QUERY PROPHECY AND IMPLICATION FOR ITINERANT PHONES

P.J.SINIJOY[1], A.MARTHE RANJANI[2]
[1]P.G student, ME Computer and Communication, Sri Sairam Engineering college, Chennai-44, India;
[2] P.G student, ME Computer and Communication, Sri Sairam Engineering college, Chennai-44, India
Email: sinijoy15@gmail.com, a.m.ranjani11990@gmail.com

Abstract— **Itinerant web search introduces new peculiarities that were not present in the traditional web search. Nowadays users always carry modern cell phones which allow them to be permanently online wherever they go, at any time. A typical mobile web search scenario consists of a user who has an information need and is probably outdoors, so he cannot access a desktop computer easily. The required effort to type the query is too big and users get easily annoyed. This leads to the mismatched query problem, mobile queries become even shorter and more ambiguous than the same queries in a static desktop environment, and they are not able to guess accurately what the user intentions are. As a consequence the returned results are poor, and the user is forced to visit a lot of links and to navigate through several result pages in order to fulfill his information need. To cope with this problem, we propose an aid tool to simplify the query construction phase and to gather as much information as possible with the minimum user effort. The impacts of this system for the common man are to facilitate query entry through query prediction system, reduction in the number of key press, providing accurate results.**

1. INTRODUCTION

There are billions content on each page is stupendous. Browsing the web on the desktop is easy mainly because of two key reasons:

(i) Providing appropriate suggestion.

(ii) Easy input mechanisms like keyboard and mouse.

However, when it comes to mobile web browsing, the above stated factors are not present. Our aim is to provide appropriate suggestions for the query typed by the user and to reduce the number of key presses. We analyze the query typed and find the relevant words. Then the HTML format of the web page is taken and then it is parsed to get the suggestions. Stemming algorithm is implemented for efficiency. In present J2ME platform the suggestions are not provided which is the drawback of the system, which makes the user restless since the user has to type the entire query. In our proposed system the suggestions are provided based on the relevant word and parsing technique is used.

2.RELATED WORKS

Mario et al. [1] proposed the importance of the query expansion using knowledge-based recommended system, which acts as aid tool in query construction phase by providing semantic suggestions and auto completion. As a result, the number of required key-presses is minimized and users can find new options. Bhogal et al. [2] proposed on query expansion and process of assigning weights to the new word which is being added to the initial word. This paper deals with various query expansion approaches including relevance feedback, corpus dependent knowledge models and corpus independent knowledge models.

Maryam and Shumeet [3] proposed an algorithm based on contextual signals. To facilitate the query entry contextual signals are considered. The query prediction system redefines the prediction dictionary after considering contextual signals such as the application being used and the location of the user to complete the query before the user types the query. Jeff and Efthimis [4] proposed query reformulation technique. The stemming algorithm is developed to reformulate the first word in the query and to provide better suggestions. We apply our reformulation technique to proper suggestions on mobile phones during web

search. Apart from the advantage of loading the pages faster, we are able to provide the users with suggestions to complete their query on their mobile devices.

3. PROPOSED SYSTEM

In our proposed system the major task is to display the suggestions to the user. This is done by mapping to a web interface. And there is also a method of developing an ontology for a particular domain so that the system can function efficiently for that particular domain. But developing an ontology for a particular domain becomes user dependent. For example if the user is interested in the domain say cricket the ontology for cricket related words will be developed and when the user types a query related to cricket user gets the suggestion from the ontology. In our system we have not developed ontology instead we have mapped to the web so that the user can get suggestions for whatever query , independent of the domain of the query. The system requires a development of a user interfaces for getting the input from the user. The user interface includes a text field where the user types the query for which the suggestions are provided and the corresponding result page is generated.

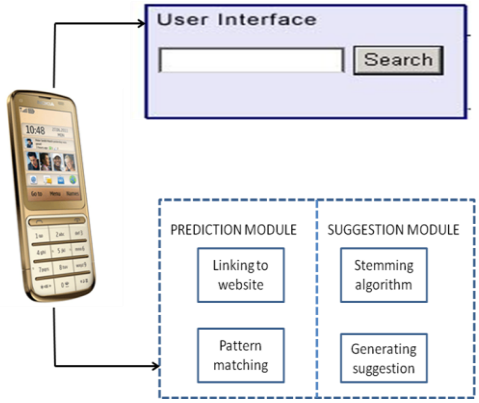

Fig 2. Overall Architecture

Prediction module has two processes as shown in Fig 2. The two processes are web interface and pattern matching. The search for the exact word is done by linking to the website. In web interface module for the first word which is being entered by the user the meaning, relevant and related words are being found from the website. The query which is typed by the user is mapped to the website. Pattern matching matches one or more alphabetic characters for the query which is being typed and the first two or three letters are matched with the words in the website. If the query and the word have the same initial pattern the corresponding word is given as suggestion.

