计算机研究与发展
Journal of Computer Research and Development
ISSN 1000-1239/CN 11-1777/TP
42(1):72~78, 2005

基于分段时间弯曲距离的时间序列挖掘

肖 辉 胡运发
(复旦大学计算机与信息技术系 上海 200433)
(miningant@hotmail.com)

Data Mining Based on Segmented Time Warping Distance in Time Series Database

Xiao Hui and Hi Yunfa
(*Department of Computer and Information Technology, Fudan University, Shanghai* 200433)

Abstract Data mining in time series database is an important task, most research work are based comparing time series with Euclidean distance measure or its transformations. However Euclidean distance measure will change greatly when the compared time series move slightly along the time-axis. It 's impossible to get satisfactory result when using Euclidean distance in many cases. Dynamic time warping distance is a good way to deal with these cases, but it 's very difficult to compute which limits its application. In this paper, a novel method is proposed to avoid the drawback of Euclidean distance measure. It first divides time series into several line segments based on some feature points which are Chosen by some heuristic method. Each time series is converted into a segmented sequence, and then a new distance measure called feature points segmented time warping distance is defined based this segmentation. Compared with the classical dynamic time warping distance, this new method is much more fast in speed and almost no degrade in accuracy. Finally, implements two completed and detailed experiments to prove its superiority.

Key words time series; dynamic time warping distance; segmented time warping distance

摘 要 在时间序列库中的数据挖掘是个重要的课题,为了在挖掘的过程中比较序列的相似性,大量的研究都采用了欧氏距离度量或者其变形,但是欧氏距离及其变形对序列在时间轴上的偏移非常敏感.因此,采用了更鲁棒的动态时间弯曲距离,允许序列在时间轴上的弯曲,并且提出了一种新的序列分段方法,在此基础上定义了特征点分段时间弯曲距离.与经典时间弯曲距离相比,大大提高了效率,而且保证了近似的准确性.

关键词 时间序列;动态时间弯曲距离;分段时间弯曲距离

中图法分类号 TP391.41

1 引 言

时间序列是一类重要的数据对象,在经济、气象等许多领域都大量存在.对这些数据进行分析,可揭示事物变化和发展的规律,为科学决策提供依据.近来在时间序列数据的挖掘方面做了很多工作,Das等人介绍了如何从时间序列中发现关联规则[1];Debregeas和Hebrail提出了针对大数据集的聚类算法[2];Keogh和Pazzani提出了一种可伸缩的时间序列分类算法[3].所有这些工作的前提都是要比较时间序列之间的相似性,一般采用的都是欧氏距离及

收稿日期:2003-09-08;修回日期:2004-02-27
基金项目:国家自然科学基金项目(60173027)

其变形.然而欧氏距离对序列在时间轴上的轻微变化非常敏感,一些轻微的变化可能会使得序列之间的欧氏距离变化很大.如图1所示,采用欧氏距离在对序列1～4聚类时,序列1和2聚为一类,序列3和4聚为一类,发生了比较大的偏差.这是因为计算欧氏距离时要求序列各点一一对应,当序列在时间轴上发生轻微偏移时欧氏距离变得很大,使得序列1,2和3没有聚在一起.动态时间弯曲距离(dynamic time warping distance)可以有效地消除欧氏距离的这个缺陷,它允许序列在时间轴上的偏移,序列各点不要求一一对应,并且能够计算不同长度序列之间的距离.图2中(a)和(b)分别显示了欧氏距离和动态时间弯曲距离计算时序列各点之间的对应关系.但是动态时间弯曲距离的计算量很大,不适合直接用于时间序列的挖掘中.为此,研究者提出了各种方法来近似计算动态时间弯曲距离[4～7].

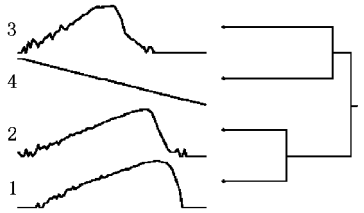


Fig. 1　Cluster using Euclidean distance measure.

