Mining Evolving User Profiles in Noisy Web Clickstream Data with a Scalable Immune System Clustering Algorithm

Olfa Nasraoui, Cesar Cardona, Carlos Rojas,
Department of Electrical
& Computer Engineering
The University of Memphis
206 Engineering Science Bldg.
Memphis, TN 38152
email: {onasraou,ccardona,crojas}@memphis.edu

Fabio Gonzalez
Department of Systems and Industrial Engineering
National University of Colombia
Bogota, Colombia
email: fgonza@ing.unal.edu.co

ABSTRACT

Web usage mining has recently attracted attention as a viable framework for extracting useful access pattern information, such as user profiles, from massive amounts of Web log data for the purpose of Web site personalization and organization. These efforts have relied mainly on clustering or association rule discovery as the enabling data mining technologies. Typically, data mining has to be completely re-applied periodically and offline on newly generated Web server logs in order to keep the discovered knowledge up to date. In addition to difficulty to scale and adapt in the face of large data and continuously evolving patterns, most clustering techniques, such as the majority of KMeans variants, also suffer from one or more of the following limitations: requirement of the specification of the correct number of clusters/profiles in advance, sensitivity to initialization, sensitivity to the presence of noise and outliers in the data, and unsuitability for sparse data sets. Hence, there is a crucial need for scalable, noise insensitive, initialization independent techniques that can continuously discover possibly evolving Web user profiles without any stoppages or reconfigurations.

In this paper, we propose a new scalable clustering methodology that gleams inspiration from the natural immune system to be able to continuously learn and adapt to new incoming patterns. The Web server plays the role of the human body, and the incoming requests play the role of foreign antigens/bacteria/viruses that need to be detected by the proposed immune based clustering technique. Hence, our clustering algorithm plays the role of the cognitive agent of an artificial immune system, whose goal is to continuously perform an intelligent organization of the incoming noisy data into clusters. Our approach exhibits superior learning abilities, while at the same time, requiring *modest* memory and computational costs. Like the natural immune system, the strongest advantage of immune based learning compared to current approaches is expected to be its ease of adaptation to the dynamic environment that characterizes several applications, particularly in mining data streams. We illustrate the ability of the proposed approach in mining user profiles from Web clickstream data in a single pass under different usage trend sequencing scenarios.

KEY WORDS

artificial immune systems, clustering, web usage mining, web personalization

1. Introduction

1.1 Mining the Web for User Profiles

Recently, data mining techniques have been applied to extract usage patterns from Web log data [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]. In [6, 8], we have proposed new robust and fuzzy relational clustering techniques that allow Web usage clusters to overlap, and that can detect and handle outliers in the data set. A new subjective similarity measure between two Web sessions, that captures the organization of a Web site, was also presented as well as a new mathematical model for “robust” Web user profiles [6] and quantitative evaluation means for their validation. Unfortunately, the computation of a huge relation matrix added a heavy computational and storage burden to the clustering process.

In [9], we presented a *quasi-linear* complexity technique, called Hierarchical Unsupervised Niche Clustering (H-UNC), for mining both user profile clusters and URL associations in a *single* step. More recently, we have presented a new approach to mining user profiles that is inspired by concepts from the natural immune system [13]. This approach proved to be successful in mining clusters and frequent itemsets from large web session data. This kind of data, which is extremely sparse, presents a real challenge to conventional clustering and frequent itemset mining techniques. Many data sets share this sparsity with clickstream data: these include text data as well as a large number of transactional databases. Unfortunately, all the above methods assume that the entire preprocessed Web session data could reside in main memory. This can be a disadvantage for systems with limited main memory in case of huge web session data, since the I/O operations would have to be extensive to shuffle chunks of data in and out, and thus compromise scalability. Today’s web sites are a source of an exploding amount of clickstream data that can put the scalability of any data mining technique into question.

Moreover, the Web access patterns on a web site are very dynamic in nature, due not only to the dynamics of Web site content and structure, but also to changes in the user’s interests, and thus their navigation patterns. The access patterns can be observed to change depending on the time of day, day of week, and according to seasonal patterns or other external events in the world. As an alternative to locking the state of the

Web access patterns in a frozen state depending on when the Web log data was collected and preprocessed, we propose an approach that considers the Web usage data as a reflection of a dynamic environment which therefore requires dynamic learning of the access patterns. An intelligent Web usage mining system should be able to continuously learn in the presence of such conditions without ungraceful stoppages, reconfigurations, or restarting from scratch. In this paper, we investigate a new immune system inspired evolutionary approach based on continuously and dynamically learning evolving Web access patterns from non-stationary Web usage environments. This evolutionary computation based approach can be generalized to fit the needs of mining dynamic data or huge data sets that do not fit in main memory.

1.2 The Immune System as A Cognitive Agent

Natural organisms exhibit powerful learning and processing abilities that allow them to survive and proliferate generation after generation in ever changing and challenging environments. The natural immune system is a powerful defense system that exhibits many signs of cognitive learning and intelligence [14, 15]. In particular the acquired or adaptive immune system is comprised mainly of lymphocytes which are special types of white blood cells (*B-cells*) that detect and destroy pathogens, such as viruses and bacteria. The features that allow the identification of a particular pathogen are the cell surface and soluble proteins called *antigens*. Special *proteins receptors* on the B-cell surface, called *antibodies* react to a particular antigen by binding to this antigen. And this binding relation is specialized so that only certain antibodies can bind and hence recognize a particular antigen. Even though there are around 10^{16} antigen varieties, the immune system is armed with only 10^8 antibody types in its repertoire at any given time. Hence lymphocytes bind only approximately to pathogens, to allow the recognition of a larger number of antigens. Lymphocytes are only activated when the bond is strong enough, and this minimum strength may be different for different lymphocytes. A stronger binding with an antigen induces a lymphocyte to clone more copies of itself, hence providing reinforcement. Furthermore, to diversify their repertoire and be able to recognize more antigens, lymphocytes undergo somatic hypermutation (10^9 times compared to evolutionary mutation in other cells) that comes in three forms: point mutations, short deletions, and insertions of random gene sequences. Mature lymphocytes become part of the long term memory of the immune system, and help recognize and fight a similar antigen that may be encountered in the future. Therefore, it can be said that the immune system possesses several desirable and powerful learning abilities allowing it to perform pattern recognition and associative memory in a continuous and decentralized manner.

Several Artificial Immune System (AIS) models [16, 17, 18] have been proposed for data analysis and pattern recognition. In the immune network based model, the data set represents the set of antigens to be recognized, and the clustering model represents a repertoire of B cells that summarize the data. Each B cell is specialized to recognize a set of antigens via a similarity measure. The presence of a large number of antigens that match or activate a B cell will stimulate it and cause it to clone into multiple copies of itself. In addition, each B cell recognizes by similarity B cells in its neighborhood. Hence similar B cells co-stimulate each other just like antigens stimulate B cells. In this way, the B cells form a network of co-stimulating cells that not only form a concise summary or synopsis of the data set, but also a form of memory of antigens seen so far. Even after an antigen has disappeared, B cells that recognized it can survive in the immune network thanks to co-stimulation with neighboring B cells. B cells that are not re-stimulated by this antigen for a long period of time will eventually die out. B cells also suppress each other as a means of controlling the inevitable explosion in the B cell population size that would result from co-stimulation only. However, in order to achieve a desired learning capability (for example detecting all clusters in a dat set), current models require the *storage and manipulation* of a large network of B Cells (with a number of B Cells often exceeding the number of data points in addition to all the pairwise links between these B Cells). Hence, current AIS models are far from being scalable, which makes them of limited use, even for medium size data sets.

