文章编号: 1001-0920(2018)07-1277-05 DOI: 10.13195/j.kzyjc.2017.0194

DTW距离的过滤搜索方法

李正欣[1,2†], 郭建胜[1], 王 瑛[1], 田 舢[1], 张晓丰[1], 李 超[1]

(1. 空军工程大学 装备管理与无人机工程学院，西安 710051;
2. 西北工业大学 光学影像分析与学习中心，西安 710072)

摘 要: 动态时间弯曲(DTW)距离支持时间序列的多种形变,具有较高的匹配精度,是一种重要的相似性度量方法. 然而,该方法计算复杂度较高,制约了其在相似性搜索中的应用. 为了平衡匹配精度与计算效率之间的矛盾,提出一种过滤搜索方法. 首先,构造一种计算代价较低的DTW下界距离,用其进行粗略过滤,得到候选集;然后,利用提前终止策略,优化计算候选集中序列的DTW距离,得到搜索结果;最后,对所提出方法进行实验验证,结果表明,该方法能够提高DTW距离的相似性搜索效率,且具有非漏报性.

关键词: 时间序列; 相似性搜索; 动态时间弯曲; 提前终止; 过滤搜索

中图分类号: TP311 **文献标志码:** A

Filtering search method for DTW distance

LI Zheng-xin[1,2†], *GUO Jian-sheng*[1], *WANG Ying*[1], *TIAN Shan*[1], *ZHANG Xiao-feng*[1], *LI Chao*[1]

(1. College of Equipment Management and UAV Engineering, Air Force Engineering University, Xi'an 710051, China;
2. Center for OPTical IMagery Analysis and Learning(OPTIMAL), Northwestern Polytechnical University, Xi'an 710072, China)

Abstract: Dynamic time warping(DTW) is an important similarity measure method, which supports a variety of deformation of time series and has high matching precision. However, the method has high computational complexity, which restricts its application in similarity search. In order to balance the contradiction between the matching precision and the computational efficiency, a filtering search method is proposed. Firstly, a lower-bounding distance for DTW is constructed, and it is used in the filtering search to obtain the candidate set. Then, the early abandon strategy is used in the candidate set to achieve the search results. Finally, the proposed method is verified by experiments. The results show that it can improve the similarity search efficiency under DTW and guarantee no false dismissal.

Keywords: time series; similarity search; dynamic time warping; early abandon; filtering search

0 引 言

近年来,随着传感器、无线通信、存储技术的发展,时间序列数据在金融、气象、天文、航空、医疗、互联网监控等领域普遍存在,且规模呈爆炸式增长[1]. 时间序列数据挖掘在图像识别、语音处理、声纳技术、遥感技术、机械工程等工程技术领域以及金融分析、人口统计等社会经济领域都具有广阔的应用前景[2]. 动态时间弯曲(Dynamic time warping, DTW)最早使用在语音识别领域,它较好地解决了语音匹配中的时间对准难题. 之后, Berndt等[3]将其引入时间序列的相似性研究中.

DTW距离定义了时间序列的最佳对齐匹配关系,支持时间轴的伸缩和弯曲,具有较高的匹配精度,已成为时间序列数据挖掘中一种重要的相似性度量方法. 然而,DTW距离的计算复杂度较高,在大规模时间序列数据集中执行相似性搜索的效率较低[4].

DTW距离的过滤搜索是一种提高相似性搜索效率的方法,可实现匹配精度与搜索效率的平衡,在时间序列数据挖掘领域具有重要的应用价值.

1 相关研究

支持DTW距离的过滤搜索主要有提前终止方法和下界距离方法.

1.1 提前终止方法

提前终止方法(Early abandon, EA)[5]在计算DTW距离的过程中,不断与给定的差异阈值进行比

收稿日期: 2017-02-27; **修回日期:** 2017-05-19.
基金项目: 国家自然科学基金项目(61502521, 71601183).
责任编委: 阳春华.
作者简介: 李正欣(1982–), 男, 讲师, 博士, 从事信息系统工程与智能决策、数据挖掘、机器学习的研究; 郭建胜(1965–), 男, 教授, 博士生导师, 从事信息系统工程与智能决策等研究.
†通讯作者. E-mail: lizhengxin_2005@163.com

较,一旦发现累积距离超过差异阈值,则断定最终的距离必将超过差异阈值,此时停止计算. 提前终止能够及时停止在不符合条件序列上的相似性计算,从而节省计算资源,提高搜索效率.

