International Journal of Computational Intelligence and Applications
Vol. 2, No. 3 (2002) 1–10
© Imperial College Press

# AN EVOLUTIONARY APPROACH TO MINING ROBUST MULTI-RESOLUTION WEB PROFILES AND CONTEXT SENSITIVE URL ASSOCIATIONS

OLFA NASRAOUI

*Department of Electrical and Computer Engineering*
*The University of Memphis, 206 Engineering Science*
*Memphis, TN, 38152/3180, USA*
*onasraou@memphis.edu*

RAGHU KRISHNAPURAM

*IBM India Research Lab, Block 1, Indian Institute of Technology*
*Hauz Khas, New Delhi, 110016, India*
*kraghura@in.ibm.com*

Received 6 November 2001
Revised 4 June 2002

We present a technique for simultaneously mining Web navigation patterns and maximally frequent context-sensitive itemsets (URL associations) from the historic user access data stored in Web server logs. A new hierarchical clustering technique that exploits the symbiosis between clusters in feature space and genetic biological niches in nature, called Hierarchical Unsupervised Niche Clustering (H-UNC) is presented. We use H-UNC as part of a complete system of knowledge discovery in Web usage data. Our approach does not necessitate fixing the number of clusters in advance, is insensitive to initialization, can handle noisy data, general non-differentiable similarity measures, and automatically provides profiles at multiple resolution levels. Our experiments show that our algorithm is not only capable of extracting meaningful user profiles on real Web sites, but also discovers associations between distinct URL pages on a site, with no additional cost. Unlike content based association methods, our approach discovers associations between different Web pages based only on the user access patterns and not on the page content. Also, unlike traditional context-blind association discovery methods, H-UNC discovers context-sensitive associations which are only meaningful within a limited context/user profile.

*Keywords*: Web mining; clustering; genetic algorithms; genetic niching; user profiles.

## 1. Introduction

In addition to its ever-expanding size and lack of structure, the World Wide Web has not been responsive to user preferences and interests. Personalization deals with tailoring a user's interaction with the Web information space based on information about him/her, in the same way that a reference librarian uses background knowledge about a person in order to help them better. For example, the phrase "theory of groups" has completely different meanings for a sociologist and a mathematician. In this case, the phrase is the same, while the contexts are different. The concept of *contexts* can be mapped to distinct user *profiles*. Mass profiling is based on general trends of usage patterns (thus protecting privacy) compiled from all users on a site, and can be achieved by mining user profiles from the historical data stored in server access logs.

Recently, data mining techniques have been applied to extract usage patterns from Web log data.[1−8] Of relevance to this paper is our previous work[6−8] where 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[6−8] 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.

Current approaches avoid the feature representation dilemma of Web data by either resorting to relational clustering[6−8] (requires the computation and storage of all pairwise similarities)or association rule discovery[9] *prior* to discovering user profiles[4] (hence relying on *two relatively expensive* data mining steps). Since URL associations tend to occur with very low support in Web log files, the frequent itemset discovery step can become prohibitively expensive. In this paper, we present a *quasi-linear* complexity technique for mining both user profile clusters and URL associations in a *single* step.

Evolutionary techniques such as Genetic Algorithms (GAs)[10] have proved effective in exploring complicated fitness landscapes and converging populations of candidate solutions to a single global optimum. Recently, Nasraoui and Krishnapuram[11] have presented a new evolutionary approach to clustering called Unsupervised Niche Clustering (UNC). UNC exploits the symbiosis between clusters in feature space and genetic biological niches in nature. It is robust to noise and makes no assumptions about the number of clusters. However, UNC was formulated for the 2-D case, based on a Euclidean metric space representation of the data.

In this paper, we propose a Hierarchical modification of UNC, called Hierarchical UNC (H-UNC), that departs from the traditional limited flat view of the data, and generates instead, a hierarchy of clusters which give more insight to the Web mining process, and speeds it up considerably. We use H-UNC as part of a complete system of knowledge discovery in Web usage data. Our new approach does not necessitate fixing the number of clusters in advance, can provide profiles to match any desired level of detail or resolution, and requires no analytical derivation of the prototypes. Thus, it can handle a vast array of general subjective, even non-metric dissimilarities, making it suitable for many applications, particularly in data and Web mining. Our web mining approach also discovers associations between different Web pages based only on the user access patterns or profiles, and not on the Web site page content. These associations are meaningful only within well defined distinct profiles/contexts (*context-sensitive*) as opposed to all or none of the data (context-blind). This approach of discovering context-sensitive associations via clustering can be generalized to other transactional data.