图1　采用欧式距离聚类结果

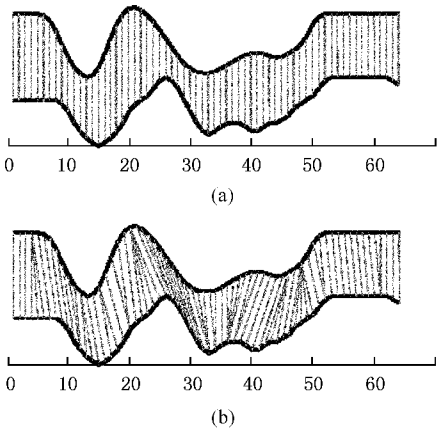


Fig. 2　Points alignment with Euclidean distance and DTW distance. (a) Euclidean distance, which assumes one-to-one correspondence and (b) DTW distance, which assumes nonlinear alignments

图2　欧式距离和动态时间弯曲距离的点对齐关系.(a)欧式距离,两序列间各点一一对应;(b)动态时间弯曲距离,两序列间各点非线性对齐

本文从序列观察的角度出发,发现序列的特征点,提出了一种新的序列分段方法(feature points segmentation),在此基础上定义了序列之间的特征点分段时间弯曲距离(feature points segmented time warping distance).采用特征点分段时间弯曲距离进行时间序列的聚类和查询,我们比较了分别采用特征点分段时间弯曲距离、欧氏距离和经典时间弯曲距离实验时的效率和准确性.结果表明,本文提出的新方法大大提高了时间序列聚类和查询的效率,而且几乎没有准确性的下降.

本文组织如下:第2节对经典的时间弯曲距离做一简要介绍;第3节提出一种新的序列分段方法FPS,以及序列之间的特征点分段时间弯曲距离FPSTWD;第4节采用几个实际数据集比较了分段时间弯曲距离FPSTWD、经典时间弯曲距离TWD和欧氏距离在序列聚类和查询时的效率和准确性;第5节是对全文的一个简单总结.

首先我们先定义一些本文使用的记号和术语如表1所示:

Table 1　List of Symbols

表1　本文定义的一些记号和术语

Symbol	Definition	Symbol	Definition
D_p	L_p distance $p=1,2,\cdots,\infty$	$\langle\rangle$	Null time series
D_{base}	Base distance, e.g., D_1, D_2	x_i	The i-th element of x
D_{warp}	Time warping distance	$\lvert x\rvert$	Length of x
D_{lb}	Lower-bound distance to D_{warp}	$head(x)$	First element of x
x	Time series	$rest(x)$	The rest elements of x but first
$x[i:j]$	Elements from the i-th to j-th of x	$x[i:-]$	Elements from the i-th to last of x
x^S	Segmentation of x	x_i^S	The i-th element of x^S

2　动态时间弯曲距离

动态时间弯曲距离在语音处理领域得到广泛的研究[8,9],并且由Berndt和Clifford首次引入到数据挖掘领域[10].到现在,动态时间弯曲距离已经在医疗信号、生物学数据以及指纹识别等领域得到快速的发展[4,11～13].下面简要介绍动态时间弯曲距离的基本定义和常用的计算方法.

定义1. 时间序列x和y之间的动态时间弯曲距离定义为

$$D_{tw}(\langle\rangle,\langle\rangle)=0,$$

$$D_{tw}(x,\langle\rangle)=D_{tw}(\langle\rangle,y)=\infty,$$

$$D_{tw}(x,y)=D_{\text{base}}(head(x),head(y))+$$

$$\min\begin{Bmatrix} D_{tw}(x,rest(y)),D_{tw}(rest(x),y), \\ D_{tw}(rest(x),rest(y)). \end{Bmatrix}$$

动态时间弯曲距离可以用动态规划的方法计算[14],时间复杂度为 $O(|x|\cdot|y|)$. 图 3 显示了计算累积距离表的过程,通过计算这个累积距离表,最后得到两个序列的动态时间弯曲距离. 在这个计算过程中,我们同时可以得到时间序列 x 和 y 的任何一个前缀 $y[1:j]$ 的时间弯曲距离,它存储在第 j 行的最后一个表格单元;同样可以得到的任何一个前缀 $x[1:j]$ 和 y 之间的时间弯曲距离,它存储在第 i 列最上面的一个表格单元.

row 6	6	16	11	12
row 5	6	13	9	10
row 4	7	10	7	8
row 3	6	6	4	5
row 2	5	3	2	3
row 1	4	1	1	2
	Y \ X	3	4	3
		col 1	col 2	col 3

Fig. 3 Cumulative distance table for computing DTW distance between $x=\langle 3,4,3\rangle$ and $y=\langle 4,5,6,7,6,6\rangle$.

