Available online at www.sciencedirect.com
SCIENCE DIRECT®
Computer Networks 50 (2006) 1488–1512
Computer Networks
www.elsevier.com/locate/comnet

A framework for mining evolving trends in Web data streams using dynamic learning and retrospective validation

Olfa Nasraoui [a,*], Carlos Rojas [a], Cesar Cardona [b,1]

[a] Department of Computer Engineering and Computer Science, University of Louisville, Louisville, KY 40292, United States
[b] Magnify Inc., Chicago, United States

Available online 27 December 2005

Abstract

The expanding and dynamic nature of the Web poses enormous challenges to most data mining techniques that try to extract patterns from Web data, such as Web usage and Web content. While scalable data mining methods are expected to cope with the size challenge, coping with evolving trends in noisy data in a continuous fashion, and without any unnecessary stoppages and reconfigurations is still an open challenge. This dynamic and single pass setting can be cast within the framework of mining evolving data streams. The harsh restrictions imposed by the "*you only get to see it once*" constraint on stream data calls for different computational models that may furthermore bring some interesting surprises when it comes to the behavior of some well known similarity measures during clustering, and even validation. In this paper, we study the effect of similarity measures on the mining process and on the interpretation of the mined patterns in the harsh single pass requirement scenario. We propose a simple similarity measure that has the advantage of explicitly coupling the precision and coverage criteria to the early learning stages. Even though the cosine similarity, and its close relative such as the Jaccard measure, have been prevalent in the majority of Web data clustering approaches, they may fail to explicitly seek profiles that achieve high coverage and high precision *simultaneously*. We also formulate a validation strategy and adapt several metrics rooted in information retrieval to the challenging task of validating a learned stream synopsis in dynamic environments. Our experiments confirm that the performance of the *MinPC* similarity is generally better than the cosine similarity, and that this outperformance can be expected to be more pronounced for data sets that are more challenging in terms of the amount of noise and/or overlap, and in terms of the level of change in the underlying profiles/topics (known sub-categories of the input data) as the input stream unravels. In our simulations, we study the task of mining and tracking trends and profiles in evolving text and Web usage data streams in a single pass, and under different trend sequencing scenarios.

Keywords: Mining evolving data streams; Web clickstreams; Web mining; Text mining; User profiles

1. Introduction

"You cannot step twice into the same stream. For as you are stepping in, other waters are ever flowing on to you" HERACLITUS, c.535–475 BC, quoted by Plato.

* Corresponding author.
E-mail addresses: olfa.nasraoui@louisville.edu (O. Nasraoui), c.rojas@louisville.edu (C. Rojas), ccardona@magnify.com (C. Cardona).
[1] This research was done while C. Cardona was at the University of Memphis.

1389-1286/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.comnet.2005.10.021

The Web has been a relentless generator of data that comes in a variety of forms, ranging from Web content data that forms the substance of most Web documents, to the daily trails left by visitors as they surf through a Website, also known as Web usage data. Hidden in this data, often lurk interesting knowledge or patterns such as Web user access trends or profiles that can be used to achieve various objectives, including supporting customer relationship management, and personalization of the user's experience on a Website.

Recently, data mining techniques have been applied to extract usage patterns from Web log data [3,6,18–21,24–26,29,30]. Most of these efforts have proposed using various data mining or machine learning techniques to model and understand Web user activity. In [29], clustering was used to segment user sessions into clusters or profiles that can later form the basis for personalization. In [21], the notion of an adaptive Website was proposed, where the user's access pattern can be used to automatically synthesize index pages. The work in [6] is based on using association rule discovery as the basis for modeling Web user activity, while the approach proposed in [3] used Markov Random Fields to model Web navigation patterns for the purpose of prediction. The work in [30] proposed building data cubes from Web log data, and later applying online analytical processing (OLAP) and data mining on the cube model. [25] presents a complete Web Usage Mining (WUM) system that extracts patterns from Web log data with a variety of data mining techniques. New relational clustering techniques with robustness to noise were used to discover user profiles that can overlap in [20,19], while a density-based evolutionary clustering technique is proposed to discover multi-resolution and robust user profiles in [18]. The K Means algorithm was used in [24] to segment user sequences into different clusters. An extensive survey of different approaches to Web usage mining can be found in [26]. It is interesting to note that an *incremental* way to update a Web usage mining model was proposed in [3]. In this approach, the user navigation records are modeled by a hypertext probabilistic grammar (HPG) whose higher probability generated strings correspond to the user's preferred trails. The model had the advantages of being self-contained (i.e., has all statistics needed to mine all the data accumulated), as well as compact (the model was in the form of a tree whose size depends on the number of items instead of the number of users, which enhances scalability). The HPG model was incremental, in the sense that when more log data became available, it could be incorporated in the model without the need of rebuilding the grammar from scratch.

Unfortunately, with the exception of [3] (which provided a scalable way to model Web user navigation, but did not explicitly address the change/evolvability aspect of this data), all the aforementioned methods assume that the entire pre-processed 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 Websites 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 Website are very dynamic in nature, due not only to the dynamics of Website content and structure, but also to changes in the users' 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. 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, 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. For all these reasons, Web usage data should be considered as a reflection of a dynamic environment which therefore requires dynamic learning of the user access patterns. This dynamic setting can be cast within the framework of mining evolving data streams. *Data streams* are massive data sets that arrive with a throughput so high that the data can only be analyzed sequentially and in a single pass. The discovery of useful patterns from data streams is referred to as stream data mining. In particular, a recent explosion of applications generating and analyzing *data streams* has added new unprecedented challenges for clustering algorithms if they are to be able to track changing clusters in streams using only the new data points because storing past data is not even an option [1,2,5,10]. Because most data streams unleash data points or measurements in a non-arbitrary order, they are inherently attached to a temporal aspect, meaning that the patterns that could be discovered from them follow dynamic

trends, and hence they are different from traditional static data sets that are very large. Such data streams are referred to as *evolving data streams*. For these reasons, even techniques that are scalable for huge data sets may not be the answer for mining evolving data streams, because these techniques always strive to work on the entire data set without making any distinction between new data and old data, and hence cannot be expected to handle the notion of emerging and obsolete patterns. The harsh restrictions imposed by the "you only get to see it once" constraint on stream data calls for different computational models and different validation methodologies that may depart from the ones used in traditional data mining. In [16], we proposed a new immune system inspired approach for clustering noisy multi-dimensional stream data, called TECNO-STREAMS (tracking evolving clusters in noisy streams), that has the advantages of *scalability*, *robustness and automatic scale estimation*. TECNO-STREAMS is a scalable clustering methodology that gleams inspiration from the natural immune system to be able to continuously learn and adapt to new incoming patterns by detecting an unknown number of clusters in evolving noisy data in a single pass. Data is presented in a stream, and is processed sequentially as it arrives, in a single pass over the data stream. A stream synopsis is learned in a continuous fashion. The *stream synopsis* consists of a set of synopsis nodes or cluster representatives with additional properties, such as spatial scale and age, that offer a summary of the data stream that is concise, and yet accurate. Because the data stream is a dynamic source of data, the stream synopsis itself is dynamic, and will change to reflect the status of the current data stream. The stream synopsis is constrained so that its size does not exceed a maximal limit that is predefined depending on the application, and preference will be given to newer parts of the data stream in occupying synopsis nodes that represent them. Obsolete parts of the summary that correspond to older parts of the data stream are gradually purged from the synopsis, and delegated to secondary storage memory.

1.1. Contributions of this paper

Within the context of mining evolving Web clickstreams, we apply the mechanics of TECNO-STREAMS to continuously discover an evolving profile synopsis, consisting of synopsis nodes. Each synopsis node is an entity summarizing a basic *Web usage trend*, also referred to as *profile*, that is characterized by the following descriptors: typical representative *user session* summary (a bag of URL indices), *spatial scale* or dispersion in the usage pattern around this representative (this is the amount of variance in the distance from compatible data stream input sessions and this node: it is also a measure of error or variance that reflects the accuracy of the synopsis node as a representative of the input data stream), and *age* (time since the profile's birth).

In this paper, we study the task of tracking emerging topics/clusters in noisy and evolving text data sets (text mining), and in mining evolving user profiles from Web clickstream data (Web usage mining) in a single pass, and under different trend sequencing scenarios. A *trend sequencing scenario* corresponds to a specific way to order the Web sessions or the text documents as they are presented as input to the stream mining algorithm. If the user sessions or the text documents are known to belong to certain profiles or classes/document categories, also heretoforth called *trends*, then one particular sequencing scenario may be obtained by presenting the sessions or documents according to a particular *sequence of the trends*, such as first the sessions/documents from the first trend, then the sessions/documents from the second trend, and so on. Reversing the order of the trends will naturally result in a different sequence. Similarly, the sessions/documents can be presented in the same order as they were received in the original data stream. In this case, we refer to the sequencing scenario as "*regular order*", "*chronological order*", or "*natural chronological order*".

We propose a validation methodology and several validation metrics that are rooted in information retrieval, and that are useful to assess the quality of the stream synopsis as a summary of an input data stream from the points of view of *precision*, *coverage* (*or recall*), *and adaptability to the evolution of the stream*. *Coverage* or *recall* of a synopsis node measures the proportion of items matching the input data in its vicinity. High coverage means that the node covers most of the items in the input. On the other hand, high *precision* means that the synopsis node covers only the correct items in the input data, and not any additional superfluous items due to noise or other artifacts. Coverage and precision generally work in contradicting ways. or example, the best coverage is obtained when a single synopsis node contains all the items in the data, but then its precision will be very low. And vice versa, a node that consists

of only one of the correct items will have 100% precision, but its coverage will suffer. Our validation procedure is useful within the framework of mining *evolving* Web data streams. Unlike existing frameworks which study mostly static data sets, our adopted validation strategy is based on taking into account *both the content and the changing nature* (hence the temporal aspect) of the stream data mining task, so that the synopsis discovered by data mining, is evaluated from the perspectives of precision and coverage *throughout the entire temporal span of the experiment, and not just at one specific time*, such as the end of the stream. This philosophy of validation gives rise to interesting *retrospective* validation measures that are rooted in some classical information retrieval metrics, but that evaluate the learned synopsis according to two different dimensions: (i) the temporal dimension which corresponds to the order in which the input data stream arrives, and (ii) the ground-truth content categories from which the input data is known to originate. In the case of Web user sessions/clickstreams, the categories correspond to user trends, profile, or classes; while in the case of text documents, the categories correspond to known document class labels. Since, the emphasis is on learning an accurate synopsis of an arbitrary and unlabeled dynamic data stream, these categories are only used during the final validation phase, and not in the actual learning/data mining.

Our previous discussion emphasized the importance of both precision and coverage for assessing the quality of the learned synopsis. This is not surprising since these measures have always been the main validation metrics used in information retrieval. However, this also led us to the ask the following question: If these metrics are so critical in assessing quality, then why not use ''them'' to guide the search for a better synopsis. Since each synopsis node is evaluated based on the density of similar data inputs in its vicinity, this led us to adapting the similarity measure to take into account the precision and coverage metrics of a candidate synopsis node. For all these reasons, we investigate a simple similarity measure that has the advantage of explicitly coupling the precision and coverage criteria to the early learning stages, hence requiring that the learned profiles are simultaneously precise and complete, with no compromises. A diagram showing the different components of the stream mining framework is shown in Fig. 1.

1.2. Organization of this paper

The rest of this paper is organized as follows. We start by reviewing the TECNO-STREAMS algorithm in Section 2. In Section 3, we describe how we can use TECNO-STREAMS to track evolving clusters in Web usage data. Then, we present our validation methodology and metrics for the evolving stream mining framework. In Section 4 we apply our validation strategy to the task of mining real evolving Web clickstream data and for tracking evolving topic trends in textual stream data, while studying the effect of the choice of different similarity measures. Finally, in Section 5, we summarize our findings and present our conclusions.

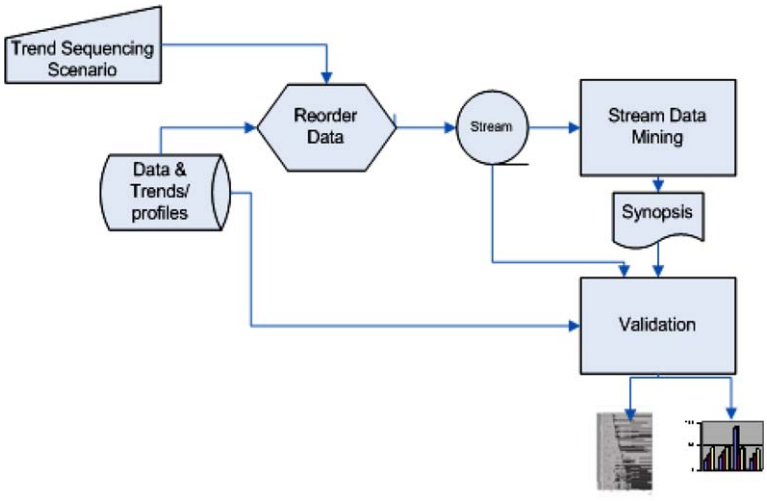

Fig. 1. Mining and validation of evolving data streams.