The remainder of this paper is organized as follows. In Sec. 2, we explain our Knowledge Discovery in Web usage data. In Sec. 3, we present the Unsupervised Niche Clustering algorithm (UNC). In Sec. 4, we present the Hierarchical Unsupervised Niche Clustering algorithm (H-UNC), and adapt it to clustering Web sessions. In Sec. 5, we present our experimental results. Finally, we present our conclusions in Sec. 6.

## 2. The Knowledge Discovery Process of Web Session Profiling

### 2.1. *Extracting web user sessions*

The access log for a given Web server consists of a record of all files accessed by users. Each log entry consists of: (i) User's IP address, (ii) Access time, (iii) URL of the page accessed, …, etc. A user session consists of accesses originating from the same IP address within a predefined time period. Each URL in the site is assigned a unique number $j \in \{1, \ldots, N_U\}$, where $N_U$ is the total number of valid URLs. Thus, the $i$th user session is encoded as an $N_U$-dimensional binary attribute vector $\mathbf{s}^{(i)}$ with the property

$$s_j^{(i)} = \begin{cases} 1 & \text{if the user accessed the } j\text{th URL during the } i\text{th session} \\ 0 & \text{otherwise.} \end{cases}$$

The ensemble of all $N_S$ sessions extracted from the server log file is denoted $\mathcal{S}$.

### 2.2. *Assessing web user session similarity*

The similarity measure between two user-sessions: $\mathbf{s}^{(k)}$ and $\mathbf{s}^{(l)}$ relies on two sub-measures.[6,8] The first measure which ignores the site structure is given by $S_{1,kl} = \frac{\sum_{i=1}^{Nu} s_i^{(k)} s_i^{(l)}}{\sqrt{\sum_{i=1}^{Nu} s_i^{(k)}}\sqrt{\sum_{i=1}^{Nu} s_i^{(l)}}}$. The second similarity measure requires the pre-computation of the similarities at the structural URL level that will be used in the computation of the similarity at the session level.

The entire Web site is modeled as a tree with the nodes representing different URLs. The tree is similar to that of a directory where an edge connects one node to another if the URL corresponding to the latter is hierarchically located under that of the former. The "syntactic" similarity between the $i$th and $j$th URLs is defined as $S_u(i,j) = \min\left(1, \frac{|p_i \cap p_j|}{\max(1, \max(|p_i|, |p_j|) - 1)}\right)$, where $p_i$ denotes the path traversed from the root node (main page) to the node corresponding to the $i$th URL, and $|p_i|$ indicates the length of this path. Note that this similarity which lies in [0, 1] basically measures the amount of overlap between the paths of the two URLs. This overlap is inferred directly from the URL address string by exploiting the one-to-one mapping between the address and the site topology. The pairwise URL similarities should be computed only once offline for a particular Web site prior to any clustering. Now the similarity on the session level which incorporates the syntactic URL similarities is computed by $S_{2,kl} = \frac{\sum_{i=1}^{N_U} \sum_{j=1}^{N_U} s_i^{(k)} s_j^{(l)} S_u(i,j)}{\sum_{i=1}^{N_U} s_i^{(k)} \sum_{j=1}^{N_U} s_j^{(l)}}$. The final similarity, given by a maximally optimisitc aggregation of $S_{1,kl}$ and $S_{2,kl}$, is $S_{kl} = \max(S_{1,kl}, S_{2,kl})$. Finally, this similarity is mapped to the dissimilarity measure $d_s^2(k,l) = (1 - S_{kl})^2$. One of the desirable properties of this Web session dissimilarity is that it becomes more stringent as the accessed URLs get farther from the root because the amount of specificity in user accesses increases correspondingly. Our syntactic similarity offers an implicit way to capture the concept hierarchy of the URLs of a Web site while mining the clusters and associations, and can be generalized to other transactional databases.

