IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 4, April 2016

Personalized Web Search using Machine Learning Technique

Pratiba D[1], Shobha G[2], Samrudh J[3]

Assistant Professor, Computer Science & Engineering, R V College Of Engineering, VTU, Bangalore, India[1]

Professor, Computer Science & Engineering, R V College Of Engineering, VTU, Bangalore, India[2]

Student, Information Science & Engineering, R V College Of Engineering, VTU, Bangalore, India[3]

Abstract: Machine learning is a field which deals with the creation and analysis of algorithms that can follow a reinforced methodology of learning. Such algorithms work on the basis of predictions made based on inputs, rather than following programmed instructions. In the existing search engines, if the user wants to know his likeness based on his previous searches, nothing except the history is present in the browser. In the proposed system, a framework has been designed to get a user specific re-ranking for the various websites and pages from an already present search engine and to implement Support Vector Machine Algorithm to find out the weightage of each concept. The main objective of the proposed system is to provide personalized results to the users based on their individual interests. It re- ranks the results for a given query obtained from existing search engine based on the user need. Profile database is got from a search engine for a given user query. Topical preferences are symbolized by words that occur regularly, referred as concepts. Weights are assigned to these concepts and this shows the user's degree of interest in that concept. To show the relations among the different concepts, required information is saved along with its strength. For every query's outcome, a score is given to each snippet based on weights of different concepts that they have and new ranking is done based on the scores. Currently web search personalization is used by social networking sites, search engines, banking services and online shopping sites.

Keywords: Machine Learning, Personalized Web Search Engine, Webpage Re-ranking, Supervised and Unsupervised Machine Learning, Support Vector Mechanism.

I. INTRODUCTION

Machine learning is a kind of artificial intelligence which gives computers the capability to learn without an explicitly written program. It focuses on the development of computer programs that can allow self-learning to occur and allows it to grow and change when exposed to new data. It has a great relation with Artificial Intelligence. It delivers methodologies, theories and applications to the field. Machine learning is used in various tasks where it is not feasible to build algorithms which are rule based and explicit. Optical Character Recognition, Computer Vision, Spam Filtering are some examples of its applications. [1].

Types of Machine Learning include Supervised Learning, Unsupervised Learning and Reinforcement Learning. Supervised learning deals with providing example inputs and desired outputs. Unsupervised and Reinforcement Learning employ algorithm based learning. Reinforcement is more dynamic in nature and goal oriented. The significance of Personalization is to customize the web for individual users by filtering out the irrelevant results and identifying only relevant ones [2]. The key steps of web personalization process includes i) Web Data Pre-processing ii) User Modelling in Personalization iii) Recommending Personalized Page Ranking Strategies. Every step of a Personalization process requires adaptability because of the change in the user's interest and instant information growth.

II. THE THREE DISTINCT PHASES

In personalized search systems, the part of user modeling influence the search in three phases, explained as follows:

- **Retrieval process:** User profiles are built into the process of searching. It is used for scoring the documents on web. It depends on results provided by an external search.
- **Re-ranking:** Here, the content and data from an ordinary non personalized search engine is taken and the content is personalized based on the interest of the user.
- **Query modification:** In this approach, the profile of the user is used to make changes to the information needs.

Considering the constraint of time enforced on the systems used for search and personalization being a process involving more time, the user profiles get better only with more time and usage. Personalization systems which provide new ranking to the documents obtained from retrieval generally utilize user profile on the client side. Also, rather than obtaining all results from the source, they re-rank only certain top ranked documents. Due to this additional time required, the process becomes considerably slow but a high level of personalization can be obtained. In query modification approach, only query representation will be altered in the profile of the user. Hence, it is less likely to affect result lists.

DOI 10.17148/IJARCCE.2016.5433

A. Personalization Approaches

There are mainly two approaches on the basis of history of search, Online and Offline. Offline approaches use the history in a unique pre-processing step, where an analysis of various relationships is done. Online approaches capture these data as soon as they are available and provide personalized results on the basis of the last interactions. Even though the latter approaches provide updated suggestions, an offline approach can implement more complex algorithms.

