AUTOMATIC UPDATION OF USER BEHAVIOR PROFILES FOR SEARCH ENGINE PERSONALIZATION

Sadesh SelvaRaj[1], Suganthe Ravichandran[2], Subathra Sengottian[3]
[1]Research Scholar and Assistant Professor, Velalar College of Engineering and Technology, Thindal, Tamil Nadu, India
[2]Professor**,** Kongu Engineering College, Perundurai, Tamil Nadu, India.
[3]M.E Student, Deptt. of Computer Science, Velalar College of Engineering and Technology, Thindal, Tamil Nadu, India

ABSTRACT

Knowledge about computer users is useful for predicting their future behaviours. An user profile is erratic, a method is needed to update the evolving user profiles. An approach used for creating and updating automatically the profile of a computer user is used called Evolving Agent behavior Classification based on Distributions of relevant events (EVABCD). This approach depends upon the observed behavior of the user .It changes search interest specified in the profile of the user automatically when the user's search interest changes and to retrieve search results based on that interest.The behavior of the user is characterized by the commands that the user types. The EVABCD approach along with proposed features is highly efficient to adapt to the evolving user profiles and return better search results according to evolving user's interest.

KEYWORDS: *User profile, EVABCD approach, Relevant events, Classifiers, User modeling.*

I. INTRODUCTION

A web search engine is used to search and retrieve information from WWW and FTP servers. The search results that are retrieved are presented in a list of results. This information may consist of web pages, images and other types of files [2]. Some search engines also mine the data available in databases. Unlike web directories that maintained by human, search engines operate algorithmically or are a mixture of algorithms and human input.

Most search engines return the same results for the same query, regardless of the user's interest. Since queries submitted to search engines are short and they will not express the user's precise needs.

- A good user profiling strategy is a fundamental component in search engine personalization. Search engine personalization is the act of gathering and interpreting the user profile information.
- Most personalization methods is based on the creation of one single profile for a user and user have to specify his search interest in that and based on that interest ,the results will be retrieved. If his search interest changes, he have to update that manually. Different queries from the user should be handled differently because a user's preferences may vary across queries [3]. For example, a user who prefers information about fruit on the query "apple" may not prefer the information about Apple Computer for the query "apple." Personalization strategies should be done such that based on user's interest, the result should be obtained.
- Based on the changing behavior of the user, profiles should be updated automatically. An adaptive approach for creating behavior profiles and recognizing different computer users called Evolving Agent behavior Classification based on Distributions of relevant events (EVABCD) is used and it is based on representing the observed behavior of an agent (computer user) as an adaptive distribution of her/his relevant atomic behaviors (events)[8].

This paper is organized as follows: Section 2 provides as overview of related work. The main improvement and the development of the personalized automatic updation of the user profile is

provided in Section 3.Section 4 describes the results and discussion. Conclusion and future work provided in section 5.

II. RELATED WORK

The novel evolving user behavior classifier is based on Evolving Fuzzy Systems and it takes into account the fact that the behavior of any user is not fixed, but is rather changing. In [5] Amandi and Godoy proposed an approach can be applied to any behavior represented by a sequence of events.

- In order to classify an observed behavior, as many other agent modeling methods, creates a library which contains the different expected behaviors. This library is not prefixed, but it is evolving and learning from the observations of the users behaviors and it will be filled from scratch by assigning temporarily to the library the first observed user behavior as prototype. The Evolving Profile Library (EPLib), is used to store the different expected user's behaviors and it is continuously changing. Trie based approach with the classifier is being used.

The following are the two main actions proposed by Gong and Pepyne [8]:
1. Creating and evolving the classifier. This action involves two subactions:
a. Creating the user behavior profiles. This subaction analyzes sequences of commands typed by the users online and creates respective profiles for the users.
b. Evolving the classifier. This subaction includes update of the classifier. The profile of the user changes automatically whenever search interest of the user changes.
2. User classification. The user profiles created are associated with one of the prototypes from the Evolving profile library. New prototypes are added and existing prototypes are removed.