### 2.3. *Interpretation and evaluation of the results*

The results of clustering the user session data are interpreted using the following quantitative measures.[6] First, the user sessions are assigned to the closest clusters based on the computed distances, $d_{ik}$, from the $i$th cluster to the $k$th session. This creates $C$ clusters $\mathcal{X}_i = \{\mathbf{s}^{(k)} \in \mathcal{S} | d_{ik} < d_{jk} \ \forall j \neq i\}$, for $1 \leq i \leq C$.

The sessions in cluster $\mathcal{X}_i$ are summarized by a typical session "profile" vector[6] $\mathbf{P}_i = (P_{i1}, \ldots, P_{iN_U})^t$. The components of $\mathbf{P}_i$ are URL relevance weights, estimated by the conditional probability of access of each URL during the sessions of $\mathcal{X}_i$, i.e. $P_{ij} = p(\mathbf{s}_j^{(k)} = 1 | \mathbf{s}_j^{(k)} \in \mathcal{X}_i) = \frac{|\mathcal{X}_{i_j}|}{|\mathcal{X}_i|}$, where $\mathcal{X}_{i_j} = \{\mathbf{s}^{(k)} \in \mathcal{X}_i | s_j^{(k)} > 0\}$. The URL weights $P_{ij}$ measure the significance of a given URL to the $i$th profile. Besides summarizing profiles, the components of the profile vector can be used to recognize an invalid profile which has no strong or frequent access pattern. For such a profile, all the URL weights will be low.

The final prototypes resulting from UNC can be evaluated based on the mean squared error or average dissimilarity, which for the $i$th cluster, is given by $\sigma_i^{*2} = \frac{\sum_{\mathbf{s}^{(k)} \in \mathcal{X}_i} d_{ik}^2}{|\mathcal{X}_i|}$. Another measure is the robust cardinality given by $N_i^* = \sum_{\mathbf{s}^{(k)} \in \mathcal{X}_i} w_{ik}$, where $w_{ik} = \exp\left(-\frac{d_{ik}^2}{2\sigma_i^{*2}}\right)$ is a robust weight (that is high for inliers/good data and low for outliers/noise). Note that the robust cardinality and robust weights can only be exploited when a "robust" clustering method is used to produce the final profiles. The robust weights can also be used to filter out the noise sessions, and thus obtain the $w_{\min}$-core of the $i$th profile, defined as $\mathcal{X}_i^* = \{\mathbf{s}^{(k)} \in \mathcal{X}_i | w_{ik} > w_{\min}\}$.

## 3. The Unsupervised Niche Clustering Algorithm (UNC)

In Ref. 11, we reformulated the clustering problem by modifying our objective from searching the solution space for $C$ clusters to searching this space for any one cluster. The fitness value, $f_i$, for the $i$th candidate center location, $\mathbf{c}_i$, is defined as the density of a hypothetical cluster at that location, defined as $f_i = \frac{\sum_{j=1}^{N} w_{ij}}{\sigma_i^2}$, where $w_{ij} = \exp - \frac{d_{ij}^2}{2\sigma_i^2}$ is a robust weight that measures how typical data point $\mathbf{x}_j$ is in the $i$th cluster, $\sigma_i^2$ is a robust measure of scale (dispersion) for the $i$th cluster, $d_{ij}^2$ is the distance from data point $\mathbf{x}_j$ to cluster center $\mathbf{c}_i$, and $N$ is the number of data points. The landscape of the density fitness function is expected to reach several suboptimal peaks (multiple modes) located at the centroids of these clusters, and their identification is a multi-modal optimization problem. Therefore, we resort to niching methods which can identify multiple optima within multimodal domains. Like in nature, niches in our context correspond to different subspaces of the environment (clusters) that can support different types of life (data samples). For the clustering problem, we found Mahfoud's[12] "deterministic crowding" (DC) to work best. DC modifies both the selection and replacement strategies in the GA. After the mating of 2 parents, DC replaces each parent by the most similar child only if the latter has higher fitness. It can easily be seen that as a variance measure, $\sigma_i^2$ is also related to the radius of the niche, since in this particular optimization problem, each cluster in the data set will generate a niche in the fitness landscape. Note that the robust weights $w_{ij}$ will be small for outliers, hence offering a means of distinguishing between good data and noise. The scale parameter that maximizes the fitness value for the $i$th cluster can be found by setting $\frac{\partial f_i}{\partial \sigma_i^2} = 0$ to obtain $\sigma_i^2 = \frac{\sum_{j=1}^{N} w_{ij} d_{ij}^2}{\sum_{j=1}^{N} w_{ij}}$. Therefore, $\sigma_i^2$ will be updated once per genereation, using the previous values of $\sigma_i^2$ to compute the weights $w_{ij}$. This *hybrid* genetic optimization converges much faster (typically 10 generations) than a purely genetic search.