1.2 下界距离方法

下界距离[6]是一种计算代价较低的DTW距离估算方法. 通常,先用下界距离在数据集中执行粗略搜索,过滤掉不满足相似性要求的序列,得到候选集;然后,在候选集中采用DTW距离进行筛选,从而获得搜索结果.

DTW下界距离应满足以下3个条件[7]:正确性:经下界距离过滤得到的候选集应包括所有满足相似性条件的序列;有效性:计算复杂度应小于DTW距离;紧致性:度量结果尽量逼近DTW距离. Yi等[8]、Kim等[9]、Keogh等[10]分别提出了各自的DTW下界距离,并且证明了对应搜索方法的非漏报性.

Yi等以一条序列作为基准序列,以另一条序列中大于基准序列最大值的点集以及小于基准序列最小值的点集作为对象,以其为基础构造下界距离LB_Yi. Kim等以时间序列的最大值点、最小值点、起始点和结束点作为特征,构造DTW下界距离LB_Kim. Keogh等利用弯曲路径的全局约束条件,提取查询序列的上下边界特征,构造下界距离LB_Keogh,并验证其过滤效果优于LB_Yi和LB_Kim. 此外,Zhu等[11]、Zhou等[12]分别对LB_Keogh进行了改进,进一步提高了过滤效果. 然而,LB_Keogh及其改进方法依赖于弯曲路径的全局约束条件,且要求被度量的序列长度相等,因而限制了其应用范围.

本文将提前终止与下界距离方法相结合,提出一种支持DTW距离的过滤搜索方法. 首先,构造一种计算代价较低的DTW下界距离,用其进行过滤,得到候选集;然后,利用提前终止策略优化计算候选集中序列的DTW距离,得到搜索结果;最后,通过实验对所提出方法进行有效性验证,得到了满意效果.

2 DTW距离的过滤搜索方法

时间序列搜索包括k近邻搜索和ε范围搜索,两种搜索在一定条件下能够相互转换. 不失一般性,文中主要针对ε范围搜索进行讨论.

2.1 提前终止策略

时间序列$Q=(q_1,q_2,\cdots,q_n)$、$C=(c_1,c_2,\cdots,c_m)$的DTW距离定义为[13]

$$D_{\rm dtw}(Q,C)=D_{\rm base}(q_1,c_1)+\min\begin{cases}D_{\rm dtw}(Q,C[2:-]);\\ D_{\rm dtw}(Q[2:-],C);\\ D_{\rm dtw}(Q[2:-],C[2:-]).\end{cases}\tag{1}$$

其中$D_{\rm base}(q_1,c_1)$表示点q_1与c_1的基距离,本文采用Minkowski距离.

计算DTW距离时,通常构造一个m行n列的累积距离矩阵,矩阵中元素$\gamma_{i,j}$定义为

$$\gamma_{i,j}=D_{\rm base}(q_i,c_j)+\min\{\gamma_{i-1,j},\gamma_{i,j-1},\gamma_{i-1,j-1}\}.\tag{2}$$

计算过程为:按照一定的方向逐行(列)计算矩阵元素$\gamma_{i,j}$,直至求出$\gamma_{m,n}$;并用它与阈值ε比较:当$\gamma_{m,n}\leqslant\varepsilon$时,$C$为搜索结果. 其中需要计算累积距离矩阵中全部$m\times n$个元素,DTW距离的计算复杂度为$O(m\times n)$.

提前终止策略表述为:在累积距离矩阵中,按一定的方向逐行(列)计算,完成一行(列)的计算时,如果该行(列)上所有元素值都大于ε,则无需进行其余计算便可知$D_{\rm dtw}(Q,C)>\varepsilon$,从而推断$C$不是搜索结果,证明见文献[5]. 这样,便减少了DTW距离的计算代价.

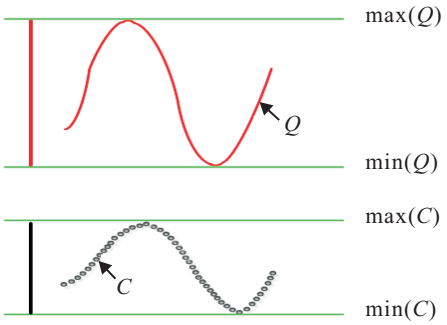
2.2 DTW下界距离

下面构造一种用于过滤搜索且计算复杂度较低的DTW下界距离. DTW距离用于确定两条序列上点之间的对齐匹配关系,每种匹配关系可以用一条弯曲路径表示. 弯曲路径满足3个基本条件:边界条件、连续性和单调性. 其中,边界条件要求两个序列的起始点与结束点对应匹配.