2.1. The Evabcd Approach

Antonio et.al [1] reported that distribution can be created from the data output stream, it is processed by the classifier. The structure of this classifier includes,
1. Classify the new sample in the library represented by a prototype.
2. Calculate the potential of the new data sample that is added.
3. Update all the prototypes comparing with the new prototype added. It is done because the density of the data space surrounding certain data sample changes. Insert the new data sample as a new prototype if needed.
4. Remove already existing prototype.
This approach uses cosine distance formula to measure similarity between two samples.

$$\text{cosDist}(x_k, x_p) = 1 - \frac{\sum_{j=1}^{n} x_{kj}, x_{pj}}{\sqrt{\sum_{j=1.}^{n} x_{kj}^2 \sum_{j=1.}^{n} x_{kj}^2}}$$

where xk and xp represents two samples to measure the distance and n is the number of different attributes in two samples.

III. OUR CONTRIBUTION

Introduces the proposed approach for clustering, classifier design, and classifies the behavior profiles of user that includes concept based and history based search facility. The History based search enable the users to navigate frequently accessed pages easily. It updates the profile of the user automatically when his/her preference of search changes. And also it deletes the unused prototypes for long time and updates with new prototypes in the evolving profile library.

3.1. Search Engine Development

Search engine is mainly used for improving the visibility of a web page. More frequently a web site appear in the hit lists, then more visitors will be navigating that page [6]. Most web sites are integrating search engine technologies to add functionality and quick navigation. The home page of the search engine contains option for new user to register and to view about the details of the search engine personalization. The custom search engine development, would pull data from the users

specific search criteria. The complexity of the search engine is associated with the amount of data and the method in which it is stored and retrieved according to the user's query.

3.2. Administrator Process

In this administrator process the administrator or authorized user adds dataset or web page details in the server data base. The administrator add web site details, all details should be valid format like title should contain the strings, numbers, special symbols. URL is in valid format and primary enrollment, decryption and tags. Administrator will be able view the profile of the user and such that able to detect any abnormalities in the user behavior. The administrator will authorize the user by validating their user name and password.

3.3. Profile Creation

A user profile (user profile, or simply profile when used in-context) is a collection of personal data associated to a specific user. A profile refers to the digital representation of a person's identity and behavior [7]. A user profile can also be considered as the representation of a user model. The creation of a user profile from command line interface should be considered as the consecutive order of the commands typed by the user. This aspect motivates the idea of automated sequence learning for computer user behavior classification by using EVABCD approach. If the features that influence the behavior of a user not known, then historical behavior of the user can be taken into account.

The normal search engine personalization techniques does not capture the new patterns that could appear in the data stream once the classifier is built. The information typed by the user tends to be short and that used to characterize the user's interest in the corresponding user's profile. Once the trie is created, the subsequences that characterize the user profile and its relevance are calculated by traversing the trie[2]. Trie data structure is used to store the commands typed by the user. For this purpose, frequency-based methods are used. In particular, EVABCD is used to evaluate the relevance of a subsequence, its relative frequency is calculated. In this case, the support is defined as the ratio of the number of times the subsequence has been inserted into the trie and the total number of subsequences of equal size inserted. User can be able to view his/her history of search and edit his profile if needed. New user must register their details and get the user account.

3.4. Profile Updating

Based on the command typed by the user, the profile of the user will be updated automatically by evolving classifier[5]. The incremental learning algorithm used satisfies the following criterias,

1. It should be able to learn additional information from new commands typed by the user.
2. If any access to the original data already exists, then it should train the existing classifier.
3. It should preserve previously acquired knowledge.
4. It should be able to accommodate new classes that may be introduced with new data.

The sample spread is determined based on the scatterred data. The equation to get the spread of the kth data sample is defined as,

$$\sigma_i(k) = \sqrt{1/k \sum_{k=0}^{n} \text{cosDist}(\text{Proti}, \text{zk})}$$

Where $\sigma i(0) = 1$

To update recursively,

$$\sigma_i(k) = \sqrt{[\sigma_i(k\text{-}1)]^2 + 1/k \ \text{cosDist}(\text{Prot}_i, z_k) - [\sigma_i(k\text{-}1)]^2}$$

where k is the number of data samples inserted.

Once the corresponding distribution has been extracted from the data output stream, then it is processed by the evolving classifier approach.

