CHAPTER

Temporal Data Mining 2

CHAPTER OUTLINE

2.1 INTRODUCTION

Temporal data mining can be defined as "process of knowledge discovery in temporal databases that enumerates structures (temporal patterns or models) over the temporal data, and any algorithm that enumerates temporal patterns from, or fits models to, temporal data is a temporal data mining algorithm" (Lin et al., 2002). The aim of temporal data mining is to discover temporal patterns, unexpected trends, or other hidden relations in the larger sequential data, which is composed of a sequence of nominal symbols from the alphabet known as a temporal sequence and a sequence of continuous real-valued elements known as a time series, by using a combination of techniques from machine learning, statistics, and database technologies. In fact, temporal data mining is composed of three major works including representation of temporal data, definition of similarity measures and mining tasks.

Temporal Data Mining via Unsupervised Ensemble Learning. http://dx.doi.org/10.1016/B978-0-12-811654-8.00002-6

2.2 REPRESENTATIONS OF TEMPORAL DATA

Representation of temporal data refers to how to represent the temporal data in an efficient way before actually mining operations takes place. There are three major methods: time domain–based representations, transformation-based representations, and generative model–based representations.

2.2.1 TIME DOMAIN–BASED REPRESENTATIONS

Time domain–based representations are the simplest way to represent the temporal data with minimal manipulation. It can keep either the original form of temporal data that is a sequence of initial samples ordered in their occurrence within the time domain or segment the temporal sequence into several parts, where each of segmentations is represented by linear functions. Generally, the time domain–based representations has the advantages of easy implementation and preventing from losing information obtained from temporal data. However, it potentially requires demanding computational power and memory resource for the mining operations and may become infeasible for the real-world applications involving the temporal data with large volume and high dimensionality.

2.2.2 TRANSFORMATION-BASED REPRESENTATIONS

Transformation-based representations aim to transfer the original temporal data into a representation space, where those features that contain the most discriminatory information are extracted and used for representing the temporal data. In general, such representations can be classified into two categories: piecewise and global representations. A piecewise representation is generated by partitioning the temporal data into segments at critical points based on a criterion then each segment will be modeled into a concise representation. As a result, all segment representations constitute a piecewise representation collectively, for example, adaptive piecewise constant approximation (Chakrabarti et al., 2002) and curvature-based Principal Component Analysis (PCA) segments (Bashir, 2005). On the other hand, a global representation is derived by modeling the temporal data via a set of basic functions, and therefore, coefficients in the parameter space forms a global representation that can be used to reconstruct the temporal data approximately. The commonly used global representations are polynomial/spline curve fitting (Dimitrova and Golshani, 1995; Chen and Chang, 2000), discrete Fourier transforms (DFTs) (Faloutsos et al., 1994), discrete wavelet transforms (DWTs) (Sahouria and Zakhor, 1997). The major advantage of transformation-based representations is to reduce high, even various dimensional temporal data to uniform lower dimensional feature space, which significantly improves computational efficiency. However, based on our previous study (Yang, 2006; Yang and Chen, 2006, 2007), we realized that no single representation technique could perfectly represent all the different temporal data set, each of them just capture limited amount of characters obtained from temporal data set.

There follow reviews of the two piecewise and two global representations applied in our proposed algorithm—Weighted clustering ensemble with multiple representations presented in Chapter 7: piecewise local statistics (PLS), piecewise discrete wavelet transform (PDWT), polynomial curve fitting (PCF), and DFTs.

Piecewise Local Statistics

Motivated by the short-term analysis in speech signal processing, discovery of time series motifs (Lin et al., 2002), and a bit-level representation of time series (Bagnall et al., 2006), we adopt a window-based statistical analysis for time series. First of all, we use a window of the fixed size to block time series into a set of segments. For each segment, we estimate the first- and second-order statistics used features of this segment. For segment n, its local statistics, mean μ_n, and standard deviation σ_n of observations are estimated by

$$\mu_n = \frac{1}{|W|} \sum_{t=1+(n-1)|W|}^{n|W|} x(t), \quad \sigma_n = \sqrt{\frac{1}{|W|} \sum_{t=1+(n-1)|W|}^{n|W|} [x(t) - \mu_n]^2} \tag{2.1}$$

where $|W|$ is the size of the window. For time series, a PLS representation of a fixed dimension is formed by the collective notation of local statistic features of all segments though the estimate might be affected at the end of time series where the window is delimited. The PLS representation would be an extension of the representation proposed in Keogh et al. (2001) where the first-order statistics only is used.