In this paper, we propose a new AIS learning approach for clustering, that addresses the shortcomings of current AIS models. Our approach exhibits improved learning abilities and *modest* complexity.

The rest of the paper is organized as follows. In Section 2., we outline our proposed immune based clustering model. In Section 3., we illustrate using the proposed Dynamic AIS model for mining Web clickstream data. Finally, in Section 4., we present our conclusions.

2. A Dynamic Artificial B-Cell Model based on Robust Weights and Immune Network Compression

In order to make the AIS approach approach scalable, we propose to reduce the storage and computational requirements related to the network structure.

2.1 A Dynamic Artificial B-Cell Model based on Robust Weights: The D-W-B-Cell Model

In a dynamic environment, the antigens are presented to the immune network one at a time, and the B-Cell parameters are updated with each presentation. It is more convenient to think of the antigen index, j, as monotonically increasing with time. That is, the antigens are presented in the following chronological order: $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N$. The Dynamic Weighted B-Cell (*D-W-B-cell*) represents an influence zone over the domain of discourse consisting of the training data set. However, since data is dynamic in nature, and has a temporal aspect, data that is more current will have higher influence compared to data that is less current/older. Quantitatively, the influence zone is defined in terms of a weight function that decreases not only with distance from the antigen/data location to the D-W-B-cell prototype / best exemplar, but also with the time since the antigen has been presented to the immune network. It is convenient to think of time as an additional dimension that is added to the D-W-B-Cell compared to the classical B-Cell, traditionally statically defined in antigen space only. For the i^{th} D-W-B-cell, DWB_i, we define the following weight/membership function:

$$w_{ij} = w_i\left(d_{ij}^2\right) = e^{-\left(\frac{d_{ij}^2}{2\sigma_{i,j}^2} + \frac{j}{\tau}\right)} \tag{1}$$

where τ controls the time decay rate of the contribution from old antigens, and hence how much emphasis is placed on the freshness of the current immune network compared to the sequence of antigens encountered so far. d_{ij}^2 is the distance from antigen $\mathbf{x}_j$ (which is the j^{th} antigen encountered by the immune network) to D-W-B-cell, DWB_i. A suitable distance measure should be chosen to reflect the clustering model being sought, such as Euclidean distance for hyperspherical shaped clusters, or a different measure for the case of clustering Web user sessions. $\sigma_{i,j}^2$ is a scale parameter that controls the decay rate of the weights along the spatial dimensions, and hence defines the size of an influence zone around a cluster prototype. Data samples falling far from this zone are considered outliers. The weight functions decrease exponentially with the order of presentation of an antigen, j, and therefore, will favor more current data in the learning process. The stimulation level, after J antigens have been presented to DWB_i, is defined as the density of the antigen population around DWB_i:

$$s_{i,J} = \frac{\sum_{j=1}^{J} w_{ij}}{\sigma_{i,J}^2}, \tag{2}$$

Since the time dependency has been absorbed into the weight function, the equations for scale updates are found by setting $\frac{\partial s_{i,J}}{\partial \sigma_{i,J}^2} = 0$, to obtain

$$\sigma_{i,J}^2 = \frac{\sum_{j=1}^{J} w_{ij} d_{ij}^2}{2\sum_{j=1}^{J} w_{ij}}. \tag{3}$$

For the purpose of computational efficiency, however, we convert the above equations to incremental counterparts as follows: Each term that takes part in the computation of the above measures (Equations (6) and (7) above) is updated individually with the arrival of each new antigen using the old values of each term in the numerator and each term in the denominator, and adding the contribution of the new antigen/data item to each one of these terms. This means that the cumulative numerator and denominator for stimulation and scale must be stored to be used in incremental updates as new antigens arrive. This results in the following incremental equations for stimulation and scale, after J antigens have been presented to DWB_i.

$$s_{ai,J} = \frac{e^{-\frac{1}{\tau}} W_{i,J-1} + w_{iJ}}{\sigma_{i,J}^2}, \tag{4}$$

$$\sigma_{i,J}^2 = \frac{e^{-\frac{1}{\tau}} \sigma_{i,J-1}^2 W_{i,J-1} + w_{iJ} d_{iJ}^2}{2\left(e^{-\frac{1}{\tau}} W_{i,J-1} + w_{iJ}\right)}. \tag{5}$$

where $W_{i,J-1} = \sum_{j=1}^{J-1} w_{ij}$ is the sum of the contributions from previous antigens, $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_{J-1}$, to D-W-B-Cell i.

2.2 Dynamic Stimulation and Suppression

We propose incorporating a dynamic stimulation factor, $\alpha(t)$, in the computation of the D-W-B-cell stimulation level. The static version of this factor is a classical way to simulate memory in an immune network by adding a compensation term that depends on other D-W-B-cells in the network [16, 17]. In other words, a group of intra-stimulated D-W-B-cells can self-sustain themselves in the immune network, even after the antigen that caused their creation disappears from the environment. However, we need to put a limit on the time span of this memory so that truly outdated patterns do not impose an additional superfluous (computational and storage) burden on the immune network. We propose to do this by an annealing schedule on the stimulation factor. This is done by allowing D-W-B-cells to have their own stimulation coefficient, and to have this stimulation coefficient decrease with their age). In the absence of a recent antigen that succeeds in stimulating a given subnet, the age of the D-W-B-cell increases by 1 with each antigen presented to the immune system. However, if a new antigen succeeds in stimulating a given subnet, then the age calculation is modifed by refreshing the age back to zero. This makes extremely old D-W-B-cells die gradually, if not re-stimulated by more recent relevent antigens.

Incorporating a dynamic suppression factor, $\beta(t)$, in the computation of the W-B-cell stimulation level is also a more sensible way to take into account internal interactions. The suppression factor is not related to memory management, but rather as a way to control the proliferation and redundancy of the D-W-B-cell population. This adaptive way to control the amount of redundancy to achieve the right balance between the needed memory and the useless redundancy can be achieved by some kind of annealing schedule on the suppression factor used in computing the D-W-B-cell stimulation levels. This is done by allowing D-W-B-cells to have their own suppression coefficient, and using an annealing schedule similar to stimulation.

In order to understand the combined effect of the proposed stimulation and suppression mechanism, we consider the following two extreme cases: (**i**) When there is positive suppression (competition), but no stimulation. This results in good population control and avoids redundancy. However, there is no memory, and the immune network will forget past encounters. (**ii**) When there is positive stimulation, but no suppression. Hence, there is good memory but no competition. This will cause the proliferation of the D-W-B-cell population or maximum redundancy. Hence, there is a natural tradeoff between redundancy/memory and competition/reduced costs.