图 3 时间序列 $x\langle 3,4,3\rangle$和 $y\langle 4,5,6,7,6,6\rangle$之间的累积距离表的计算过程

定理 1. 如果时间序列 x 和 y 的任何一个前缀 $y[1:j]$ 的时间弯曲距离大于 ε,那么 x 和 y 之间的时间弯曲距离一定也大于 ε.

证明. 定理的正确性是显然的,从图 3 容易看出,$D_{tw}(x,y)\geqslant D_{tw}(x,y[1:j])$.

3 序列分段及分段时间弯曲距离

由于动态时间弯曲距离计算的复杂度太大,不适合大数据集的挖掘. 本节我们通过序列分段线性化,可以有效地降低计算复杂度,并且保证了近似的准确性. 根据生理实验表明,人类的视觉系统将平滑的曲线分为多个直线段处理,对序列第 1 眼印象最深的点就是序列的极值点[15]. 从这点出发,我们选择序列中那些对序列形状影响最大的点称为特征点(feature points). 通过连接这些特征点将序列线段化,在此基础上定义了新的特征点分段时间弯曲距离(feature points segmented time warping distance).

定义 2. 时间序列 x 的特征点是指满足以下两个条件的点:①该点必须是序列的极值点;②该极值点保持极值的时间段(即该点与前极值点及后极值点的时间段)与该序列长度的比值必须大于某个阈值 C.

注 1. 序列的起始点和结束点自动成为特征点;

注 2. 参数 C 的解释:该极值点的影响因子. 取值和领域知识、序列长度以及用户关注角度有关,一般取 0.01~0.1 之间.

注 3. 参数 C 的意义:在股票的技术分析方法中,威廉指标 $W\%R$ 和随机指标 KDJ 分别采用 12 天和 9 天中的极大值和极小值进行技术分析. 定义 2 中的 C 也可以改为该极值点保持极值的时间段(比如取 12 或 9).

特征点分段(FPSegmentation)如图 4 所示.

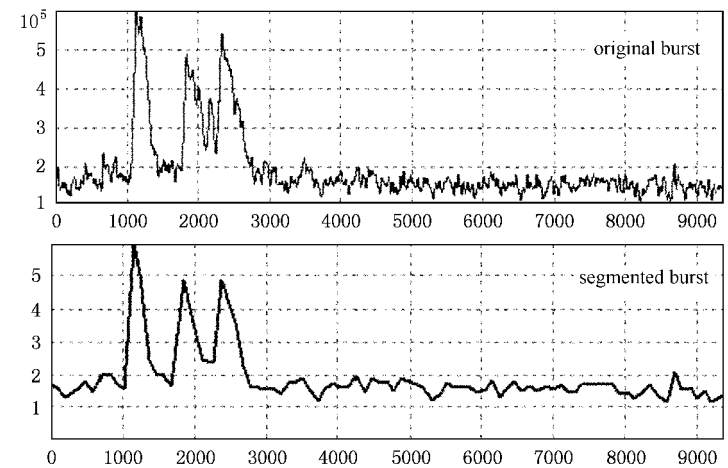
```
FPSegmentation(x,C)
begin
  Init_MM = FindMaxMinPoint(x);
  Sel_MM = SelectFeaturePoint(Init_MM,C);
  TSSegmentation(x,Sel_MM,x^S);
  return x^S;
end;
FindMaxMinPoint(x);
begin
  k = 1;
  Init_MM[k] = 1;
  for i = 2 to Length(x) - 1 do
  if ((x[i] > x[i - 1]) and (x[i] > x[i + 1])) or ((x[i] < x
    [i - 1]) and(x[i] < x[i + 1]) then
  begin
    Inc(k); Init_MM[k] = i;
  end;
  Init_MM[k + 1] = Length(x);
end;
SelectFeaturePoint(Init_MM,C);
begin
  k = 1;
  Sel_MM[k] = 1;
  for i = 2 to length(Arr_MM) - 1 do
  if (Arr_MM[i + 1] - Arr_MM[i - 1]) ≥ Length(x) * C then
  begin
    Inc(k); Sel_MM[k] = i;
  end;
  Sel_MM[k] = Length(Arr_MM);
end;
TSSegmentation(x,Sel_MM,x^S);
begin
  for i = 1 to length(Sel_MM) do
    x_i^S = x_Sel_MM[i];
end;
```

Fig. 4 FPSegmentation algorithm.