Piecewise Discrete Wavelet Transforms

DWT turns out to be an effective multiscale analysis tool. Like the preprocessing in the PLS representation, time series, $\{x(t)\}_{t=1}^{T}$, is blocked into a set of segments with a window of size $|W|$. In the proposed weighted clustering ensemble model presented in Chapter 7, we apply the Daubechies wavelets to each segment for a multiscale analysis in order to capture local details in a more accurate way, for example, abrupt changes, that often fail to be characterized accurately by local statistics in our PLS representation. For the nth segment with a multiscale analysis of J levels, the application of the DWT leads to a piecewise representation with all coefficients collectively:

$$\{x(t)\}_{t=(n-1)|W|}^{n|W|} \Rightarrow \left\{ \Psi_L^J, \left\{ \Psi_H^j \right\}_{j=1}^{J} \right\} \tag{2.2}$$

The DWT decomposes each segment of time series via the successive use of low-pass and high-pass filtering at appropriate levels. At level j, $|W| * 2^{-j}$ coefficients of high-pass filters, Ψ_H^j, encode the detailed information, while those of low-pass filters, Ψ_L^j, characterize coarse information.

Polynomial Curve Fitting

The objective of curve fitting is to find a mathematical equation that describes the signal and that is minimally influenced by the presence of noise. The most common

approach is the least-squares polynomial method, which is capable of finding the coefficients of polynomial equations that are a best fit to the sequential data. In the study by Policker and Geva (2000), time series is modeled by fitting it to a parametric polynomial function

$$x(t) = \alpha_P t^P + \alpha_{P-1} t^{P-1} + \cdots + \alpha_1 t + \alpha_0. \tag{2.3}$$

Here α_p ($p = 0, 1, \ldots, P$) is the polynomial coefficient of the pth order. In general, fourth order of polynomial coefficient gives best performance according the empirical study, higher order does not make significant improvements. The fitting is carried out by minimizing a least-square error function by considering all sequential points of time series and the polynomial model of a given order, with respect to α_p ($p = 0, 1, \ldots, P$). All coefficients obtained via optimization constitute a PCF representation, a sequential point location-dependent global representation of time series. Normally, the complex structure of time series, which has a large amount of abrupt changes along the observed points, would require the higher order polynomial curve for appropriate approximation. Although the PCF is only good at obtaining the global information from time series, the important local information such as the abrupt changes could not be captured by the PCF.

Discrete Fourier Transforms

The previous PCF representation bears the general information of time series trend, which analyze the trajectory in time domain. However, Fourier transform decomposes the time series in frequency domain. The Fourier analysis is among the most widely used tools for transforming data sequences and functions, from the time domain to their representation in the frequency domain. Analysis of sequences in the frequency domain, can uncover important properties which are not readily observable in the time domain. Basically, the Fourier transforms are classified into continues Fourier transform and DFT. The continuous Fourier transform decomposes a continuous waveform into a continuous spectrum of its frequency components, and the inverse transform synthesizes a function from its spectrum of frequency components. In contrast, the DFT is defined for discrete sampled single. For the purpose of temporal data representation, based on discrete sequences of observations from time series, we are particularly interested in the DFT. The DFT maps a discrete sequence in the time domain (observations) to a discrete sequence in the frequency domain (frequency coefficients).

DFTs have been applied to derive a global representation of time series in frequency domain (Faloutsos et al., 1994). The DFT of time series $\{x(t)\}_{t=1}^{T}$ yields a set of Fourier coefficients:

$$a_k = \frac{1}{T} \sum_{t=1}^{T} x(t) \exp\left(\frac{-j2\pi kt}{T}\right), \quad k = 0, 1, \ldots, T-1 \tag{2.4}$$

In order to form a robust representation in the presence of noise, only few top K ($K << T$) coefficients corresponding to low frequencies are collectively used to

form a Fourier descriptor, a sequential point location independent global representation of time series. According to our previous study, top 16 coefficients corresponding to low frequencies generally capture most of feature obtained from frequency component; more DFT coefficients do not make significant improvements.