2. Tecno-streams (tracking evolving clusters in noisy streams)

In this section we present the main features of TECNO-STREAMS that are relevant to this paper, leaving most of the detail in [16]. The immune system (lymphocyte elements) can behave as an alternative biological model of intelligent machines, in contrast to the conventional model of the neural system (neurons). In particular, the artificial immune network (AIN) model is based on Jerne's Immune Network theory [11,27]. The system consists of a network of artificial B cell lymphocytes, $\mathscr{X}_B$, that summarize the learned model, hence playing the role of synopsis nodes. In addition to the B Cells, the immune network consists of stimulating and suppressing links between these cells. Learning takes as input a set of input data (external antigenic agents), $\mathbf{X}_a$, and tries to learn an optimal immune network consisting of linked B Cells based on cloning operations. Each B Cell represents a learned pattern that could be matched to a data point or another B Cell in the network. A link between two B Cells gets stronger if they are more similar. This results in co-stimulation between similar cells. Because an excess of co-stimulation between similar B Cells can cause an explosive growth of the B Cell population in a small local neighborhood (by cloning), there is another phenomenon, known as co-suppression, which acts to balance the influence of close B Cells. In addition to controlling population growth and enabling memory in the learned immune network, co-stimulation and co-suppression define implicit pathways of communication between the different elements (synopsis nodes) of the immune network which act like an adaptive and distributed set of agents that track the distribution of the input data stream. Without this collaborative or cooperative component in the learning, every synopsis node will act as an isolated element with no visibility of the neighboring cells. It has lately been recognized that a richer and higher level of intelligence can emerge from collaborative behavior between even the simplest agents. This phenomenon is frequently referred to as *emergence*, where complex and organized global behavior can arise from the interaction of simple local rules. Examples can be found in ant colonies, bee swarms and bird flocks [8,9,22]. In this specific context, this kind of collaborative behavior is expected to enhance memory in a distributed manner, while affecting the dynamics of learning. These crucial characteristics may well be essential to learning and adaptation in a single-pass setting, just as they are crucial to the survival of natural organisms in dynamic environments. Data from the input stream is matched against a B Cell or synopsis node based on a properly chosen similarity measure. This affects the synopsis node's stimulation level, which in turn affects both its outlook for survival, as well as the number of clones that it produces. Because clones are similar to their spawning parent, they together form a network of co-stimulated cells that can sustain themselves even long after the disappearance of data that has initiated the cloning. However, this network of synopsis nodes will slowly wither and die if it is no longer stimulated by the data for which it has specialized, hence gradually forgetting old encounters. This forgetting is the reason why the immune system needs periodical reminders in the form of re-vaccination. The combined recall and forgetting behavior in the face of external antigenic agents forms the fundamental principle behind the concept of emerging or dynamic memory in the immune system. This is specifically the reason why the immune system metaphor offers a very competitive model within the evolving data stream framework. In the following description, we present a more formal treatment of the intuitive concepts explained above.

We will use TECNO-STREAMS to continuously and dynamically learn evolving patterns from dynamic Web data. To summarize our approach: (1) The input data can be extracted from Web log data (a Web log is a record of all files/URLs accessed by users on a Web site), or from a collection of text documents, (2) the data is pre-processed (e.g., via cookies or IP address and time out window for Web logs) to produce session lists: A session list $\mathbf{s}_t$ for user number t is a list of indices of the URLs that were visited by the same user, represented as a binary vector (1 if the URL was visited during this session, and 0 otherwise). Each session will represent a data record from the input stream. In the case of text documents, a similar representation is obtained by replacing the notion of a URL by a keyword or term and a session by a document or Web page content. Hence a data object would correspond to one document represented as a list of terms; (3) the ith B Cell plays the role of a synopsis node that represents the ith candidate profile $\mathbf{p}_{s_i}$ and encodes relevant URLs (or keywords for text data), which are the attributes in this case, as well as the additional measures of scale ($\sigma^2_{i,J}$) and age (t_{si}) at

any point J (after J inputs have been processed) in the stream sequence.

In a dynamic environment, the objects $\mathbf{x}_t$ from a data stream $\mathbf{X}$ are presented to the immune network one at a time, with the stimulation and scale measures updated incrementally with each new data object. It is more convenient to think of the data index, t, as monotonically increasing with time. That is, the N_X data points are presented in the following chronological order: $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{N_X}$. Each *synopsis node* represents an influence zone over the input data space. 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. Quantitatively, the influence zone is defined in terms of a weight function that decreases not only with distance from the data location to the synopsis node prototype, but also with the time since the data has been presented to the immune network. It is convenient to think of time as an additional dimension that is added to the synopsis node compared to the classical static B Cell, traditionally defined in the data space only [17].

Robust weight/activation function: For the ith synopsis node, $\mathbf{p}_{\mathbf{s}_i}$, $i = 1, \ldots, N_{P_s}$, we define the activation caused by the tth data point (received at sequential order or time t), after a total of J inputs have been received from the data stream as

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

where τ is a *forgetting time constant* that controls the time decay rate of the contribution from old data points, and hence how much emphasis is placed on the currency of the stream synopsis compared to the sequence of data points encountered so far. d_{it}^2 is the distance from data $\mathbf{x}_t$ (which is the tth data encountered by the stream synopsis) to synopsis node, $\mathbf{p}_{\mathbf{s}_i}$. $\sigma_{i,J}^2$ 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 a data point, t, and therefore, will favor more current data in the learning process.

Influence zone: The ith synopsis node represents a soft influence zone, $\mathbf{IZ}_i$, that can be interpreted as a robust zone of influence.

$$\mathbf{IZ}_i = \{\mathbf{x}_t \in \mathbf{X} | w_{it} \geqslant w_{\min}\}. \tag{2}$$

Each synopsis node is allowed to have is own zone of influence with radial size proportional to $\sigma_{i,J}^2$, that is dynamically estimated. Hence, outliers are easily detected as data points falling outside the influence zone of all synopsis nodes or through their weak activations ($w_{it} < w_{\min}$, $\forall i$).

Stimulation/optimization criterion: The stimulation level, after J data points have been presented to synopsis node $\mathbf{p}_{\mathbf{s}_i}$, is defined as the density of the *data* population around $\mathbf{p}_{\mathbf{s}_i}$ (i.e., an estimate of the spatial density at the synopsis node as measured by the number of points in the influence zone, divided by the radius of this influence zone):

$$\delta_{i,J} = \frac{\sum_{t=1}^{J} w_{it}}{\sigma_{i,J}^2}. \tag{3}$$

2.1. Cloning in the dynamic immune system

The synopsis nodes are cloned in proportion to their stimulation levels relative to the average network stimulation by creating N_{clones_i} clones or duplicates of the ith node, where $N_{\mathrm{clones}_i} = K_{\mathrm{clone}} \frac{\delta_{i,J}}{\sum_{k=1}^{N_{P_s}} \delta_{k,J}}$. When the synopsis node population size ($N_{P_s}(t)$) at any time t exceeds a pre-specified maximum ($N_{P_{\max}}$), the synopsis nodes are sorted in ascending order of their stimulation levels, and the top ($N_{P_s}(t) - N_{P_{\max}}$) synopsis nodes (with lowest stimulation) are killed or archived in long term secondary storage in case of low stimulation cells that are mature or old.

2.2. Learning new data and relation to emerging trend detection

Somatic hyper-mutation is a powerful natural exploration mechanism in the immune system, that allows it to learn how to respond to new data that has 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 data duplication whenever a data point is encountered that fails to activate the entire stream synopsis. A new data, $\mathbf{x}_t$ is said to activate the ith synopsis node, if it falls within its influence zone, $\mathbf{IZ}_i$, essentially meaning that its activation of this synopsis node, w_{it} exceeds a minimum threshold $w_{\min}$.

Potential outlier: A *Potential outlier* is a data point that fails to activate the entire synopsis, i.e., $w_{it} < w_{\min}$, $\forall i = 1, \ldots, N_{P_s}$. The outlier is termed

potential because, initially, it may either be an outlier or a new emerging pattern. It is only through the continuous learning process that lies ahead, that the fate of this outlier will be decided. If it is indeed a true outlier, then it will form no mature nodes in the stream synopsis.

2.3. Tecno-streams: tracking evolving clusters in noisy data streams with a scalable immune system learning model

The following algorithm is only an intuitive list of steps for learning a dynamic synopsis from an evolving input data stream. More details can be found in [16].

TECNO-STREAMS algorithm: (optional steps are enclosed in [])

INPUT: data stream $\mathbf{x}_t$
OUTPUT: up to a maximum of $N_{P_{\max}}$ synopsis nodes $\mathbf{p}_{\mathbf{s}_i}$ in the stream synopsis
Fix the maximal population size or number of synopsis nodes, $N_{P_{\max}}$;
Initialize synopsis node population and $\sigma_i^2 = \sigma_{\text{init}}$ *using the first* $N_{P_{\max}}$ *input data;*
Compress stream synopsis into K subnetworks, with centroid, $\mathbf{C}_k$, $k = 1,\ldots,K$, *using 2 iterations of K Means;*
Repeat for each incoming data $\mathbf{x}_t$ {
  Present data to each subnetwork centroid, $\mathbf{C}_k$, $k = 1,\ldots,K$ *in network : Compute distance, activation weight,* w_{kt} *and update subnetwork's scale* σ_k^2 *incrementally;*
  Determine the most activated subnetwork (the one with maximum w_{kt}*);*
  IF All synopsis nodes in most activated subnetwork have $w_{it} < w_{\min}$ (*data does not sufficiently activate subnetwork*) *THEN*{
    Create by duplication a new synopsis node $= \mathbf{p}_{\mathbf{s}_{\text{new}}} = \mathbf{x}_t$ and $\sigma_{\text{new}}^2 = \sigma_{\text{init}}$;
  }
  ELSE {
    Repeat for each synopsis node $\mathbf{p}_{\mathbf{s}_i}$ *in most activated subnetwork* {
      IF $w_{it} > w_{\min}$ (*i.e., data activates synopsis node* $\mathbf{p}_{\mathbf{s}_i}$) *THEN*
        Refresh age ($t_{s_i} = 0$) *for synopsis node* $\mathbf{p}_{\mathbf{s}_i}$;
      ELSE
        Increment age (t_{s_i}) *for synopsis node* $\mathbf{p}_{\mathbf{s}_i}$;
      Compute distance from data $\mathbf{x}_j$ *to synopsis node* $\mathbf{p}_{\mathbf{s}_i}$;
      Compute synopsis node $\mathbf{p}_{\mathbf{s}_i}$*'s stimulation level;*
      Update synopsis node $\mathbf{p}_{\mathbf{s}_i}$*'s scale* σ_i^2;
    }
  }
  Clone and mutate synopsis nodes;
  IF synopsis size $N_{P_s}(t) > N_{P_{\max}}$ *Then* {
    IF (*Age of* i^{th}*synopsis node* $t_{s_i} < t_{\min}$) *THEN*
      Temporarily scale synopsis node's stimulation level to the network average stimulation;
    Sort synopsis nodes in ascending order of their stimulation level;
    Kill worst excess (*top* $(N_{P_s}(t) - N_{P_{\max}})$ *according to previous sorting) synopsis nodes;*
    [or move oldest/mature synopsis nodes to secondary (long term) storage];
  }

  Compress stream synopsis periodically (after every T data points), into K subnetworks using 2 iterations of K Means with the previous centroids as initial centroids;
}

2.4. Example of learning an evolving synopsis from a noisy data stream

To illustrate how the synopsis if formed dynamically as the input data stream arrives, we show the results on a noisy 2-D data set with three clusters (or trends) and 1400 data points because the results can be inspected visually and easily. The implementation parameters were $N_{P_{\max}} = 30$, $\tau = 100$, $w_{\min} = 0.2$, and compression with rate $K = 10$ after every $T = 40$ inputs have been processed. The evolution of the synopsis, limited to a maximum size of 30 nodes, for three noisy clusters, when the input data is presented in the order of the clusters or trends is shown in Fig. 2, where the synopsis nodes are superimposed on the original data set seen so far, at

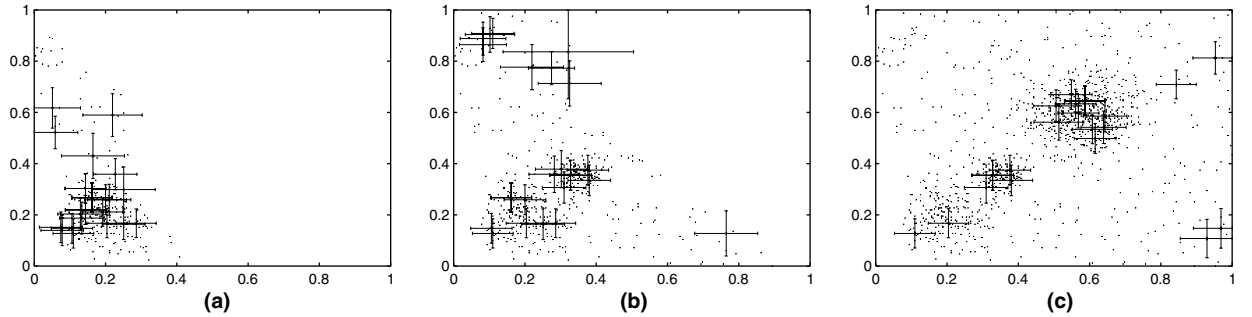

Fig. 2. Single pass results on a noisy data stream presented one input at a time in the same order as the trends/clusters: location of synopsis nodes and estimated scales for a noisy stream with three clusters after processing (a) 250 samples, (b) 500 samples, and (c) all 1134 samples.

different snapshots corresponding to distinct milestones within the data stream. The three milestones approximate the instant following the presentation of the inputs from the first cluster, then the second cluster, and finally the third cluster. Each pair of crossing vertical and horizontal lines is centered on a node location, and their length represents the diameter of this node's influence zone, **IZ**, ($\approx 3\sigma_{s_i}$). This trend sequencing scenario is the most difficult (worst) case for single-pass learning, as it truly tests the ability of the system to memorize the old patterns, adapt to new patterns, and still work within the constraints of a small synopsis. We emphasize that the final results after all inputs have been processed is equivalent to a *single* pass, resulting in a small synopsis size of only 30 nodes, that evolves together with the original input data stream to form a dynamic summary of its distribution. Note how the noise is ignored after a sufficient number of inputs, except for the last snapshot that shows the synopsis capturing a few noise points that have just been received as input. Hence they are expected to be removed like all the previous noise points if the stream were continued.