2.3 Organization and Compression of the Immune Network

In order to deal with the overhead of storing and manipulating the immune network, we propose using an alternative scheme to store the network. Instead of storing all the pairwise distance values or links (and hence all internal reaction terms in the immune network), the D-W-B-cell population (with size N_B) is divided into groups or *subNetworks*, such that the members of the same group/subnet are most likely considered as close neighbors. For a given D-W-B-cell, the D-W-B-cells in the same group can be considered as most influential in determining the suppression and stimulation terms. Hence, only these D-W-B-cells are considered for the summation. This provides an efficient and accurate *intra-subnetwork interaction* model. A fast way to organize the immune network into subnetworks can be achieved by a few iterations of the K Means clustering algorithm applied to the D-W-B-cell space to divide the D-W-B-cell population into K clusters. This divide and conquer strategy can have significant impact on the number of interactions that need to be processed in the immune network. We define *external interactions* as those occuring between an antigen (external agent) and the D-W-B-cell in the immune network. We define *internal interactions* as those occuring between one D-W-B-cell and all other D-W-B-cells in the immune network. Figure 1 illustrates internal (relative to D-W-B-cell$_k$) and external interactions (caused by an external agent called "Antigen"). Note that the number of possible interactions is immense, and this is a serious bottleneck in the face of all existing immune network based learning techniques [16, 17, 18]. Suppose that the immune network is compressed by clustering the D-W-B-cells using a linear complexity scalable approach such as K Means. Then the immune network can be divided into several *subnetworks* that form a parsimonious view of the entire network. For global low resolution interactions, such as the ones between D-W-B-cells that are very different, only the *inter-subnetwork interactions* are germane. For higher resolution interactions such as the ones between similar D-W-B-cells, we can drill down inside the corresponding subnetwork and afford to consider all the *intra-subnetwork interactions*. Similarly, the external interactions can be compressed by considering interactions between the antigen and the subnetworks instead of all the D-W-B-cells in the immune network. Note that the centroid of the D-W-B-cells in a given subnetwork/cluster is used to summarize this subnetwork, and hence to compute the distance values that contribute in the internal and external interaction terms.

The savings in computation can be significant. Assuming that the network is divided into roughly K equal sized subnetworks, then the number of internal interactions in an immune network of N_B D-W-B-cells, can drop from N_B^2 in the uncompressed network, to $\left(\frac{N_B}{K}\right)^2$ *intra-subnetwork interactions* and $K-1$ *inter-subnetwork interactions* in the compressed immune network. This clearly **can approach linear complexity as** $K \rightarrow \sqrt{N_B}$. Figure 2(b) illustrates the reduced internal (relative to D-W-B-cell$_k$) interactions in a compressed immune network. Similarly the number of external interactions relative to each antigen can drop from N_B in the uncompressed network to K in the compressed network. Figure 2(a) illustrates the reduced external (relative to external agent "Antigen") interactions. Furthermore, the compression rate can be modulated by choosing the appropriate number of clusters, $K \approx \sqrt{N_B}$, when clustering the D-W-B-cell population, to maintain linear complexity, $O(N_B)$.

Sufficient summary statistics for each cluster of D-W-B-cells are computed, and can later be used as approximations in lieu of repeating the computation of the entire suppression/stimulation sum. The summary statistics are in the form of average dissimilarity within the group, cardinality of the group (number of D-W-B-cells in the group), and density of the group. This approach can be seen as forming sub-networks of the immune network with sufficient summary statistics to be used in evolving the entire immune network.

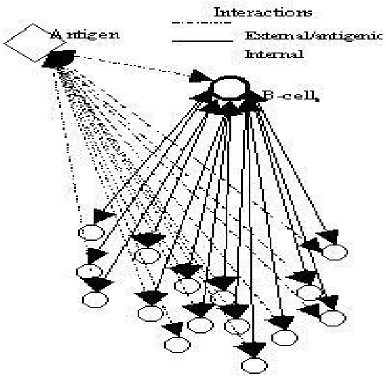
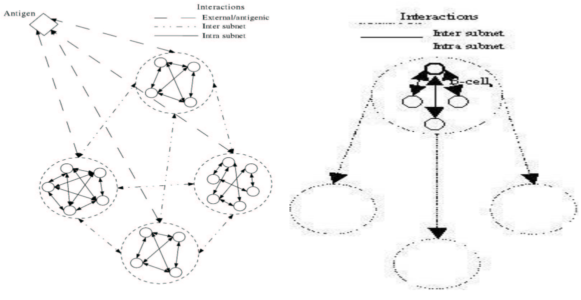


Figure 1. Immune network interactions without compression



(a) (b)

Figure 2. Immune network interactions with compression: (a) Internal and External interactions , (b) Internal interactions

2.4 Effect of the Network Compression and Dynamic Stimulation and Suppression on Interaction Terms

The D-W-B-cell specific computations can be replaced by subnet computations in a compressed immune network. Instead of taking into account all possible $(N_B)^2$ interactions between all N_B B-cells in the immune network, only the intra-subnetwork interactions with the N_B^j B-cells inside the nearest subnetwork are taken into account. For even more accuracy, inter-subnetwork interactions may also be taken into account. In this case only one interaction term per subnetwork will result in at most K interaction terms, one with each subnetwork representative B-cell. In case K-Means is used, this representative as well as the organization of the network into subnetworks is a by-product. For more complex data structures, a reasonable representative prototype (such as a medoid) can be chosen. Taking these modifications into account, the stimulation and scale values that take advantage of the compressed network are given by

$$s_i = s_{i,J} + \alpha\left(t\right) \frac{\sum_{l=1}^{N_B^j} w_{il}}{\sigma_i^2} - \beta\left(t\right) \frac{\sum_{l=1}^{N_B^j} w_{il}}{\sigma_i^2}, \qquad (6)$$

where $s_{i,J}$ is the pure antigen stimulation after encountering J antigens, given by (4) for D-W-B-cell$_i$; and N_B^j is the number of B-cells in the subnetwork that is closest to the j^{th} antigen. This will modify the D-W-B-cell scale update equations to become

$$\sigma_i^2 = \frac{1}{2} \frac{\sum_{j=1}^{N} w_{ij} d_{ij}^2 + \alpha\left(t\right) \sum_{l=1}^{N_B^j} w_{il} d_{il}^2 - \beta\left(t\right) \sum_{l=1}^{N_B^j} w_{il} d_{il}^2}{\sum_{j=1}^{N} w_{ij} + \alpha\left(t\right) \sum_{l=1}^{N_B^j} w_{il} - \beta\left(t\right) \sum_{l=1}^{N_B^j} w_{il}}. \qquad (7)$$

Note that the centroid of the D-W-B-cells in a given subnetwork is used to summarize this subnetwork, and hence to compute the distance values that contribute in the internal intra-subnetwork and external interaction terms.

2.5 Cloning in the Dynamic Immune System

The D-W-B-cells are cloned in proportion to their stimulation levels relative to the average stimulation in the immune network. However, to avoid preliminary proliferation of good B-Cells, and to encourage a diverse repertoire, new B-Cells do not clone before they are mature (their age, t_i exceeds a lower limit t_{min}). They are also not removed from the immune network regardless of their stimulation level. Similarly, B-cells with age $t_i > t_{max}$ are frozen, or prevented from cloning, to give a fair chance to newer B-Cells. This means that

$$N_{clones_i} = K_{clone} \frac{s_i}{\sum_{k=1}^{N_B} s_k} \text{ if } t_{min} \leq t_i \leq t_{max}, \qquad (8)$$

where K_{clone} is a constant multiplier that moderates the intensity of cloning. After cloning, B-cells may undergo mutation. For this purpose, a z-score is generated randomly from a standard normal distribution. A multiple of this number (we used 1.5) is rounded, resulting in the number of URLs N_{mut} that will be affected by mutation. Finally N_{mut} URLs are picked randomly from the set of all URLs. A selected URL is inserted in the B-cell profile if its not already present, and either deleted from the B-cell profile if it is present.