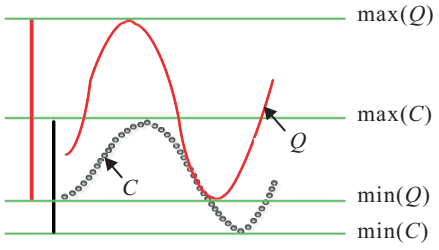
对于时间序列Q和C,假设$\max(Q)>\max(C)$,两序列之间的位置关系共有3种:1) 相离,$\min(Q)>\max(C)$(见图1 (a));2) 相交,$\min(Q)\leqslant\max(C)$、$\min(Q)>\min(C)$(见图1(b));3) 包容,$\min(Q)\leqslant\max(C)$、$\min(Q)\leqslant\min(C)$(见图1(c)).

DTW下界距离可以构造为

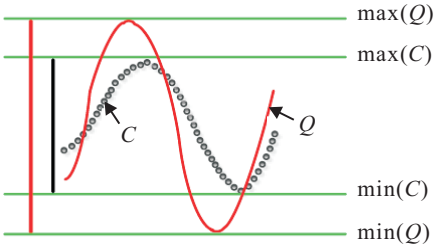
$$D_{lb}(Q,C)=|q_1-c_1|+|q_n-c_m|+\begin{cases}1)\ \max\Big\{\sum\limits_{i=2}^{n-1}|q_i-\max(C)|,\ \sum\limits_{j=2}^{m-1}|\min(Q)-c_j|\Big\};\\ 2)\ \sum\limits_{i=2}^{n-1}\phi(q_i,\max(C))+\sum\limits_{j=2}^{m-1}\phi(\min(Q),c_j);\\ 3)\ \sum\limits_{i=2}^{n-1}\phi(q_i,\max(C))+\sum\limits_{i=2}^{n-1}\phi(\min(C),q_i).\end{cases}\tag{3}$$



(a) 序列相离



(b) 序列相交



(c) 序列包容

图 1　两条序列之间的 3 种位置关系

$$\phi(x,y)=\begin{cases}|x-y|, & x>y;\\ 0, & \text{otherwise.}\end{cases} \tag{4}$$

其中: $|q_1-c_1|+|q_n-c_m|$ 表示 Q、C 的起始点与结束点对应的累积距离, 1)～3) 分别表示序列相离、相交和包容的情况. 可以看出, $D_{lb}(Q,C)$ 的计算复杂度为 $O(m+n)$, 与序列长度呈线性关系.

根据下界距离引理[14], 如果有

$$D_{lb}(Q,C)\leqslant D_{\text{dtw}}(Q,C), \tag{5}$$

则使用 D_{lb} 进行过滤搜索不会产生漏报.

下面证明式 (5) 成立. 计算 Q、C 的 DTW 距离时, 由边界条件可知, 起始点与结束点对应匹配. 因此, $|q_1-c_1|+|q_n-c_m|$ 是 $D_{\text{dtw}}(Q,C)$ 的一部分, 它是 Q、C 起始点与结束点对应的累积距离.

再根据序列之间的 3 种位置关系, 分析除了起始点和结束点之外, 其余点的情况.

首先考虑序列相离的情况. 可以从两个视角理解 $D_{\text{dtw}}(Q,C)$ 的计算: 以 Q 为基准, 序列 Q 上的每一个点都与序列 C 上的一个或多个点匹配; 以 C 为基准, 序列 C 上的每一个点都与序列 Q 上的一个或多个点匹配.

当 Q、C 的位置为相离时, 以 Q 为基准的情况下, $\sum_{i=2}^{n-1}|q_i-\max(C)|$ 不大于 $D_{\text{dtw}}(Q,C)$ 中, Q 除起始点和结束点之外, 其余点与 C 上的点对累积距离. 以 C 为基准的情况下, $\sum_{j=2}^{m-1}|\min(Q)-c_j|$ 也不大于 $D_{\text{dtw}}(Q,C)$ 中, C 除起始点和结束点之外, 其余点与 Q 上的点对累积距离. 因此, 在序列相离的情况下, 有式 (5) 成立.