3. Mining evolving user profiles from noisy Web clickstream data

3.1. Similarity measures used in the learning phase of single-pass mining of clusters in Web data

For many data mining applications such as clustering *text* documents and other *high dimensional* data sets, the Euclidean distance measure is not appropriate. This is due mainly to the high dimensionality of the problem, and the fact that two documents may not be considered similar if keywords are missing in both documents. More appropriate for this application, is a distance based on the cosine similarity measure between data item $\mathbf{x}_t$ and a learned synopsis node profile $\mathbf{p}_{\mathbf{s}_i}$, which in the simplest case, can both be defined as binary vectors of length N_I, the total number of items (URLs or keywords), [12],

$$S_{\cos}(i,t) = \frac{\sum_{k=1}^{N_I} x_{t,k} \times p_{si,k}}{\sqrt{\sum_{k=1}^{N_I} x_{t,k} \sum_{k=1}^{N_I} p_{si,k}}}. \tag{4}$$

It is easy to show that the cosine similarity is related to the well known information retrieval measures of precision and coverage as follows:

$$S_{\cos}(i,t) = \sqrt{Prec_{i,t}^{L} Covg_{i,t}^{L}}, \tag{5}$$

where the *precision* in the learning phase, $Prec_{i,t}^{L}$ describes the accuracy of the learned synopsis node profiles $\mathbf{p}_{\mathbf{s}_i}$ in representing the data $\mathbf{x}_t$, or the ratio of the number of matching items (URLs or terms) between the learned profile and the data (session or document) to the number of items in the learned profile:

$$Prec_{i,t}^{L} = \frac{\sum_{k=1}^{N_I} x_{t,k} \times p_{s_{i,k}}}{\sum_{k=1}^{N_I} p_{s_{i,k}}}, \tag{6}$$

while the *coverage* (also known as *recall*) in the learning phase, $Covg_{ij}^{L}$ describes the completeness of the learned synopsis node profiles $\mathbf{p}_{\mathbf{s}_i}$ in representing the stream data point $\mathbf{x}_t$, or the ratio of the number of matching items (URLs or terms) between the learned profile and the data (session or document) to the number of items in the data:

$$Covg_{i,t}^{L} = \frac{\sum_{k=1}^{N_I} x_{t,k} \times p_{s_{i,k}}}{\sum_{k=1}^{N_I} x_{t,k}}. \tag{7}$$

In light of (5), we can see that using the cosine similarity as a basis for the distance used to compute the density around each synopsis node, given by the

stimulation equation in (3) results in optimizing both precision and coverage equally by combining them through the geometrical average. However, we noticed that when learning in a single-pass framework, this tends to favor longer profiles that tend to match more data, while compromising precision. Without loss of generality, if we confine ourselves to the simplest type of recommendation strategy or information retrieval scheme, we can see that compromising precision can have a pernicious effect on the learned profiles, especially when these are viewed as the cluster or profile summaries that will be used later in a recommendation system based on recommending the nearest profile, or in an information retrieval system based on matching a user query to the nearest cluster representative centroid. In order to circumvent this problem, one can simply disregard the coverage component from the cosine similarity, hence using only precision as a similarity measure. However, we noticed that this would tend to suffer from the other extreme, resulting in very short profiles that completely ignore coverage. For this reason, we propose to use different combination strategies of precision and coverage, not necessarily limited to the geometrical average. It can be shown that the most conservative aggregation that places harsh demands on both precision and coverage *simultaneously* must be given by the following pessimistic aggregation,

$$S_{\min}(i,t) = \min\left\{Prec^L_{i,t}, Covg^L_{i,t}\right\}. \tag{8}$$

Therefore, we will compare learning the profiles using cosine similarity $S_{\cos}$ to learning using the most pessimistic aggregation of precision and coverage, called *Min-Of-Precision-Coverage* or *MinPC*, $S_{\min}$. Note that the distance used to compute the nodes' stimulation/density values in (3) is given by

$$d^2_{it} = 1 - S_{\cos}(i,t), \tag{9}$$

in case the cosine similarity is used, or as

$$d^2_{it} = 1 - S_{\min}(i,t) \tag{10}$$

in case MinPC is used.

3.2. Validation metrics for single-pass mining of evolving Web data streams

We propose several validation metrics that are rooted in information retrieval, and that are useful to assess the quality of the stream synopsis as a summary of an input data stream from the points of view of *precision*, *coverage* (*or recall*), *and adaptability to the evolution of the stream*. In evaluating the goodness of the learned synopsis node profiles that make up the stream synopsis model, we recall that the *ideal* synopsis node profiles should represent the input data stream with respect to its subcategories or ground-truth trends *as accurately as possible*, and *as completely as possible*, and that the distribution of the learned repertoire of synopsis node profiles should mirror the incoming stream of evolving data as represented by the ground truth profiles/topic representatives. *Accuracy* can be measured based on the *precision* of the learned synopsis node profiles, $\mathbf{p}_{\mathbf{s}_i}$ relative to the ground truth profile for category c, $\mathbf{p}_\mathbf{c}$, while *completeness* can be measured based on the *coverage* of the learned synopsis node profiles, $\mathbf{p}_{\mathbf{s}_i}$ relative to the ground truth profile for category c, $\mathbf{p}_\mathbf{c}$. Here, precision in the validation phase, describes the accuracy of the synopsis node profiles in representing the ground truth profiles in terms of the number of matching items (URLs or terms) between the learned synopsis profile and the ground truth profiles. Let k denote the item index, and N_I denote the total number of items in the data stream (such as URLs in clickstreams or terms in text documents). Let t denote the time index or the order of the most recent/current input $\mathbf{x}_t$ from the data stream. t is assumed to increase by 1 with each new input from the data stream. Here $x_{t,k} = 1$ if $\mathbf{x}_t$ contains the kth item, and 0 otherwise. Let c denote the category index for the cth ground truth profile $\mathbf{p}_\mathbf{c}$. Here $p_{c,k} = 1$ if $\mathbf{p}_\mathbf{c}$ contains the kth item, and 0 otherwise. Let i denote the index for the ith synopsis node and $N_{P_s}(t)$ denote the total number of synopsis nodes at time t. Let $\mathbf{p}_{\mathbf{s}_i}(t)$ denote the ith synopsis node at time t. $p_{s_{i,k}}(t) = 1$ if $\mathbf{p}_{\mathbf{s}_i}(t)$ contains the kth item, and 0 otherwise. Note that the previous binary vector notations facilitate our presentation of the validation metric equations below. However, for implementation purposes, the synopsis node profiles, the input sessions or documents, and the ground truth profiles are represented as lists of items to avoid the lengthy and yet very sparse vector representation, and thus save significant amounts of memory and computations. Let $\mathbf{P_S}(t)$ be the set of all $N_{P_s}(t)$ synopsis nodes at time t. Then the precision of the ith synopsis node $\mathbf{p}_{\mathbf{s}_i}(t)$, relative to the ground truth profile for the cth category $\mathbf{p}_\mathbf{c}$, is defined as

$$Prec(\mathbf{p}_{\mathbf{s}_i}(t), \mathbf{p}_\mathbf{c}) = \frac{\sum_{k=1}^{N_I} p_{s_{i,k}}(t) \times p_{c,k}}{\sum_{k=1}^{N_I} p_{s_{i,k}}(t)}, \tag{11}$$

while the coverage of the ith synopsis node $\mathbf{p}_{\mathbf{s}_i}(t)$, relative to the the cth ground truth profile $\mathbf{p}_\mathbf{c}$, is defined as

$$Covg(\mathbf{p}_{\mathbf{s}_i}(t), \mathbf{p}_\mathbf{c}) = \frac{\sum_{k=1}^{N_I} p_{s_{i,k}}(t) \times p_{c,k}}{\sum_{k=1}^{N_I} p_{c,k}}, \tag{12}$$

The above measures evaluate the quality of an individual synopsis node $\mathbf{p}_{\mathbf{s}_i}(t)$ at time t, and need to be aggregated over the entire synopsis $\mathbf{P_S}(t)$ to assess its quality as follows in terms of precision and in terms of coverage, respectively.

$$S_P^{\alpha_P}(t,c) = Prec^{\alpha_P}(\mathbf{P_s}(t), \mathbf{p_c}) = \begin{cases} 1 & \text{if } \max_{i=1}^{N_{P_s}(t)}\{Prec(\mathbf{p}_{\mathbf{s}_i}(t), \mathbf{p}_\mathbf{c})\} > \alpha_P, \\ 0 & \text{otherwise.} \end{cases} \tag{13}$$

$$S_C^{\alpha_C}(t,c) = Covg^{\alpha_C}(\mathbf{P_s}(t), \mathbf{p_c}) = \begin{cases} 1 & \text{if } \max_{i=1}^{N_{P_s}(t)}\{Covg(\mathbf{p}_{\mathbf{s}_i}(t), \mathbf{p}_\mathbf{c})\} > \alpha_C, \\ 0 & \text{otherwise.} \end{cases} \tag{14}$$

$S_P^{\alpha_P}(t,c)$ is a matrix set or more formally a binary relation matrix that describes the *distribution of the precise synopsis nodes per input category c at time t*, relative to a minimum quality level α_P, while $S_C^{\alpha_C}(t,c)$ is a matrix set or more formally a binary relation matrix that describes the *distribution of the complete synopsis nodes per input category c at time t*, relative to a minimum quality level α_C. The two measures can be combined to get an overall quality measure that summarizes the distribution of the learned synopsis nodes that simultaneously achieve a precision level exceeding α_P and a coverage level exceeding α_C, at any time t, as follows

$$S^{\alpha_C,\alpha_P}(t,c) = S_P^{\alpha_P}(t,c) \times S_C^{\alpha_C}(t,c). \tag{15}$$

In order to have an objective evaluation of the synopsis, we must compare the above two matrices with the ones that would result if the actual input data stream object $\mathbf{x}_t$ was used instead of the synopsis at any point t in the data stream sequence, as follows. First we compute analogous quality metrics to the ones defined in (13) and (14), but that take the original data stream as input, while accounting for the past Δt inputs from the stream:

$$D_P^{\alpha_P}(t,c,\Delta t) = Prec^{\alpha_P}(\mathbf{x}_t, \mathbf{p_c}, \Delta t) = \begin{cases} 1 & \text{if } \exists t' \in [t-\Delta t, t] | Prec(\mathbf{x}_{t'}, \mathbf{p}_\mathbf{c}) > \alpha_P, \\ 0 & \text{otherwise.} \end{cases} \tag{16}$$

$$D_C^{\alpha_C}(t,c,\Delta t) = Covg^{\alpha_C}(\mathbf{x}_t, \mathbf{p_c}, \Delta t) = \begin{cases} 1 & \text{if } \exists t' \in [t-\Delta t, t] | Covg(\mathbf{x}_{t'}, \mathbf{p}_\mathbf{c}) > \alpha_C, \\ 0 & \text{otherwise.} \end{cases} \tag{17}$$

Δt is called the *retrospective time span*, and it captures the state of the data stream not only at a single instant t, but rather the cumulative *retrospective* distribution throughout a window spanning the past Δt inputs from the stream. Ideally, Δt should be related to the memorization time span τ. For example, $\Delta t = 0$ would capture an instantaneous snapshot of the input stream with no retrospection into the recent past, while $\Delta t = K\tau$ retrospectively summarizes the recent past $K\tau$ inputs from the data stream. The two previous measures in (16) and (17) can be combined to get an overall reference measure that summarizes the *retrospective* distribution of the input data stream at any time t, as follows

$$D^{\alpha_P,\alpha_C}(t,c,\Delta t) = D_P^{\alpha_P}(t,c,\Delta t) \times D_C^{\alpha_C}(t,c,\Delta t). \tag{18}$$

Finally, we can define two global measures that respectively assess the overall level of precision and the overall level of coverage of the entire synopsis throughout the entire stream lifetime or number of individual data points N_X, and all N_C categories. The *retrospective macro precision* of the learned synopsis relative to the past Δt inputs from the stream is defined as

$$\mathscr{P}(\Delta t) = \frac{\sum_{t=1}^{N_X}\sum_{c=1}^{N_C} S^{\alpha_C,\alpha_P}(t,c) \times D^{\alpha_P,\alpha_C}(t,c,\Delta t)}{\sum_{t=1}^{N_X}\sum_{c=1}^{N_C} S^{\alpha_C,\alpha_P}(t,c)}, \tag{19}$$

while the *retrospective macro coverage* of the learned synopsis relative to the past Δt inputs from the stream is defined as

$$\mathscr{C}(\Delta t) = \frac{\sum_{t=1}^{N_X}\sum_{c=1}^{N_C} S^{\alpha_C,\alpha_P}(t,c) \times D^{\alpha_P,\alpha_C}(t,c,\Delta t)}{\sum_{t=1}^{N_X}\sum_{c=1}^{N_C} D^{\alpha_P,\alpha_C}(t,c,\Delta t)}, \tag{20}$$

$\mathscr{P}(\Delta t)$ measures the proportion of the learned synopsis nodes that are accurate representations of the past Δt inputs from the stream relative to all the synopsis nodes, while $\mathscr{C}(\Delta t)$ measures the proportion of the past Δt inputs from the stream that have been accurately summarized by the learned synopsis.

In addition, we can focus only on the *item-wise* quality of each synopsis node compared to the ground truth profiles, and define two global

measures that respectively assess the overall level of precision and the overall level of coverage of the entire synopsis *at the item level* throughout the entire stream lifetime or number of individual data points N_X, and all N_C categories, as follows:

$$\mathscr{P}_\mu(\Delta t) = \frac{\sum_{t=1}^{N_X}\sum_{c=1}^{N_C} S_P^{\alpha_P}(t,c) \times D^{\alpha_P,\alpha_C}(t,c,\Delta t)}{\sum_{t=1}^{N_X}\sum_{c=1}^{N_C} D^{\alpha_P,\alpha_C}(t,c,\Delta t)}, \quad (21)$$

$$\mathscr{C}_\mu(\Delta t) = \frac{\sum_{t=1}^{N_X}\sum_{c=1}^{N_C} S_C^{\alpha_C}(t,c) \times D^{\alpha_P,\alpha_C}(t,c,\Delta t)}{\sum_{t=1}^{N_X}\sum_{c=1}^{N_C} D^{\alpha_P,\alpha_C}(t,c,\Delta t)}, \quad (22)$$

$\mathscr{P}_\mu(\Delta t)$ is called *retrospective micro precision*, and it measures the proportion of the recent input data stream that has been *precisely* summarized by the learned synopsis at the *item level*; while $\mathscr{C}_\mu(\Delta t)$ is called *retrospective micro coverage*, and it measures the proportion of the recent input data stream that has been *completely* summarized by the learned synopsis at the *item level*. The measures are referred to as *micro* metrics because they assess the individual precision and coverage separately for the synopsis nodes *at the item level*, while the *macro* metrics assess the quality of the entire synopsis in summarizing the input data stream.