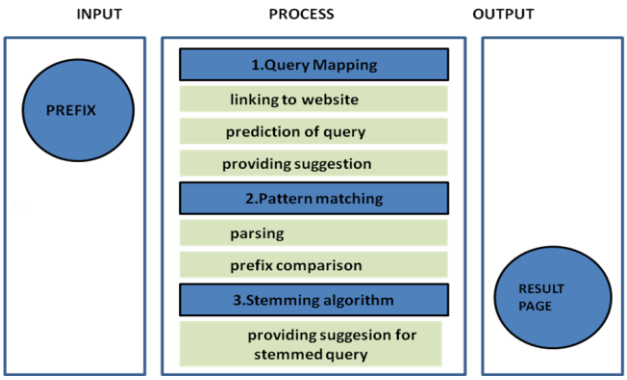

Fig 1. Schematic representation

The Suggestion module includes stemming algorithm which gives a precise suggestion. Affix removal conflation techniques are referred to as stemming algorithms and can be implemented in different methods. All remove suffices and/or prefixes are an attempt to reduce a word to its stem. The algorithms that are discussed in the following sections, and those that is implemented in this project is all suffix removal stemmers. During the development of a stemmer the issues of iteration and context awareness must be addressed. Suffices that are concatenated to words are often done in a certain order, such that a set of order-classes will exist among suffices. An iterative stemming algorithm will remove suffices one at a time, starting at the end of the word and working towards the beginning

An issue also exists about whether a stemmer should be context-free or context-sensitive. A context-sensitive algorithm involves a number of qualitative contextual restrictions that are developing to prevent the removal of endings that, in certain situations, can lead to erroneous stems being produced. A context free algorithm removes endings with no restrictions placed. In this paper stemming algorithm is implemented for the query which is typed by the user. A stemming reformulation involves changing the word stem in the first query. Suggestions are provided for the stemmed word. Stemming algorithm is implemented for the query which is typed by the user. The stemming algorithm is given below :

1. If the query ends with ‘ies’ but not ‘eies’ or ‘aies’ then ‘ies’ is replaced by ‘y’.
2. If the query ends with ‘es’ but not ‘ees’ or ‘aes’ then replace ‘es’ by ‘e’.

3. If the query ends with the suffix ‘full’ then stem ‘full’ and if the last letter after stemming is ‘i’ then replace ‘i’ by ‘y’ .
4. If the query ends with the suffix ‘ive’ then stem the suffix ‘ive’.

IF the first word ends with ‘ies’ and not with ‘aies’ or ‘eies’

THEN

Replace ‘ies’ by ‘y’

ENDIF

IF the first word ends with ‘es’ and not with ‘aes’ or ‘ees’

THEN

Stem ‘es’

IF the last letter after stemming is‘s’ THEN

Insert ‘e’

ENDIF

ENDIF

Fig 3. Stemming algorithm

4.IMPLEMENTATION AND TESTING

The algorithm explained above has been implemented and tested for 100 words on an average in 10 different J2ME enabled mobile phones. The result is that on usage users found it to be easy to implement such algorithm and search the key word. More relevant and appropriate key words were suggested.

A sample of the algorithm works has been shown below in the output Fig 4.

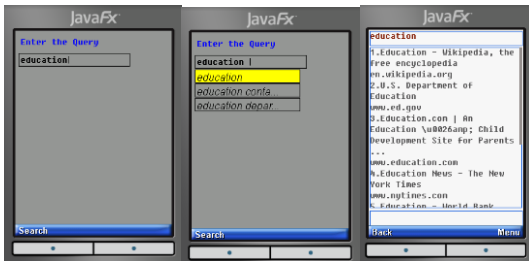

Fig 4. Ouput

After implementation the system has been tested with the user group. They were given a set of questionnaire and were asked for their feedbacks. The collective feedback is brought in a pictorial format in Table 1.

Query	**Time to enter query in existing system**	**Time to enter query in proposed system**
Q1	0.25	0.15
Q2	0.25	0.16
Q3	0.28	0.20
Q4	0.21	0.15

Table 1. User Experience

5. CONCLUSION AND FUTURE WORKS

The query is typed into the application installed in the mobile phones and the suggestions are provided. When the user selects a suggestion the result page is generated. With the exponential growth of information available on the World Wide Web, users increasingly turn to search engines to help them navigate through a sea of content in order to find the information they are looking for. The application can be extended with minimum effort to provide a wider scope without decreasing accuracy. The mobile landscape changes faster than the desktop's. It is quite natural to stretch the capabilities of the application by adding support for more and more mobile operating systems in the future. Also, the application has the potential to make use of all the useful APIs newer OS will offer since the application is basically built on top of a minimalist user interface and programmed in java, which is a very flexible language. More features like support for polished transitions and graphical elements can be added in the future.

6.ACKNOWLEDGEMENTS

The authors like to express their thanks to the management of Sri SaiRam Engineering College and the department of Information Technology for their encouragement and support during this work. Special thanks to Dr. T.Sheela, HOD of IT

department for his continuous help in pursuing our work.

7.REFERENCES

[1]. Mario Arias, Jose M. Cantera, Pablo de la Fuente, Cesar Llamas, and Jesus Vegas. (2010) 'Knowledge-Based Thesaurus Recommender System in Mobile Web Search', CERI, pp. 77-87.

[2]. J. Bhogal, A. Macfarlane, P. Smith. (2007) 'A review of ontology based query expansion', Information Processing Management 43, pp. 866-886.

[3]. Maryam Kamvar, Shumeet Baluja. (2007) 'The Role of Context in Query Input: Using contextual signals to complete queries on mobile devices', International conference on Human computer interaction with mobile devices and services ACM New York, NY, USA.

[4]. Jeff Huang, Efchimif N.Efchimiadis. (2009) 'Analyzing and Evaluating Query Reformulation Strategies in Web Search Logs', University of Washington.

[5]David, M and Liar okapis, F (2007) 'Mixed reality (MR) interfaces for mobile information systems' Aslib Proceedings, pp.422-436.

[6] Wei che, H. ;Trotman,A and Geva,S (2008) ' Efficient information seeking on a mobile Device' Proceedings of the SIGIR workshop on Mobile Information Retrieval.

