

## A Novel Method for Creating and Recognizing User Behavior Profiles

Boyilla Sudheer<sup>1</sup>, Syed Sadat Ali<sup>2</sup>

<sup>1</sup>M.Tech, Nimra College of Engineering & Technology, Vijayawada, A.P., India.

<sup>2</sup>Assoc.Professor & Head, Dept.of CSE, Nimra College of Engineering & Technology, Vijayawada, A.P., India.

**ABSTRACT:** User profile can be defined as the description of the user interests, behaviors, characteristics, and preferences. User profiling is the practice of gathering, organizing, and interpreting the users profile information. In this paper, we propose an adaptive approach for creating user behavior profiles and recognizing computer users. We call this approach Evolving Agent behavior Classification based on Distributions of relevant events (EVABCD) and it is based on representing the observed behavior of a computer agent as an adaptive distribution of his/her relevant atomic behaviors (events). Once the model has been created, EVABCD presents an evolving method for updating and evolving the user profiles and classifying an observed user. The approach we present is generalizable to all kinds of computer user behaviors represented by a sequence of events.

**Keywords:** Classifier, EVABCD, Profiles, Trie.

### I. INTRODUCTION

A web search engine is used to search and retrieve required information from WWW and FTP servers. The needed search results that are retrieved are presented in a list of results. This information may consist of web pages, audio, video, images and other types of files [1]. Some search engines also mine the data available in various databases. Unlike web directories that maintained by human, search engines operate algorithmically or are a mixture of several algorithms and human input. Most search engines return the same results for the same query, regardless of the user's interest in that area. Since queries submitted to search engines are very short and they will not express the user's precise needs. A good user profiling strategy is a fundamental component in the search engine personalization. Search engine personalization is the act of gathering and interpreting the users profile information. Most personalization methods is based on the creation of one single profile for a user and then user have to specify his search interest in that and based on that interest ,the results will be retrieved. If his/her 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 different queries [2]. For example, a user who prefers information about a fruit on the query "apple" may not prefer the information about Apple Computer for the query "apple." Personalization strategies should be done such that it is based on user's interest, the result should be obtained. Based on the changing behavior of the user, their 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 a computer agent as an adaptive distribution of her/his relevant atomic behaviors [3].

This paper is organized as follows: Section II provides as overview of background work. The main improvement and the development of the personalized automatic updation of the user profile are provided in Section III. Section IV describes the conclusion.

### II. BACKGROUND WORK

Various approaches have been proposed as literature point of view that the user profile usually changes to recognize behavior of others in real-time. To predict, to coordinate, and to recognize human brain capacity for future actions. Different methods have been used to find out the relevant information in computer user behavior in different computer areas:

#### A. Discovery of navigation patterns

In [4], the authors present the Web Utilization Miner (WUM), a mining system for discovering interesting navigation patterns in website. WUM prepares the web log data for mining process and the language MINT mining the aggregated data according to the directives of the human expert [5].

#### B. Web recommender systems

In [6], the authors propose a system (WebMemex) that provides recommended information based on the captured history of navigation from a list of known users. WebMemex captures information such as IP addresses, user Ids and the URL accessed for future analysis.

#### C. Web page filtering

In [7], the authors present a technique to generate readable user profiles that accurately capture interests by observing their behavior on the Web. The proposed method is built on the Web Document Conceptual Clustering algorithm, with which profiles without an a priori knowledge of user interest categories can be acquired.

#### D. Computer security

In [8], the authors describe a method using queuing theory and logistic regression modeling methods for profiling computer users based on simple temporal aspects of their behavior.

### III. PROPOSED WORK

In this paper, we propose a novel method for creating behavior profiles and recognizing computer users. We call this method Evolving Agent behavior Classification based on Distributions of relevant events (EVABCD) and it is based on representing the observed behavior of a computer agent as an adaptive distribution of his/her relevant atomic events. The goal of our work in the UNIX environment can be divided into two phases:

- Creating and updating user behavior profiles from the commands the users typed in a UNIX shell.
- Classifying a new sequence of UNIX commands into the predefined profiles.