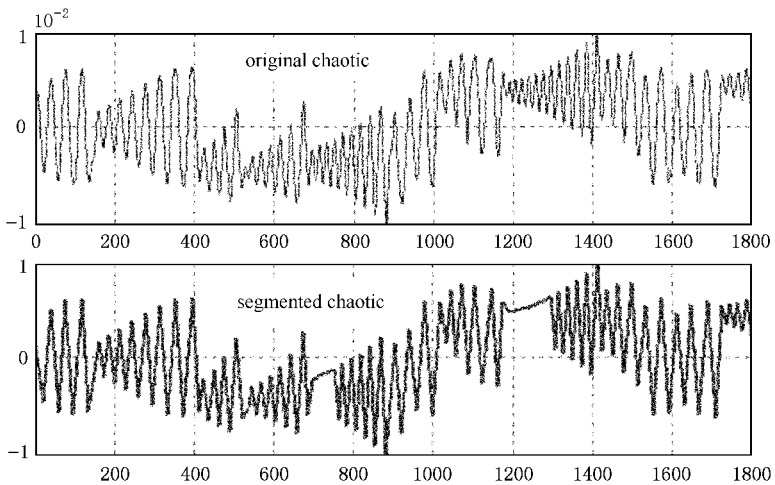
图 4 特征点分段算法

算法时间复杂度为 $o(|x|)$. 经过优化,只需扫描序列一遍就可以得到分段结果.

Fink 和 Pratt 提出了基于重要点的序列分段方法(为了叙述上的简单起见,本文把他们的方法称为 IPSegmentation)[5],和他们的方法相比,我们提出的 FPSegmentation 在分段速度上及近似质量都和 IPSegmentation 相当. 但是我们的方法适用面更广,对噪声的处理较好. 在我们的实验中,采用了以下网址(http://www.cs.ucr.edu/~eamonn/TSDMA/datasets.html)的 10 个数据集,即 burst, burstin, chaotic, darwin, earthquake, leleccm, ocean, powerplant, speech, tide 对 IPSegmentation 和 FPSegmentationli 两种方法进行了比较(序列长度从 1020~50000 不等). 结果发现, IMSegmentation 方法对数据集 chaotic, earchquake, ocean, powerplant, tide 无法有效地分段,无论如何调整它的参数,只能得到极少的几条线段来近似原序列,显然这样的近似结果是无法接受的; FPSegmentation 方法在所有的数据集上都有良好的表现,在压缩率 95% 以上仍然可以很好地表现原序列的形态. 图 5 是 FPSegmentation 在数据集 burst 和 chaotic 上的分段效果.



(a)



(b)

Fig. 5　Segmentation of time series using FPSegmentation method. (a) segmentation of burst time series using FPSegmentation method with parameter $C=0.01$, compression ratio equals to 99.1% and (b) Segmentation of chaotic time series using FPSegmentation method with parameter $C=0.01$, compression ratio equals to 95.4%.

图 5　FPSegmentation 方法用于时间序列的分段. (a) 时间序列 burst 用 FPSementation 分段前后图示,分段参数 $C=0.01$,压缩率为 99.1%;(b) 时间序列 chaotic 用 FPSementation 分段前后图示,分段参数 $C=0.005$,压缩率为 95.4%

在上面基于特征点的序列分段的基础上,我们定义了序列之间的特征点分段时间弯曲距离 FPSTWD.

定义 3. 假设时间序列 x 和 y 线段化后分别是

x^S 和 y^S,它们之间的特征点分段时间弯曲距离(FPSTWD)定义如下:

$$D_{tw}(\langle\rangle,\langle\rangle)=0,$$

$$D_{tw}(x^S,\langle\rangle)=D_{tw}(\langle\rangle,y^S)=\infty,$$

$$D_{tw}(x^S,y^S)=D_{\text{base}}(x_1^S,y_1^S)+\min\begin{Bmatrix}D_{tw}(x^S,rest(y^S)),D_{tw}(rest(x^S),y^S),\\ D_{tw}(rest(x^S),rest(y^S)),\end{Bmatrix}$$

其中,

$$D_{\text{base}}(x_1^S,y_1^S)=(x_1^S[1]-y_1^S[1])^2+(x_1^S[|x_1^S|]-y_1^S[|y_1^S|])^2.$$

Park 等人也定义了分段时间弯曲距离[5],但是却有一个严格的条件限制:序列线段化后的分段数目必须相同. 我们的特征点分段时间弯曲距离 FPSTWD 可以针对分段数目不相同的序列进行计算,这保证了序列分段的最佳效果,而不用因为追求相同的分段数目牺牲分段质量. FPSTWD 的时间复杂度是 $O(|x^S|\,|y^S|)$,这比经典动态时间弯曲距离的 $O(|x|\,|y|)$ 减少了一个很大的常数倍,这个常数主要依赖于分段的压缩率. 当压缩率达到 80% 时,时间复杂度减少了 25 倍;当压缩率达到 90% 时,时间复杂度减少了 100 倍.

