A Representation of Time Series for Temporal Web Mining *

Mireille Samia
Institute of Computer Science
Databases and Information Systems
Heinrich-Heine-University Düsseldorf
D-40225 Düsseldorf, Germany
samia@cs.uni-duesseldorf.de

Abstract

Data with temporal information is constantly generated, sampled, gathered, and analyzed in different domains, such as medicine, finance, engineering, environmental sciences, and earth sciences. Temporal Web mining extends temporal data mining and Web mining, and concerns the Web mining of data with significant temporal information. Its main goal is to query local and Web data in real time, analyze these temporal sequences in order to discover previously unknown important temporal information. Using temporal data as temporal sequences without any preprocessing fails to extract key features of this data. For this reason, before applying mining techniques, an appropriate representation of temporal sequences is needed. This paper emphasizes on the representation of time series for temporal Web mining.

1 Introduction

Temporal Web mining (TWM) concerns the Web mining of data with significant temporal information. The data can contain temporal knowledge, but it is often treated as static. Hence, in order to extract important features of this data, we need to find an appropriate representation of our temporal sequences before applying mining techniques. In the representation of time series for TWM, we define two main types of data band ranges. Our first data band range is called the *Dangerous Data Band* (DDB). DDB consists of the data during the occurrence of a significant event, such as the values of the rise in sea-level. Our second data band range, the *Risky Data Band* (RDB), consists of the data before an important event (such as flooding) occurs. RDB can be a clue to forecast a significant event, such as flooding. The main advantages of using DDB and RDB are that we can estimate the weight of each segment and predict its closeness to DDB and RDB in order to find any early sign that can warn of a crucial event (such as flooding).

In this paper, section 2 provides a brief overview of temporal Web mining, and discusses related works. In section 3, we present our representation of time series for temporal Web mining. Section 4 concludes this paper and points out our directions for future work.

2 Overview and Related Work

This section provides an overview of temporal Web mining, and discusses further related work.

2.1 Overview of Temporal Web Mining

Following [9], temporal Web mining is an extension of temporal data mining and Web mining. It can be used in different domains, such as finance, engineering, environmental sciences, medicine, and earth sciences. Temporal data discovered by the application of temporal data mining techniques is used in the Web mining process in order to retrieve useful data with temporal information in real time over the Web. The derived useful data with temporal information

[*] **This work is supported by the German Academic Exchange Service (Deutscher Akademischer Austausch Dienst DAAD).**

discovered by the application of Web mining techniques is used again in the temporal data mining process. Thus, we define Temporal Web Mining (TWM) as the process of discovering, extracting, analyzing, and predicting data with significant temporal information from the temporal information discovered by the application of temporal data mining techniques, and applying Web mining techniques to this data in real time over the Web [9].

According to [10], TWM supports the temporal aspect of Web data by mining Web data with temporal information as temporal data, and not as static data. Its purpose is to introduce prediction as a main issue in Web mining, specifically Web content mining. In other words, TWM aims at predicting temporal data from the content of the Web data. Furthermore, TWM uses Web data, such as temporal data from the Web, in the temporal data mining process [10].

2.2 Further Related Work

In temporal data mining, the representation of data takes place before defining the similarity measures between sequences and applying actual data mining techniques [1]. The data is represented into time series by either keeping it in its original form ordered by their instant of occurrence and without any pre-processing [8], or subsequences of a sequence are obtained using windowing and by finding a piecewise linear function able to approximately describe the entire initial sequence [2, 3]. Another approach [1, 6] to represent data into time series data is segmenting a sequence by iteratively merging two similar segments, that are choosen based on the squared error minimization. An extension to this method is to associate with each segment a weight value, in order to define the importance of each segment according to the entire sequence[5].

In the representation of time series for TWM, we define two main types of data band ranges. Our first data band range, *Dangerous Data Band* (DDB), consists of the data during the occurrence of a significant event, such as the values of the rise in sea level. Our second data band range is called the *Risky Data Band* (RDB). RDB consists of the data before an important event (such as flooding) happens. RDB can be a clue to forecast a significant event, such as flooding. According to DDB and RDB, we can estimate the weight of each segment and predict its closeness to DDB and RDB that help us to find any early sign of a crucial event (such as flooding).

In our TWM representation of time series, we subdivide a sequence to subsequences. Each subsequence is represented as a continuous function defined on a closed interval. Each subsequence is divided into equidistant subintervals in order to represent it by a sequence of straight-line segments. By computing the area under a subsequence and comparing it to the area that separates it from the *Dangerous Data Band* or from the *Risky Data Band*, we can estimate the weight of each segment of a subsequence. Moreover, we can predict its closeness to DDB and RDB in order to discover any early warning of a crucial event (such as flooding).