## 4. Hierarchical Unsupervised Niche Clustering and Its Adaptation to Web Usage Mining

We retain the principal structure of UNC presented in Sec. 3. The solution space for possible session prototypes consists of binary chromosome strings which are defined to be the binary session attribute vectors $\mathbf{s}_i$ defined in Sec. 2.1. The Web session dissimilarity measure, defined in Sec. 2.2, is used instead of the Euclidean distance to compute the fitness measures.

The computational time of genetic optimization can be significantly reduced if we perform clustering in a hierarchical mode. In other words, we could cluster smaller subsets of the data using a smaller population size, $N_P$ (typically $< \frac{N}{10000}$ to $\frac{N}{1000}$, at multiple levels, instead of clustering the entire data set on a single level which would necessitate a larger population size. The computational complexiy of UNC at each level is $\mathcal{O}(N_P.N)$. This quasi-linear complexity is much lower than that of relational clustering techniques such as Agglomerative Hierarchical Clustering (AHC),[13] $\mathcal{O}(N^2 \log N)$ and the closely related graph theoretic based Minimum Spanning Tree (MST),[13] $\mathcal{O}(N^2)$. We further exploit the fact that Web sessions are extremely sparse (typically $< 10$ URLs per session), and reduce the distance computation complexity to practically point-size with respect to the total number of URLs which can be huge, by using list data structures to represent the Web sessions. That is why the number of URLs, no matter how huge, does not contribute to the computational complexity. This can be seen as a case of subspace clustering, since the distance computations are done only in the subspaces relevent to the Web sessions being considered.

The hierarchical clustering is performed recursively starting from the top level (lowest resolution) until a termination criterion, based on the minimum acceptable size of a cluster, $N_{\text{split}}$, and its maximum allowable mean squared error, $\sigma^2_{\text{split}}$, is met. The hierarchical clustering procedure using UNC for Web mining is given below.

```
                Hierarchical clustering using UNC (H-UNC algorithm)
Fix population size, number of generations, and maximum number of levels (L);
Set starting level $l = 1$, and initial number of clusters $|\mathcal{C}_{(l-1)}| = 1$;
Set initial data set to be clustered $\mathcal{X}_{(l-1)} = \mathcal{X}_{(l-1)_1} = \mathcal{X}$;
Set initial set of prototypes $\mathcal{C}_{(l-1)} = \emptyset$;
Initialize final list of prototypes $\mathcal{P} = \emptyset$;
Initialize the set of mean squared errors $\Sigma_{(l-1)} = \{\sigma^{*2}_{(l-1)_1} = 1\}$;
Cluster_Recursively ($\mathcal{X}_{(l-1)}, \mathcal{C}_{(l-1)}, \Sigma_{(l-1)}, l$);
Assign all data points in $\mathcal{X}$ to closest prototype $P_i \in \mathcal{P}$;
Recompute $\sigma^{*2}_i$ and $N^*_i$ as explained in Sec. 2.3;
```

```
            Cluster_Recursively ($\mathcal{X}_{l-1}, \mathcal{C}_{l-1}, \Sigma_{l-1},$)
FOR $i = 1$ TO $|\mathcal{C}_{l-1}|$ DO { /* Each prototype in $\mathcal{C}_{l-1}$ */
   IF ($l = 1$) or ($|\mathcal{X}_{(l-1)_i}| > N_{split}$ and $\sigma^{*2}_{(l-1)_i} > \sigma^2_{split}$ and $l \leq L$) THEN
   {
     Perform UNC clustering on data subset $\mathcal{X}_{(l-1)_i}$;
        /* will result in extracted prototypes set, $\mathcal{C}_{(l)_i}, = \{P_{l_1}, \ldots, P_{l_{|\mathcal{C}_l|}}\}$
        partionned data set $\mathcal{X}_{li} = \mathcal{X}_{l1} \bigcup \cdots \bigcup \mathcal{X}_{l|\mathcal{C}_l|}$, and $\Sigma_{li} = \{\sigma^{*2}_{l_1}, \ldots, \sigma^{*2}_{l_{|\mathcal{C}_l|}}\}$
        computed for each subset of this partition */
   Cluster_Recursively ($\mathcal{X}_{li}, \mathcal{C}_{li}, \Sigma_{li}, l + 1$);
}
   ELSE {
        Add $i$th prototype to final list of prototypes: $\mathcal{P} \leftarrow \mathcal{P} \bigcup P_{(l-1)_1}$;
   }
}
```