B. Online Approach

An enhanced personalized search engine builds the user profile by means of implicit feedback techniques. In particular, the system records a trail of all queries and the web sites the user has selected from the results, building an internal representation of his needs assigning a higher score to the resources related to what the user has seen in the past. Leung et.al [3], proposed a search system, which improves search accuracy by creating user profiles from their query histories and examined search results. These profiles are used to re-rank the results returned by an external search service by giving more importance to the documents related to topics contained in their user profile. In this approach, user profiles are represented as weighted concept hierarchies. The Open Directory Project (ODP) is used as the reference concept hierarchy for the profiles. For each individual user, two different types of information are collected, the submitted queries for which at least one result was visited and the snippets such as titles and textual summaries. A classifier trained on the ODP's hierarchy, chooses the concepts most related to the collected information, assigning higher weights to them. After a query is submitted to the wrapper, the search result snippets are classified into the same reference concept hierarchy. A matching function calculates the degree of similarity between each of the concepts associated with result snippet j and the user profile i.

The degree of similarity between each concepts is calculated using the similarity formula given by the following equation:

$$sim(user_i, doc_j) = \sum_{k=1}^{N} wp_{i,k} \cdot wd_{j,k}$$

In this similarity formula, wp(i,k) is the weight of the concept k-in of the user profile i, wd (j,k) is the weight of the concept k-in of the document j and N is the number of concepts.

The final weight of the document is used for re-ranking which is calculated using the weighting scheme given below:

$$match(user_i, doc_j) = \alpha \cdot sim(user_i, doc_j) + (1 - \alpha) \cdot googlerank(doc_j)$$

In the equation, α gets values between 0 and 1. When α is 0, conceptual rank is not given any weight and the match is equivalent to the original rank assigned by GOOGLE. If α has a value of 1, the search engine ranking is ignored and pure conceptual match is considered. Obviously, the conceptual and search engine-based rankings can be blended in different proportions by varying the value of α.

C. Offline Approach

An innovative personalized search algorithm is the Cube SVD algorithm, introduced by Sun et.al [4], and based on the click-through data analysis. This technique is suitable for the typical scenario of web searching, where the user submits a query to the search engine, the search engine returns a ranked list of the retrieved web pages and finally the user clicks on pages of interest. After a period of usage, the system will have recorded useful click-through data represented as triples such as user, query, visited page that could be assumed to reflect users' interests. The proposed algorithm aims to model the users' information needs by exploiting such data. It addresses two typical challenges of web search. The first concerns the study of the complex relationship between users, the query and the visited web pages. The proposed framework is to capture the latent associations among the objects. The second challenge is the problem of data sparseness. A user generally submits a small number of queries compared with the size of the query set submitted by all the users and visits few pages. In this case, recognizing relationships among the data becomes very essential. For this, a unified framework to model a click-through element as a 3-order tensor, i.e., a higher order generalization of a vector (first order tensor) and a Matrix (second order tensor), on which 3-mode analysis is performed using the Singular Value Decomposition(SVD) technique can be used.

III. REQUIREMENTS AND SPECIFICATIONS

The personalized search engine is developed for general end users who have little knowledge of computer. Hence it follows a set of standard requirements to design and run successfully. It includes the user characteristics, general constraints and assumption dependency.

D. User Characteristics

$$sim(user_i, doc_j) = \sum_{k=1}^{N} wp_{i,k} \cdot wd_{j,k}$$

This project provides customized results to the end users. The outcome of the project will benefit users in getting the result of his interest. The interface is designed so as to minimize the depth of knowledge required by the end user.

E. Constraints, Assumption, Dependencies

The development of the project is limited by few conditions and confined to certain circumstances [5].

- The project gets the required real time data as input from the search engine like Google. So the system must be connected to the internet and client must have a valid user ID and password.
- The system does not run successfully without server. So any servers such as Tomcat Apache server should be included in client system.

DOI 10.17148/IJARCCE.2016.5433

The Personalized Search Engine depends on certain datum and few assumptions are made to personalize the search.