4 实验结果

在时间序列挖掘中,序列匹配是个基本而且重要的问题. 它通常包括 2 个方面:

(1) 全序列匹配. 给定查询序列 x,需要在相同长度的序列库中找到和 x 最相似的序列或者与 x 距离小于某个阈值 ε 的所有序列.

(2) 子序列匹配. 给定查询序列 x,y 是比 x 长得多的序列,需要在 y 上找到子序列和 x 最相似或者与 x 距离小于某个阈值 ε.

针对这两个问题,本节设计了两个实验,分别对欧氏距离、经典 TWD 距离和我们提出的 FPSTWD 距离进行了时间和准确性的比较.

4.1 聚 类

聚类实验可以比较 3 种距离在全序列匹配问题上的性能. 本节我们分别采用欧氏距离、经典 TWD 以及 FPSTWD 进行聚类实验比较聚类的时间和质量. 聚类方法采用层次聚类法,数据集采用 system control chart (http://www.cs.ucr.edu/~eamonn/TSDMA/cluster.html). 该数据集包括 600 个样本,每个样本 60 个点,共 6 类,每个类都是 100 个样本.

表 2 比较了在 3 种距离下的聚类时间及正确聚类数.

Table 2 Comparison of Cluster Experiment Using Euclidean Distance, Classical TWD and FPSTWD

表 2 欧氏距离、经典 TWD、FPSTWD 在聚类实验中的性能比较

Distance Measure	Mean Time (second)	Correct Clustering
Euclidean	10.2	1
Classical TWD	328.7	4
FPSTWD	15.3	4

图 6 是对第 1 类样本 1~5 和第 2 类样本 6~10 做的一个层次聚类表示,结果显示,FPSTWD 的效果明显好于基于欧氏距离的聚类.

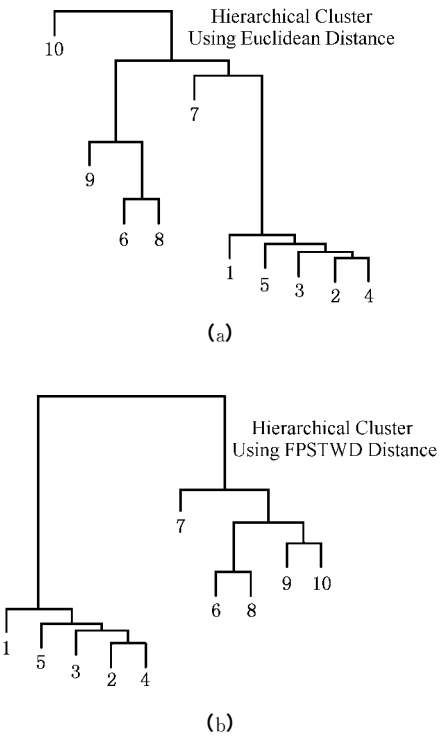


Fig. 6 Hierarchical cluster results using euclidean distance and FPSTWD distance. (a) Hierarchical cluster results using euclidean distance and (b) Hierarchical cluster results using FPSTWD distance

图 6 采用欧式距离和 FPSTWD 距离的层次聚类结果. (a) 采用欧式距离的层次聚类结果;(b) 采用 FPSTWD 距离的层次聚类结果

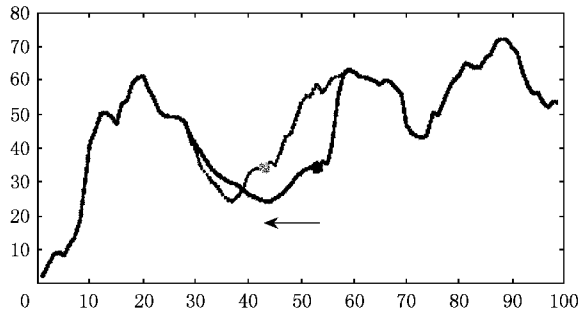
尽管采用欧氏距离聚类速度最快,但是效率最低,几乎无法聚类. 采用 TWD 和 FPSTWD 聚类质量几乎一样,但是 FPSTWD 比 TWD 速度提高了近 21 倍.