### 4.1. *Comparison with conventional hierarchical clustering*

Our approach is substantially different from classical divisive hierarchical clustering techniques.[13] This is because our approach relies on robust weights to suppress the influence of outliers and data from other clusters, and seeks multiple clusters in parallel at each level. This means that at any given level of recursive clustering, even if the population size is too small, H-UNC is expected to identify as many good clusters as possible, while classical hierarchical approaches are expected to yield the optimal cluster prototypes only at the optimal level of the partition that corresponds to the known number of clusters. This is why H-UNC performs well even with anomalously small population sizes. Also, H-UNC re-partitions the data at the very end of clustering. Hence, there is no final commitment of the data at each level. This avoids one of the well known pitfalls of hierarchical clustering techniques, and also allows H-UNC to yield better partitions, and hence more accurate Web profiles.

## 5. Web Usage Mining Experimental Results

Our new approach to profiling Web users based on Hierarchical UNC was applied to extract the typical Web session profiles from the log data of five real Web sites with number of sessions extracted after preprocessing varying from 1638 to 29,876, and number of valid URLs varying between 369 and 17,665. Because of space limitations, we present profiling results for only one data set. Our results are not sensitive to crossover and mutation probability parameters, as long as they are within reasonable bounds. That is, $P_c > 0.8$ and $P_m < 10^{-2}$, UNC used 10 generations per clustering with a population size, $N_P = 10$, and $N_{\min} = 10$. Since all session dissimilarities are confined in $[0, 1]$, it is reasonable to choose $\sigma^2_{\max} = 0.95$, $\sigma^2_{\text{split}} = 0.3$, $N_{\text{split}} = 30$. After clustering, the sessions were assigned to the closest cluster and only the session clusters or profiles with cardinalities exceeding 20 were considered sufficiently strong. The profile vectors were computed as explained in Sec. 2.3, where only the significant URLs ($P_{ij} > 0.15$) were retained, and the individual components are displayed in the format $\{P_{ij}$ - $j$th URL$\}$ in Table 1 illustrating some profiles. The tables that summarize the description of some profiles also list their cardinality, $|\mathcal{X}_i|$, core cardinality, $|\mathcal{X}_i^*|$ (or "–" if $|\mathcal{X}_i^*| < 20$ sessions), robust cardinality, $N_i^*$, and average dissimilarity, $\sigma_i^{*2}$.

Table 1. Examples of core profiles discovered by H-UNC from MU-CECS1 data at $L = 3$ and $w_{\min} = 0.6$.

| $i$ | $\mathbf{P}_i$ |
|---|---|
| 1 | {.83 – /CECS_computer.class} {.95 – /courses.html} {.95 – /courses100.html} {.95 – /courses_index.html} {.19 – /courses200.html} {.19 – /people.html} {.19 – /people_index.html} {.19 – /faculty.html} {.93 - /} |
| 3 | {1.00 – /} {.67 – /CECS_computer.class} |

We describe the results obtained on the 1703 sessions and 369 URLs extracted from 12-day Web logs of the Department of Electrical and Computer Engineering at the University of Missouri-Columbia. The results at $L = 3$ levels are summarized in Table 3 which shows that H-UNC succeeded in delineating many real profiles reflecting typical access patterns — the general "outside visitor" is captured in profiles 1 and 3; prospective students in profiles 2 and 4, CECS 227 students in profile 10, etc. The quality of these clusters is confirmed by their low average dissimilarity ($\sigma_i^{*2}$) compared to the maximal value of one.