All the metrics above are *retrospective* in the sense that they take into account the *recent past* when comparing data stream and synopsis. We do this because while the learned synopsis is expected to adapt to the current input, it is also expected to keep some memory of the recent past. In a sense, this is also done to capture *the conflicting aspects of stability and volatility of the synopsis*, with more emphasis towards the latter as Δt is decreased.

3.3. Validation methodology for single-pass mining of evolving Web data streams

In order to take advantage of the above metrics, we propose a validation methodology that is most useful within the framework of mining *evolving* Web data streams. If this were a simpler non-dynamic framework, then all that would need to be measured would be the precision and coverage/recall of the learned synopsis as a faithful representation of the input data stream. However, in this case, we have to test an additional feature of learning which is the adaptability in the face of evolution. Evolution can be simulated easily if an input data set is pre-partitioned into several subsets, one in each known category. The categories can correspond to class labels available with the data, or they can be categories that are computed and validated using an external technique. In the case of labeled data, such as the 20-newsgroup text data set, the categories or trends are the 20 classes (or newsgroups). In the case of Web clickstreams or user sessions, the categories have been pre-discovered and validated using a third method that mines user profiles from Web user session data. Once the data has been divided into several categories, which are heretoforth called *trends*, different *trend sequencing scenarios* can be formed simply by presenting the data as a stream in the order of the trends. This is captured quantitatively by the retrospective temporal-trend distribution $D^{\alpha_P,\alpha_C}(t,c,\Delta t)$ of the input data stream at time/sequence order point t with respect to the different categories or trends (c). Hence every trend sequencing scenario, which corresponds to a different permutation of the order of presentation of the trends/categories, will be entirely captured in one metric: $D^{\alpha_P,\alpha_C}(t,c,\Delta t)$.

The main idea in the validation procedure is to compare the distribution of the learned synopsis against that of the original input data stream, under a given trend sequencing scenario. This amounts to contrasting the temporal-trend distributions of the learned synopsis from a precision point of view ($S_P^{\alpha_P}(t,c)$) and from a coverage point of view ($S_C^{\alpha_C}(t,c)$) against the retrospective temporal-trend distribution $D^{\alpha_P,\alpha_C}(t,c,\Delta t)$ of the input data stream under that sequencing scenario. For data sets of modest size and modest number of categories or trends (20 newsgroup data), a visualization of the synopsis distribution matrices $S_P^{\alpha_P}(t,c)$ and $S_C^{\alpha_C}(t,c)$ in comparison to the original data distribution $D^{\alpha_P,\alpha_C}(t,c,\Delta t)$ is sufficient. However, in cases, where a visual comparison is not easy, such as for the larger Webclickstream data, that contains 92 trends, this comparison is further captured by the aggregate retrospective micro precision and coverage metrics $\mathscr{P}_\mu(\Delta t)$ and $\mathscr{C}_\mu(\Delta t)$, given in (21) and (22), respectively.

4. Single-pass mining of evolving topics in text data

Clustering is an important task that is performed as part of many text mining and information retrieval systems. Clustering can be used for efficiently finding the nearest neighbors of a document [4], for improving the precision or recall in information retrieval systems [13,28], for aid in browsing a collection of documents [7], for organizing search engine results [31], and lately for the personalization

of search engine results [15]. Most current document clustering approaches work with what is known as the vector-space model, where each document is represented by a vector in the term-space. The latter generally consists of the keywords important to the document collection. For instance, the respective term frequencies (TF) [12] in a given document can be used to form a vector model for this document. In order to discount frequent words with little discriminating power, each term/word can be weighted based on its inverse document frequency (IDF) [12,15] in the document collection. The detection of cluster representatives in text data proceeds in a similar way to the Web usage data. The presence of a URL in Web sessions is analogous to the presence of a keyword in a text document, and a Web user profile is analogous to a topic profile which corresponds to a set of keywords present in a synopsis node profile. In other words, we do not change the learning model, and even use the simplest document vector representation, i.e., binary, to really be able to assess the ability of TECNO-STREAMS to track different topics without any extra help from pre-processing. Of course a generalization to alternate document vector formats is trivial, and would not require any modification to the proposed approach. Here again, we have the choice of using the cosine similarity or the *MinPC* similarity in the learning process, as for the case of tracking evolving Web usage trends.

4.1. Simulation results on the 20 newsgroups data

The 20 mini newsgroups data set [14] is a collection of 2000 messages, collected from 20 different netnews newsgroups. One hundred messages from each of the 20 newsgroups were chosen at random and partitioned by newsgroup name. The list of newsgroups from which the messages were chosen is shown in Table 1. The documents were first pre-processed: This included stripping each news message from the e-mail header and special tags, then eliminating stop words and finally stemming words to their root form using the *rainbow* software package [14]. After pre-processing, 395 words were selected based both on both inverse document frequency (IDF) and picking the top words based on information gain with the class attribute. Consequently, there were 1969 documents with at least one of these selected keywords. We opt to use the simplest document vector representation, i.e., binary. In order to evaluate the ability of TECNO-STREAMS to learn synopsis node profiles that can track and evolve with evolving data, we present the document collection one newsgroup or category/trend at a time (from topics 0 to 19). The ground-truth profiles consist of the set of keywords that occur in at least 20 documents for each newsgroup category separately. We perform learning using the cosine similarity $S_{\cos_{ij}}$ in learning as given by (4), and then again using the *MinPC* similarity $S_{\min_{ij}}$, as given by (8). The control parameter for compression was $K = 10$, and periodical compression every $T = 10$ sessions. The activation threshold was $w_{\min} = 0.375$, the maximum synopsis size $N_{P_{\max}}$ was 50, and forgetting time constant τ was 100. We track the number of synopsis nodes that succeed in learning each one of the 20 ground truth topic profiles after each document is presented, by plotting the matrices that describe the distribution of precise synopsis nodes $S_P^{\alpha_P}(t,c)$ and the distribution of complete synopsis nodes $S_C^{\alpha_C}(t,c)$, given in (13) and (14), respectively.

Table 1
Names of the 20 newsgroups

Class	Class descriptions
0	alt.atheism
1	comp.graphics
2	comp.os.ms-windows.misc
3	comp.sys.ibm.pc.hardware
4	comp.sys.mac.hardware
5	comp.windows.x
6	misc.forsale
7	rec.autos
8	rec.motorcycles
9	rec.sport.baseball
10	rec.sport.hockey
11	sci.crypt
12	sci.electronics
13	sci.med
14	sci.space
15	soc.religion.christian
16	talk.politics.guns
17	talk.politics.mideast
18	talk.politics.misc
19	talk.religion.misc

Because this data set is notorious for its large amounts of noise, overlap, and even mislabelings in the documents, tracking the evolving topics one at a time is expected to be challenging. Fig. 3 shows that with the *MinPC* similarity, most of the topics can be detected with decent coverage and precision, as they are gradually presented in a single pass, hence resulting in a staircase pattern except for some synchronizations between several related

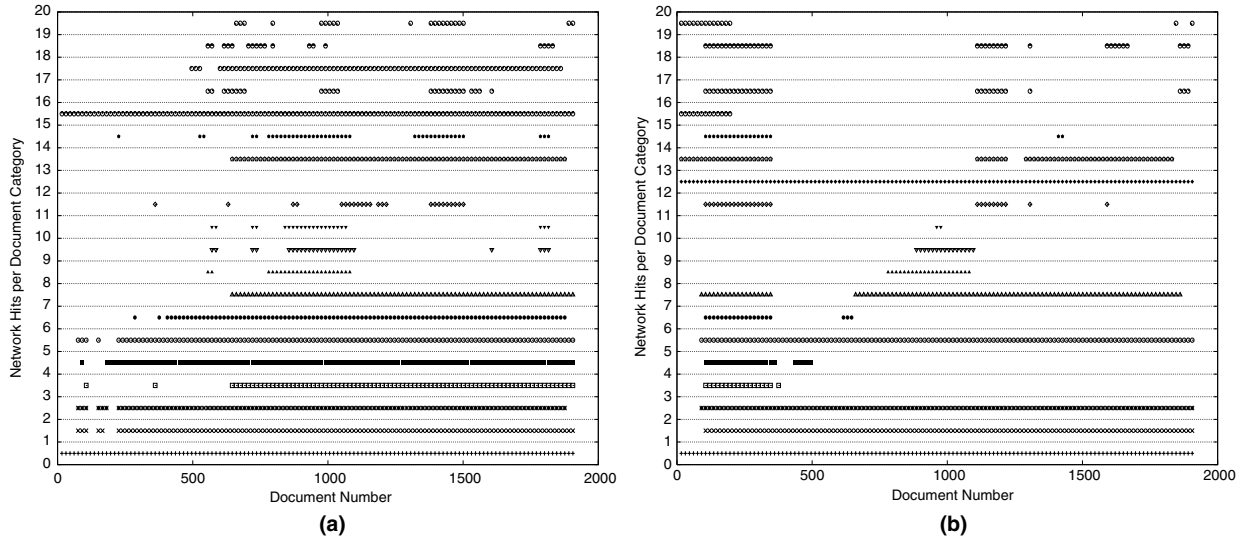

Fig. 3. Hits per usage trend (c) versus document number (t) when documents are presented in order of trend 0 to trend 19 and *MinPC* similarity is used: (a) Precision $S_P^{0.3}(t,c)$, (b) coverage $S_C^{0.5}(t,c)$, showing more hits compared to the results with the cosine similarity in Fig. 4.

topics. For instance, topics 0 (alt.atheism) and 15 (soc.religion.christian) are found to be related because of the overlap in their documents, particularly, e-mail messages containing religious arguments and debates. Similarly, topics 1–5 (all the comp.newsgroups) are found to be related as expected. The differences between Fig. 3(a) and (b) indicate that most of the overlapping topics are synchronized with respect to coverage, but not with respect to precision. This is a very desirable property that further asserts the importance of both precision and coverage in evaluating the learned profile summaries and how they interact. While overlapping subjects may register high coverage with respect to each other, precision should be more restrictive, to keep a better distinction between the specific categories.

On the other hand the cosine similarity manages well in detecting high precision profiles as shown in Fig. 4(a). However, with respect to coverage, it can-

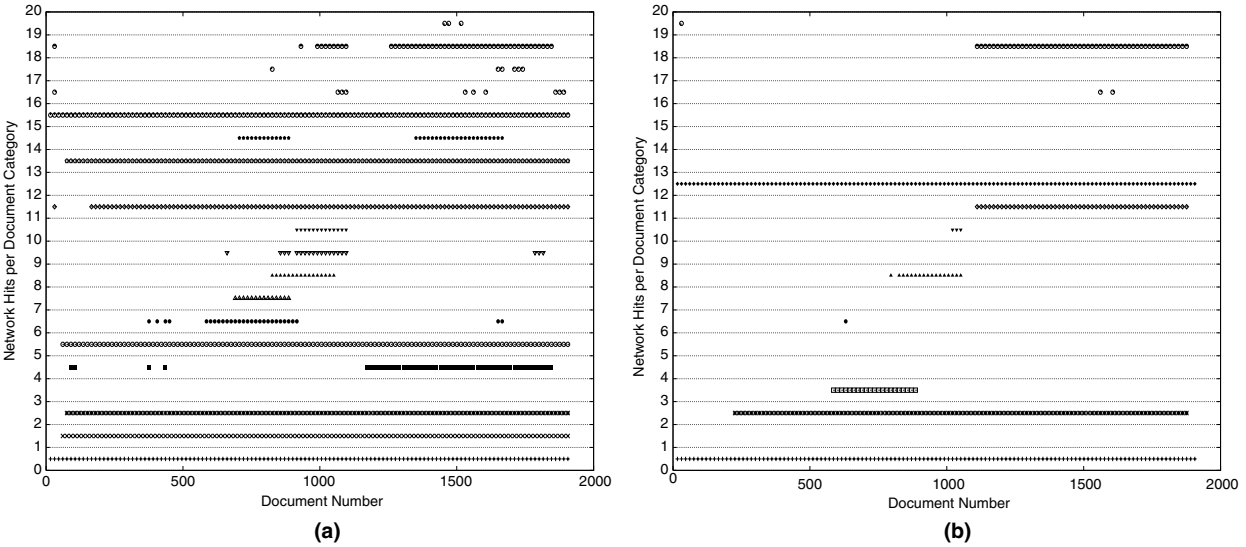

Fig. 4. Hits per usage trend (c) versus document number (t) when documents are presented in order of trend 0 to trend 19 and cosine similarity is used: (a) Precision $S_P^{0.3}(t,c)$, (b) coverage $S_C^{0.5}(t,c)$.

not succeed to help learn most of the newsgroup topics, as shown in Fig. 4(b). This confirms that the performance of the *MinPC* similarity is generally better than the cosine similarity, and that this outperformance can be expected to get even higher for data sets that are more challenging in terms of the amount of noise or overlap, or in terms of the sequencing of the topics with respect to each other. It is important to note that our results correspond to a very challenging scenario, where each datum (a user session or a text document) is processed only once.

