A Prototype for Detecting Users Interests Based on Search Behavior

GRADUATE PROJECT REPORT

Submitted to the Faculty of
The School of Engineering & Computing Sciences
Texas A&M University-Corpus Christi
Corpus Christi, TX

In Partial Fulfillment of the Requirements for the Degree of
Master of Science in Computer Science

By

Aravind Kumar Thalakokkula
Summer 2015

Committee Members

Dr. Long Zhuang Li  _____________________________
Committee Chairperson

Dr. Dulal Kar  _____________________________
Committee Member
ABSTRACT

The modern day world relies primarily on the Internet for many needs. The number of Internet users has grown tremendously recent decade. To serve the users there were many search engines exist on Web to retrieve the information to the users for their search.

The rapid propagation technology and increase in the use of Internet has changed the life of the modern day world. As the Internet is the interconnection of different networks, it is considered as the deep web and retrieving structured data from deep web pages is very challenging due to the complex nature of the information in the pages. Incorporating users behavior data can improve the ordering of top results on web search. As the number of users to the Internet has grown, it is difficult in some of the cases to detect the individual interests of the users. The better way to solve this problem is to provide personalize service to each user. For this it is better to develop users personal profile on the Web, based on his/her browsing behavior on the Internet. In this project, the browsing content of the user is analyzed to determine their interests.
# TABLE OF CONTENTS

Abstract .................................................................................................................. 2

Table of Contents .................................................................................................. 3

List of Figures ........................................................................................................ 5

1. Background and Rationale .................................................................................. 7
   1.1 Introduction .................................................................................................... 7
   1.2 Evolution of Data Mining .............................................................................. 8
   1.3 Personalized web Search ............................................................................... 9
   1.4 User Profile Modeling .................................................................................. 9
   1.5 Existing Methods ......................................................................................... 10
   1.6 Eclipse IDE .................................................................................................. 11
   1.7 Rationale ...................................................................................................... 13

2. Narrative ............................................................................................................ 14
   2.1 Problem Statement ....................................................................................... 14
   2.2 Motivation .................................................................................................... 14
   2.3 Project Objective ......................................................................................... 14
   2.4 Project Scope ............................................................................................... 15
3. Proposed System Design

3.1 System Design and Architecture

3.2 Dataflow Diagram

3.3 System Requirements

4. Implementation

4.1 Apriori Algorithm

4.2 Environment

4.3 H2 Database

4.4 WordNet

5. Testing and Evaluation

5.1 Software Testing

5.2 Unit Testing

5.3 System Testing

5.4 Test cases for the Application

5.4.1 Testing home page

5.4.2 Registration page

5.4.3 Login page

5.4.4 Search page
5.4.5 Searching keywords ................................................................. 24

5.4.6 Getting results from Google .................................................... 25

5.4.7 Searching for different keywords ............................................ 26

5.4.8 Displaying interests ................................................................. 28

5.4.9 Displaying different interests .................................................. 30

5.4.10 Displaying suggestions ........................................................ 30

6. Future Work ............................................................................. 32

7. Conclusion .............................................................................. 33

Bibliography and References ...................................................... 34
List Of Figures

Figure 1: System Architecture. ..............................................................16

Figure 2: Dataflow in the Application...............................................18

Figure 3: Home Page...........................................................................22

Figure 4: Registration Page..............................................................22

Figure 5: Login Page.........................................................................23

Figure 6: Search Field......................................................................24

Figure 7: Searching from the Application...........................................24

Figure 8: Results Page.....................................................................25

Figure 9: Resultant Webpage..............................................................26

Figure 10: Searching for Other Keywords.............................................26

Figure 11: Resultant Webpage..............................................................27

Figure 12: Searching in Different Way...............................................27

Figure 13: Displaying Interests............................................................28

Figure 14: Word count in the Database..............................................29

Figure 15: Working of WordNet..........................................................29

Figure 16: Generating Different Interests............................................30

Figure 17: Displaying Suggestions......................................................34
1. BACKGROUND AND RATIONALE

1.1 Introduction

Rapid popularity of the Internet along with evolution of information technology has changed nature of many businesses. In most of the cases, the large amount of transactional data collected from the information systems helps any organization to better understand the users needs and can improve the market plans [11].