3.5. Concept Based Search Engine Process

One criticism of search engines is that when queries are issued, most return the same results to users. In fact, the majority of queries to search engines are short and ambiguous and will not express the user's real interest. Different users may have completely different information needs and goals when using precisely the same query. For example, a person may query "cookie" to get information about

snacks or junk foods, while programmers use the same query to find information about computer programs[1]. When such a query is issued, search engines will return a list of documents that mix different search results. User profiling is a fundamental component of any personalization applications. Develop a search engine which is used to search under user preference i.e. wanted and unwanted preference [9]. Initially the user/client get the user registration in that user must specify the interested preferences according to that the search results must be positive to the user. The interested preference must be editable. In earlier personalization techniques, the search interest of the user should be manually updated but by using evolving classifier approach that will be able to update the interest of the user automatically when the search interest of the user changes. It is based on concept based method by extracting concepts from web snippets. Letter pair similarity algorithm is used to find similarity between the commands typed by the user and the snippets in the search results. Only close proximity results are retrieved and presented to the user by setting threshold limit for similarity existence.

3.6. History Based Search Engine Process

A framework is proposed that enables large-scale evaluation of personalized search. In this framework, a click through a data is recorded in search engine logs to simulate user experiences in Web search [11]. In general, when a user submits a query, the user u checks the search result list from top to bottom. The user clicks one or more documents that look relevant and skips those documents that the user is not interested in. This personalization method along with the history based search can re rank relevant documents for a user higher in results list, then user would be able to navigate frequently accessed pages more easily. Therefore, user clicks are utilized as relevant judgments to evaluate search accuracy. Since click-through data can be collected at low cost, it is possible to do large-scale evaluation under this framework. Thus in history based search, the preference will be given for the documents that user clicks**.** It is based on number of times particular document clicked by the user. It estimates the user's document preferences.

IV. RESULTS AND DISCUSSION

The output design was done so that results of processing could be communicated to the users. Computer output is the most important and direct source of information to the user. Good and effective output design will improve the systems relationships with the user. Output requirements are designed during system analysis. For efficient output design, the System Flow Diagram (SFD) can be used. Human factors reduce issues for design involves addressing internal controls to ensure readability. In View link**,** According to user search the contents and the link must displayed fewer than two preferences called positive and negative. The profiles that capture both user's positive and negative results reflects the user's behavior and provides best personalization technique. It is designed in such a way that whenever the user logs into his/her account, the current interest of the user will be displayed such that it will be easy for the user to follow and it will be changed automatically whenever their search interest changes.