2.2.3 GENERATIVE MODEL—BASED REPRESENTATIONS

Generative model—based representations treat that the temporal data are obtained from a statistical or deterministic model such as Hidden Markov Model (HMM), mixture of first-order Markov chain (Smyth, 1999), dynamic Bayesian networks (Murphy, 2002), or Autoregressive Moving Average Model—ARMA (Xiong and Yeung, 2002), hence entire temporal data set can be represented by a mixture of these models with appropriate model parameters, which is as,

$$p(x|\theta) = \sum_{k=1}^{K} w_k p(x|\theta_k) \tag{2.5}$$

where $\theta = \{\theta_k\}_{k=1}^{K}$ is the unknown model parameters, w_k is the prior probability (also known as mixing or weighting coefficient), and satisfies the requirements $0 \leq w_k \leq 1$, and $\sum_{k=1}^{K} w_k = 1$. K is the number of component models used for representing the entire data set.

As an important model-based representation approach, HMM has outstanding ability in capturing temporal features whose values change significantly during the observation period, thereby satisfying the Markov property. Essentially, temporal data can be represented by the HMM model, which describes an unobservable stochastic process consisting of a finite number of states, each of which is related to another stochastic process that emits observation. Initially, an observation is emitted with an emission probability b_j at the state j, which is selected according to the initial probability π_j. The next state i is decided by the state transition probability a_{ij} and also generates a symbol based on emission probability b_i. The process repeats until reaching a stop criterion. The entire process produces a sequence of observations instead of states, from where the name "hidden" is drawn. The complete set of model parameters describing HMM are given by the triplet $\lambda = \{\pi, A, B\}$, where$\pi = \{\pi_j\}$, $A = \{a_{ij}\}$, $B = \{b_j\}$. For continuous-valued temporal data such as time series, the emission probability function of each state is defined by multivariate Gaussian distribution. However, for application of HMM involved in our simulations, the emission distribution function of continuous-valued temporal data is normally modeled as a single Gaussian distribution $b_j = \left\{\mu_j, \sigma_j^2\right\}$ in order to reduce the computational cost and prevent the risk of overfitting on the limited available data set.

For temporal data, the entire data set can be represented as a set of K HMM $\{\lambda_1, \lambda_2, \ldots, \lambda_K\}$ with M states based on single Gaussian distributed observations. Each component consists of the following parameters:

- An M-dimensional initial state probability vector π
- An $M \times M$ state transition matrix A
- Mean vector $\{\mu_1, \mu_2, \ldots, \mu_M\}$
- Variance vector $\left\{\sigma_1^2, \sigma_2^2, \ldots, \sigma_M^2\right\}$

For most applications of HMM, we have to solve three major problems:

1. Given the model parameters $\lambda = \{\pi, A, B\}$, compute the probability $p(x|\lambda)$ of a particular sequence of observations $x = \{x(t)\}_{t=1}^{T}$. This problem is solved by the forward and backward algorithms (Baum and Eagon, 1967; Baum and Sell, 1968).
2. Given the model parameters $\lambda = \{\pi, A, B\}$, find the most likely sequence of hidden states which could have generated a given sequence of observations $x = \{x(t)\}_{t=1}^{T}$. Solved by the Viterbi algorithm (Viterbi, 1967; Forney Jr., 1973).
3. Given a sequence of observations $x = \{x(t)\}_{t=1}^{T}$, find the most likely model parameters $\lambda = \{\pi, A, B\}$. Solved by the expectation-maximization algorithm (Dempster et al., 1977).

It is obvious that the HMM-based representation of temporal data could be achieved by solving problem 3.

Although the nature of generative model−based representations would facilitate to identify the data dependency and regularity behind the dynamic behaviors of temporal data, this approach always causes the high-computational cost resulted in mining operations for temporal data with high dimensionality and large volume. Therefore, it may become infeasible in the real-world applications.

2.3 SIMILARITY MEASURES

Once an appropriate representation obtained from temporal data, another interesting problem is to find whether the different temporal data behaves similarly in the representation space, which is referred to similarity measure. Although many similarity measures have been proposed for temporal data mining, there is a strong relation between the representation method and definition of similarity measure, which is mainly governed by the objectives of mining operations. As suggested by Bagnall and Janacek (2005), the similarity measures can be classified into three categories based on different objectives, which are Similarity in time, Similarity in shape and Similarity in change.

2.3.1 SIMILARITY IN TIME

Normally the aim of similarity measures is primarily to measure the similarity between two temporal data related to time and find whether the value of instance varies similarly on each of time intervals. This objective can be achieved by applying a correlation or Euclidean distance metric on the either time domain−based representations or transformation-based representations of temporal data.