- The extensions that are required for connecting to the storage arrays such as the Heidi SQL must be present at the client system.
- The storage area should have sufficient capacity to accommodate the result data collected by the API's.
- Information on various storage areas such as the server credentials, namespaces, URLs of the graphical interfaces representing the data etc. must be available to the client.

IV. HIGH LEVEL DESIGN

This section discusses the High Level Design that will be used in the development of the personalized search engine models. It highlights the techniques and approaches used during development of these models and provide detailed analysis.

F. Development Method

The project is designed using Eclipse IDE. The user interface is to be coded in Java along with the basic classes and methods. The front end is designed using JSP along with CSS and JavaScript which will be presented to the user. The major classes that are being used are Controller class, Delegate class, Service class and DAO (Data Access Object) class. Other than these, there are few other classes that are used to connect, manage jobs and provide helper functions.

G. Architectural Strategy

The architectural strategies include the preliminary decisions that need to be taken for implementation such as programming languages considered, the development phases, the user paradigms such as graphical user interface decisions and many other important features which are covered in this section.

1) Programming Language

Java was the effective option among others to be chosen as programming language. Java has strong base for Object Oriented Designing. Also the polymorphism feature of Java provides effective solution for version specific design issues. JSP has been used for designing web interface.

2) Error Detection and Recovery

The main source of error arises due to wrong user input. In order to negate this, several error control mechanisms that detect errors before they cause failure have been put in place.

This includes parameter checks, boundary condition checks, try-catch block for exception handling. Also a log file is maintained that records the stepwise activity of the system and it can be used to track the errors.

3) Data Storage Management

Data storage management is required for competent nature of the program. It must be ensured that all real time variables and objects are properly cleared and cleaned up. Various techniques must be used to guarantee a certain amount of available RAM to the program memory space.

H. System Architecture

Figure 1 represents the architectural diagram of personalized web search. The system consists of three main modules, namely SVM computation, Concept Building and Re-Ranking. The following sections explain each module of the system architecture.

1) SVM Computation

The Data is collected from the user search history. The raw data is categorized into different categories like URL, description, title and query. The browsed data is cleaned so that all the meaningless terms is removed. The cleaned data is passed for tokenization process where each word is converted into tokens. Frequency is computed by counting the number of times a token is presented in a web snippet.

2) Support Value Computation & Concept Building

Support value of each token is calculated using the formula given below:

$$\textbf{Support }(\mathbf{C_i}) = (\textbf{sf}(\mathbf{c_i})\ ||\mathbf{c_i}||\ \acute{ø}) / \textbf{n}$$

where Support(c) = Support Value
Sf = snippet frequency for concept c_i
$|c_i|$ = number of words
n = number of web snippets
ǿ = threshold value $0.1 \leq ǿ \leq 1$

If the support value of any token is greater than manual threshold value, it becomes a concept.

3) Re-ranking

Search results are displayed in descending order of support value which is different for different users.

V. DETAILED DESIGN AND STRUCTURE CHART

Software design has become the most essential part of the overall system development in all software development projects[6]. An informal design is prepared from the initial set of requirements which is usually in natural language. All the modules involved are described in detail along with implementation in terms of data structure and algorithm.

I. Structure Chart

A Structure Chart shows the flow of control among the different modules in a system along with the interaction between them. It shows the input for every module and gives the output. Figure 2 is a structure chart that represents all the modules and data flow within them. The sequence of the data as well as control flow is also presented in the structure chart. The core functionality of the system can be divided into five modules which are explained further.

1) Data Cleaning

This module takes search result's description as input and provides cleaned data which is free from stopwords as output.

2) Tokenization

- **Purpose**: The purpose of this module is to convert the raw data into tokens so that it can be used for further processing.
- **Functionality**: The functionality of this module includes stopwords analysis and tokenization of clean data.
- **Input**: The input is the query of user choice and predefined stopwords which should be removed from clean data.
- **Output**: The output is token of each word present in the description of the clean data.