By representing a subsequence as a continuous function, each sequence is represented as a sequence of the functions of its subsequences. Storing only the functions of the sequence can reduce the storage requirements. Furthermore, because the functions are continuous, new unsampled points can be to a certain extent deduced.

3 TWM Representation of Time Series

This section provides our representation of time series for temporal Web mining.

Specifying the Dangerous Data Band and the Risky Data Band Because we are dealing with temporal data, which can be an early warning sign to predict to a certain extent some important events (such as flooding), we define two main types of data band ranges. In Figure 1, our first data band range, called *Dangerous Data Band* (DDB), consists of the data during the occurrence of a crucial event, such as the values of the rise in sea level. DDB varies between β and λ; i.e. $\beta \leq \text{DDB} \leq \lambda$, where $\beta < \lambda$.

Our second data band range is the *Risky Data Band* (RDB). RDB consists of the data before an important event (such as flooding) occurs. RDB is normally close to DDB. RDB varies between α and β; i.e. $\alpha \leq$ RDB $< \beta$, where $\alpha < \beta$. The data outside DDB or RDB are called Out-Of-Band Data.

According to DDB and RDB, we can specify if a data point (x_i, y_i) is a dangerous point, a risky point or an out-of-band point (i.e. a normal point). More clearly,
if (x_i, y_i) is a dangerous point, then $\beta \leq y_i \leq \lambda$, where $\beta < \lambda$ and $0 \leq i \leq n$
if (x_i, y_i) is a risky point, then $\alpha \leq y_i < \beta$, where $\alpha < \beta$ and $0 \leq i \leq n$
if (x_i, y_i) is an out-of-band point, then ($y_i < \alpha$, where $0 \leq i \leq n$) or ($y_i > \lambda$, where $0 \leq i \leq n$)

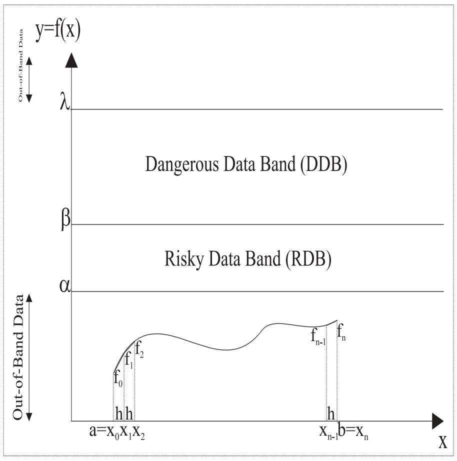

Figure 1: General Representation of a Subsequence in TWM

Finding the Function of a Subsequence In our TWM representation of time series, we subdivide sequences into meaningful subsequences. In Figure 1, we represent each subsequence as a continuous function defined on a closed interval [a, b].

Suppose that a subsequence consists of a set of (n+1) data values ((x_0, y_0), (x_1, y_1), ..., (x_n, y_n)), where x is the controllable variable (such as time) and y is the measured variable (such as temperature). It is generally assumed that such data values may be described by a function. Then, we represent each subsequence as a continuous function defined on a closed interval [a, b]. Weierstrass approximation theorem states that any continuous function defined on a closed interval [a, b] can be uniformly approximated as closely as desired by a polynomial function [7]. Then, from the set of data values of a subsequence, we can get the following unique Newton's polynomial of degree at most n that passes through the (n+1) data points:

$$P_n(x) = a_0 + a_1(\text{x}-x_0) + a_2(\text{x}-x_0)(\text{x}-x_1) + ... + a_n(\text{x}-x_0)(\text{x}-x_1)...(\text{x}-x_{n-1}),$$

where $a_k = \text{f}[x_0, ..., x_k]$, for $k = 0, 1, ..., n$.

$P_n(x)$ represents the polynomial function of a subsequence. By representing a subsequence as a continuous function, each sequence is represented as a sequence of the functions of its subsequences. Because the functions are continuous, they allow prediction of new unsampled points.