2.6 Learning New Antigens and Relation to Outlier Detection

Somatic hypermutation is a powerfull natural exploration mechanism in the immune system, that allows it to learn how to respond to new antigens that have never been seen before. However, from a *computational* point of view, this is a very costly and inefficient operation since its complexity is exponential in the number of features. Therefore, we model this operation in the artificial immune system model by an instant antigen duplication whenever an antigen is encountered that fails to activate the entire immune network. A new antigen, $\mathbf{x}_j$ is said to activate the i^{th} B-Cell, if its contribution to this B-Cell, w_{ij} exceeds a minimum threshold w_{min}. Hence, an antigen that fails to activate the immune network is one, such that $w_{ij} < w_{min}$, $\forall i = 1, \cdots, N_B$. Such an antigen can be considered as a potential outlier. However, in a continuous stream learning environment, such an antigen may also represent the onset of an emerging pattern to be learned, if more similar antigens are subsequently encountered. In this case, the dynamnics of the immune learning will cause the creation of a new set of specialized B-Cells, just like the previously encountered patterns. If this antigen happens to be an isolated case, then its specialized B-cells will gradually die out by lack of stimulation. When this occurs, the defunct specialized B-cell can be considered as an outlier representative, and it can either be discarded or moved to secondary storage. To further take advantage of the compressed immune network, only the N_B^j D-W-B-cells within the closest subnetwork (say the i^{th} subnetwork) to the current antigen are considered. This results in activation iff. $w_{ij} < w_{min}$, $\forall i = 1, \cdots, N_B^j$. We refer to the new antigen duplication operation, as *dendritic injection*, since it mimics the action of dendritic cells that teach the immune system new antigens that have never been seen before.

2.7 Proposed Scalable Immune Learning Algorithm For Clustering Evolving Data

Scalable Immune Based Clustering for Evolving Data: (optional steps are enclosed in [])

```
Fix the maximal population size $N_B$;
Initialize D-W-B-cell population and $\sigma_i^2 = \sigma_{init}$ using the first batch of the input antigens/data;
Compress immune network into $K$ subnets using 2-3 iterations of K Means;
Repeat for each incoming antigen $\mathbf{x}_j$ {
    Present antigen to each subnet centroid, $\mathbf{C}_k, k = 1, \cdots, K$ in network : Compute distance, activation weight, $w_{kj}$ and update $\sigma_k^2$ incrementally using (5);
    Determine the most activated subnet;
    IF antigen activates closest subnet Then {
        Present antigen to each D-W-B-cell, $D - W - B - cell_i$, in closest immune subnet;
        Refresh this D-W-B-cell's age ($t = 0$) and update $w_{ij}$ using (1);
        [Update the compressed immune network subnets incrementally;]
    }
    [ELSE IF antigen activates any subnetwork in secondary storage (long term memory) THEN ]
        [Retrieve this old subnetwork into active memory (becomes part of the current immune network); ]
    IF All B-cells in most activated subnet have $w_{ij} < w_{min}$ (antigen does not activate subnet) THEN{
        Create by dendritic injection a new D-W-B-cell = $\mathbf{x}_j$ and $\sigma_i^2 = \sigma_{initial}$;
        [Flag this antigen as potential outlier that can grow into a new trend;]
    }
    Repeat for each D-W-B-cell$_i$ in closest subnet only {
        Increment age (t) for D-W-B-cell$_i$;
        Compute D-W-B-cell$_i$'s stimulation level using (6);
        Update D-W-B-cell$_i$'s $\sigma_i^2$ using (7);
    }
    Clone and mutate D-W-B-cells;
    IF population size $> N_B$ Then
        Kill worst excess D-W-B-cells, or leave only subnetwork representatives (single cell prototype) of oldest/mature subnetworks in main memory and move oldest/mature subnetwork D-W-B-Cells to secondary (long term) storage;
    Compress immune network periodically (after every $T$ antigens), into $K$ subnets using 2-3 iterations of K Means with (1 - cosine similarity) as distance measure, and the previous centroids as initial centroids;
}
```

3. Application to Mining User Profiles from Web Clickstream Data

3.1 General Framework

An intelligent Web usage mining system should be able to continuously learn evolving usage trends without ungraceful stoppages, reconfigurations, or restarting from scratch. This can be accomplished using the proposed Dynamic Immume Learning approach to clustering and tracking dense evolving patterns in massive changing data sets. The proposed algorithm for mining Web user profiles from clickstream data can be summarized as follows:

Table 1. Summary of 20 usage trends previously discovered using Hierarchical Unsupervised Niche Clustering [9] (only URLs with top 3 to 4 relevance weights shown in each profile)

i	$\|P_{T\,i}\|$	$P_{T\,i}$
0	106	{0.99 - /people_index.html}, {0.98 - /people.html}, {0.97 - /faculty.html}
1	104	{0.99 - /}, {1.00 - /cecs_computer.class}
2	177	{0.90 - /courses_index.html}, {0.88 - /courses100.html}, {0.87 - /courses.html} , {0.81 - /}
3	61	{0.80 - /}, {0.48 - /degrees.html}, {0.23 - /degrees_grad.html}
4	58	{0.97 - /degrees_undergrad.html}, {0.97 - /bsce.html}, {0.95 - /degrees_index.html}
5	50	{0.56 - /faculty/springer.html}, {0.38 - /faculty/palani.html}
6	116	{0.91 - /~saab/cecs333/private}, {0.78 - /~saab/cecs333}
7	51	{0.90 - /~saab/cecs303}, {0.84 - /~saab/cecs303/private}
8	134	{0.74 - /~joshi/courses/cecs352}, {0.35 - /~joshi/courses/cecs352/slidesindex.html}
9	41	{0.51 - / joshi/courses/cecs438}, {0.37 - / joshi/courses/cecs438/proj.html}
10	95	{0.48 - /~joshi}, {0.16 - /~joshi/sciag}
11	185	{0.84 - /~c697168/cecs227}, {0.74 - /~c697168/cecs227/left.html}, {0.73 - /~c697168/cecs227/head.html}
12	74	{0.57 - /~shi/cecs345}, {0.45 - /~shi/cecs345/java_examples}, {0.46 - /~shi/cecs345/Lectures/07.html}
13	38	{0.82 - /~shi/cecs345}, {0.47 - /~shi}, {0.34 - /~shi/cecs345/references.html}
14	33	{0.55 - /~shi/cecs345}, {0.55 - /~shi/cecs345/java_examples}, {0.33 - /~shi/cecs345/Projects/1.html}
15	51	{0.92 - /courses_index.html} , {0.90 - /courses100.html}, {0.86 - /courses.html}, {0.78 - /courses200.html}
16	77	{0.78 - /~yshang/CECS341.html} , {0.56 - /~yshang/W98CECS341}, {0.29 - /~yshang}
17	68	{0.22 - /~jiang/lana.html} , {0.19 - /~jiang/chinadoc.html}
18	65	{0.22 - /~manager/LAB/motif.html} , {0.20 - /~manager/LAB/unix.html}
19	120	{0.27 - /access} , {0.23 - /access/details.html}

(1) The Web server plays the role of the human body, and the incoming requests play the role of antigens that need to be detected,

(2) The input data is similar to web log data (a record of all files/URLs accessed by users on a Web site),

(3) The data is continuously pre-processed to produce session lists: A session list $\mathbf{s}_i$ for user i is an item list of URLs visited by the same user. In discovery mode, a session is fed to the learning system as soon as it is available,

(4) The i^{th} B-Cell, D-W-B-cell$_i$, represents the i^{th} candidate profile and encodes a list of relevant URLs. It is matched to incoming sessions using (1-Cosine similarity) as a distance measure,

(5) Each profile has its own influence zone defined by σ_i^2. This measures the average dissimilarity between the i^{th} candidate profile and the sessions/antigens that activate the i^{th} B-Cell, D-W-B-cell$_i$.