3) Frequency Computation

- **Purpose**: The purpose of this module is to get token with their respective frequency. Frequency is the number of occurrence of a token in its own web snippet.
- **Functionality**: The functionality of the module is to compute the number of occurrences of a token.
- **Input**: The input is token.
- **Output**: The output is the tokens of cleaned data with their respective frequency.

4) Support Function

This module is used to compute the support value of each token using the formula explained in Section IV – C 2

5) Concept Building

- **Purpose**: The purpose of this module to get the tokens with their respective support value and to build the concept out of that.
- **Functionality**: The functionality of this module is to find the support value of each token and to build the concept of all the tokens whose frequency is greater than threshold frequency.
- **Input**: The input is tokens with respective frequency value.
- **Output**: The output is personalized search results arranged in descending order of support value.

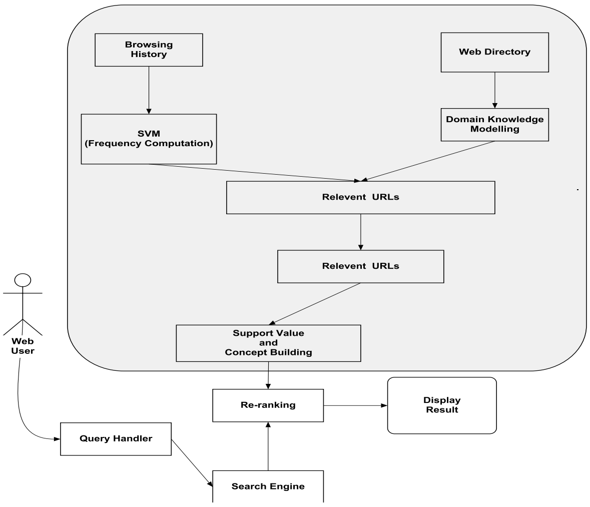

Fig 1 System Architecture for personalized web search

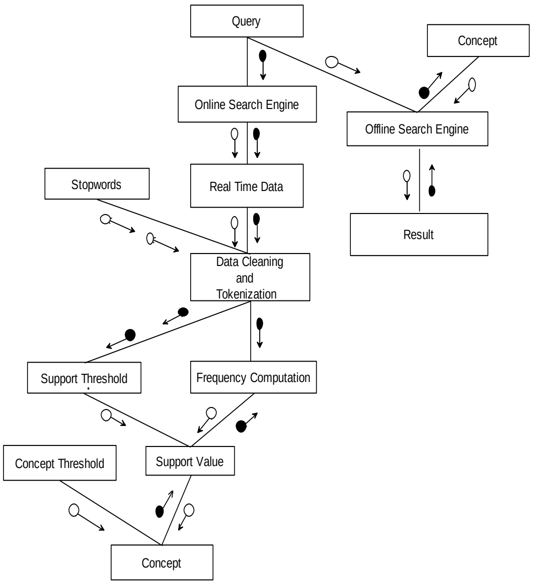

Fig 2 Structure Chart for personalized web search

VI. EXPERIMENTAL ANALYSIS AND RESULTS OF PERSONALIZED WEB SEARCH ENGINE

The Personalized web search operations were carried out on different queries, which have different meanings in different contexts as part of an experiment to evaluate its working. It is successfully experimented for different queries.

This section highlights the outcome of the experiment conducted and the inference that was drawn after the tests. The evaluation metrics are as listed and the quantification of the respective outcomes has been done. The results that were got from the experiment describe the general performance trend of the personalization.

J. Evaluation Metric

The main aim of personalized web search engine is to provide the customized results to the end users. Hence the performance evaluation is done based on criteria of support values of different queries.

The most vital criterion for this particular project is the time taken to sort the result and data to the respective category. The four key criteria are :

- **Accuracy:** It is the ratio of the number of correct cases to total.
- **Precision:** It is the part of the retrieved instances which are pertinent to the given query.
- **Recall:** It is the percentage of correct items that are selected from among the retrieved data.
- **F Measure:** A metric that combines precision and recall metrics. It is the weighted harmonic mean.