Euclidean distance metric has been used for the similarity measure involved in our proposed algorithms that are Iteratively constructed clustering ensemble presented in Chapter 5 and Weighted clustering ensemble with multiple representations presented in Chapter 7. For Iteratively constructed clustering ensemble, we treat the sequence of instance value obtained from original temporal data with identical length as a vector and directly calculate the Euclidean distance between pair of temporal data. For Weighted clustering ensemble with multiple representations, a feature vector is extracted from original temporal data by the corresponding transformation-based representation method; therefore, we simply calculate the Euclidean distance between their feature vector representing the temporal data instead of the original form of temporal data. The Euclidean distance metric is defined as following:

$$D(x, y) = \sqrt{(x, y)\Sigma^{-1}(x - y)^T}, \tag{2.6}$$

where $x = \{x(t)\}_{t=1}^{T}$ and $y = \{y(t)\}_{t=1}^{T}$ represent a pair of temporal data with length of T.

2.3.2 SIMILARITY IN SHAPE

Actually similarity in sharp is a more general case of similarity in time, its objective is more likely to find the similarity between two temporal data varied in time or speed by detecting common trend occurred at different time or similar subpattern in the data, for example, finding the similar motion path from several persons who are walking at different speed or time duration. As well known, Dynamic time warping is always used in order to achieve this objective of similarity measure on the temporal data.

In general, Dynamic time warping aims to align pair of sequences such as time series following a warping path so that predetermined distance metric such as Euclidean distance is minimized. Given two time series $x = \{x(i)\}_{i=1}^{I}$ and $y = \{y(j)\}_{j=1}^{J}$, the warping path, $z = \{z_l\}_{l=1}^{L}$ $(\max(I, J) \leq L \leq I + J - 1)$ can be constructed by satisfy three conditions:

1. Boundary condition: $z_1 = \{x(1), y(1)\}$ and $z_L = \{x(I), y(J)\}$, this requires the warping path to start and finish at beginning and end instances between two sequences.
2. Continuity condition: given $z_l = \{x(i), y(j)\}$ then $z_{l-1} = \{x(i'), y(j')\}$ where $i - i' \leq 1$ and $j - j' \leq 1$. This restricts the allowable steps in the warping path to adjacent manner.
3. Monotonicity condition: given $z_l = \{x(i), y(j)\}$ then $z_{l-1} = \{x(i'), y(j')\}$ where $i - i \geq 0$ and $j - j' \geq 0$. This forces the pair of instances in Z to be monotonically spaced in time.

Then the DTW distance can be defined as following:

$$D^{DWT}(x,y) = \min\left(\frac{1}{L}\sum_{l=1}^{L} D^{Euclidean}(z_l)\right) \tag{2.7}$$

2.3.3 SIMILARITY IN CHANGE

The objective of similar in change is to find the similar dynamic behavior between temporal data whose values change significantly during the observation period. As mentioned in Section 2.2.3, some of generative model−based representations such as HMM and ARMA have outstanding ability in capturing such character. Based on the log-likelihood of each sequence given the model generated for the other sequence, a symmetric distance between two sequences $x = \{x(t)\}_{t=1}^{T}$ and $y = \{y(t)\}_{t=1}^{T}$ has been proposed by Juang and Rabiner (1985), its equation is shown as following:

$$D^{sym}(x,y) = \frac{1}{2}[LL_{xy} + LL_{yx}] \tag{2.8}$$

Alternatively, Panuccio et al. (2009) introduced a similar log-likelihood-based distance metric named BP, which is formulated by:

$$D^{BP}(x,y) = \frac{1}{2}\left[\frac{LL_{xy} - LL_{xx}}{LL_{xx}} - \frac{LL_{xy} + LL_{yy}}{LL_{yy}}\right] \tag{2.9}$$

where $LL_{xy} = \log(p(X|\theta_y))$, given the parameters θ_y of model generating sequence $y = \{y(t)\}_{t=1}^{T}$.

Based on the log-likelihood of each sequence given the corresponding generator model and another model, the Kullback-Leibler (KL) distance (Juang and Rabiner, 1985; Sinkkonen and Kaski, 2002) between both of component models can be computed by using the single linkage, complete linkage, and average linkage methods (Zhong and Ghosh, 2003), which are defined as

$$D^{MinKL}(\lambda_i,\lambda_j) = \min_{x\in C_i}(\log p(x|\lambda_i) - \log p(x|\lambda_j)) \tag{2.10}$$

$$D^{MaxKL}(\lambda_i,\lambda_j) = \max_{x\in C_i}(\log p(x|\lambda_i) - \log p(x|\lambda_j)) \tag{2.11}$$

$$D^{BoundaryKL}(\lambda_i,\lambda_j) = \frac{1}{|B_x|}\sum_{x\in C_i}(\log p(x|\lambda_i) - \log p(x|\lambda_j)) \tag{2.12}$$

where x is the item grouped into the cluster C_i. B_x is the fraction of items x grouped into the cluster C_i that has smallest value $\log p(x|\lambda_x) - \log p(x|\lambda_y)$ and the value of 0 for $\log p(x|\lambda_i) - \log p(x|\lambda_j)$ determines the boundary between cluster i and j.