4.2. Simulation results with single-pass mining of user profiles from real Web clickstream data

Profiles were mined from a clickstream data set consisting of 1,484,449 distinct hits to the main Website of the University of Missouri-Columbia. After pre-processing as explained in [18], 23,938 sessions (a session consists of consecutive and close requests from the same IP address) were extracted accessing a total of 17,595 URLs. The control parameter for compression was $K = 5$, and periodical compression every $T = 10$ sessions. The activation threshold was $w_{\min} = 0.375$. It is interesting to note that the *memorization span* of the network is affected by the parameter τ which affects the rate of forgetting in the stream synopsis. A low value will favor faster forgetting, and therefore a more current set of profiles that reflect the most recent activity on a Website, while a higher value will tend to keep older profiles in the network for longer periods. Another important parameter is the *maximum synopsis size* of the network (maximum number of nodes) $N_{P_{\max}}$ which can be considered as the number of resources available to make up the stream synopsis. A low value will require a stream synopsis of

Table 2
A sample of the 93 profiles discovered by H-UNC from the MU main Website data

c	$\lvert P_c \rvert$	Description
8	111	Accesses to /~engmo/amlit.html: English professor's American literature page
11	64	Accesses to /~elliswww pages: Ellis library electronic catalog
16	103	Accesses to /mu/academic.html and /~regwww pages: academic course registration
24	59	Accesses to / and /~regwww/admission pages: main page, admissions, application
35	36	Accesses to / and /~komu pages: local public broadcasting TV station
39	83	Accesses to /~c639692/blend.html (student offering color blending program)
40	51	Accesses to / and /~c639692/exp pages: student pages about the Euler number
51	166	Accesses to /~c717733/funnies (student offering jokes' page)
53	158	Accesses to /~c617756 pages (student dedicating page to actor Antonio Banderas)
60	257	Accesses to /~c641644 pages: student dedicating page to music group Nirvana
66	161	Accesses to / and /~jschool pages: Journalism school
75	260	Accesses to /~c690403/dmb pages (student dedicating page to music band)

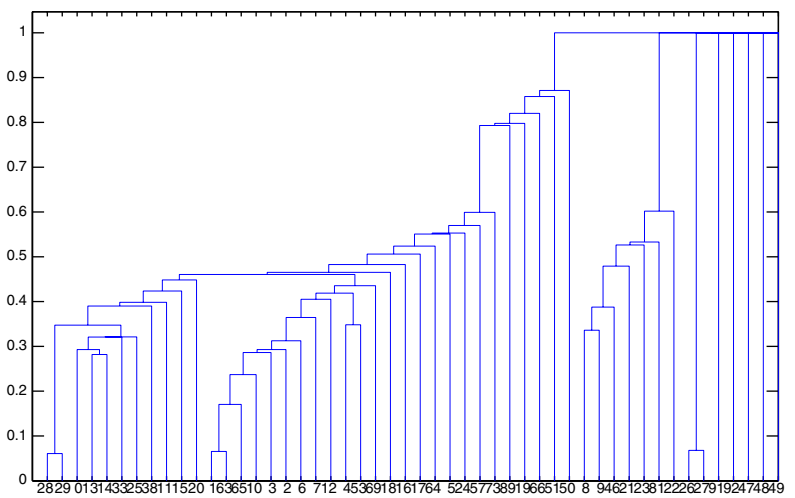

Fig. 5. Dendrogram of the ground-truth profiles based on inter-class distances.

more modest size that can fit in smaller memory size (hence more useful for stream-mining applications), while a higher value will tend to require more memory, and is therefore more costly. Hence the maximum synopsis size $N_{P_{\max}}$ and forgetting time constant τ were set to different combinations to test the effect of the amount of resources available to the synopsis in terms of *space constraints* ($N_{P_{\max}} = 150$, 300) and *memorization span* ($\tau = 50{,}250$).

We illustrate the *continuous* learning ability of the proposed technique using the following simulations:

Scenario 1: Ascending order/drastic changes: We partition the Web sessions into 93 distinct sets of sessions, each one assigned to the closest of 93 profiles previously discovered and validated using Hierarchical Unsupervised Niche Clustering (HUNC) [18], and listed in Table 2. Then we presented these sessions to TECNO-STREAMS one profile at a time: sessions assigned to trend 0, then sessions assigned to profile 1,..., etc. This scenario emphasizes *drastic changes* in user access patterns, where the user activity changes from one category to a different one at certain points in time.

Scenario 2: Regular or natural order/mild changes: The Web sessions are presented in their natural chronological order exactly as received in real time by the Web server. This scenario generally results in more continuous and less drastic changes compared to scenario 1, and is therefore termed *mild changes*.

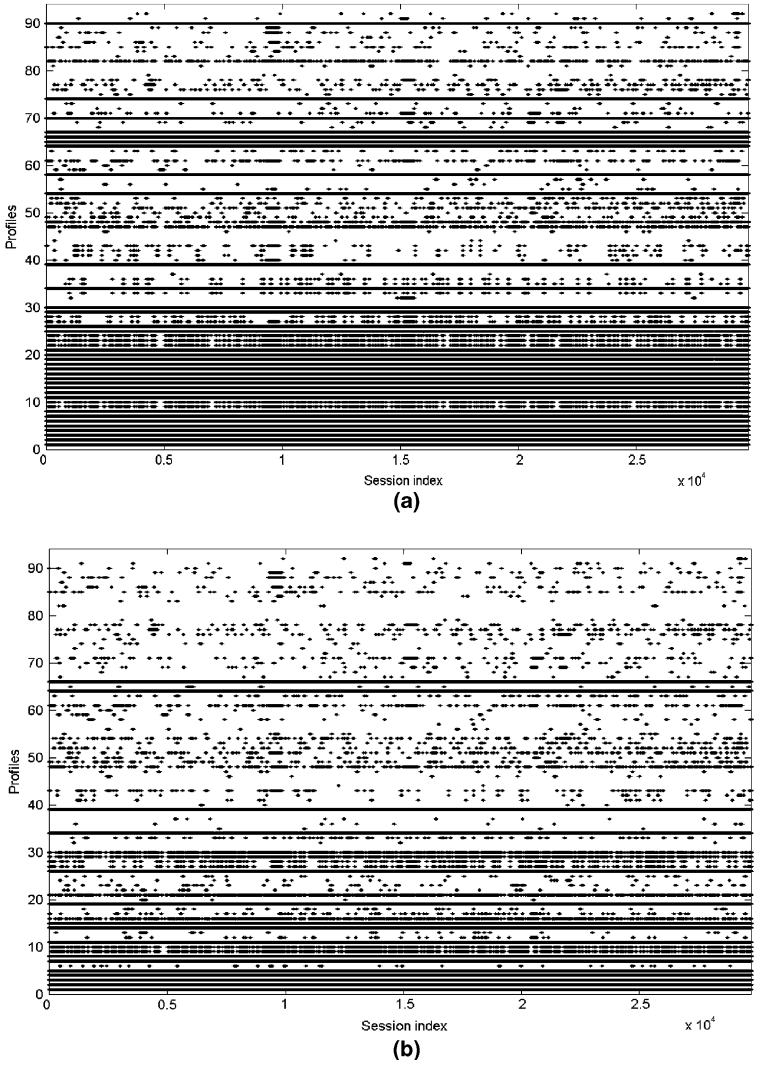

Fig. 6. Hits per usage trend (c) versus session number (t) when sessions are presented in *natural order: mild changes* (scenario 2) and *MinPC* similarity is used: (a) Precision $S_P^{0.3}(t,c)$, (b) coverage $S_C^{0.3}(t,c)$.

For each of the above scenarios, we repeated the experiment using cosine similarity $S_{\cos_{ij}}$ in learning as given by (4), and then again using the *MinPC* similarity $S_{\min_{ij}}$ as given by (8).

Fig. 5 shows a dendrogram of the classes based on the inter-class distance values, where the distance is given by (1 minus cosine similarity). Thus, classes 28 and 29 (which have the same URLs with different weights) appear the closest. Here we can see three big groups: one including Class 0, another including Class 1, and another including Classes 8 and 9. From here we could expect some interactions in learning the classes within each group.

The immune clustering algorithm could learn the user profiles in a single pass with a maximum synopsis size of 150 nodes and $\tau = 50$. A single pass over all 23,938 Web user sessions (with non-optimized Java code) took 6 min on a 2 GHz Pentium 4 PC running on Linux. With an average of *0.02 s per user session*, our profile mining system is suitable for use in a real time personalization system to constantly and continuously provide a fresh and current list of an unknown number of evolving 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

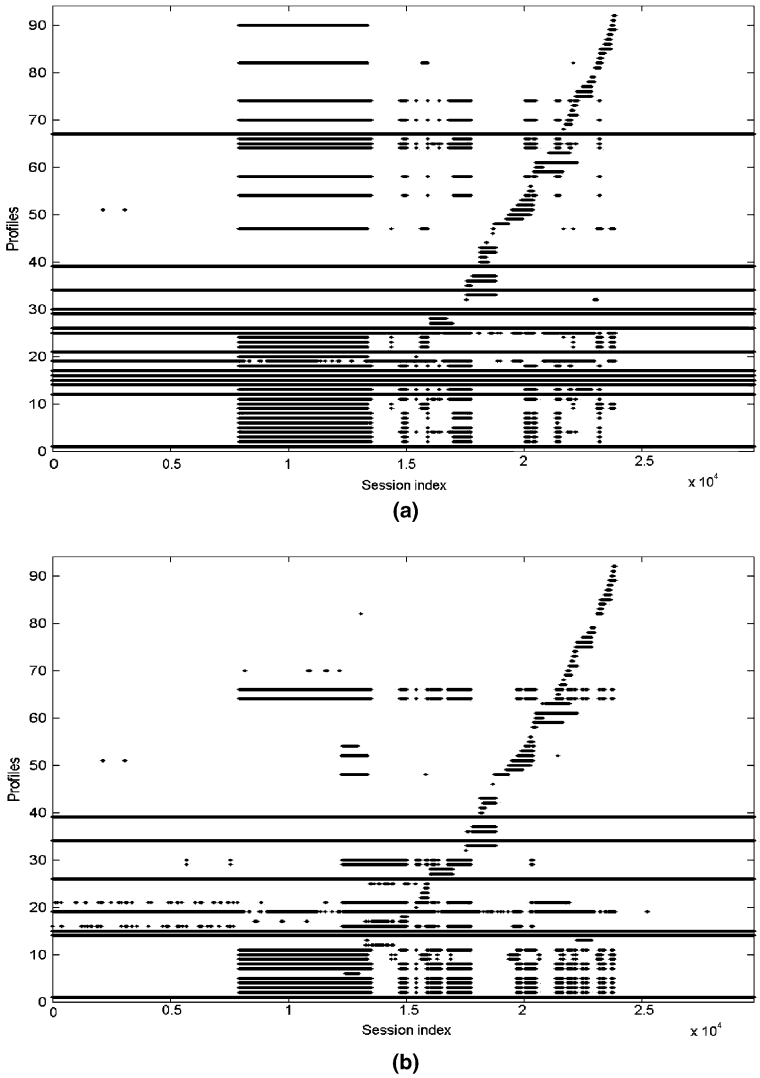

Fig. 7. Hits per usage trend (c) versus session number (t) when sessions are presented in *ascending order: drastic changes* (scenario 1) and *MinPC* similarity is used: (a) Precision $S_P^{0.3}(t,c)$, (b) coverage $S_C^{0.3}(t,c)$.

if discarded, older profiles that re-emerge later, would be re-learned from scratch just like new profiles. Hence the logistics of maintaining old profiles are less crucial compared to existing techniques.

We plot the matrices that describe the distribution of precise synopsis nodes $S_P^{\alpha_P}(t,c)$ and the distribution of complete synopsis nodes $S_C^{\alpha_C}(t,c)$, given in (13) and (14), respectively. This provides an evolving number of hits per profile relative to each of the above criteria, as shown in Figs. 6 and 7, for the two different trend sequencing scenarios respectively. The y-axis is split into 93 intervals, with each interval devoted to the trend/profile number (c) indicated by the lower value (from 0 to 92). A non-zero value in the matrix $S_P^{\alpha_P}(t,c)$ or $S_C^{\alpha_C}(t,c)$ can be interpreted as a hit for the cth category/trend for session No. t, and is shown using a dot symbol in these figures at location (t,c). The presence of a dot symbol indicates that at least one synopsis node profile has achieved the desired threshold in precision or coverage.

Fig. 7(a) and (b) show the distribution of complete synopsis nodes $S_C^{\alpha_C}(t,c)$, and the distribution of precise synopsis nodes $S_P^{\alpha_P}(t,c)$, respectively, for

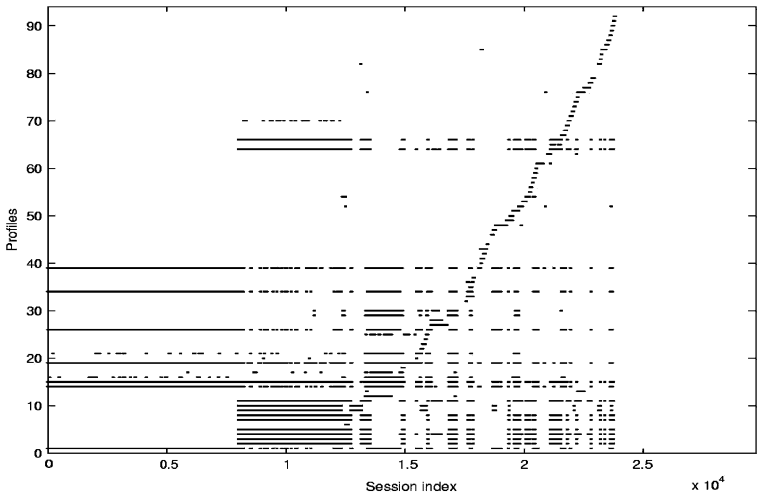

Fig. 8. Temporal-trend distribution $D^{\alpha_P,\alpha_C}(t,c,0)$ shown as hits in usage trend (c) versus session number (t) of the input data stream when sessions are presented in *ascending order: drastic changes* (scenario 1).