(i) ***Robust profiling:*** When only sessions with weights exceeding 0.6 are considered, profiles Nos. 8 and 13 end up having less than 20 members, hence making weak profiles. Not only do the robust weights enable filtering the profiles of noise-induced URLs, but they can also be used to compute robust goodness measures such as core cardinality, $|\mathcal{X}_i^*|$, robust cardinality, $N_i^*$, and average dissimilarity, $\sigma_i^{*2}$ as explained above. In particular, the agreement between $|\mathcal{X}_i^*|$ and $N_i^*$ is a good indicator of the accuracy of the robust weights as true quantitative discriminators between sessions that are relevent to each profile, and those that are not.

(ii) ***Multiresolution profiling:*** Note how profile 2 (at $L = 1$) in Table 2 is split into many profiles with distinct user interests (at $L = 2$) (profiles Nos. 6, 8,

Table 2. 4 of the 7 profiles discovered by H-UNC from MU-CECS1 data at $L = 1$.

| $i$ | $\lvert\mathcal{X}_i\rvert$ | $\lvert\mathcal{X}_i^*\rvert$ | $N_i^*$ | Description | $\sigma_i^{*2}$ |
|---|---|---|---|---|---|
| 1 | 572 | 312 | 362.2 | main page, courses, people and degree pages | 0.32 |
| 2 | 305 | 170 | 191.0 | CECS 333 and CECS 352 course pages | 0.54 |
| 3 | 185 | 111 | 124.0 | Accesses to the CECS227 class pages | 0.20 |
| 4 | 162 | 84 | 102.2 | Dr Shi's CECS345 pages | 0.37 |
| 5 | 73 | 56 | 51.0 | Dr Shang's course pages | 0.08 |

Table 3. Some of the 16 profiles discovered by H-UNC from MU-CECS1 data at $L = 2$ and $L = 3$.

| $i$ | $\lvert\mathcal{X}_i\rvert$ | $\lvert\mathcal{X}_i^*\rvert$ | $N_i^*$ | Description | $\sigma_i^{*2}$ |
|---|---|---|---|---|---|
| 1 | 219 | 132 | 140.5 | main page, class list, course enquiries and people | 0.16 |
| 2 | 119 | 73 | 77.0 | main page, course and undergraduate degree enquiries | 0.27 |
| 3 | 140 | 85 | 91.6 | Short sessions mostly limited to main page | 0.13 |
| 4 | 129 | 71 | 80.7 | main page, people, faculty, research and graduate degree pages | 0.39 |
| 6 | 133 | 80 | 85.2 | CECS333 (long detailed sessions) | 0.46 |
| 8 | 47 | – | 29.4 | CECS333 pages (short sessions) | 0.16 |
| 9 | 53 | 28 | 33.4 | CECS303 pages | 0.19 |
| 10 | 184 | 111 | 123.3 | Accesses to the CECS227 class pages | 0.20 |
| 11 | 77 | 49 | 49.3 | Dr Shi's CECS345 (Java examples) | 0.27 |
| 12 | 47 | 30 | 30.0 | Dr Shi's CECS345 (long detailed sessions) | 0.26 |
| 13 | 34 | – | 22.5 | Dr Shi's CECS345 (short sessions to main page) | 0.19 |

Table 4. Some of the 12 user session profiles discovered by NERF (a relational clustering approach) for MU-CECS1 data.

| $i$ | $\lvert\mathcal{X}_i\rvert$ | Description | $\overline{D}_{Wi}$ |
|---|---|---|---|
| 1 | 70 | Dr Shang's pages | 0.20 |
| 4 | 213 | General course inquiries | 0.21 |
| 5 | 109 | Main page, people, research, and faculty pages | 0.78 |
| 9 | 172 | Accesses to the CECS227 class pages | 0.28 |
| 12 | 458 | Mixture of unrelated accesses | 0.87 |

and 9) as shown in Table 3. The first cluster (general inquiries about the CECS department) gets split, at level 2, into profiles Nos. 1, 2, 3, and 4, with each such profile showing a more specific interest in the department.

(iii) ***Inferring associations between different URLs:*** Profile 2 (at $L = 1$) in Table 2 contains accesses to two different courses taught by different professors, signaling an association. It was later revealed that one of the courses (CECS 352: Operating systems) relies for the implementation of its projects on C++ which is taught in the other course (CECS 333: Object Oriented Design).