对于序列相交和包容的情况, 也可以用类似方法进行证明.

2.3　过滤搜索算法及其复杂度分析

过滤搜索算法通过计算代价较低的 D_{lb} 过滤掉部分序列, 再使用提前终止方法进行筛选, 避免直接计算 DTW 距离, 从而提高搜索效率. 该算法的程序如下:

Giver Q, $R=\{\}$;
/* 下界距离过滤 */
foreach sequence C_i in the dataset T
　if $D_{lb}(Q,C_i)\leqslant\varepsilon$
　then add C_i to the candidate set B;
end
/* 提前终止策略 */
foreach sequence S_j in the candidate set B
　if EA-DTW $(Q,S_j)\leqslant\varepsilon$
　then add S_j to R;
end
return R.

下面比较 DTW、提前终止方法 (EA-DTW)、Yi 等[8] 所提方法 (LB_Yi+DTW) 和本文方法 (D_{lb}+EA-DTW) 的计算复杂度.

数据集中序列数为 N, 序列均长为 L; 提前终止方法平均计算到第 $K(K<L)$ 行结束计算; 下界距离 D_{lb}、LB_Yi 过滤掉的序列数分别为 V、$V'(V<N, V'<N)$.

它们的计算代价分别为 $N\times L\times L$, $N\times K\times L$, $N\times L\times 2+(N-V)\times K\times L$, $N\times L\times 2+(N-V')\times L\times L$. 分别用后 3 种方法除以 DTW 的计算代价, 有如下公式:

$$\frac{f_{\text{EA-DTW}}}{f_{\text{DTW}}}=\frac{K}{L}, \tag{6}$$

$$\frac{f_{D_{lb}+\text{EA-DTW}}}{f_{\text{DTW}}}=\frac{1}{L}\times\left[2+K\left(1-\frac{V}{N}\right)\right], \tag{7}$$

$$\frac{f_{\text{LB\_Yi+DTW}}}{f_{\text{DTW}}}=\frac{1}{L}\times\left[2+L\left(1-\frac{V'}{N}\right)\right]. \tag{8}$$

3 实验与分析

选取3组公开的时间序列数据集作为实验对象[15]: Adiac、Beef和Gun_Point, 详细信息见表1. 实验环境为: Matlab R2010a, Windows 7, Intel(R) Core(TM) i7-3770 CPU, 4 G RAM. 取数据集前5个序列, 依次作为查询序列, 执行相似性搜索, 比较4种方法的计算代价, 以5次结果的均值作为比较依据.

表1 实验数据集

数据集	序列数	序列均长	分类数
Adiac	781	176	37
Beef	60	470	5
Gun_Point	200	150	2

LB_Yi+DTW和D_{lb}+EA-DTW都利用下界距离过滤, 现对其过滤效果进行比较, 见图2.

随着ε的增大, LB_Yi和D_{lb}过滤掉的序列数都逐渐减少, 但D_{lb}过滤掉的序列数始终不小于LB_Yi. 这是因为, 与LB_Yi相比, 构造D_{lb}时, 利用了DTW距离计算的边界条件, 把起始点与结束点对应的累积距离加入到下界距离中, 提高了紧致性.

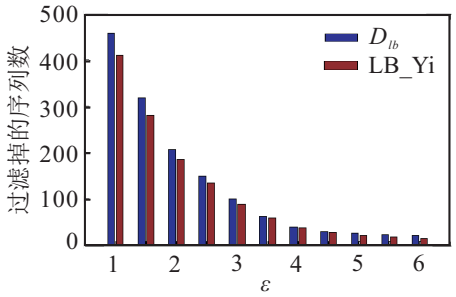
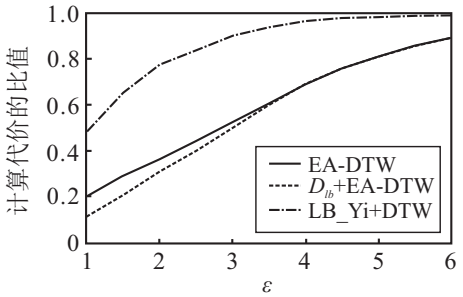