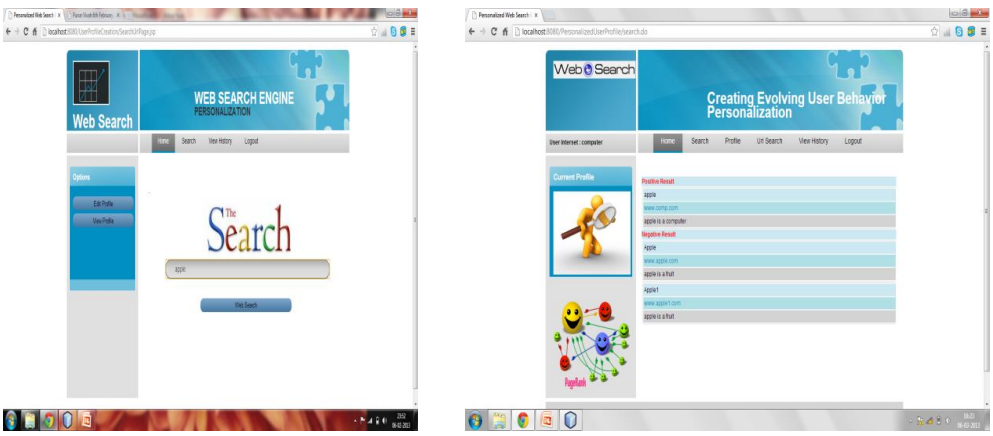

Figure 1. Concept based search engine process

V. CONCLUSION AND FUTURE WORK

An important result from the experiments is that profiles with negative preferences can increase the separation between similar and dissimilar queries. And also search results retrieved based on history based search to help user to navigate the pages easily. A generic approach, EVABCD, to model and classify user behaviors from a sequence of events. EVABCD is recursive, and it can be used in an interactive mode; therefore, it is computationally efficient and fast in updating the profile of the user. In addition, its structure is simple and interpretable. This personalization technique can also be used to monitor and detect abnormalities based on a time-varying behavior of same users and to detect masqueraders [8]. The performance analysis shows that EVABCD along with proposed features is highly efficient to adapt to the evolving user profiles and return better search results according user's interest. In order to have fast retrieval of search results than prescribed above, the classifier can be further studied and improved in future.

REFERENCES

[1]. Agapito Ledezma, Araceli Sanchis, Jose Antonio Iglesias, Plamen Angelov (2012), 'Creating Evolving User Behavior Profiles Automatically', Ieee Transactions on knowledge engineering, VOL. 24, no. 5.
[2]. Alaniz-Macedo A., Camacho-Guerrero J.A., Graca-Pimentel M. and Truong K.N. (2003), 'Automatically Sharing Web Experiences through a Hyperdocument Recommender System', Proc. ACM Conf. Hypertext and Hypermedia (HYPERTEXT '03), pp. 48-56.
[3]. Amandi A. and Godoy D. (2005), 'User Profiling in Personal Information Agents: A Survey', Knowledge Eng. Rev., vol. 20, no. 4, pp. 329-361.
[4]. Angelov P. and Zhou.X (2008), 'Evolving Fuzzy Rule-Based Classifiers from Data Streams' IEEE Trans. Fuzzy Systems: Special Issue on Evolving Fuzzy Systems, vol. 16, no. 6, pp. 1462-1475.
[5]. Branch J.W, Breimer E., Coull S.E and Szymanski.B.K. (2003), 'Intrusion Detection: A Bioinformatics Approach', Proc. Ann. Computer Security Applications Conf. (ACSAC), pp. 24-33.
[6]. Chai A., Chieu H.L.(2002), 'Bayesian Online Classifiers for Text Classification and Filtering', Proc. Int'l Conf.Research and Development in Information Retrieval (SIGIR), pp. 97-104, 2002.
[7]. Fredkin E. (1960), 'Trie Memory', Comm. ACM, vol. 3, no. 9, pp. 490-499
[8]. Gong W., Hu J., Pepyne D.L. (2009), 'User Profiling for Computer Security' Proc. Am. Control Conf., pp. 982-987.
[9]. Iglesias J.A., Ledezma A., and Sanchis A.(2007), 'Sequence Classification Using Statistical Pattern Recognition', Proc. Int'l Conf. Intelligent Data Analysis (IDA), pp. 207-218.
[10]. Iglesias J.A., Ledezma A., and Sanchis A. (2009), 'Creating User Profiles from a Command-Line Interface: A Statistical Approach', Proc. Int'l Conf. User Modeling, Adaptation, and Personalization (UMAP), pp. 90-101.

AUTHORS BIOGRAPHY

S.Sadesh is currently working as a Assistant professor in the Department of Computer Science and Engineering, Velalar College of Engineering & Technology, Erode, Tamilnadu, India. He has completed his B.E and M.E. degree in Computer Science and Engineering from Velalar College of Engineering & Technology, Erode, Tamilnadu, India. He is currently doing his PhD degree in Anna University, Chennai, Tamilnadu, India. His research interests include Data warehousing, Data mining and Web mining.

R. C. Suganthe is currently working as a Professor in the Department of Computer Science and Engineering, Kongu Engineering College, Perundurai, Tamilnadu, India. She has completed her Ph.D degree in the year 2010 in the area of Adhoc networks. She have published 8 research articles in National/International Journals and 30 research papers in national /international conferences. She has guiding 7 PhD scholars in the areas like Data mining, Image Processing and Networking.

S. Subathra, received B.Tech in Information Technology in 2011 from Kongu Engineering College Perundurai, Tamil Nadu, India. Currently pursuing M.E degree in Computer Science and Engineering in velalar college of Engineering and Technology, Erode,Tamil Nadu, India.