文章编号:1001 - 9081(2007)08 - 1976 - 04

多数据流的实时聚类算法

陈 崚[1,2],邹凌君[1],屠 莉[3]
(1. 扬州大学 信息工程学院, 江苏 扬州 225009;
2. 南京大学 计算机软件新技术国家重点实验室, 南京 210093;
3. 南京航空航天大学 信息与科学技术学院, 南京 210093)
(lchen@yzcn.net)

摘 要:针对当前对多条数据流的聚类算法不能兼顾质量和效率的矛盾,提出了基于相关系数的多条数据流的聚类算法,实现固定长度的在线动态聚类。算法引入衰减系数提高聚类质量,以相关系数作为流数据间相似度的度量标准,将数据流划分若干个数据段,以各数据流的相关统计信息进行聚类,得到实时的聚类结构。实验结果表明,算法有较高的效率、聚类质量和稳定性。

关键词:聚类;流数据;相关系数
中图分类号: TP311 **文献标志码**:A

Real-time clustering algorithm for multiple data streams

CHEN Ling[1,2], ZOU Ling-jun[1], TU Li[3]
(1. *Department of Computer, Yangzhou University, Yangzhou Jiangsu* 225009, *China*;
2. *State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing Jiangsu* 210093, *China*;
3. *Institute of Information Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing Jiangsu* 210093, *China*)

Abstract: To overcome the imbalance between clustering quality and efficiency in current multiple data streams clustering algorithms, a clustering algorithm based on correlation coefficient was proposed. The algorithm can dynamically discover the clusters in the data streams over a fixed time period. The attenuation coefficient was introduced to improve the performance of clustering and the correlation coefficient was used to measure the similarity between data streams. In the algorithm, the time horizon was divided into several equal segments and statistical information was computed for stream data in each time segment. The algorithm can modify the clustering structure according to the statistical information in real time. Experimental results show that the algorithm has higher efficiency, clustering quality and stability than other methods.

Key words: clustering; data streams; correlation coefficient

0 引言

流数据是近年来基于实时应用产生的一类新的数据对象,它是一种大量的连续到达、时间有序的、快速变化、潜在无限的数据[1,2]。在流数据模型中,只能按数据的到达顺序依次访问,不能随机存取数据。相对于大量、潜在无限的流数据,内存的容量很小,只能存储有限的信息。且对流数据的访问只能一次或有限次,多次访问需要的代价很高。目前对流数据挖掘的研究方向主要集中在流数据的分类[3,4]、频繁模式挖掘[5-7]和聚类[8-18]方面,其中聚类问题是流挖掘的研究热点,有很大的应用前景。

目前对流数据的聚类的研究是对单条流数据中的数据成员进行聚类。然而在一些应用中需要对多条数据流进行聚类。对多条数据流聚类是将每一数据流看成一个聚类对象,考虑的是数据流间的相似度。目前在这方面的研究较少,文献[16]用带权重的快照差的和作为流数据间距离的度量,不能反映流数据间趋势变化的相似度。文献[17]通过对流数据标准化等预处理后用离散傅立叶变换减少噪声,用增量在线的 k-means 算法进行聚类。算法质量和执行效率都依赖于 DFT 系数个数,难以在效率和质量间达到平衡。文献[18]提出了一种自适应的聚类多条流数据的算法,在线阶段使用一种分层机制保存概要信息。离线阶段设计了一种自适应聚类算法,但算法的在线部分计算统计信息的时间较长。

针对以上算法的局限性,本文提出基于相关系数的多数据流聚类算法 CORREL-cluster,使用相关系数作为距离标准,引入衰减系数提高聚类质量。以各数据流的统计信息对数据聚类,克服了现有算法中效率和质量的矛盾。实验证明了 CORREL-cluster 算法有较高的聚类质量和稳定性。

1 问题描述和相关概念

假设在时间 t 有 n 条数据流 $\{X_1, X_2, \cdots, X_n\}$,其中 $X_i = (x_{i1}, x_{i2}, \cdots, x_{im})(1 \leqslant i \leqslant n)$,$x_{ij}$ 是流数据 X_i 在时间 j 到达的数值。对多条数据流在时间 t、跨度 L 上的聚类,就是要将流数据分为 k 类:$C_1(L), C_2(L), \cdots, C_k(L)$,使得评价聚类质量的目