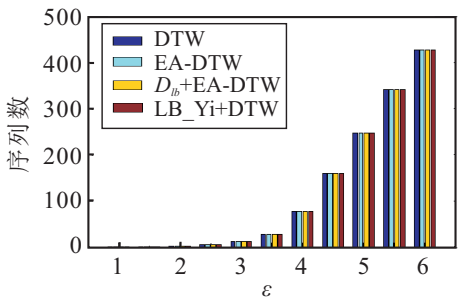
图2 下界距离在Adiac上过滤效果的比较

图3(a)记录了Adiac数据集上4种方法的计算代价. 总体来看, ε在一定范围内取值时, EA-DTW、LB_Yi+DTW和D_{lb}+EA-DTW的计算复杂度都低于DTW. 随着ε的增大, 3种方法的计算复杂度逐步提高, LB_Yi+DTW接近甚至高于DTW方法. 这是由于LB_Yi的过滤效果逐渐减弱, 大部分序列都通过随后的DTW进行相似性度量.

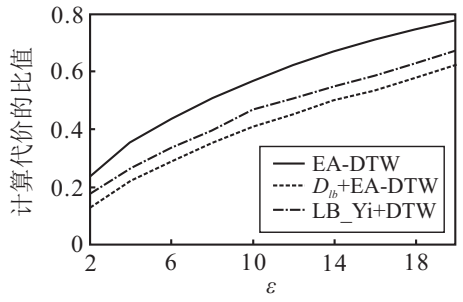
当ε较小时, D_{lb}+EA-DTW的计算代价最低; 当ε增大到一定程度时, 其计算代价逐步接近甚至略高于EA-DTW. 这是因为随着ε的增加, D_{lb}的过滤效果逐渐减弱, 大部分序列要靠随后的EA-DTW进行筛选.



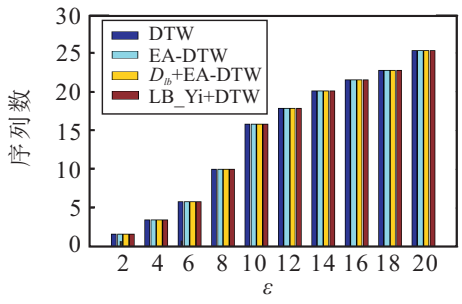
(a) Adiac上计算代价的比较



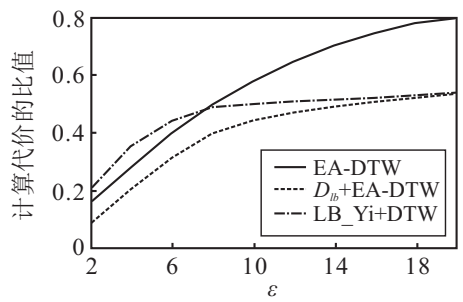
(b) 4种方法在Adiac上搜索到的序列数



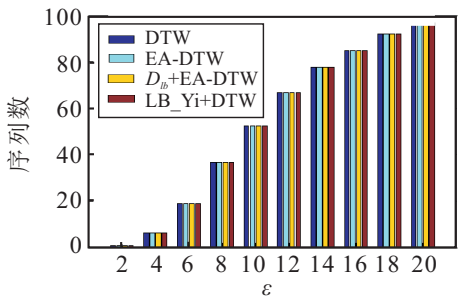
(c) Beef上计算代价的比较



(d) 4种方法在Beef上搜索到的序列数



(e) Gun_Point上计算代价的比较



(f) 4种方法在Gun_Point上搜索到的序列数

图3 3组数据集上的实验结果

图3(b)记录了ε取不同值时,4种方法在5次搜索中得到的平均序列数. 当$\varepsilon = 4$时,平均序列数为78.6,约占Adiac数据集序列总数的10%,可满足大多数范围查询的需求. 而在$\varepsilon < 4$时,D_{lb}+EA-DTW计算代价最低,因此,D_{lb}+EA-DTW在Adiac数据集上是适用的.

同时可以看出,ε取不同值时,D_{lb}+EA-DTW得到的序列数与其他3种方法相同,即找到了满足相似性要求的全部序列,验证了其非漏报性.

在其他数据集上,分别比较4种方法的计算代价和搜索到的序列数,见图3(c)～图3(f).

Beef数据集上,LB_Yi+DTW和D_{lb}+EA-DTW的计算代价均低于DTW和EA-DTW,表明下界距离LB_Yi和D_{lb}的过滤效果比较明显.