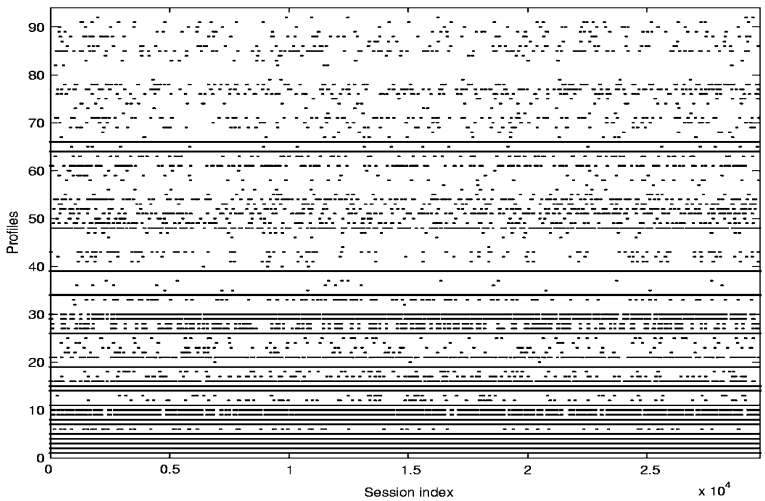

Fig. 9. Temporal-trend distribution $D^{\alpha_P,\alpha_C}(t,c)$ shown as hits in usage trend (c) versus session number (t) of the input data stream when sessions are presented in *natural order: mild changes* (scenario 2).

the *MinPC* similarity, when scenario 1 is deployed for sequencing the usage trends. They both exhibit an expected *staircase pattern*, similar to the original data distribution $D^{\alpha_P,\alpha_C}(t,c,0)$, in Fig. 8, hence proving the gradual learning of emergent usage trends as these are experienced by the stream synopsis in the order from trend 0 to 92. The plot shows some peculiarities, for example trend 28 records hits at the same time as trend 29. Fig. 5 shows that these trends do indeed share a high similarity. Typical cross reactions between similar patterns are actually desired and illustrate a certain tolerance for inexact matching.

Finally Fig. 6(a) and (b) show the distribution of complete synopsis nodes $S_C^{\alpha_C}(t,c)$, and the distribution of precise synopsis nodes $S_P^{\alpha_P}(t,c)$, respectively, for the *MinPC* similarity, when the sessions are presented in their regular or natural chronological order corresponding to scenario 2. In this case, the order of presentation of the trends is no longer sequenced in the order of the trend number. Instead, the user sessions are presented in completely natural

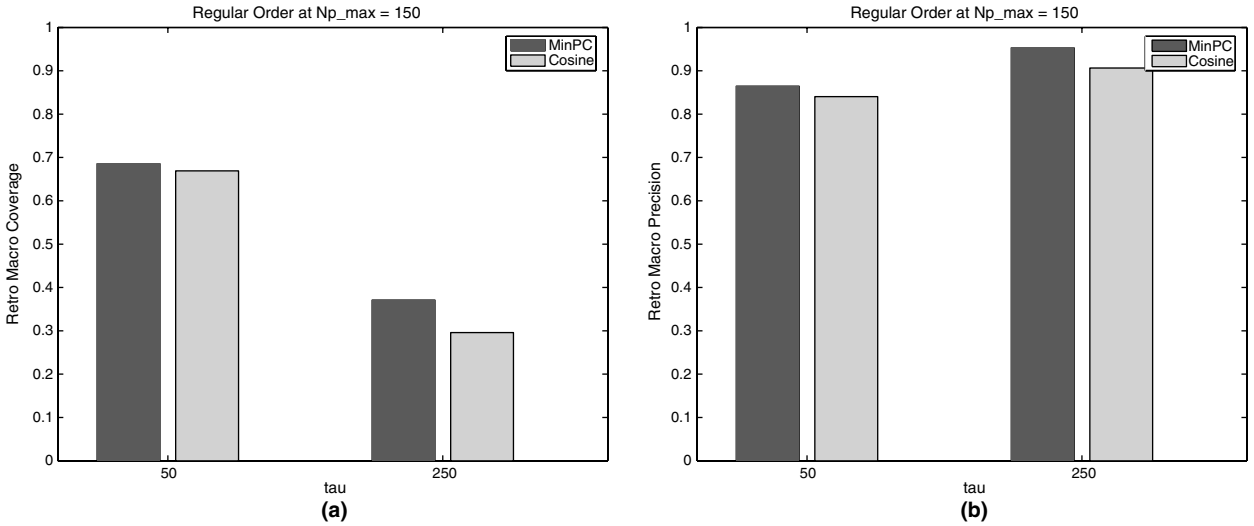

Fig. 10. Average Retro Macro metrics versus τ for *natural order* at $N_{P_{\max}} = 150$: (a) Coverage $\mathscr{C}(\tau)$, and (b) precision $\mathscr{P}(\tau)$.

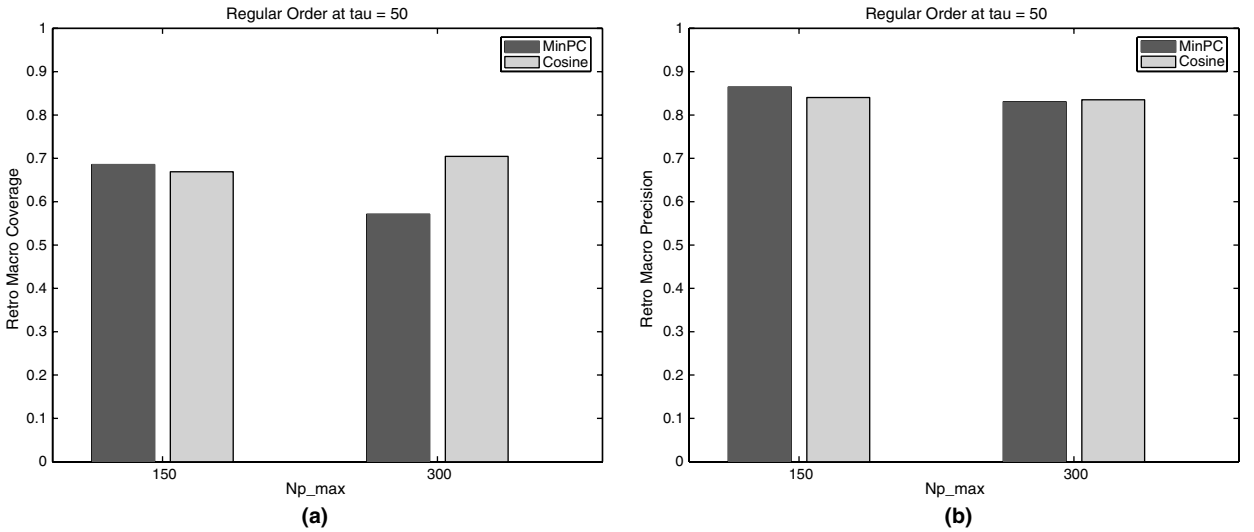

Fig. 11. Average Retro Macro metrics versus $N_{P_{\max}}$ for *natural order* at $\tau = 50$: (a) Coverage $\mathscr{C}(\tau)$, and (b) precision $\mathscr{P}(\tau)$.

(chronological) order, exactly as received by the Web server in real time. So we cannot expect a staircase pattern. In order to visualize the *expected pattern*, we simply plot the distribution of the original input sessions, as captured by $D^{\alpha_P,\alpha_C}(t,c,0)$, in Fig. 9. This figure shows that the session data is quite noisy, and that 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 are not synchronized with others, and how some of the trends (e.g.: No. 40–45) are weak and noisy. Such weak profiles can be even more elusive to discover in a real time Web mining system. While Fig. 6(a) and (b) show the high coverage and high precision synopsis node distribution with time, Fig. 9 shows the distribution of the input data with time. The fact that all these figures show a striking similarity in the emergence

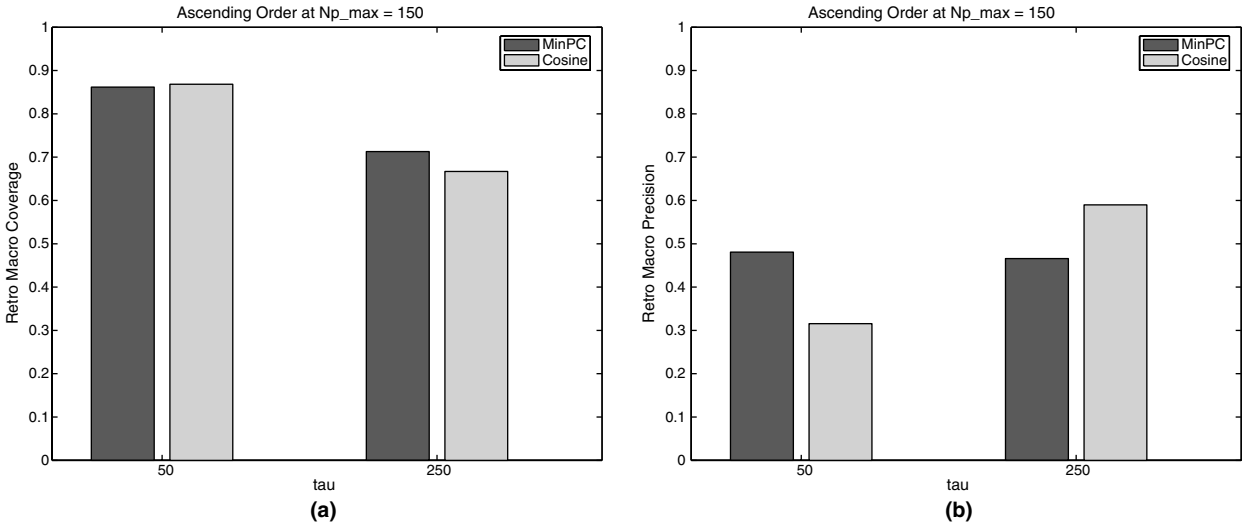

Fig. 12. Average Retro Macro metrics versus τ for *ascending order* at $N_{P_{\max}} = 150$: (a) Coverage $\mathscr{C}(\tau)$, and (b) precision $\mathscr{P}(\tau)$.

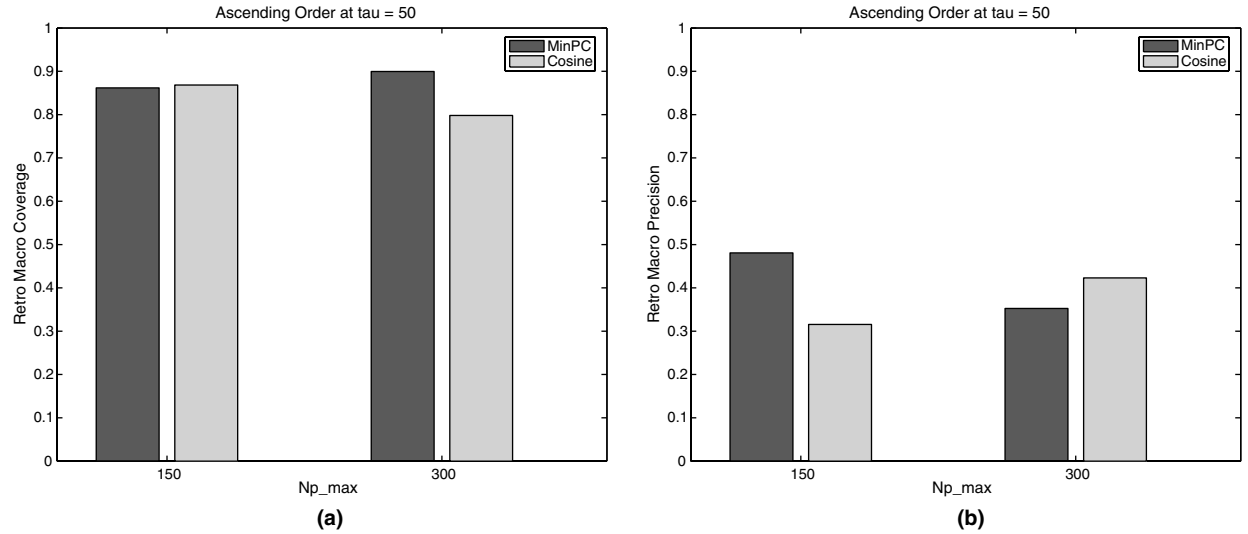

Fig. 13. Average Retro Macro metrics versus $N_{P_{\max}}$ for *ascending order* at $\tau = 50$: (a) Coverage $\mathscr{C}(\tau)$, and (b) precision $\mathscr{P}(\tau)$.

patterns of the trends, attests to the fact that the stream synopsis is able to form a reasonable dynamic synopsis of the usage data, even after a *single pass* over the data.

To further assess the results for this data set, the retrospective macro metrics $\mathscr{P}$ and $\mathscr{C}$ at $\Delta t = \tau$ were averaged over all quality levels (0.1–0.9 in 0.2 increments) of α_P amd α_C respectively, and then plotted versus the forgetting time constant τ in Fig. 10, and versus the maximum synopsis size $N_{P_{\max}}$ in Fig. 11 for the *regular or natural order* (*mild changes*). They are also plotted versus the forgetting time constant τ in Fig. 12, and versus the maximum synopsis size $N_{P_{\max}}$ in Fig. 13 for the *ascending order* (*drastic changes*). From these figures, we see that in the *regular or natural order* (*mild changes*), the MinPC similarity achieves better macro coverage and precision levels, particularly for smaller synopsis size ($N_{P_{\max}} = 150$) compared to the cosine similarity, which only achieves higher coverage for twice the synopsis size (300). This means that under harsh constraints that limit the resources in terms of space requirements to learn a good synopsis, the MinPC has a slight advantage. This is a desired

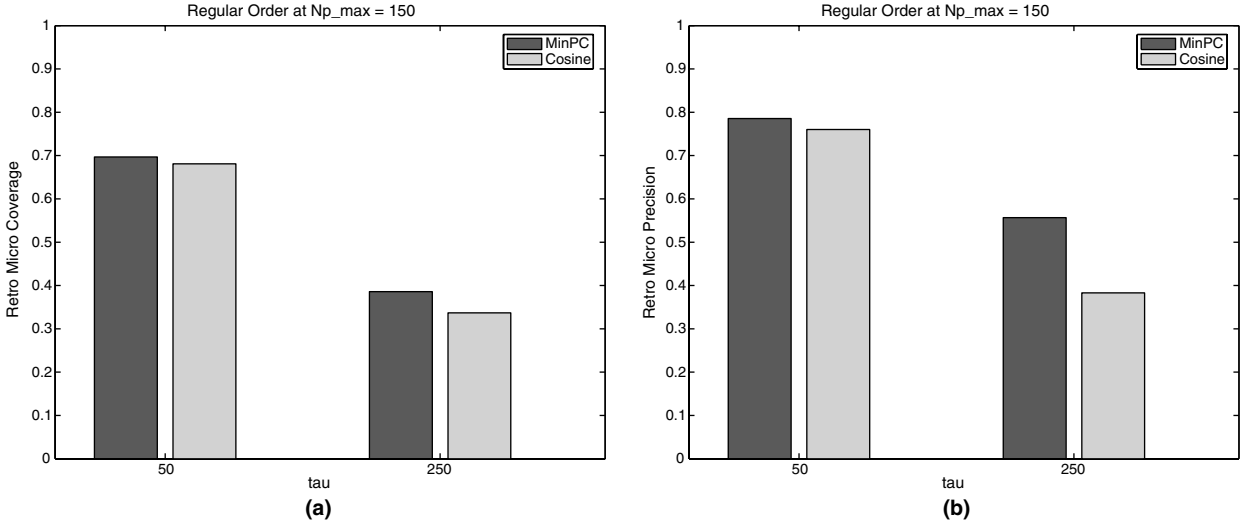

Fig. 14. Average Retro Micro metrics versus τ for *natural order* at $N_{P_{\max}} = 150$: (a) Coverage $\mathscr{C}_\mu(\tau)$, and (b) precision $\mathscr{P}_\mu(\tau)$.