Segmenting a Subsequence From the previous subsection, we have the function of a subsequence *y=f(x)* defined on a closed interval [a, b]. In Figure 1, we subdivide the interval [a, b] into n subintervals of length *h*. We calculate the function of every straight line joining two subsequent points (i.e. segment) in order to represent the subsequence by a chain of straight-line segments. The segment joining the data points (x_i, f(x_i)) and (x_{i+1}, f(x_{i+1})) of the subsequence can be evaluated using the following equation:

$$\text{y} = f_i + \frac{1}{h}\,(f_{i+1} - f_i)(x - x_i), \text{ where } f(x_i) = f_i \text{ and } f(x_{i+1}) = f_{i+1}$$

A weight value w_i is assigned to every segment. The weight of the subsequence is the sum of products of the weight of each segment with the straight-line segment of each subinterval, and

can be defined as following:
$\sum_{i=0}^{n-1} w_i(f_i + \frac{1}{h}(f_{i+1} - f_i)(x - x_i))$

After all the subsequences of a sequence are segmented, all the weights are initialized to 1. Consequently, if any of the weights are changed, the weights are renormalized [5]. More clearly, if one of the weights is changed, all the weights are redistributed. For example, if the weight of a segment is decreased, all other segments will have their weight slightly increased [4].

Approximating the Area under a Subsequence To estimate the weight of a subsequence, we compute the area that estimates the area under this subsequence f(x). In Figure 2, subdividing the subsequence into n subintervals of length h and then finding the straight-line segment of each subinterval leads to n trapezoids. Hence, to approximate the area under a subsequence, we can use the trapezoidal rule. The area of the n trapezoids $T_{f_{i-1}f_ix_ix_{i-1}}$ is calculated, and then, added as follows:
$\sum_{i=1}^{n} A_i = \frac{h}{2}(f_0 + f_n + 2\sum_{i=1}^{n-1} f_i)$, where A_i is the area of the i^{th} trapezoid.
Thus, the area under a subsequence is estimated using the trapezoidal rule in the form:
$\sum_{i=1}^{n} A_i = \frac{h}{2}\ (f_0 + f_n) + h\sum_{i=1}^{n-1} f_i$

In Figure 2, we note that the weight of the area of each subinterval A_i is equal to the weight of its straight-line segment $f_{i-1}f_i$.

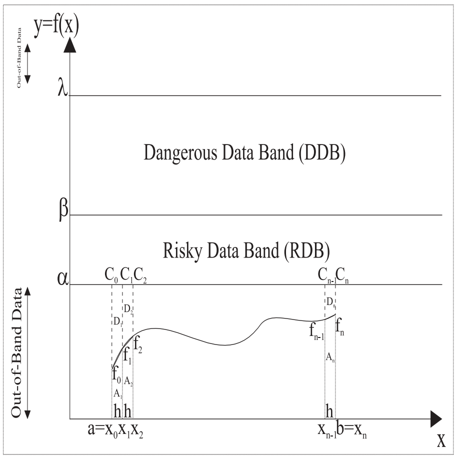

Figure 2: The segments are below RDB and DDB

Estimating the Weight of a Segment To find the weight of a specific segment, we estimate the closeness of this segment to the *Dangerous Data Band* (DDB) or the *Risky Data Band* (RDB) (cf. Figure 2). The closer the segment of a subsequence is to DDB or to RDB, the greater its weight is according to the weight of the entire sequence.

In Figure 2, the closeness of a segment of a subsequence to DDB or to RDB can be estimated by comparing the area A_i of the trapezoid $T_{f_{i-1}f_ix_ix_{i-1}}$ and the area D_i of the trapezoid $T_{f_{i-1}f_iC_iC_{i-1}}$ (where i varies between 1 and n). For instance, the greater the area of the trapezoid $T_{f_{i-1}f_ix_ix_{i-1}}$ is according to the area of the trapezoid $T_{f_{i-1}f_iC_iC_{i-1}}$, the closer the segment $f_{i-1}f_i$ is to DDB or to RDB. In other words, the greater the weight of this segment is according to the weight of the whole subsequence.
In the previous subsection, we found the area A_i of the trapezoid $T_{f_{i-1}f_ix_ix_{i-1}}$. To compute the area D_i of the trapezoid $T_{f_{i-1}f_iC_iC_{i-1}}$, we calculate the area of the rectangle $R_{x_{i-1}x_iC_iC_{i-1}}$. Then, we substract the area A_i of the trapezoid $T_{f_{i-1}f_ix_ix_{i-1}}$ from the area of the rectangle $R_{x_{i-1}x_iC_iC_{i-1}}$. We get the following:

$$D_i = \left| h\psi - \frac{h}{2}(f_i + f_{i-1}) \right| = \left| h(\psi - \frac{1}{2}(f_i + f_{i-1})) \right|,$$

where ψ is the height of the rectangle $R_{x_{i-1}x_iC_iC_{i-1}}$ and $1 \leq i \leq n$
From the value of the maximum of f_{i-1} and f_i, we determine the value of ψ to calculate the area

