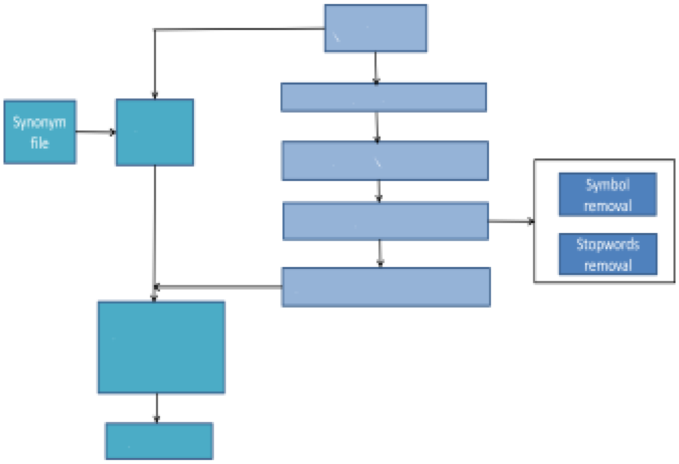

Figure 1. General architecture of the proposed approach

I.II.III. THE SORTING ALGORITHM

Key Phrase:-

Phase1:-for getting links and snippets

1) Input: Query.

2) Search: search query in URl via Mozilla agent to request for data.

3) Store: link and Snippet in text file.

Phase2:- for pre-processing

1)take a link from files and we are searching back in search engine.

2) get paragraphs for each link

3) Store the paragraph in a text file.

4) Apply Stemming and stopping on data.

5) Remove the special symbols, stop words.

9) Afer removing stop words store the words in a text file(eg.word.txt).

Phase3:- tf-idf:

1) take a text file 'word.txt' as a input and calculate term frequency of each word of every link separately.

2) if (count >= 2)

{

3)then add that word in text file(eg.wordcount.txt).

4) Compare this wordcount.txt file with the dictionary file(i.e. designation file).

5)If words matches in both the files store that word in database(e.g. matchword.txt).

6) if words matches more in the particular link, count of that link will be incremented.

}

Phase3:- To show the re-ranking link.

The links are retrieved from database on the basis of term frequency. According to tf-idf concept whose throughput is greater, those links will b retrieved and displayed to the user.

II. RESULTS

From Fig.2 we can see that how re-ranking is applied to unsorted links (that is on the left side). For example, when the query is "cancer" Google is returning the documents related to about the word cancer. Cancer means zodiac sign also, and cancer means disease also. If the user's designation is doctor, the results of cancer disease must be retrieved. And as we can see the sorting is performed on each and every link. We know that the results retrieved by user are dependent upon the number of visits on that page rather than the content what user is actually searching for.

Figure 2. Architecture of proposed ranking system

III. CONCLUSION

The purpose of this project was to develop a system that optimizes the search results. Introducing an approach, by which the ambiguities will be removed. Hence, we also tried to remove irrelevant documents and personalize the results and sort the links according to user behavior.

To make searching more useful we are providing a searching strategy based on interested area provided by the user in the login section. If the user is new to the project the user has to register first. Thereafter searching for the result we are considering the personal information such as hobbies, interest and designation of the user, which will make the further classification and sorting more relevant. This feature will show the most relevant document according to user behavior.

Further implementation can be based on creating history of the user. Which links the user has visited the most, and giving recommendations of similar links to the user.

IV. REFERENCES

[1] Roufan Wang, Shan Jiang and Yan Zhang, "Re-ranking Search Results Using Semantic Similarity", *Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, pp.1047-1051, 2011.

[2] Sanjeev Patel, Kriti Khanna and Vishnu Sharma, "Documents Ranking Using New Learning Approach*", International Conference on Computing Communication and Automation (ICCCA),* pp. 65-70, 2016.

[3] Poonam Chahal, Manjeet Singh and Suresh Kumar "Ranking of Web Documents using Semantic Similarity", *International Conference on Information Systems and Computer Networks (ISCON)*, pp.145-150, 2013.

[4] Abdelkrim Bouramoul, Mohamed-Khireddine Kholladi and Bich-Lien Doan, "An ontology-based approach for semantics ranking of the web search engines results" , *International Conference on Multimedia Computing and Systems (ICMCS),* 2012.

[5] Ashlesha Gupta, Ashutosh Dixit and A. K. Sharma, "Relevant Document Crawling with Usage Pattern and Domain Profile Based Page Ranking", *International Conference on Information Systems and Computer Networks* , pp.119-124, 2013.

[6] Chunchen Liu and Jianqiang Li, "Semantic-based Composite Document Ranking", *IEEE Sixth International Conference on Semantic Computing,* pp.126-129, 2012.

[7] Shashi Shekhar, K. V. Arya, Rohit Agrawal and Rakesh Kumar, "A WEBIR Crawling Framework for Retrieving Highly Relevant Web Documents: Evaluation Based on Rank Aggregation and Result Merging Algorithms", *International Conference on Computational Intelligence and Communication Systems*, pp.83-88, 2011.

[8] Ching-Yang Tseng, ChangChun Lu and Cheng-Fu Chou, "Efficient Privacy-Preserving Multi-keyword Ranked Search Utilizing Document Replication and Partition", *12th Annual IEEE Consumer Communications and Networking Conference (CCNC),* pp.671-676, 2015.

[9] Kozhushko O.A. and Tarkov M.S. "Using Hierarchical Temporal Memory for Document Ranking System Identification", *International Siberian Conference on Control and Communications (SIBCON),* 2015.

[10] Syandra Sari and Mima Adriani, "Learning to rank for determining relevant document in Indonesian-English cross language information retrieval using BM25*" International Conference on Advanced Computer Science and Information Systems (ICACSIS),* pp.309-314, 2014.

[11] Ajni.K.Ajai and R.S. Rajesh, "Hierarchical Multi-keyword Ranked Search for Secured Document Retrieval in Public Clouds", *International Conference on Communication and Network Technologies (ICCNT)*, pp.33-37, 2014.

[12] R.Sivashankari and Dr. B.Valarmathi, "An Empirical Semi-Supervised Machine Learning
Approach on Extracting and Ranking Document Level Multi-Word Product Names Using Improved C-value Approach*", International Conference on Advances in Computing, Communications and Informatics (ICACCI),* pp.770-775, 2016.

[13] Azam Feyznia, Mohsen Kahani and Reza Ramezani, "A Link Analysis Based Ranking Algorithm for Semantic Web Documents", *6th Conference on Information and Knowledge Technology (lKT),* pp.123-127,2014.

[14] Veningston .K and Dr.R.Shanmugalakshmi, "Enhancing personalized web search re-ranking algorithm by incorporating user profile", *Third International Conference Computing Communication & Networking Technologies (ICCCNT),* 2012.

[15] Patel Jay, Pinal Shah, Kamlesh Makvana and Parth Shah, "Review on web search personalization through semantic data", *International Conference on Electrical, Computer and Communication Technologies (ICECCT),* 2015.

[16] Ahmad Hawalah and Maria Fasli, "A Hybrid Re-ranking Algorithm Based on Ontological User Profiles", *3rd Computer Science and Electronic Engineering Conference (CEEC),* pp.50-55, 2011.