收稿日期:2007 - 01 - 29;**修回日期**:2007 - 04 - 12。 **基金项目**:国家科技攻关项目(2003BA614A-14);国家自然科学基金资助项目(60673060);江苏省自然科学基金资助项目(BK2005047);南京大学软件新技术国家重点实验室开放基金资助项目。

作者简介:陈崚(1951 -),男,江苏宝应人,教授,博士生导师,主要研究方向:数据挖掘和并行计算; 邹凌君(1984 -),女,江苏六合人,硕士研究生,主要研究方向:数据挖掘和并行计算; 屠莉(1980 -),女,江苏江阴人,博士研究生,主要研究方向:数据挖掘和并行计算。

标函数 G 在 $[t_{now}-L+1,t_{now}]$ 内最大。且满足:$\bigcap_{j=1}^{k} C_j(L)=\varnothing$,同时,$\bigcup_{j=1}^{k} C_j(L)=\{X_1(L),X_2(L),\cdots,X_n(L)\}$。其中 $X_q(L)=\{x_{q(t_{now}-L+1)},x_{q(t_{now}-L+2)},\cdots,x_{qt_{now}}\}$,$1\leqslant q\leqslant n$。

在流数据中,较"旧"的数据对于现在聚类结果的影响应该较小,因为查询总是对较新的数据感兴趣。所以在聚类时要对数据加上时间权值 λ(如取 $\lambda=0.99$),称为衰减系数。这样,流数据 x_i 在时间 t 参与聚类处理时的实际值为 $\lambda^{t-i}x_i$,记为 $x_i(t)$。

由于流数据的特点,我们通常仅对一个长为 L 的最新的时间片内的数据进行聚类。算法在时间为 $t-L+1$ 至 t 的时间片里,对流数据实际上处理的是 $[x_{t-L+1}(t),\cdots,x_t(t)]$。

为了方便计算和更新汇总信息,算法 CORREL-cluster 将长度为 L 的时间片里的数据分为 m 段,每段长为 l 个单位时间。在任意时刻,算法保存 m 个数据段。在每一个数据段加入后,将其有关信息保存到存储区中,同时去掉最"旧"的一个数据段。

2　流数据的处理

算法 CORREL-cluster 使用相关系数作为相似度的度量标准对多条数据流进行聚类。这种距离度量的标准比使用欧氏距离等方法在某些应用中更加精确。因为对于多条数据流而言,每条数据流中的数据都带有时间戳,这不同于静态数据的各个分量,用欧氏距离来度量它们之间的相关程度,失去了相同时间数据间的对应关系信息,不能反映各个数据流随时间动态变化的趋势。而利用相关系数可以衡量数据流之间发展趋势的一致性。

对于序列 $X=(x_1,x_2,\cdots,x_n)$,$Y=(y_1,y_2,\cdots,y_n)$,其相关系数为 $\rho_{xy}=\dfrac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2(y_i-\bar{y})^2}}$,其中 $\bar{x}=\dfrac{x_1+x_2+\cdots+x_n}{n}$,$\bar{y}=\dfrac{y_1+y_2+\cdots+y_n}{n}$。

由上述定义可知,$|\rho_{xy}|\leqslant 1$。ρ_{xy} 反映了 X,Y 之间的相关程度,ρ_{xy} 值越大,相关程度就越高。

流数据的瞬时性和无限性,使得在对它们进行聚类时,算法只能存储和使用部分新数据或者某些概要信息来实时追踪聚类的变化。在算法 CORREL-cluster 中,为了计算相关系数 ρ,要对每一条数据流 X 记录统计信息 $\sum x_i$,$\sum x_i^2$;对每一对数据流 X、Y 记录统计信息 $\sum x_iy_i$。因而算法在读入每一个数据 x_i 后,将其转换成 $x_i(t)$,并进行信息的累加,得到统计信息 $\sum x_i$,$\sum x_i^2$,$\sum x_iy_i$。在一定的时间间隔以后,算法根据统计信息进行聚类。

定理　对于 X,Y 序列,保存 $\sum x_i$,$\sum y_i$,$\sum x_iy_i$,$\sum x_i^2$,$\sum y_i^2$ 就可以计算出相关系数 ρ_{xy}。