4.2 子序列查询

从前面对聚类实验的结果分析得知,FPSTWD

在全序列匹配上的应用是成功的,性能比欧氏距离及 TWD 都要好. 这里我们针对子序列匹配,进一步比较这 3 种距离度量的性能.

实验任务描述如下:给定查询序列 x,y 是比 x 长得多的序列,需要在 y 上找到子序列和的距离最近并且返回该子序列的位置. 如果采用欧氏距离,我们可以利用多维空间索引来加速查询速度[16,17]. 由于动态时间弯曲距离不符合三角不等式,所以无法使用类似的索引技术,这里我们采用顺序扫描和滑动窗口技术进行子序列匹配,但是不像通常那样每次窗口只滑动一个点,由于我们从序列中提取了那些对序列形状影响最大的特征点,可以认为窗口只有在经过一个特征点时,匹配的子序列才会发生明显的变化,所以每次我们让窗口滑动到下一个特征点,以加快顺序扫描的速度. 实验数据使用 eamonn 提供的 Earthquake 数据集(http://lib.stat.cmu.edu/general/tsa/tsa.html),该数据集包含了一条 4096 个点的序列. 查询序列是从该序列中随机抽取一段长度为 100 的子序列,并且在该子序列的某处加以时间轴上的弯曲,如图 7 所示(将该点向左或右平移 w 个时间单位,该点前后极值点之间的所有点也通过插值的方法随着移动,实验中我们设置 w 分别为 5,10,15 进行了系列实验).



Solid line denotes original time series and
dashed line denotes time warped time series.

Fig. 7　Query sequence produced by moving and interpolating.

图 7　通过平移和插值得到查询序列

表 3 是欧氏距离,TWD,FPSTWD 3 种距离度量在实验中查询准确率和查询时间的比较结果:

Table 3　Comparison of Query Experiment Using Euclidean Distance, Classical TWD and FPSTWD

表 3　欧式距离、经典 TWD、FPSTWD 在序列查询实验中的性能比较

Distance Measure	Mean Accuracy(%)			Mean Time
	$w=5$	$w=10$	$w=15$	(s)
Euclidean	15	10	0	0.5
Classical TWD	100	80	50	12.5
FPSTWD	100	75	50	0.6

虽然欧氏距离在查询时间上最快,但是查询准确率却最差. FPSTWD 和 TWD 相比,查询准确率相当,但是查询速度加快了近 20 倍.

5　结论和进一步研究

时间序列之间的距离度量是时间序列挖掘任务的研究基础. 寻求一种好的距离度量对于提高这些挖掘任务的效率和准确性有着至关重要的意义. 我们提出了一种新的序列分段的方法 FPS,并且定义了基于特征点的分段时间弯曲距离 FPSTWD. 与以往的分段方法 IPS 相比,FPS 在分段速度及近似质量上相当,但是适用面更广,对噪声的处理更好;我们提出的新的距离度量 FPSTWD 不要求序列的分段数目一致,从而可以在分段时得到最好的近似质量.

实验证明,和动态时间弯曲距离相比,我们的方法在不降低性能的同时,在聚类和相似性查询方面速度有很大的提高;和欧氏距离相比,我们的方法大大提高了聚类和相似性查询的性能,而且在速度上只是稍微慢一点. 以后的工作我们将试图解释时间序列的特征点的物理意义,进一步研究在噪声环境下时间序列的分段算法,同其他的分段方法进行全面的比较,并且推广到多变量的时间序列和时间序列流.