3.2 Single-Pass Mining of User Profiles from Real Web Clickstream Data

Profiles were mined continuously and in a single pass over the 12-day clickstream data (from 1998) corresponding to 1704 sessions and 343 URLs from the website of the department of Computer Engineering and Computer Science at the University of Missouri. This data set is used to benchmark the proposed algorithm against a variety of clustering techniques proposed in [6, 9]. The profiles that were discovered using the proposed scalable immune approach in a single pass are comparable to the ones previously obtained using a variety of different, but less scalable techniques [6, 9]. The maximum population size was 100, the control parameter for compression was varied between $K = 1$ and 50, and periodical compression every $T = 40$ sessions. The activation threshold was $w_{min} = 0.6$, and $\tau = 20$. In order to illustrate the *continuous* learning ability of the proposed technique, we have performed the following three simulations:

scenario 1: We used 20 profiles previously discovered using Hierarchical Unsupervised Niche Clustering (HUNC) [9], and listed in Table 1, to partition the Web sessions into 20 distinct sets of sessions, each one assigned to the closest profile. Then we presented these sessions to the immune clustering algorithm one profile at a time. That is, we first present the sessions assigned to ground truth profile/trend 0, then the sessions assigned to profile 1, $\cdots$ etc.

scenario 2: In this scenario we used the same pre-partitioned session data set as the previous scenario, but presented the profiles in reverse order. That is, we first present the sessions assigned to trend 19, then the sessions assigned to profile 18, $\cdots$ etc, ending with sessions from trend 0.

scenario 3: The Web sessions are presented in their natural chronological order exactly as they were received in real time by the web server.

In scenario 1 and 2, sessions within the same profile are not re-ordered. That is, they are presented in their natural chronological order, as logged by the server. In each scenario, we track the actual composition of the B cells in the immune network, i.e., the URLs present in the B cell. This will show how each B cell represents a detected user profile as desired. We also track the number of B cells that succeed in learning each one of the 20 ground truth profiles after each session is presented. This is computed by counting the number of hits per ground truth usage trend or profile. This number is the number of B-cells within 0.4 radius of the ground truth profile (distance is computed as the square of 1 - cosine similarity). It provides us with an evolving number of hits per profile. These hits are shown in Figures 3, 4, and 5, for the three above scenarios respectively. The y-axis is split into 20 intervals, with each interval devoted to the trend/profile number indicated by the upper value (from 0 to 19). A hit for the i^{th} profile for session No. t is shown in these figures at location (t, i).

In each scenario the sessions were presented one time only. Hence we tested the proposed immune clustering algorithm's ability to learn the user profiles in a single pass. A single pass over all 1704 Web user sessions (with non-optimized Java code) took less than 7 seconds on a 2 GHz Pentium 4 PC running on Linux. This attests to the system's ability to learn an unknown number of evolving profiles *in real time*. With an average of *4 milliseconds per user session*, the proposed profile mining system is suitable for use in a real time personalization system to constantly and continuously provide the recommendation engine with a fresh and current list of user profiles. Old profiles can be handled in a variety of ways. They may either be discarded, moved to secondary storage, or cached for possible re-emergence. Even if discarded, older profiles that re-emerge later, would be re-learned from scratch just like completely new profiles. Hence the logistics of maintaining old profiles are much less crucial with our approach than with most existing techniques.

Figure 3 exhibits an expected staircase pattern showing the gradual learning of emergent usage trends as these are experienced by the immune network in the order from trend 0 to 19. The plot shows some peculiarities, for example at trend 15 since it records a short lived hit at the same time as trend 2. Table 1 shows that trends 2 and 15 do indeed share many similarities, with the main difference being that trend 2 includes visits to the freshmen courses and the department's main page, while trend 15 represents visits to the freshmen and sophomore course pages without the main page. Typical cross reactions between similar patterns are actually desired and illustrate a certain tolerance for inexact matching that is very similar to cross reactions in detecting related viruses in the immune system. Trends 0 through 4 (general and course pages) survive for a long period due to their interaction and subsequent reinforcement with trend 15 and with each other.

Figure 4 shows an interesting inversed staircase pattern due to the sessions being presented in reverse order of the trends: from 19 to 0. Here again, cross reactions are obvious. For example, trend 13 reacts with trends 12 and 14, and its is clear how trends 1, 2, 4, and 15 interact since accesses to degree pages often overlap with accesses to course pages.

Finally Figure 5 shows the trend B-cell hits versus session number when the sessions are presented in their original chronological order corresponding to scenario 3. Here, there is no staircase pattern, since the sessions are not presented in any controlled order of usage trend. In this case the immune network simply responds to the stream of usage data that is encountered with all trends emerging without any particular order other than the natural occurence pattern.

Figure 7 shows the distribution of the input noise sessions which are sessions with low robust weight (less than 0.6) over the usage trends estimated by HUNC [9]. This plot shows that the number of sessions that are not at the core of the 20 ground truth usage trends is quite significant.

On the other hand, Figure 6 is analogous to Figure 5 in the sense that it shows the distribution of the original input sessions, but with all the noise sessions excluded. Hence Figure 6 provides a more faithful visualization of the ground truth usage trends, and when their user sessions are actually experienced by the website. These figures attest to the fact that the session data is quite noisy, and the arrival sequence and pattern of sessions belonging to the same usage trend may vary in a way that makes incremental tracking and discovery of the profiles even more challenging than in a batch style approach where the sessions can be stored in memory, and a standard iterative approach is used to mine the profiles. It also shows how some of the usage trends (e.g: No. 13, 14, 15) are not synchronized with others, and how some of the trends are weak and noisy based on the distribution of their core (non-noisy) sessions. Such weak profiles can be even more elusive to discover in a real time system such as the one proposed herein. While Figure 5 shows the B-cell distribution with time, Figure 6 shows the distribution of the input data with time. The fact that both figures show some similarity and the emergence patterns of the trends, attests to the fact that the immune network is able to form a reasonable dynamic synopsis of the usage data.