The dynamically generated web is called Deep web, which is generated from the data sources such as file systems or the databases. There is a difference between the deep web and the surface web. If we consider the surface web, the data is mostly available through URL but in contrast, the data present in the deep web is guarded by the search interface. We can conclude the meaning of deep web as the process of collecting hidden data by giving keywords as queries through various search interfaces. Web services, HTML forms and programmable web APIs are mostly considered as the sources for deep web data. It is not possible for any of the users to get an access to the information, which is present in the Internet through static links [12].

In this modern day world, every individual is gaining much useful and additional knowledge from the Internet. Internet usage has become a daily routine to most of the modern day individuals. Especially for the students, Internet web search has become essential to expand their knowledge areas. To provide services to the users on large scale, there were millions of websites exist on the Internet. By looking at all the possible situations, there exists an approach called “Personalized web search”. To which, the main intention is to provide the users with more relevant data. PWS tries to eliminate
irrelevant data in most of the cases. In PWS a users query is gathered and analyzed to provide more appropriate results to the users query [10].

1.2 Evolution of Data Mining

The evolution of databases and information technology started in early 60’s and they are systematically evolved from an under developing file processing systems to complicated database. The research and development of the databases from the next decade has progressed from hierarchical databases to relational databases where the data are stored in relational table structure [3].

Data mining is about the analysis of raw data and turning it into useful information and knowledge. The extracted information is used in making predictions and decision making for the benefit of the organization or the applications for that matter. The information industry is more attracted towards data mining because of the availability of wide range of data and need for turning this available data into useful information and knowledge. The information gathered, can be used in applications irrespective of their field in which they are [2] [7].

Data can be accessed through query languages, optimized query processing and transaction management, which are flexible and convenient to the users. The evolution of information technology resulted in data mining. Database industry has witnessed an evolutionary path in the development of data collection, creation, data management analysis and understanding. Query and transaction processes are offered by numerous database systems as a common practice, so data mining and data warehousing has become primary targets to the detailed solution. In response to the above cases, data
mining uses software techniques to find patterns and regularities in sets of data and analyzes this data, to get the desired results for the users in most of the techniques [4].

1.3 Personalized Web Search

Personalized web search is considered as an important field for making the existing traditional Information Retrieval information into a more focused Information retrieval system. There are many problems for the Internet users for example we have different user from different backgrounds and different interests their search interest are varied in every aspect. As the users are not static and the information present in the Internet has to be modified time to time to improve the quality of the search. The PWS can help the users by constructing their profile and enhancing the search results by retrieving more focused information depending on the users interests. This can be used for suggesting good web pages for the user based on his search query or the keyword and his background knowledge [1].

1.4 Users Profile Modeling

The main intension of behind building a separate users profile is to reflect users interests and tries to estimate a prediction on the users interests on new queries. The users profile is used to improve the performance of personalized web search. [1][5].
1.5 Existing Methods

There were some existing methods such as creating users profile for using web usage mining, efficient deep web data extraction, discovering users interest from browsing behavior and improving web search ranking incorporating users behavior information.

Taking the users search history on the web, analyzing the links he/she searched on the internet and ordering the most visited web pages or most common links which are searched and ranking them according to the priority will lead to the efficient way of web usage. There exists another practice to increase the web usage, which takes the feedbacks of the resultant search pages queried by the user. In this approach, it is mainly concentrated on taking feedbacks of the users on their search results and analyzing them whether the retrieved information of their search behavior is appropriate for their interests or not [10].

On the other hand, creating the users profile using web usage is mainly concerned about the classification of the navigation patterns of the web users. This approach includes taking web server logs and building users profiles on web. Here, all the search information is stored and analyzed with users interests. Web mining is carried out using data mining techniques, so we can automatically extract the information from web services and documents. The web mining is categorized into four types such as:

1. Finding information or requested information is extracted from the textual documents, which are present on web.
2. Selecting information and pre-processing, the processing is performed automatically on the retrieved information.