参　考　文　献

1　G. Das, K. Lin, H. Mannila, *et al*.. Rule discovery from time series. In: Proc. of the 4th Int'l Conf. of Knowledge Discovery and Data Mining. Menlo Park, CA: AAAI Press, 1998. 16～22

2　A. Debregeas, G. Hebrail. Interactive interpretation of Kohonen maps applied to curves. In: Proc. of the 4th Int'l Conf. of Knowledge Discovery and Data Mining. Menlo Park, CA: AAAI Press, 1998. 179～183

3　E. Keogh, M. Pazzani. An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback. In: Proc. of the 4th Int'l Conf. of Knowledge Discovery and Data Mining. Menlo Park, CA: AAAI Press, 1998. 239～241

4　Z. M. Kovacs-Vajna. A fingerprint verification system based on triangular matching and dynamic time warping. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2000, 22(11): 1266～1276

5　S. Park, S. Kim, W. Chu. Segment-based approach for subsequence searches in sequence databases. The 16th ACM Symp on Applied Computing, Las Vegas, NV, 2001

6 S. Kim, S. Park, W. Chu. An index-based approach for similarity search supporting time warping in large sequence databases. The 17th Int'l Conf. on Data Engineering, Heidelberg, Germany, 2001

7 Zeng Haiquan. Research on mining and similarity searching in time series database: [Ph. D. dissertation]. Shanghai: Fudan University, 2003 (in Chinese)
(曾海泉. 时间序列挖掘与相似性查找技术研究:[博士论文]. 上海:复旦大学, 2003)

8 L. Rabiner, B. H. Juang. Fundamentals of Speech Recognition. Englewood Cliffs, NJ: Prentice-Hall, 1993

9 H. J. L. M. Vullings, M. H. G. Verhaegen, H. B. Verbruggen. ECG segmentation using time warping. In: Proc. of 2nd Int'l Symposium on Advances in Intelligent Data Analysis, 1997. 275~285

10 D. J. Berndt, J. Clifford. Using dynamic time warping to find patterns in time series. Working Notes of the Knowledge Discovery in Databases Workshop, Seatle, WA, 1994

11 D. M. Gavrila, L. Davis. Towards 3-d model-based tracking and recognition of human movement: A multi-view approach. IEEE Int'l Conf. on Automatic Face and Gesture Recognition, Zurich, Switzerland, 1995

12 A. Kassidas, J. F. MacGregor, P. A. Taylor. Synchronization of batch trajectories using dynamic time warping. American Institute of Chemical Engineers. 1998, 44: 864~874

13 M. E. Munich, P. Perona. Continuous dynamic time warping for translation-invariant curve alignment with applications to signature verification. The 8th IEEE Int'l Conf. on Computer Vision, Corfu, Greece, 1999

14 D. J. Berndt, J. Cliord. Finding patterns in time series: A dynamic programming approach. Advances in Knowledge Discovery and Data Mining. Menlo Park, CA: AAAI Press, 1996. 229~248

15 F. Attneave. Some information aspects of visual perception. Psychology Review, 1954, 61(3): 183~193

16 C. Faloutsos, M. Ranganathan, Y. Manolopoulos. Fast subsequence matching in time series databases. In: Proc. of ACM SIGMOD Conf. on Management of Data, 1994. 419~429

17 E. Keogh, M. Pazzani. An indexing scheme for fast similarity search in large time series databases. In: Proc. of the 11th Int'l Conf. on Scientific and Statistical Database Management. Los Alamitos, CA: IEEE Computer Society Press, 1999. 56~67

Xiao Hui, born in 1978. Received the B.A's and M. A's. degrees in mathematics from the University of Nanchang, Jiangxi, China, in 1999 and 2002 respectively. Since 2002, he has been a Ph.D. degree candidate in computing science from the University of Fudan, shanghai, China. His current research interests include data mining, time series analysis and outlier detection.

肖 辉,1978 年生,博士研究生,主要研究方向为数据挖掘、空间索引模型、时间序列.

Hu Yunfa, born in 1940. He has been professor of Fudan University since 1993. His main research interests are data engineering and knowledge engineering.

胡运发,1940 年生,教授,博士生导师,主要研究方向为数据与知识工程、自然语言处理、数据挖掘.

Research Background

Time series data mining (TSDM), including classification, cluster, detection of patterns in large database, has found application in a huge range of problem domains, such as Web mining, meteorology, medical analysis (ECG/EEG/MRI), gene expression analysis, and economics. In this paper, concentrating on how the DM algorithms employed are driven by the need to measure similarity between time series and to represent data, we introduced a new segmentation method for time series and similarity measure for TSDM, called STWD (segmented time warping distance), which is very effective and efficient when used in TSDM algorithm. Its effectiveness is driven by the nonlinear alignments betweens time series with STWD distance measure and its efficient is driven by the segmentation method for time series. Our work is supported by the National Science Foundation of China (60173027).