2.4 MINING TASKS

After representing the temporal data in a suitable form and defining the appropriate similarity measure, An algorithm would be used for a particular temporal data task, which is also called mining operation. According to various objectives of temporal data tasks based on a wide range of applications, Lin et al. (2002) suggest that these tasks can be classified into five categories: prediction, classification, clustering, search & retrieval, and pattern discovery. However, the work presented in this book mainly focuses on clustering task.

Prediction: It is one of the most important problems in mining operations, which has to do with forecasting the evolution of temporal data such as time series based on its past samples. In order to do this, generative models are normally built for representing the predictive temporal data. However, in many cases, prediction problems may be formulated as classification, association rule finding, or clustering problems.

Classification: It is one of the most typical operations in supervised learning, but most of classification algorithms have been adapted with special treatment due to the nature of temporal data. In temporal data classification, each data presented to the system are assumed to belong to one of finitely many classes or categories, which are predefined or trained on the given training sets, and these tasks aim to automatically determine the corresponding class or category for the given input data.

Clustering: It is to partition a collection of time series into several groups called clusters such that items with similar characteristics are grouped together. Because clustering provides an unsupervized leaning approach to automatically determine the underlying structure in temporal data that would be always difficult to summarize or visualize due to the nature of temporal data such as large volume, high dimensionality, it has been specially paid attention by many researchers and applied for a wide range of applications. There are two major problems in clustering: finding the intrinsic clusters numbers and properly grouping the temporal data based on a meaningful similarity measure.

Search and retrieval: It can be simply defined as detecting an objective activity from large archives in a general term, but searching for temporal data in large databases is becoming an important task in temporal data mining due to the dramatic growth of temporal data in our daily life. Temporal data search and retrieval techniques play an important role in interactive explorations of large sequential databases such as online media library. The problem is concerned with efficiently locating subsequences referred to as queries in large archives of sequences or a single long sequence.

Pattern discovery: It aims to discover interesting patterns, which include periodic and abnormal patterns, from temporal data. The discovery of interesting patterns has become one of the most important data-mining tasks, and it can be applied to many domains. Particularly when the domain expert-derived patterns do not exist or are not complete, an algorithm to discover specific patterns or shapes automatically

from the temporal data is necessary. Such an algorithm is noteworthy in that it does not assume prior knowledge of the number of interesting structures nor does it require an exhaustive explanation of the patterns being described.

2.5 SUMMARY

For temporal data mining, we presented three major phases of temporal data mining: representation of temporal data, definition of similarity measures, and mining tasks. Representation of temporal data refers to how to represent the temporal data in an efficient way before actually mining operations takes place. Three major methods including time domain−based representations, transformation-based representations, and generative model−based representations have been described and discussed in their advantages and disadvantages. In the transformation-based representations, four representations including PLS, PDWT, PCF, and DFT, which applied in our proposed algorithm—Weighted clustering ensemble with multiple representations presented in Chapter 7, have been described in detail. In generative model−based representations, HMM as an important model-based representation approach has been described with answering three questions related to its application of temporal data. After that, we described the similarity measures of temporal data mining based on different objectives: Similarity in time, similarity in shape, and similarity in change. For similarity in time, we described the Euclidean distance, which has been used for the similarity measure involved in our proposed algorithms that are Iteratively constructed clustering ensemble presented in Chapter 4 and Weighted clustering ensemble with multiple representations presented in Chapter 7, and explained how to apply such similarity measure on the different representations. For similarity in shapes, we described the DTW distance in detail. For similarity in change, a symmetric model−based distance measure and its variant named BP has been presented, and the KL distance which was used for our proposed *HMM-based meta clustering ensemble* presented in Chapter 5, has been further described here. After representing the temporal data in a suitable form, and defining the appropriate similarity measure, an algorithm would be used for a particular temporal data mining tasks, which is also called mining operation. Therefore, five mining tasks including prediction, classification, clustering, search & retrieval, and pattern discovery has been briefly described.