The proposed work includes at each step the following two main actions:

- Creating the user behavior profiles and evolving the classifier
- User classification

#### Construction of the user behavior profile

In order to construct a user behavior profile in online mode from the data stream, extract an ordered sequence of acknowledged events. Commands are inherently in order, it is considered in the modelling process. According to this aspect, in order to get the most representative set of subsequence's from a sequence, we propose the use of a tree data structure called trie. Building of a user profile from a single sequence of commands is done by a 3 step process:

- Segmentation of the sequence of commands.
- Storage of the subsequence's in a tree.
- Creation of the user profile.

#### B. Segmentation of commands and Trie creation

First, the sequence is divided into subsequences of equal length from the first to the last element. Thus, the sequence  $A = A_1A_2 \dots A_n$  (where  $n$  is the number of unix commands of the sequence) will be segmented in the subsequences described by  $A_i \dots A_{i+length} \forall i, i=[1, n-length+1]$ , where length is the size of the subsequences created. The subsequences of unix commands are stored in a trie data structure. When a new model needs to be constructed, we create an empty trie, and insert each subsequence of behaviors into it, such that all possible subsequences are accessible and explicitly represented. Every trie node represents an event appearing at the end of a subsequence, and the trie nodes children represent the events that have appeared following this event. Also, each trie node keeps track of the number of times a command has been recorded into it. When a new subsequence is inserted into the trie, the existing nodes are modified and/or new nodes are created. Considering the example, the first subsequence ( $\{ls \rightarrow date \rightarrow ls\}$ ) is added as the first branch of the empty trie (Figure 1a). Each node is labeled with the number 1 which indicates that the command has been inserted in the node once (in Figure 1, this number is enclosed in square brackets). Then, the suffixes of the subsequence ( $\{date \rightarrow ls\}$  and  $\{ls\}$ ) are also inserted (Figure 1b). Finally, after inserting the three subsequences and its corresponding suffixes, the completed trie is obtained (Figure 1c).

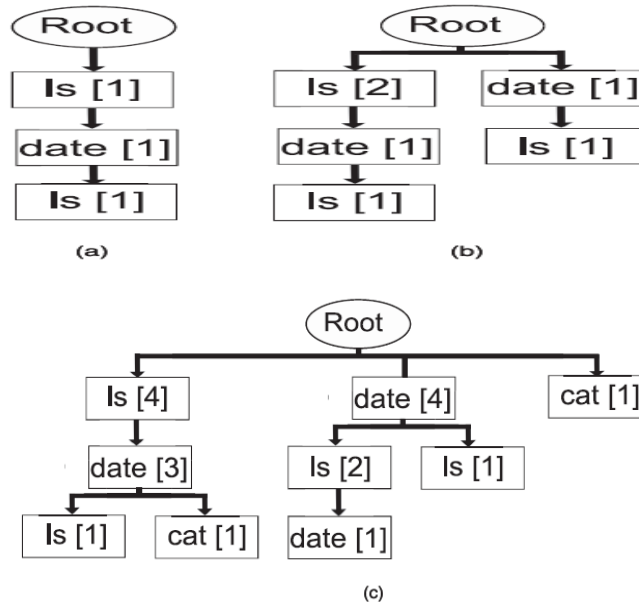


Figure1: Steps of creating an example trie

#### C. Creation of the User Profile

Once the trie structure is created, the subsequences that characterize the user profile and its relevance are calculated by traversing the trie. For this purpose, frequency-based methods are often used. In particular, in EVABCD, to evaluate the relevance of the subsequence, its relative frequency or support is calculated. In this step, the trie structure can be transformed into a set of subsequences labeled by its support value. In EVABCD, this set of subsequences is represented as the distribution of relevant subsequences. Thus, we assume that user behavior profiles are  $n$ -dimensional matrices, where each dimension of the matrix will represent a particular subsequence of commands. Once a user behavior profile has been created, it is then classified and used to update the Evolving-Profile-Library.