Table 3 shows the typical D-W-B-Cell profiles activated by more than 8 antigen sessions, together with their σ^2 values and age, after all 1704 sessions have been presented following scenario 2 (i.e. in reverse order of trend 19 to 0). It is clear that the σ^2 values are reasonably small for all profiles, indicating a compact trend in their usage patterns. The profiles are also ordered according to decreasing age in this table, meaning that older profiles appear closer to the top. This table shows a snapshot of the B-cell contents at session No. 1704, corresponding to the last (rightmost) point on the x-axis of Figure 4. Since the sessions are presented one trend at a time starting from trend 19, until trend 0, older B-cell profiles that are weak expire by the end of the simulation pass. Similarly, Table 4 shows the D-W-B-Cell profiles activated by more than 8 antigen sessions, after all 1704 sessions have been presented following scenario 3 (i.e. in natural order). Trend 11 from Table 1, which has most of the input sessions, gets split in several more specific profiles in Table 4. On the other hand, some of the profiles that are weak disappear by the end of the simulation pass, even though they are detected earlier as shown in Figure 5. Examples include trend 13, 14, and 15, as well as 8, 9, and 10 from Table 1 which are weak as can be verified from the amount of noise in Figure 7.

It is interesting to note that the *memory span* of the network is affected by the parameter τ which affects the rate of forgetting of B-cells in the immune network. Higher values result in slower forgetting (longer term memory). In practice, the value of τ should be calibrated to correspond with the dynamics of the website, and the desired amount of forgetting in the network. A low value will favor faster forgetting, and therefore a fresher or more current set of profiles that reflect the most recent access activity on a website, while a higher value will tend to keep older profiles in the immune network for longer periods.

Finally we note, that regardless of age, the profiles discovered using the proposed immune clustering technique compare with the ones discovered using HUNC [9] even though the immune based technique takes only one pass through the web sessions. Figure 8 shows how the number of discovered profiles varies with the network compression rate K, indicating between 12 and 17 profiles on average. Figure 9 shows the final number of subnetworks versus the prespecified compression rate K, indicating a maximum value of $K = 20$ subnetworks for the data presented in scenario 3, after which many subnetworks become depleted.

4. Conclusion

In this paper, we have introduced a new scalable artificial immune system approach to stream clustering in a noisy environ-

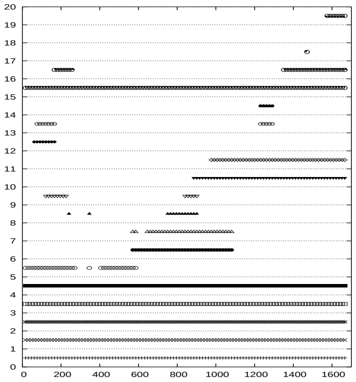


Figure 3. Hits per usage trend versus session number when sessions are presented in order of trend 0 to trend 19, $K = 10$, $\tau = 20$

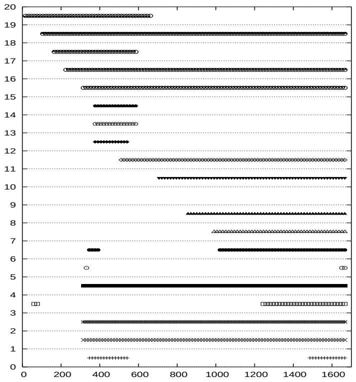


Figure 4. Hits per usage trend versus session number when sessions are presented in reverse order from trend 19 to trend 0, $K = 10$, $\tau = 20$

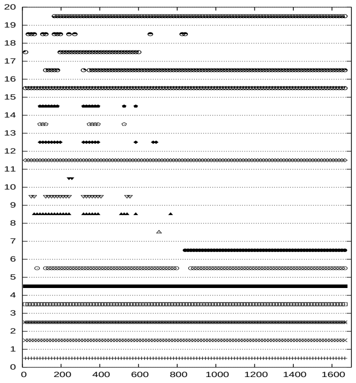


Figure 5. Hits per usage trend versus session number when all sessions are presented in natural chronological order: $K = 25$, $\tau = 20$

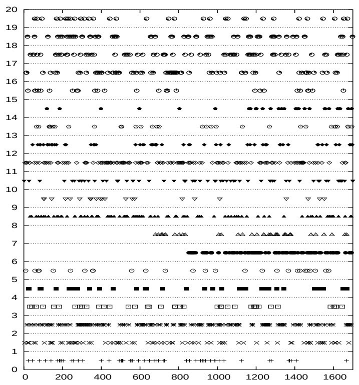


Figure 6. Distribution of input sessions over usage trend versus session number when only non-noisy ($w_{ij} > 0.6$) sessions are presented in natural chronological order: Trends 5, 9, 13, 14, 15, and 19 appear to be weakier and noisier. Also trends 6 and 7 emerge late in the 12-day access log, while trend 0 weakens in the last days.

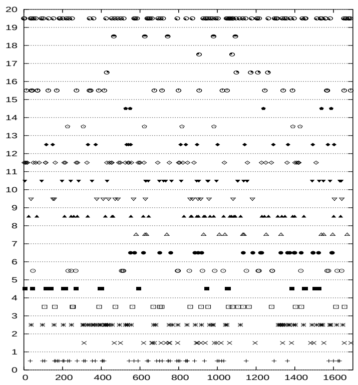


Figure 7. Distribution of input sessions over usage trends versus session number when only noisy ($w_{ij} < 0.6$) sessions are presented in natural chronological order. The union of this figure with Fig. 6 represents the *complete* set of sessions.

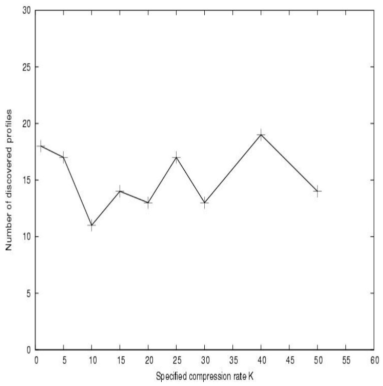


Figure 8. Number of discovered profiles versus specifed compression rate K, when sessions are presented in natural chronological order

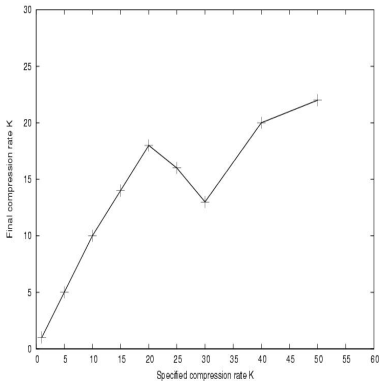


Figure 9. Final value of K (number of non empty subnetworks) versus specifed compression rate K, when sessions are presented in natural chronological order

Table 2. Comparison of proposed Immune Clustering Approach (*AIS**) with Clustering based on Other Learning Paradigms: Existing AIS (*AIS*) , Genetic Algorithms (*GA*), Hill Climbing (*HC*), and Parallel Hill Climbing (*PHC*). Search Operators are abbreviated: *cl* = cloning, *mut* = mutation, *loc* = local optimization, *sel* = fitness based selection, *Xv* = Crossover)