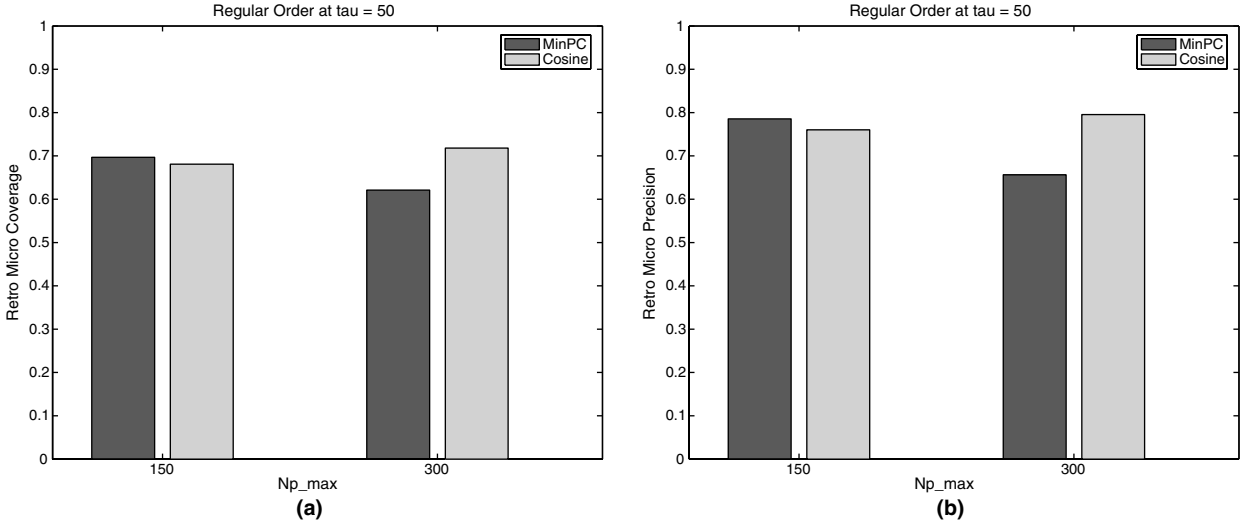

Fig. 15. Average Retro Micro metrics versus $N_{P_{\max}}$ for *natural order* at $\tau = 50$: (a) Coverage $\mathscr{C}_\mu(\tau)$, and (b) precision $\mathscr{P}_\mu(\tau)$.

property in most massive data streaming applications, where memory space tends to be severely limited compared to the data throughput. It is also possible to see that in the case of *ascending order* (*drastic changes*), the MinPC similarity achieves better macro coverage compared to the cosine similarity. However, macro precision is better only with smaller synopsis size and with a shorter memorization span, because, as these parameters increase, the amount of memory (of older profiles) kept in the synopsis is also increased. This in turn can adversely affect the macro precision metric because in addition to the *current* synopsis nodes adapted to the current environment, there is also a bulk of redundant and older synopsis nodes that are kept in the synopsis for sometime. It appears that, when given twice the resources in terms of space (synopsis size) and a five times longer memorization span, cosine overcomes its limitations and achieves the same level of quality as MinPC, but only from a precision point of view. Coverage remains lower.

The Retro Micro metrics $\mathscr{P}_\mu$ and $\mathscr{C}_\mu$ at $\Delta t = \tau$ were also averaged over all quality levels α_P amd α_C respectively, and then plotted versus the forget-

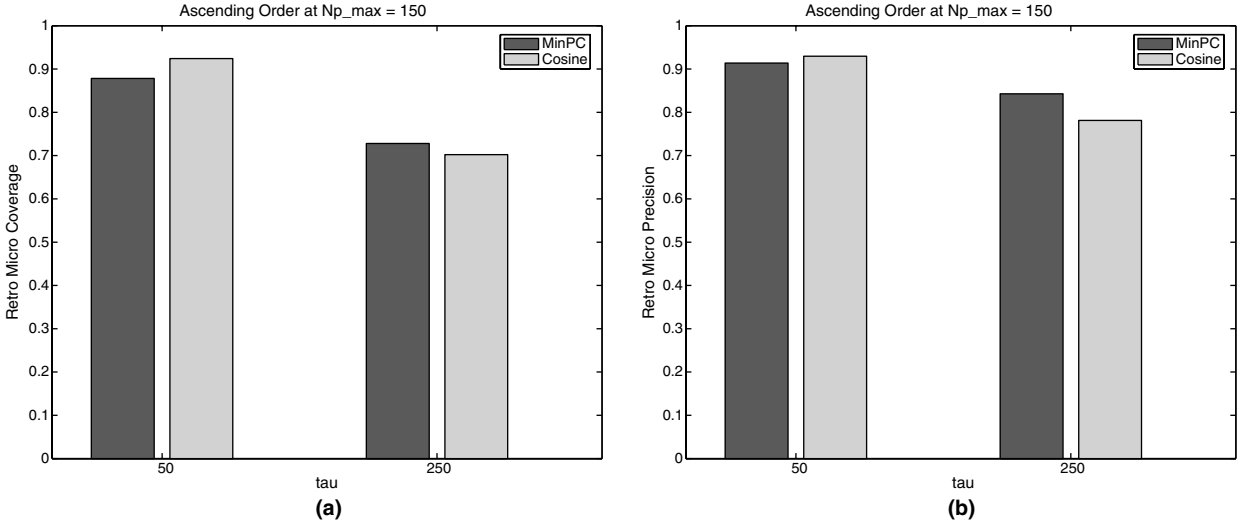

Fig. 16. Average Retro Micro metrics versus τ for *ascending order* at $N_{P_{\max}} = 150$: (a) Coverage $\mathscr{C}_\mu(\tau)$, and (b) precision $\mathscr{P}_\mu(\tau)$.

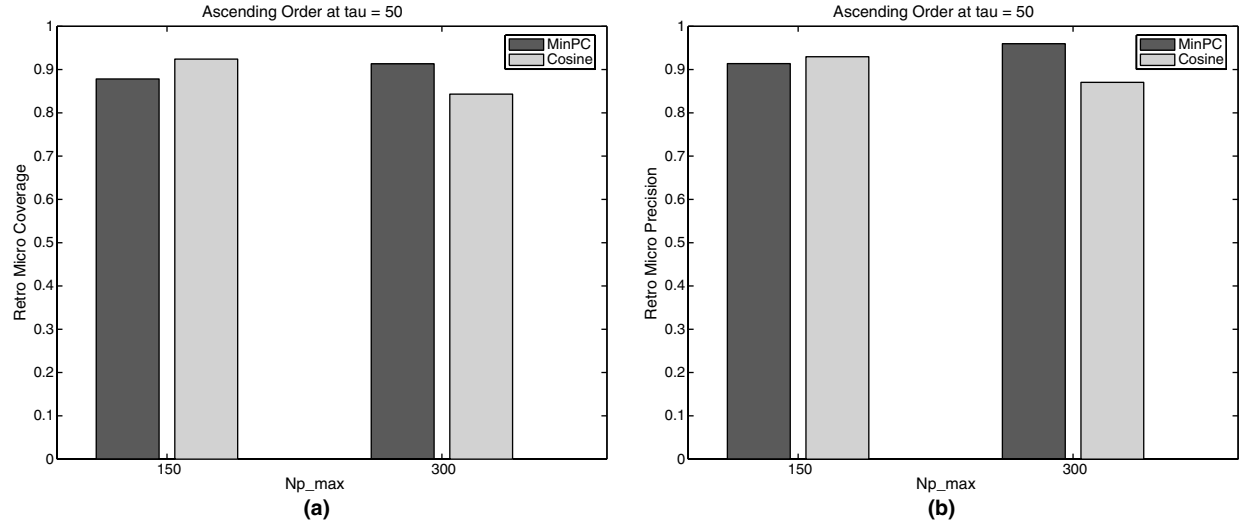

Fig. 17. Average Retro Micro metrics versus $N_{P_{\max}}$ for *ascending order* at $\tau = 50$: (a) Coverage $\mathscr{C}_\mu(\tau)$, and (b) precision $\mathscr{P}_\mu(\tau)$.

ting time constant τ in Fig. 14, and versus the maximum synopsis size $N_{P_{\max}}$ in Fig. 15 for the *regular or natural order* (*mild changes*). They are also plotted versus the forgetting time constant τ in Fig. 16, and versus the maximum synopsis size $N_{P_{\max}}$ in Fig. 17 for the *ascending order* (*drastic changes*). From these figures, we see that in the *regular or natural order* (*mild changes*), both the MinPC and cosine similarity measures achieve similar micro coverage and precision. However, for MinPC, the micro metrics are generally higher with smaller synopsis size ($N_{P_{\max}} = 150$). This means that under harsh constraints that limit the resources in terms of space requirements to learn a good synopsis, the MinPC has a slight advantage. It is also possible to see that in the case of *ascending order* (*drastic changes*), the MinPC similarity achieves better micro coverage and micro precision compared to the cosine similarity. This means that under harsh constraints and drastic changes, the MinPC has the advantage.

We can conclude that the gap between the *MinPC* and cosine similarities, in the number and fidelity of learned high-precision and high-coverage profiles compared to the incoming stream of evolving trends, gets wider when the trends are presented one at a time (scenario 1: drastic changes) as opposed to when they are presented in a more random, alternating order (scenario 2: mild changes). Note that scenario 1 is much more challenging than scenario 2, and it was simulated intentionally to test the ability to learn completely new and unseen patterns (usage trends, topics, etc.), even after settling on a stable set of learned patterns before. In other words, this scenario represents an extreme test of the adaptability of the single-pass Web mining system.

Figs. 18 and 19 show the effect of the past retrospective span Δt on the retro metrics versus $(\Delta t/\tau)$, showing that with increasing retrospection into the past, coverage decreases, since we are comparing the synopsis against an entire span of the input data stream, while the precision increases, showing that the synopsis consists of not only instantly adapted nodes, but also nodes that form a memory of the recent past. However, the gap between MinPC and Cosine gets wider with increasing retrospection into the past, even though they both start at the same level with no retrospection ($\Delta t = 0$). This shows that the quality of the synopsis with MinPC is slightly better from a persistence/memory point of view, while keeping the same adaptation level. In other words, Cosine results in a more volatile synopsis.

We finally comment on the relatively low macro precision values without retrospection into the recent past, i.e., with $\Delta t = 0$, by noting that the search for good synopsis nodes in TECNOSTREAMS is based on an evolutionary type of optimization strategy inspired by the immune system. This strategy is based on cloning and a population of candidate synopsis nodes. Hence, redundancy of the nodes is a natural by-product which actually enhances the search process with a pool of candidates that further cooperate, instead of an individual candidate. In addition to the

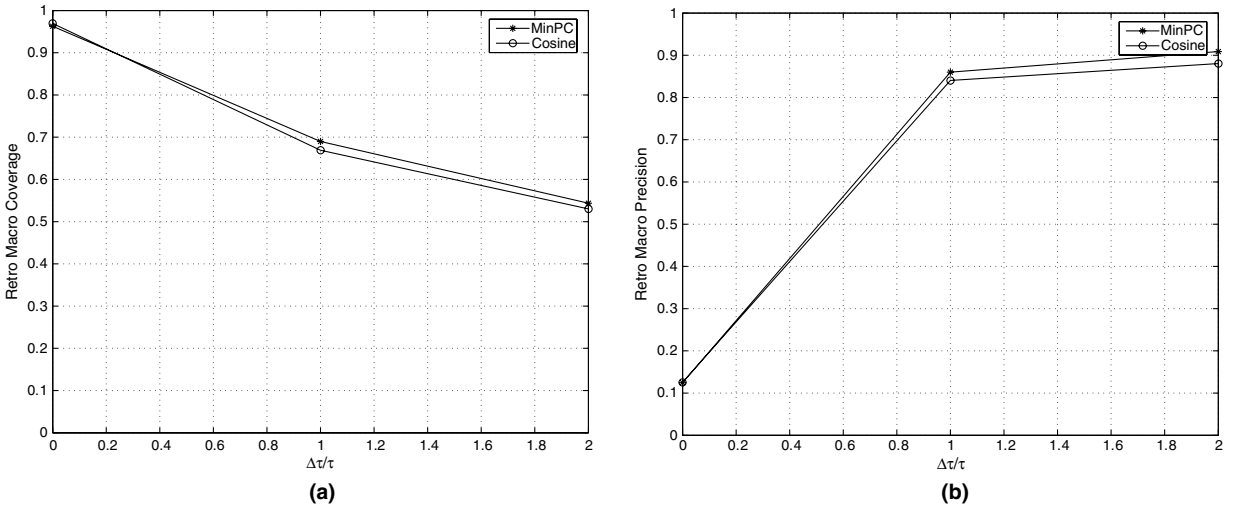

Fig. 18. Average Retro Macro metrics versus $\Delta t/\tau$ for *natural order* at $N_{P_{\max}} = 150, \tau = 50$: (a) Coverage $\mathscr{C}(\Delta t)$, and (b) precision $\mathscr{P}(\Delta t)$.