of the rectangle $R_{x_{i-1}x_iC_iC_{i-1}}$. The length of ψ can be equal to our risky point α, our dangerous point β or our dangerous point λ. More clearly,
if $\max(f_{i-1},f_i) < \alpha$, then $\psi = \alpha$; i.e. the segments are below RDB and DDB (cf. Figure 2).
If $\max(f_{i-1},f_i) > \lambda$, then $\psi = \lambda$; i.e. the segments are above DDB and RDB.
If $\alpha \leq \max(f_{i-1},f_i) < \beta$, then $\psi = \beta$; i.e. the segments belong to RDB.
If $\beta \leq \max(f_{i-1},f_i) \leq \lambda$, then $\psi = \lambda$; i.e. the segments belong to DDB.
Note that if f_{i-1} is equal to f_0 and f_i is equal to f_n, then we can estimate the weight of the whole subsequence corresponding to RDB and DDB.

By defining our two main bands the *dangerous data band* (DDB) and the *risky data band* RDB), we assign for every segment of a subsequence a specific type. In other words, if a segment is in RDB, then, its a risky segment. According to its closeness to *risky data band*, the importance of its weight is considered. If the segment is in DDB, then it is a dangerous segment. The importance of its weight is estimated according to its closeness to the *dangerous data band* limit. If a segment is above DDB or below RDB, then it is a normal segment. In other words, its data values are out-of-band data values. Estimating its closeness to RDB or DDB helps to deduce how the next segment of the same subsequence can be in order to discover any early warning of a significant event (such as flooding).

4 Conclusion and Outlook

Because the interest in extracting and analyzing temporal hidden information grows, temporal Web mining (TWM) extends temporal data mining and Web mining. Its primary goal is to deal with temporal data, such as local and Web data, in real time over the Web.
In our TWM representation of time series, we define two types of data band range, which are called the *Dangerous Data Band* (DDB) and the *Risky Data Band* (RDB). According to DDB and RDB, we can estimate the weight of each segment and forecast its closeness to DDB and RDB in order to find any early sign that can be a clue to a crucial event (such as flooding).

In this paper, we provide a brief overview of temporal Web mining and Web mining, and discuss related work. Then, we present our time series representation for TWM.

A future work includes the definition of similarity measures between sequences in TWM. Enhancing the quality of temporal data improves the data representation of time series.

References

[1] Claudia Antunes and Arlindo Oliveira. Temporal Data Mining: an Overview. In *KDD Workshop on Temporal Data Mining*, pages 1–13, San Francisco, 2001.

[2] Mayur Datar and S. Muthukrishnan. Estimating Rarity and Similarity over Data Stream Windows. In *Proceedings of the 10th European Symposium on Algorithms*, Rome, Italy, September 2002.

[3] Christos Faloutsos, M. Ranganathan, and Yannis Manolopoulos. Fast Subsequence Matching in Time Series Databases. In *ACM SIGMOD International Conference on Management of Data*, pages 419–429, Minneapolis, USA, 1994.

[4] Eamonn Keogh and Michael Pazzani. An Enhanced Representation of Time Series which Allows Fast and Accurate Classification, Clustering and Relevance Feedback. In *Fourth International Conference on Knowledge Discovery and Data Mining (KDD'98)*, pages 239–241, New York City, NY, August 1998. ACM Press.

[5] Eamonn Keogh and Michael Pazzani. Relevance Feedback Retrieval of Time Series Data. In *In Proceedings of SIGIR '99*, Berkley, CA USA, August 1999.

[6] Eamonn Keogh and Padhraic Smyth. A Probabilistic Approach to Fast Pattern Matching in Time Series Databases. In *Third International Conference on Knowledge Discovery and Data Mining*, pages 24–30, Newport Beach, CA, USA, 1997.

[7] H. Poincaré. L'oeuvre mathématique de Weierstrass (The Mathematical Work of Weierstrass). In *Acta Mathematica*, volume 22, pages 1–18, 1899.

[8] Tore Risch and Ling Lin. Querying Continuous Time Sequences. In *VLDB*, pages 170–181, Newport Beach, CA, USA, 1998.

[9] Mireille Samia. Temporal Web Mining. In *15. GI-Workshop über Grundlagen von Datenbanken (15th GI-Workshop on the Foundations of Databases)*, pages 27–31, Tangermünde, Germany, June 2003.

[10] Mireille Samia and Stefan Conrad. From Temporal Data Mining and Web Mining To Temporal Web Mining. In *Proceedings of the Sixth International Baltic Conference on Databases and Information Systems (BalticDB&IS'2004) (to appear)*, Riga, Latvia, June 2004.