Approach →	*AIS**	*AIS*	*GA*	*HC*	*PHC*
Reliabibilty/Insensitivity to initialization	yes	yes	yes	no	yes
Robustness to noise	yes	no	no	no	no
Scalability	yes	no	no	no	no
Diverse Repertoire	yes	yes	yes	no	yes
Population based?	yes	yes	yes	no	yes
Evolving data	yes	no	no	no	no
Adaptive Population size?	yes	yes	no	no	no
Level of Evolution	Cell	Cell	Organism	NA	NA
Search Operators	*cl*/*mut* /*loc*	*cl*/*mut*	*sel* /*Xv*/*mut*	*loc*	*loc*

ment. Intelligent search mechanisms are crucial in Web mining because of the large combinatorial optimization nature of many problems. An artificial immune system can act as a continuous monitoring and learning system in the face of a stream of incoming data with an unknown number of clusters. We have proposed an approach to reduce the immune interactions' complexity from quadratic to linear complexity by using a compression parameter $K > \sqrt{(N_B)}$. The proposed approach is favorable from the points of view of scalability, as well as quality of learning. Quality comes in the form of diversity and continuous adaptation as new patterns emerge. Table 2 compares the proposed scalable immune based learning, (*AIS**), to other learning pradigms: Existing AIS (*AIS*), Genetic Algorithms (*GA*), Hill Climbing (*HC*) which is the basic search method used in most existing clustering techniques such as variants of K Means and K medoids, and Parallel Hill Climbing (*PHC*), which is the search method used in techniques based on repetitive runs such as in bootstrapping. This helps situate the immune based approach within the realm of evolutionary computation as well as general search methods based on hill climbing. It is clear that the immune based approach possesses several advantages, especially from the point of view of scalability, adaptation to evolving patterns, and robustness to noise. The main factor behind

Table 3. Profiles ordered by age (from oldest to most recently discovered) and activated by 8 sessions or more, after presenting all sessions in reverse order from trend 19 to trend 0, $K = 10$, $\tau = 20$

Description
Profile : $\sigma^2 = $ 0.0552 age=77 num sessions=21
/~joshi/courses/cecs35
Profile : $\sigma^2 = $ 0.05664 age=69 num sessions=12
/,/cecs_computer.class,/courses.htm /courses_index.html,/courses100.html,/courses300.htm /~joshi/courses/cecs301/projects/proj2.htm
Profile : $\sigma^2 = $ 0.05549 age=46 num sessions=25
/~saab/cecs333,/~saab/cecs333/private,/~saab/cecs333/private/textbook_program /~saab/cecs333/private/lecture_programs,/~saab/cecs333/special_needs.html,/~saab/cecs333/final.htm /~saab/cecs333/grading.htm
Profile : $\sigma^2 = $ 0.05747 age=43 num sessions=18
/~yshang/CECS341.html,/~yshang/W98CECS34
Profile : $\sigma^2 = $ 0.05381 age=31 num sessions=28
/,/cecs_computer.class,/courses.htm /courses_index.html,/courses100.html,/courses300.htm
Profile : $\sigma^2 = $ 0.05519 age=19 num sessions=27
/,/cecs_computer.class,/courses.htm /courses_index.html,/courses100.html,/degrees.htm
Profile : $\sigma^2 = $ 0.05085 age=16 num sessions=15
/~c697168/cecs227/index.html,/~c697168/cecs227/left.html,/~c697168/cecs227/main.htm /faculty/tyrer.html,/~c697168/cecs227/info.html,/~c697168/cecs227/handouts.htm /~c697168/cecs227,/~c697168/cecs227/announce.html,/~c697168/cecs227/head.htm /~c697168/cecs227/homeworks.html,/~c697168/cecs227/labs/index.html,/~c697168/cecs227/labs/head.htm /~c697168/cecs227/labs/left.html,/~c697168/cecs227/labs/main.html,/~c697168/cecs227/labs/lab1.htm
Profile : $\sigma^2 = $ 0.05241 age=16 num sessions=10
/,/cecs_computer.class,/courses.htm /courses_index.html,/courses100.html,/courses_webpg.htm
Profile : $\sigma^2 = $ 0.05584 age=16 num sessions=9
/~c697168/cecs227/left.html,/~c697168/cecs227/main.html,/faculty/tyrer.htm /~c697168/cecs227/info.html,/~c697168/cecs227/handouts.html,/~c697168/cecs22 /~c697168/cecs227/announce.html,/~c697168/cecs227/head.html,/~c697168/cecs227/homeworks.htm /~c697168/cecs227/labs/index.html,/~c697168/cecs227/labs/head.html,/~c697168/cecs227/labs/left.htm /~c697168/cecs227/labs/main.html,/~c697168/cecs227/labs/lab1.htm
Profile : $\sigma^2 = $ 0.05063 age=15 num sessions=52
/,/cecs_computer.clas
Profile : $\sigma^2 = $ 0.0612 age=13 num sessions=8
/~c697168/cecs227/left.html,/~c697168/cecs227/main.html,/~c697168/cecs227/handouts.htm /~c697168/cecs227,/~c697168/cecs227/announce.html,/~c697168/cecs227/head.htm /~c697168/cecs227/homeworks.html,/~c697168/cecs227/labs/index.html,/~c697168/cecs227/labs/head.htm /~c697168/cecs227/labs/left.html,/~c697168/cecs227/labs/main.html,/~c697168/cecs227/labs/lab1.htm
Profile : $\sigma^2 = $ 0.05296 age=12 num sessions=10
/~c697168/cecs227/left.html,/~c697168/cecs227/main.html,/~c697168/cecs22 /~c697168/cecs227/head.html,/~c697168/cecs227/labs/index.html,/~c697168/cecs227/labs/head.htm /~c697168/cecs227/labs/left.html,/~c697168/cecs227/labs/main.html,/~c697168/cecs227/labs/lab1.htm
Profile : $\sigma^2 = $ 0.05023 age=4 num sessions=15
/,/cecs_computer.class,/people.htm /people_index.html,/faculty.htm

the ability of the immune clustering method to learn in a single pass lies in the richness of the immune network structure that forms a dynamic synopsis of the data. Such complex immune network structures have the reputation of being huge and thus hard and time consuming to manipulate. The proposed compression mechanism and the dynamic weighted B cell model are exactly what make this network small and manageable, and continuous learning possible.

With an average of *4 milliseconds per user session*, the proposed profile mining system is suitable for use in a real time personalization system to constantly and continuously provide the recommendation engine with a current set of user profiles. The logistics of maintaining, caching, or discarding old profiles are much less crucial with our approach than with most existing techniques. Even if discarded, older profiles that re-emerge later, would be re-learned from scratch just like completely new profiles. Like the natural immune system, the strongest advantage of our approach is expected to be its ease of adaptation in dynamic environments such as the World Wide Web. We are currenly performing more experiments to study the effect of the algorithm parameters on the evolution of the B-cell network in response to different input scenarios. We plan to continue applying the proposed immune clustering approach to extract patterns from continuous and evolving clickstream, text, and network data.

5. Acknowledgments

This work is supported by a National Science Foundation CAREER Award IIS-0133948 to Olfa Nasraoui, and by a grant from the Southeastern Consortium for Electrical Engineering Education (SCEEE). Fabio Gonzalez is supported by the National University of Colombia.

References

[1] T. Yan, M. Jacobsen, H. Garcia-Molina, and U. Dayal, “From user access patterns to dynamic hypertext linking,” in *Proceedings of the 5th International World Wide Web conference*, Paris, France, 1996.

[2] O. Zaiane, M. Xin, and J. Han, “Discovering web access patterns and trends by applying olap and data mining technology on web logs,” in *Advances in Digital Libraries*, Santa Barbara, CA, 1998, pp. 19–29.

[3] M. Perkowitz and O. Etzioni, “Adaptive web sites: Automatically synthesizing web pages,” in *AAAI 98*, 1998.

[4] R. Cooley, B. Mobasher, and J. Srivastava, “Data preparation for mining world wide web browsing patterns,” *Journal of knowledge and information systems*, vol. 1, no. 1, 1999.