### D. Evolving UNIX User classifier

A classifier is defined as a mapping from the feature space to the class label space. In the proposed classifier, the feature space is defined by the distributions of subsequences of events. Thus, a distribution in the class label space represents a specific behavior which is one of the prototypes of the EPLib. EVABCD receives observations in real time from the environment to analyze. In our case, these observations are UNIX commands and they are converted into the corresponding distribution of the subsequences online. In order to classify a UNIX user behavior, these distributions must be represented in a data space. Figure 2 explains graphically this novel idea. In this example, the distribution of the first user consists of five subsequences of commands (ls, ls-date, date, cat, and date-cat); therefore, we need a 5-dimensional data space to represent this distribution because each different subsequence is represented by one dimension. If we consider the second user, we can see that 3 of the five previous subsequences have not been typed by this user (ls-date, date, and date-cat), so these values are not available. To sum up, the dimensions of the data space represent the different subsequences typed by the computer users and they will increase according to the different new subsequences obtained.

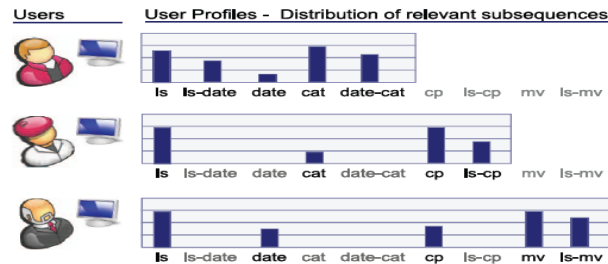


Figure 2: Distributions of subsequences of events in an evolving system approach

### E. Structure of the EVABCD

Once the corresponding distribution has been created from the online stream, it is processed by the classifier. The structure of this classifier includes the following:

1. Classify the new sample in a class represented by a prototype.
2. Calculate the potential of the new data sample to be a prototype.
3. Update all the prototypes considering the new data sample. It is done because the density of the data space surrounding certain data sample changes with the insertion of each new data sample. Insert the new data sample as a new prototype if needed.
4. Remove any prototype if needed.

## IV. CONCLUSION

An important result from the experiments is that user 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 web pages easily. Our proposed method, EVABCD, to model and classify user behaviors from a sequence of events. EVABCD is recursive, and it can be used in an interactive mode; thus, it is computationally efficient and fast in updating the profile of the user. In addition, its structure is simple as well as interpretable. This personalization technique can also be used to monitor and also to detect abnormalities based on a time-varying behaviour of same users and to detect masquerades.

## REFERENCES

- [1] 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.
- [2] Amandi A. and Godoy D. (2005), 'User Profiling in Personal Information Agents: A Survey', Knowledge Eng. Rev., vol. 20, no. 4, pp. 329-361.
- [3] Gong W., Hu J., Pepyne D.L. (2009), 'User Profiling for Computer Security' Proc. Am. Control Conf., pp. 982-987.
- [4] Spiliopoulou, M., Faulstich, L.C.: Wum: "A web utilization miner," In: Proceedings of EDBT Workshop WebDB 1998, pp. 109-115. Springer, Heidelberg (1998).
- [5] Wexelblat, A.: "An environment for aiding information browsing tasks." In: Proc. Of AAAI Spring Symposium on Acquisition, Learning and Demonstration: Automating Tasks for Users. AAAI Press, Menlo Park (1996).
- [6] Macedo, A.A., Truong, K.N., Camacho-Guerrero, J.A., da GraC,a Pimentel, M.: "Automatically sharing web experiences through a hyperdocument recommender system," In: HYPERTEXT 2003, pp. 48-56. ACM, New York (2003).
- [7] Godoy, D., Amandi, A.: "User profiling for web page filtering," IEEE Internet Computing 9(4), 56-64 (2005).
- [8] Pepyne, D.L., Hu, J., Gong, W.: "User profiling for computer security," In: Proceedings of the American Control Conference, pp. 982-987 (2004).