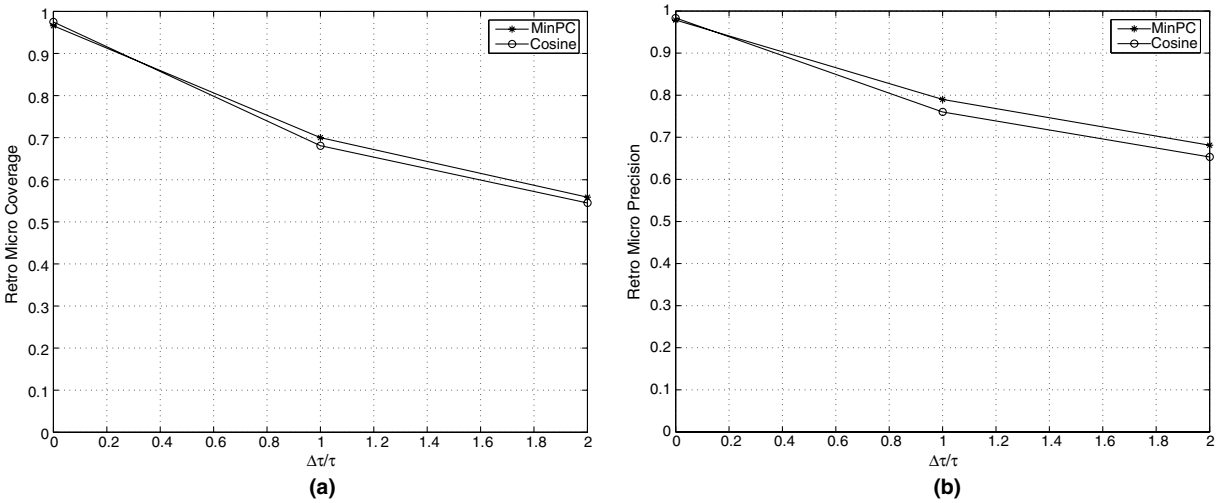

Fig. 19. Average Retro Micro metrics versus $\Delta t/\tau$ for *natural order* at $N_{P_{\max}} = 150, \tau = 50$: (a) Coverage $\mathscr{C}_{\mu}(\Delta t)$, and (b) precision $\mathscr{P}_{\mu}(\Delta t)$.

redundancy, even as the synopsis tries to keep up with new and emerging trends, it still maintains some of the older profiles simply by virtue of being a form of evolving memory of the input data. Yet, when we evaluate the synopsis from a macro precision point of view, we compare all the synopsis nodes against the input data distribution at the current time t only. For this reason, the macro precision can vary depending on the number of current or active trends at time t. The fact that there is a tradeoff between the memorization of older profiles and the adaptation to new profiles is confirmed by the observation that the macro precision seems to be adversely affected by a larger synopsis (higher $N_{P_{\max}}$).

5. Conclusions

We investigated a recently proposed robust and scalable algorithm (TECNO-STREAMS) and the impact of similarity measures on mining an unknown number of evolving profiles or trends in a noisy Web data stream. The main factor behind the ability of the proposed method to learn in a single pass lies in the richness of the immune network structure that forms a dynamic synopsis of the data. TECNO-STREAMS adheres to all the requirements of clustering data streams [2]: *compactness of representation*, *fast incremental processing of new data points*, and *clear and fast identification of outliers*. This is mainly due to the compression mechanism and the dynamic synopsis node model that make the immune network manageable, and continuous learning possible. Furthermore the co-stimulation and co-suppression define implicit pathways of communication between the different elements (synopsis nodes) of the immune network which act like an adaptive and distributed set of agents that track the distribution of the input data stream. This leads to a phenomenon known as *emergence*, where complex and organized global behavior can arise from the interaction of simple local components. Examples can be found in ant colonies, bee swarms and bird flocks [8,9,22]. In the context of mining evolving data streams, this kind of collaborative behavior is expected to enhance memory in a distributed manner, while affecting the dynamics of learning. These crucial characteristics may well be essential to learning and adaptation in a single-pass setting, just as they are crucial to survival in dynamic environments.

Even though measures such as the cosine and Jaccard similarities have been prevalent in the majority of Web clustering approaches, they may fail to explicitly seek profiles that achieve high coverage and high precision *simultaneously*. Because of the paucity of space, we only presented results comparing MinPC with the cosine similarity. However, this is supported by previous research that has shown that the difference between cosine and Jaccard, and most other popular measures tends to be slim in information retrieval applications because they are all monotonic with respect to each other [23]. The MinPC similarity on the other hand, is based on two non-monotonic measures (precision and coverage), and is expected to break this bond

of monotonicity. Our simulations confirmed that the *MinPC* similarity does a better job than cosine similarity in learning from a stream of evolving data in a single pass setting, when the data stream exhibits drastic changes, and under restrictive memory/space allocation (a small synopsis size). This is a desired property in most massive data streaming applications, where memory space tends to be severely limited compared to the data throughput. It is important to note that our results correspond to a very challenging scenario, where each datum (a user session or a text document) is processed only once. It is also important to note that using the minimum of precision and coverage or in fact the minimum of any two drastically different measures leads to a non-differentiable optimization criterion (because of the *Minimum* operator) that rules out using other well known unsupervised learning techniques such as K Means and most of its variants. Because our stream mining approach is not based on gradients for estimating cluster representatives, it can handle such scenarios.

Being able to evaluate and compare different methods in the dynamic stream framework can be a painstaking effort, especially for large data sets with many topics/trends. Hence, we have also presented an innovative strategy to evaluate the discovered profiles/trends using specialized metrics and a simple visualization method showing the hits based on the two criteria of high precision and high coverage separately. The differences between the visualizations with these two criteria (such as between Fig. 4(a) and (b)) indicate that most of the overlapping topics should be synchronized with respect to coverage, but not so much with respect to precision. This is a very desirable property that further asserts the importance of both precision and coverage in evaluating the learned profile summaries and how they interact. While overlapping topics may register high coverage with respect to each other, it is preferable that precision be more restrictive, lest the distinction between the specific categories be compromised. Hence even as a visualization strategy, our plots can provide rich information in terms of both overlap and specificity of the evolving trends in a dynamic scenario.

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. Our approach is modular and generic enough that it can be extended to handle richer Web object models, such as more sophisticated Web user profiles and Web user sessions, or more elaborate text document representations. The only module to be extended would be the similarity measure that is used to compute the stimulation levels controlling the survival, interaction, and proliferation of the learned synopsis node profiles.

In the future, we plan to further investigate validation metrics that are specifically targeted at the framework of mining evolving data streams, and study their prediction ability for the performance of real-time dynamic personalization strategies, such as recommender systems that rely on the continuously evolving stream synopsis as a knowledge base of dynamic profiles.

Acknowledgement

This work is supported by the National Science Foundation CAREER Award IIS-0133948 to O. Nasraoui and by a fellowship from the Logistics and Distribution Institute (LODI) at the University of Louisville for C. Rojas.

References

[1] S. Babu, J. Widom, Continuous queries over data streams, in: SIGMOD Record'01, 2001, pp. 109–120.
[2] D. Barbara, Requirements for clustering data streams, ACM SIGKDD Explorations Newsletter 3 (2) (2002) 23–27.
[3] J. Borges, M. Levene, Data mining of user navigation patterns, in: H.A. Abbass, R.A. Sarker, C. Newton (Eds.), Web Usage Analysis and User Profiling, Lecture Notes in Computer Science, Springer-Verlag, 1999, pp. 92–111.
[4] C. Buckley, A. Lewit, Optimizations of inverted vector searches, in: SIGIR '85, 1985, pp. 97–110.
[5] Y. Chen, G. Dong, J. Han, B.W. Wah, J. Wang, Multi-dimensional regression analysis of time-series data streams, in: 2002 International Conference on Very Large Data Bases (VLDB'02), Hong Kong, China, 2002.
[6] R. Cooley, B. Mobasher, J. Srivastava, Data preparation for mining world wide web browsing patterns, Journal of Knowledge and Information Systems 1 (1) (1999).
[7] D. Cutting, D.R. Karger, J.O. Pedersen, J.W. Tukey, Scatter/gather: a cluster-based approach to browsing large document collections, in: SIGIR '92, 1992, pp. 318–329.
[8] M. Dorigo, V. Maniezzo, A. Colorni, Ant system: optimization by a colony of cooperating agents, IEEE Transactions on Systems Man and Cybernetics—Part B 26 (1) (1996) 29–41.
[9] R.C. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: Sixth International Symposium on Micro-machine and Human Science, Nagoya, Japan, 1995, pp. 39–43.
[10] S. Guha, N. Mishra, R. Motwani, L. O'Callaghan, Clustering data streams, in: IEEE Symposium on Foundations of Computer Science (FOCS'00), Redondo Beach, CA, 2000.

[11] N.K. Jerne, The immune system, Scientific American 229 (1) (1973) 52–60.
[12] R.R. Korfhage, Information Storage and Retrieval, Wiley, 1997.
[13] G. Kowalski, Information Retrieval Systems—Theory and Implementations, Kluwer Academic Publishers, 1997.
[14] A.K. McCallum, Bow: a toolkit for statistical language modeling, text retrieval, classification and clustering. Available from: <http://www.cs.cmu.edu/mccallum/bow> 1996.
[15] D. Mladenic, Text learning and related intelligent agents, IEEE Expert, July 1999.
[16] O. Nasraoui, C. Cardona-Uribe, C. Rojas-Coronel, Tecnostreams: tracking evolving clusters in noisy data streams with a scalable immune system learning model, in: IEEE International Conference on Data Mining, Melbourne, Florida, November 2003.
[17] O. Nasraoui, D. Dasgupta, F. Gonzalez, An artificial immune system approach to robust data mining, in: Genetic and Evolutionary Computation Conference (GECCO) Late breaking papers, New York, NY, 2002, pp. 356–363.
[18] O. Nasraoui, 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.
[19] O. Nasraoui, R. Krishnapuram, H. Frigui, A. Joshi, Extracting web user profiles using relational competitive fuzzy clustering, International Journal of Artificial Intelligence Tools 9 (4) (2000) 509–526.
[20] O. Nasraoui, R. Krishnapuram, 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.
[21] M. Perkowitz, O. Etzioni, Adaptive web sites: automatically synthesizing web pages, in: AAAI 98, 1998.
[22] C.W. Reynolds, Flocks, herds, and schools: a distributed behavioral model, Computer Graphics (ACM SIGGRAPH '87) 21 (4) (1987) 25–34.
[23] G. Salton, M. McGill, Introduction to Modern Information Retrieval, Mc Graw Hill, 1983.
[24] C. Shahabi, A.M. Zarkesh, J. Abidi, V. Shah, Knowledge discovery from users web-page navigation, in: Proceedings of workshop on research issues in Data engineering, Birmingham, England, 1997.
[25] M. Spiliopoulou, L.C. Faulstich, Wum: a web utilization miner, in: Proceedings of EDBT workshop WebDB98, Valencia, Spain, 1999.
[26] J. Srivastava, R. Cooley, M. Deshpande, P.-N. Tan, Web usage mining: discovery and applications of usage patterns from web data, SIGKDD Explorations 1 (2) (2000) 1–12.
[27] J. Timmis, M. Neal, J. Hunt, An artificial immune system for data analysis, Biosystems 55 (1/3) (2000) 143–150.
[28] C. VanRijsbergen, Information Retrieval, Buttersworth, London, 1989.
[29] T. Yan, M. Jacobsen, H. Garcia-Molina, U. Dayal, From user access patterns to dynamic hypertext linking, in: Proceedings of the 5th International World Wide Web conference, Paris, France, 1996.
[30] O. Zaiane, M. Xin, 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.
[31] O. Zamir, O. Etzioni, O. Madani, R. Karp, Fast and intuitive clustering of web documents, in: KDD'97, 1997, pp. 287–290.

Olfa Nasraoui is an Endowed assistant professor and Director of the Knowledge Discovery and Web Mining Lab at the University of Louisville. Her research activities have included Data Mining, Web mining, Personalization, and Computational Intelligence, with a more recent focus on mining evolving data streams, particularly in the Web domain. She received her PhD from the University of Missouri-Columbia in 1999, and was at the University of Memphis between 2000 and 2004. She is a recipient of the *National Science Foundation CAREER Award*, and of the *best paper award in theoretical developments in computational intelligence* at the Artificial Neural Networks In Engineering conference (ANNIE 2001). Her research is funded by NSF and by NASA. She has served on the program committees of several data and Web mining conferences, and served in organizing and program chairing several conferences and workshops, including notably the WebKDD 2004 and WebKDD 2005 workhops on Web Mining and Web Usage Analysis, that were held in conjunction with the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD). Her URL is http:/www.louisville.edu/~o0nasr01.

Carlos Rojas received his B.S. degree (with honors) in Computer Engineering from the "Universidad Nacional de Colombia", Bogota, Colombia in 1999 and his M.S. degree in Electrical and Computer Engineering from The University of Memphis in 2004. His Masters Thesis work involved the study of novel statistically inspired robust stream clustering techniques. He is currently pursuing a PhD degree in Computer Engineering and Computer Science at the University of Louisville. His research interests include artificial intelligence, machine learning, and data mining.

Cesar Cardona received his B.S. degree (with honors) in Computer Science from the "Universidad Nacional de Colombia", Bogota, Colombia in 1999 and his M.S. degree in electrical and computer engineering from The University of Memphis in 2004, where his Masters Thesis involved new techniques for unsupervised mining of evolving data streams, under the supervision of Professor Olfa Nasraoui.

He joined Magnify Inc. in 2004 to develop decision making tools applied to fraud detection, list selection, offer targeting, and customer retention, particularly using large, complex, and disparate data sets. His research interests include data mining, machine learning, and bio-inspired computing.