[5] C. Shahabi, A. M. Zarkesh, J. Abidi, and V. Shah, “Knowledge discovery from users web-page navigation,” in *Proceedings of workshop on research issues in Data engineering*, Birmingham, England, 1997.

[6] O. Nasraoui, R. Krishnapuram, and A. Joshi, “Mining web access logs using a relational clustering algorithm based on a robust estimator,” in *8th International World Wide Web Conference*, Toronto, Canada, 1999, pp. 40–41.

[7] Jaideep Srivastava, Robert Cooley, Mukund Deshpande, and Pang-Ning Tan, “Web usage mining: Discovery and applications of usage patterns from web data,” *SIGKDD Explorations*, vol. 1, no. 2, pp. 1–12, Jan 2000.

[8] O. Nasraoui, R. Krishnapuram, H. Frigui, and Joshi A., “Extracting web user profiles using relational competitive fuzzy clustering,” *International Journal of Artificial Intelligence Tools*, vol. 9, no. 4, pp. 509–526, 2000.

[9] O. Nasraoui and R. Krishnapuram, “One step evolutionary mining of context sensitive associations and web navigation patterns,” in *SIAM conference on Data Mining*, Arlington, VA, 2002, pp. 531–547.

[10] Bettina Berendt, Bamshad Mobasher, Miki Nakagawa, and Myra Spiliopoulou, “The impact of site structure and user environment on session reconstruction in web usage analysis,” in *WebKDD workshop on Knowledge Discovery in the Web*, Edmonton, Alberta, Canada, 2002, pp. 115–129.

[11] Hui Yang, Srinivasan Parthasarathy, and Sandeep Reddy, “On the use of constrained association rules for web mining,” in *WebKDD workshop on Knowledge Discovery in the Web*, Edmonton, Alberta, Canada, 2002, pp. 77–90.

[12] Weinan Yang and Osmar Zaiane, “Clustering web sessions by sequence alignment,” in *Third International Workshop on Management of Information on the Web in Conjunction with 13th International Conference on Database and Expert Systems Applications*, Conference on Database and Expert Systems Applications, 2002, pp. 394–398.

[13] O. Nasraoui, D. Dasgupta, and F. Gonzalez, “An artificial immune system approach to robust data mining,” in *Genetic and Evolutionary Computation Conference (GECCO)*, New York, NY, 2002, pp. 356–363.

[14] D. Dasgupta, “Artificial immune systems and their applications,” Springer Verlag, 1999.

[15] I. Cohen, *Tending Adam’s Garden*, Springer Verlag, 2000.

[16] J. Hunt and D. Cooke, “An adaptative, distributed learning system, based on immune system,” in *IEEE International Conference on Systems, Man and Cybernetics*, Los Alamitos, CA, 1995, pp. 2494–2499.

[17] J. Timmis, M. Neal, and J. Hunt, “An artificial immune system for data analysis,” *Biosystems*, vol. 55, no. 1.

[18] L. N. De Castro and F. J. Von Zuben, “Ainet: An artificial immune network for data analysis,” in *Data Mining: A Heuristic Approach*, R. A. Sarker H. A. Abbass and C.S. Newton, Eds. Idea Group Publishing, USA, 2001.

Table 4. Profiles (activated by 8 sessions or more) after presenting all sessions in natural chronological order with $K = 25, \tau = 20$

Profile : $\sigma^2 = $ 0.05594 age=50 num sessions=30
/˜saab/cecs333,/˜saab/cecs333/private,/˜saab/cecs333/private/textbook_program /˜saab/cecs333/private/lecture_programs,/˜saab/cecs333/academic_honesty.html,/˜saab/cecs333/special_needs.htm /˜saab/cecs333/final.html,/˜saab/cecs333/grading.htm
Profile : $\sigma^2 = $ 0.05279 age=38 num sessions=8
/˜c697168/cecs227/left.html,/˜c697168/cecs227/main.html,/˜c697168/cecs227/handouts.htm /˜c697168/cecs227,/˜c697168/cecs227/announce.html,/˜c697168/cecs227/head.htm /˜c697168/cecs227/homeworks.html,/˜c697168/cecs227/labs/index.html,/˜c697168/cecs227/labs/head.htm /˜c697168/cecs227/labs/left.html,/˜c697168/cecs227/labs/main.htm
Profile : $\sigma^2 = $ 0.11668 age=21 num sessions=14
/,/cecs_computer.class,/research.htm
Profile : $\sigma^2 = $ 0.06302 age=13 num sessions=10
/˜c697168/cecs227/left.html,/˜c697168/cecs227/main.html,/˜c697168/cecs227/handouts.htm /˜c697168/cecs227,/˜c697168/cecs227/announce.html,/˜c697168/cecs227/head.htm /˜c697168/cecs227/homeworks.htm
Profile : $\sigma^2 = $ 0.10249 age=11 num sessions=18
/˜c697168/cecs227/index.html,/˜c697168/cecs227/left.html,/˜c697168/cecs227/main.htm /faculty/tyrer.html,/˜c697168/cecs227/info.html,/˜c697168/cecs227/handouts.htm /˜c697168/cecs227,/˜c697168/cecs227/announce.html,/˜c697168/cecs227/head.htm /˜c697168/cecs227/homeworks.htm
Profile : $\sigma^2 = $ 0.0544 age=8 num sessions=22
/,/cecs_computer.class,/courses.htm /courses_index.html,/courses100.html,/courses300.htm /courses200.htm
Profile : $\sigma^2 = $ 0.1402 age=7 num sessions=42
/,/cecs_computer.class,/degrees_undergrad.htm /degrees_index.html,/bsce.html,/bscs.htm /courses.html,/courses_index.html,/courses100.htm /degrees.htm
Profile : $\sigma^2 = $ 0.0547 age=4 num sessions=18
/˜c697168/cecs227/left.html,/˜c697168/cecs227/main.html,/˜c697168/cecs227/handouts.htm /˜c697168/cecs227,/˜c697168/cecs227/head.htm
Profile : $\sigma^2 = $ 0.0505 age=4 num sessions=17
/˜yshang/CECS341.htm
Profile : $\sigma^2 = $ 0.05121 age=1 num sessions=33
/˜yshang/CECS341.html,/˜yshang/W98CECS34
Profile : $\sigma^2 = $ 0.05373 age=0 num sessions=48
/,/cecs_computer.clas
Profile : $\sigma^2 = $ 0.05721 age=0 num sessions=40
/,/cecs_computer.class,/courses.htm /courses_index.html,/courses100.htm
Profile : $\sigma^2 = $ 0.08292 age=0 num sessions=27
/access,/access/details.htm
Profile : $\sigma^2 = $ 0.0596 age=0 num sessions=21
/˜c697168/cecs227/left.html,/˜c697168/cecs227/main.html,/˜c697168/cecs22 /˜c697168/cecs227/head.html,/˜c697168/cecs227/homeworks.html,/˜c697168/cecs227/labs/index.htm /˜c697168/cecs227/labs/head.html,/˜c697168/cecs227/labs/left.html,/˜c697168/cecs227/labs/main.htm /˜c697168/cecs227/labs/lab1.htm
Profile : $\sigma^2 = $ 0.06169 age=0 num sessions=11
/,/cecs_computer.class,/courses.htm /courses_index.html,/courses100.htm