Gun_Point数据集上,$\varepsilon = 6$时,得到的平均序列数约占序列总数的10%. 当$\varepsilon < 6$时,D_{lb}+EA-DTW计算代价最低;当$\varepsilon > 6$时,LB_Yi+DTW与D_{lb}+EA-DTW的计算代价接近. 这是由于当$\varepsilon < 6$时,下界距离LB_Yi和D_{lb}均能过滤掉大部分序列,但LB_Yi+DTW用DTW处理候选集,增加了计算代价. 当$\varepsilon > 6$时,下界距离LB_Yi和D_{lb}仅能过滤掉部分序列,大部分序列进入候选集. 随着ε的增加,与DTW相比,EA-DTW的提前终止效果逐步减弱.

4 结 论

本文构造了一种紧致性高于LB_Yi的DTW下界距离D_{lb},结合提前终止和下界距离方法,提出了一种支持DTW距离的过滤搜索方法,能够提高相似性搜索效率,且具有非漏报性. 与LB_Keogh方法相比,所提出方法不依赖于弯曲路径的全局约束条件,且不要求被度量的序列长度相等.

参考文献(References)

[1] Saeed Aghabozorgi, Ali Seyed Shirkhorshidi, Teh Ying Wah. Time-series clustering — A decade review[J]. Information Systems, 2015, 53: 16-38.

[2] Joan Serrà, Josep Ll Arcos. An empirical evaluation of similarity measures for time series classication[J]. Knowledge-Based Systems, 2014, 67(3): 305-314.

[3] Berndt D J, Clifford J. Using dynamic time warping to find patterns in time series[C]. Proc of the Workshop on Knowledge Discovery in Databases. Seattle, 1994: 229-248.

[4] Romain Tavenard, Laurent Amsaleg. Improving the efficiency of traditional DTW accelerators[J]. Knowledge and Information Systems, 2015, 42(1): 215-243.

[5] Kim Man-Soon, Kim Sang-Wook, Shin Miyoung. Optimization of subsequence matching under time warping in time-series databases[C]. ACM Symposium on Applied Computing. New Mexico, 2005: 581-586.

[6] 李正欣,张凤鸣,李克武,等. 一种支持DTW距离的多元时间序列索引结构[J]. 软件学报, 2014, 25(3): 560-575.
(Li Z X, Zhang F M, Li K W, et al. Index structure for multivariate time series under DTW distance metric[J]. J of Software, 2014, 25(3): 560-575.)

[7] Wong Teddy Siu Fung, Wong Man Hon. Efficient subsequence matching for sequences databases under time warping[C]. Proc of the 7th Int Database Engineering and Applications Symposium. Hong Kong, 2003: 139-148.

[8] Yi B, Jagadish H V, Faloutsos C. Efficient retrieval of similar time sequences under time warping[C]. Proc of the 14th Int Conf on Data Engineering. Washington DC, 1998: 201-208.

[9] Kim Sang-Wook, Sanghyun Park, Chu Wesley W. An index-based approach for similarity search supporting time warping in large sequence databases[C]. Proc of the 17th Int Conf on Data Engineering. Heidelberg, 2001: 607-614.

[10] Keogh E, Ratanamahatana C. Exact indexing of dynamic time warping[J]. Knowledge and Information Systems, 2005, 7(3): 358-386.

[11] Zhu Yunyue, Shasha Dennis. Warping indexes with envelope transforms for query by humming[C]. Proc of the 2003 ACM SIGMOD Int Conf on Management of Data. San Diego, 2003: 181-192.

[12] Zhou Mi, Wong Man Hon. Boundary-based lower-bound functions for dynamic time warping and their indexing[J]. Information Sciences, 2011, 181(19): 4175-4196.

[13] 李海林,梁叶. 分段聚合近似和数值导数的动态时间弯曲方法[J]. 智能系统学报, 2016, 11(2): 249-256.
(Li H L, Liang Y. Dynamic time warping based on piecewise aggregate approximation and data derivatives[J]. CAAI Trans on Intelligent Systems, 2016, 11(2): 249-256.)

[14] Faloutsos C, Ranganathan M, Manolopoulos Y. Fast subsequence matching in time-series databases[C]. Proc of the ACM SICMOD Conf on Management of Data. New York, 1994: 419-429.

[15] Keogh E, Xi X, Wei L, et al. The UCR time series classification/clustering homepage[EB/OL]. (2016-10-16). http: //www.cs.ucr.edu/～eamonn/time_series_data/.

(责任编辑: 李君玲)