### 5.1. *Comparison with NERF C-means algorithm*

After computing all pairwise dissimilarities, the Non-Euclidean Relational Fuzzy $C-$Means (NERF)[14] was used to cluster the sessions relation matrix with $C = 30$ clusters, and resulted in only 12 significant profiles (Table 4). NERF completely missed seven of the profiles found by H-UNC. Moreover, NERF's clusters tend to contain more irrelevent sessions or noise because it uses no robust weights.

## 6. Conclusion

For Web usage mining, the session dissimilarity measure is not a distance metric, and dealing with relational data[6–8] is impractical given the huge size of the data sets. Therefore, we presented an adaptation of UNC, called Hierarchical Unsupervised Niche Clustering (H-UNC) which is considerably faster than its non-hierarchical counterpart. H-UNC does not necessitate fixing the number of clusters in advance, can provide profiles to match any desired level of resolution, and requires no analytical derivation of the prototypes. Thus, it can handle a vast array of subjective, even non-metric dissimilarities, making it suitable for many applications in data and Web mining. Also, associations between different URL addresses are discovered with no additional cost. Moreover, they are meaningful only within well defined distinct profiles/contexts (context-sensitive) as opposed to all or none of the data (context-blind). Unlike *Lamarckian learning*,[15] our dynamic approach to estimate the scale mathematically during genetic optimization of the cluster representatives *does not disrupt the genotype* of the solutions. However, it improves *individual learning* by dynamically modifying the *fitness landscape* in a way that will make it easier to maintain diversity and to converge closer to the niche peaks. This can be seen as introducing a *Baldwin learning effect* [15] into the evolution. We are currently investigating different ways to make our approach scalable to large data sets, using it for clustering text documents/Web content, and incorporating Web content into Web user profiling.

## Acknowledgment

This work was supported by the National Science Foundation (CAREER Award IIS-0133948 to O. Nasraoui and NSF Grant IIS-9800899 to R. Krishnapuram).

## References

1. T. Yan, M. Jacobsen, H. Garcia-Molina and U. Dayal, From user access patterns to dynamic hypertext linking, *5th World Wide Web Conf.* Paris, 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, *Advances in Digital Libraries*, Santa Barbara, CA (1998) 19–29.
3. M. Perkowitz and O. Etzioni, Adaptive web sites: Automatically synthesizing web pages, *AAAI'98*, 1998.
4. R. Cooley, B. Mobasher and J. Srivastava, Data preparation for mining world wide web browsing patterns, *Knowledge Inf. Syst.* **1**, 1 (1999).
5. C. Shahabi, A. M. Zarkesh, J. Abidi and V. Shah, Knowledge discovery from users web-page navigation, *Proc. Workshop Res. Issues Data Eng.*, 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, *NAFIPS Conf.*, New York (June 1999) 705–709.
7. O. Nasraoui, H. Frigui, R. Krishnapuram and A. Joshi, Mining web access logs using relational competitive fuzzy clustering, *8th Int. World Wide Web Conf.*, Toronto, Canada, 1999.
8. O. Nasraoui, R. Krishnapuram, H. Frigui and A. Joshi, Extracting web user profiles using relational competitive fuzzy clustering, *Int. J. Artif. Intell. Tools* **9**, 4, (2000) 509–526.
9. R. Agrawal and R. Srikant, Fast algorithms for mining association rules, *20th VLDB Conf.*, Santiago, Chile, 1994, 487–499.
10. J. H. Holland, *Adaptation in Natural and Artificial Systems* (MIT Press, 1975).
11. O. Nasraoui and R. Krishnapuram, A novel approach to unsupervised robust clustering using genetic niching, *9th IEEE Int. Conf. Fuzzy Syst.*, San Antonio, TX, May 2000, 170–175.
12. S. W. Mahfoud, Crowding and preselection revisited, *Parallel Problem Solving from Nature, PPSN'92*, Brussels, 1992.
13. R. Duda and P. Hart, *Pattern Classification and Scene Analysis*, Wiley, New York, 1973.
14. R. J. Hathaway and J. C. Bezdek, Nerf c-means: Non-euclidean relational fuzzy clustering, *Pattern Recognition*, **27**, 3 (1994) 429–437.
15. D. Whitley, S. Gordon and K. Mathias, Lamarckian evolution, the baldwin effect and function optimization, *PPSN III*, Jerusalem, 1994.