3. Generalizing is done in order to detect the patterns of the personal web sites.

4. Analyzing the extracted patterns and estimating their validity.

On the other hand, web content mining is done to discover the effective and useful information, which is present in the web texts. This is also called web text mininng [10]-[13].

**1.6 Eclipse IDE**

In computer programming, Eclipse is an integrated development environment (IDE), contains a base workspace and extensible plug-in system for customizing the environment. Written mostly in java, Eclipse mostly can be used to develop applications. By means of various plug-ins, Eclipse may also be used to develop applications in other programming languages.

The Eclipse software development kit (SDK), which includes java application tools, is meant for java application developers. Users can extend its abilities by installing plug-ins, written for the Eclipse platform, such as development tool kits for other programming languages, and can write and contribute their own plug-in models.

Eclipse supports development for Tomcat, Glass Fish and many other servers and is often capable of installing the required server directly from the IDE. It supports remote debugging, allowing the users to watch variables and step through the code of an application that is running on the attached server.
The Eclipse SDK includes the Eclipse java development tools (JDT), offering an IDE with a built-in incremental java compiler and a full model of java source files. This allows for advanced refactoring techniques and code analysis. The Eclipse Public Licensing (EPL) is the fundamental license under which Eclipse projects are released. The Eclipse was originally released under the common public license, but was later re-licensed under the Eclipse Public License [2].
1.7 Rationale

The existing methods in personalized web search help the Internet users in most of the cases. But face some challenges like the information or the URL’s retrieved through the web may sometime does not have much relation with their sub-topics, and hence, the users profile can be built in a better way [8]. As the personalized web search is built on users profiles, as the profiles are online, and hence they keep on updating themselves giving enhanced users profiles. But here in this case, it does not care about the users personal preferences [6].

When comes to Deep web data, the main problem in data extraction is, that the data is always extracted or retrieved and analyzed based in HTML source code. In this system, the approach is html independent and the storing of the data in different forms to the database is a challenging process. A better clustering technique may be incorporated for an accurate data or information retrieval [12].

Another approach states that discovering the users interests is a very important task in providing personalized services. But this approach is concentrated on discovering users interests only in news depending on their browsing behavior [13].

To overcome all the above limitations, we propose a new system, which automatically detects the users interests depending on their search behavior to make. And making the users feel comfortable in detecting their day-to-day search interests.
2. NARRATIVE

2.1 Problem Statement

The above discussed existing methods or the algorithms, mainly focused on the generalization of a particular query. In many of the cases, the information, which is retrieved from the web, is related to some particular keyword searched by the user.

However when the user wants the more likely information he/she wants to search, then there exists the problem for the users to decide what exactly his/her interest is about. The previous approaches tend to give an answer to this problem to some extent. Such a one approach is to find the users interest on what kind of news he/she is most likely interested in, and thus, the existing application can only be served as an Internet news recommendation system.

2.2 Motivation

The main motivation on building this application is to increase the search effectiveness of the keywords, which are queried by the users, and allowing the user to build his/her own profile to make personal preferences of his/her own choice and interests. It is not only to the recommendation for the news or some other keywords, but to all the keywords he/she is searching for.

2.3 Project Goal

The main goal of this project is to maintain the users interest on his/her profile, depending on his/her search behavior. The application builds a customer’s profiles from
his/her browsing behavior recorded by the database and tries to recommend personal services to the profiles.

Once the user is done with his/her web browsing, and all the browsed links are saved and analyzed, and the algorithm, in the backend, will analyze all the links, which are clicked and automatically count which keyword is appearing the largest number of times. Now the most likely appeared keyword matches with the list of predefined keywords in the database and displays on the users account on later login of the user.

2.4 Project Scope

In this application, users are allowed to build his/her profiles. He/she is provided the access to Google search. User is are freely allowed to web browsing. He/she can search his/her interested keywords.