K. Experimental Dataset

The stated implementation was tested on search results for about 50 different queries collected from different users having different meaning in different context.

DOI 10.17148/IJARCCE.2016.5433

L. Performance Analysis

Table I
Various Classifiers & Performance analysis

Classifier	Accuracy (%)	Classification Time (sec)
Naïve Bayes	86.2546	0.900
K-NN	67.5490	2.200
SVM	97.7227	3.800

The performance analysis of different classifiers such as Naïve Bayes, K-NN and SVM are shown above. It is clear from the table, that SVM classifier is the best classifier for the personalization of real time data. Although the classification time for the SVM is more it has very high accuracy.

ACKNOWLEDGMENT

A number of people, in their own capacities have helped us in carrying out this work. We would like to thank them all. The College and the Principal of our college, R.V.C.E have been a moral support. Thanks to the various professors of out Department for the support. Thanks to our peers and friends who provided us their encouragement. Thanks to all.

REFERENCES

[1]. Tarannum Bibi, Pratiksha Dixit, Rutuja Ghule, Rohini Jadhav, “Web Search Personalization Using Machine Learning”, Proceeding of the 2014 IEEE International Advance Computing Conference (IACC), 2014, pp 1296-1299.

[2]. K. Collins Thompson, P. N. Bennett, R. W. White, S. de la Chica, and D. Sontag, “Personalizing Web search results by reading level,” Proceeding of the ACM International Conference on Information and Knowledge Management(CIKM ’11), New York, NY, USA, 2011, pp. 403-412.

[3]. K.W.T. Leung, W. Ng, and D.L. Lee, “Personalized Concept-Based Clustering of Search Engine Queries”, Proceeding of the IEEE Trans. Knowledge and Data Eng., Vol. 20, (11) ,Nov. 2008, pp. 1505-1518.

[4]. J.T. Sun, H. J. Zeng, H. Liu, Y. Lu, and Z. Chen, “CubeSVD: A novel approach to personalized Web search”, Proceeding of the 14th International Conference on World Wide Web, ACM Press, 2005, pp. 382-390.

[5]. Cordón, Oscar, Enrique Herrera-Viedma, and Marıa Luque, "Fuzzy Linguistic Query-based User Profile Learning by Multi objective Genetic Algorithms", Proceeding of the Evolving Fuzzy Systems, 2006 International Symposium, IEEE 2006, pp. 261-266.

[6]. Kamlesh Makvana, Pinal Shah, Parth Shah, “A Novel Approach to Personalize Web Search through User Profiling and Query Reformulation”, Proceeding of the 2014 IEEE, pp. 22-36.

[7]. Kim H. R, and Philip K. Chan, "Personalized ranking of search results with learned user interest hierarchies from bookmarks", Proceeding of the WEBKDD, 2005, pp. 32-43.

[8]. Sieg, B Mobasher, and R. Burke, “Ontological user profiles for representing context in web search”, Proceeding of the Web Intelligence and Intelligent Agent Technology Workshops, 5-12 Nov. 2007, pp.91–94.

[9]. Michlmayr E, Cayzer S & Shabajee P, “Learning User Profiles from Tagging Data and Leveraging them for Personal(ized) Information Access Tagging and Metadata for Social Information Organization” ,Proceeding of a workshop at WWW , 2007, pp.1123-1130.

[10]. N. Matthijs, and F. Radlinski, “Personalizing Web search using tong term browsing,history”, Proceedings of the ACM WSDM Conference on Web Search and Data Mining, 2011, pp. 25–34.

[11]. Peng, Xueping, ZhendongNiu, Sheng Huang, and YuminZhao, "Personalized Web Search Using Clickthrough Data and Web Page Rating", Proceeding of Journal of Computers 7, 2012, pp. 2578-2584.

[12]. Liu, Fang, Clement Yu, and WeiyiMeng, “Personalized web search for improving retrieval effectiveness", Proceeding of the Knowledge and Data Engineering, 2004, pp. 28-40.

DOI 10.17148/IJARCCE.2016.5433