As soon as the users click on different links, the information in the web pages is scraped into the database and the text on those web pages is analyzed to get the most happening keyword. That most happening word is then sent to the WorldNet to search for the synonym of that particular word. Then, the user is automatically shown with the interests of his search.
3. PROPOSED SYSTEM DESIGN

3.1 System Design and Architecture

The proposed system architecture for the application is shown as in Figure 1.

![System Architecture Diagram]

Figure 1: System Architecture
Here as we can see in the architecture, the user first logs into the application, and builds his/her profile with the appropriate user name and password. All the information is stored into the database.

Now when the user does the search, he/she gets the results from the Google search engine. By clicking of any URL, he/she navigated to the respective web page. Now the information/content, which is present in those web pages, is automatically stored into the database. In the backend, an algorithm named Apriori algorithm is applied on the stored web content. This algorithm extracts the most happening keyword from those content and gives it to WordNet.

The above process continues for each and every clicked URL link. It means the contents of the web pages are continuously processed while the user is searching. So the user is displayed with different interests as he/she search different keywords.

The user is even provided with the suggestions for every interest he/she is displayed in interests tab.

### 3.2 Dataflow Diagram

As shown in Figure 2, it can be seen that how dataflow is occurring in the application. The user is logged into the application, searches keywords through search field. Clicks different links and the web content of those clicked web pages are analyzed through the data processing technique and the most happening keyword is sent to WordNet to display Interests to the users profile.
3.3 System Requirements

Software:

Front End Development. : Java Struts 1.3

Database Connection. : JDBC

Client Requirements. : HTML and Java Script

Hardware:

Inter Core 2 duo processor or above

RAM capacity 1 GB or more

Hard Drive capacity 250 GB or more
4. IMPLEMENTATION

From the architecture and flow chart, we can say that the application is mainly built for the search purpose as users are provided with a search field. The search field in this application is similar to the Google search field as the results are fetched from Google API. When user clicks any link, the information or the web content in the resultant pages are stored into the database and automatically processed by the Apriori algorithm.

4.1 Apriori Algorithm

Apriori algorithm is used for frequent item set mining and association rule learning over transactional databases. It proceeds by identifying the frequent individual items in the database and extending them to larger and larger item sets as long as those item sets appear sufficiently often in the database. Apriori uses a "bottom up" approach. The groups of candidates are tested against the data. The algorithm terminates when no further successful extensions are found. Below is the snippet for Apriori Algorithm.

```java
public Map<String, Integer> getFrequentPatterns(String value) {
    Map<String, Integer> wordMap = new HashMap<String, Integer>();
    StringTokenizer st = new StringTokenizer(value, " ");
    while (st.hasMoreTokens()) {
        String tmp = st.nextToken().toLowerCase();
        if (tmp.length() > 3 && !stringContainsItemFromList(tmp, StopWords.stopWordsofwordnet)) {
            if (wordMap.containsKey(tmp)) {
                wordMap.put(tmp, wordMap.get(tmp) + 1);
            } else {
                wordMap.put(tmp, 1);
            }
        }
    }
    return wordMap;
}
```
4.2 Environment

The system is implemented using Java in Eclipse IDE environment, as they are more suitable for programming. Apache tomcat along with H2 database is used to store the entire searched web content and do the data mining.

4.3 H2 Database

H2 is a relational database management system written in Java and can be embedded in java applications or can be run in the client server mode. This is available as open source software.

In this database, it is possible to create both in memory tables and disk-based tables. All the data manipulation tables are transactional. Table level locking and multi version concurrency control are available in this database.

4.4 WordNet

WordNet is a lexical database for the English language. It groups English words into sets called Synsets. This relates the word sets and groups them together to a required category. Words from same lexical category are mostly synonyms and grouped into Synsets. Synsets mostly contain simplex words collected or grouped together [2].

Other than these above approaches, there were some other approaches such as supporting privacy protection to the users data. Other approaches include setting users interests to the users personal instance of the database and searching. With this approach the users can get the exact information of what he/she is needed exactly without retrieving any irrelevant data.
5. TESTING AND EVALUATION

5.1 Software Testing

The process of evaluating a software product to identify the differences between given input and expected output is called as software testing. The quality of a product can be judged by testing.

5.2 Unit Testing

Unit testing is a type of testing where a single unit or groups of related units are tested. It is tested for the expected outputs for given inputs of the unit that is implemented. The Apriori algorithm is tested for finding out the most occurring keyword set from the webpage.

5.3 System Testing

System testing usually verifies that the system being delivered meets the scope of the product along with the specifications and its purpose [17]. In this project, one of the system testing tasks could be, how many users can be registered with this product? The testing part for this project can be evaluated by trying with different keywords for the target event based on the expected output.

5.4 Test Cases for the Application

5.4.1 Testing home page

This test case detects the home page of the application. As our application is for automatic detection of the user interests depending on the users search behavior, the user must create an online profile for the application.
5.4.2 Registration Page
The application has a registration page so that user can register into the application by providing necessary details. This form helps users to build their online profile.

5.4.3 Login Page

![Login Page](image)

**Figure 5: Login Page**

Once the user is registered with the application, he/she can login into the application with appropriate login details. This feature allows the user to go into the application.

5.4.4 Search Page

Once the user logged into the application he/she is provided with the search field.

This field is similar to the Google search field where a user can enter the keywords of his/her choice.
5.4.5 Searching keywords

Figure 6: Search Field

Figure 7: Searching from the Application
In the Figure 7, it is clearly shown that the search results are live results and fetched from Google. It shows the search field in the application working same as the Google search.

5.4.6 Getting results from Google

![Figure 8: Results page](image-url)

When a user clicks on any of the link or the URL from the results page, the application shows a pop-up to the user that some URL is clicked and is redirected to that web page. This is shown in Figure 8.

Once the URL is clicked the user is navigated to the respective webpage. This is shown in Figure 9.
5.4.7 Searching for different keywords
We can see different results for different search criteria are shown in Figure 10. And on click of any URL, the results are fetched from Google and are shown in below Figure 11.
Figure 12 shows that the application can also retrieve the results for any different type of searching. Many of the Internet users search the web by typing the phrase into the search field. This test case passes by proving the application similar to the Google Search.

5.4.8 Displaying interests

![Figure 13: Displaying Interests](image)

As shown above Figure 13, shows the resultant interests of a particular search.

Here the application behaves in such a way that, when a user searches for something, the web content of the web page is analyzed by the Apriori algorithm and the most happening keyword is extracted and sent to WordNet for processing. In the same way, when a user searches for other keywords, the algorithm processes the data automatically.
Here the main point to be understood is, if the most happening keyword from the second search results does not exceed the most happening keyword of the previous search, then the interest bar does not change. It only changes whenever it gets the most occurring pattern than the previous.

These results are due to the item set count by the Apriori algorithm as shown below in Figure 14.

**Figure 14: Word count in the Database**

Working of WordNet is shown below in Figure 15.
Figure 15: Working of WordNet

5.4.9 Displaying different Interests

Figure 16 shows application displaying different results. This is due to the count of the keyword greater than that of the previous search.

5.4.10 Displaying Suggestions

In the following Figure 17, it shows how different suggestions are displayed for the user. The suggestions are calculated in the descending order of the keywords.

This suggestions tab helps user in finding out their previous searches.
Figure 17. Displaying Suggestions
In future, the same application can be built using different word processing techniques. The word processing tool, WordNet, can process keywords that are other than the personalities or human related keywords.

In future, this limitation can be overcome by using more powerful word processing tools so that the application can get more appropriate interests depending on the users search behavior.

The suggestions tab can be improved by providing the users with appropriate URL links so that the users can be easily directed to the suggested web pages.
7. CONCLUSION

This application is about automatic detection of user interests depending on user search. This approach outperforms the previous methods such as finding interests based on click-log based methods etc.

The WordNet tool, which is used in this application works better for the keywords, which are other than the personality behavior keywords. The suggestions, which are presented in the descending order of the search, also helps the application users in easy browsing.
BIBLIOGRAPHY AND REFERENCES


[10] Improving web Search ranking by incorporating users behavior information, September 2014.