证明　因为 $\sum(x_i-\bar{X})(y_i-\bar{Y})=\sum x_iy_i-\frac{1}{n}\sum x_i\sum y_i$。因此,$\rho_{xy}$ 的分子可由 $\sum x_i$、$\sum y_i$ 及 $\sum x_iy_i$ 计算出来。因为 $\sum(x_i-\bar{X})^2=\sum x_i^2-\frac{1}{n}(\sum x_i)^2$,由此可见 $\sum(x_i-\bar{X})^2$ 可以由 $\sum x_i$ 及 $\sum x_i^2$ 计算出来。同理,$\sum(y_i-\bar{Y})^2$ 也可以由 $\sum y_i$ 及 $\sum y_i^2$ 计算得到。因此,ρ_{xy} 的分母可由 $\sum x_i$,$\sum x_i^2$,$\sum y_i$,$\sum y_i^2$ 算得。　证毕。

为了计算 ρ_{xy},算法 CORREL-cluster 对每一数据流 X 在每一数据段所保存的统计信息仅为:$(\sum x_i,\sum x_i^2,l_c,t_c)$,其中,$t_c$ 为该组中的最后一个元素的到达时间,l_c 为长度。对于相应时间段中的每一对数据流 X、Y,保存的统计信息为 $\sum x_iy_i$。

对于以前在 t_c 时刻处理过的数据段的统计信息,在 t 时刻($t_c<t$)使用时,由于它们当时的数据为 $x_{ki}(t_c)$,现在需要转换成 $x_{ki}(t)$ 才能使用,因而要对统计信息作相应的更新。设 $t-t_c=t_\Delta$,则 $x_{ki}(t)=\lambda^{t_\Delta}\cdot x_{ki}(t_c)$。因此,在 t 与 t_c 时刻的相应的统计信息 $\sum_i x_{ki}(t)$ 与 $\sum_i x_{ki}(t_c)$ 之间有如下的关系:

$$\sum_{i=t_c-l_c+1}^{t_c} x_{ki}(t)=\lambda^{t_\Delta}\sum_{i=t_c-l_c+1}^{t_c} x_{ki}(t_c)$$

$$\sum_{i=t_c-l_c+1}^{t_c} x_{ki}^2(t)=\lambda^{2t_\Delta}\sum_{i=t_c-l_c+1}^{t_c} x_{ki}^2(t_c)$$

$$\sum_{i=t_c-l_c+1}^{t_c} x_{ki}(t)y_{ji}(t)=\lambda^{2t_\Delta}\sum_{i=t_c-l_c+1}^{t_c} x_{ki}(t_c)y_{ji}(t_c)$$

对数据流进行聚类时,算法 CORREL-cluster 要对不同时间的数据段里的统计信息进行合并。设 $t_1<t_2<t_3$,$l_1=t_2-t_1$,$l_2=t_3-t_2$,则相邻时间段 (t_1+1,t_2),(t_2+1,t_3) 相应的统计信息:

$$\left[\sum_{i=t_1+1}^{t_2} x_{ki}(t_2),\ \sum_{i=t_1+1}^{t_2} x_{ki}^2(t_2),\ \sum_{i=t_1+1}^{t_2} x_{ki}(t_2)y_{ji}(t_2),k,\ j=1,\cdots,n,k\neq j,t_2,l_1\right]$$

以及:

$$\left[\sum_{i=t_2+1}^{t_3} x_{ki}(t_3),\ \sum_{i=t_2+1}^{t_3} x_{ki}^2(t_3),\ \sum_{i=t_2+1}^{t_3} x_{ki}(t_3)y_{ji}(t_3),k,\ j=1,\cdots,n,k\neq j,t_3,l_2\right]$$

合并后成为:

$$\left[\sum_{i=t_1+1}^{t_3} x_{ki}(t_3),\ \sum_{i=t_1+1}^{t_3} x_{ki}^2(t_3),\ \sum_{i=t_1+1}^{t_3} x_{ki}(t_3)y_{ji}(t_3),k,\ j=1,\cdots,n,k\neq j,t_3,l_1+l_2\right]$$

其中:

$$\sum_{i=t_1+1}^{t_3} x_{ki}(t_3)=\lambda^{l_2}\sum_{i=t_1+1}^{t_2} x_{ki}(t_2)+\sum_{i=t_2+1}^{t_3} x_{ki}(t_3)$$

$$\sum_{i=t_1+1}^{t_3} x_{ki}^2(t_3)=\lambda^{2l_2}\sum_{i=t_1+1}^{t_2} x_{ki}^2(t_2)+\sum_{i=t_2+1}^{t_3} x_{ki}^2(t_3)$$

$$\sum_{i=t_1+1}^{t_3} x_{ki}(t_3)y_{ji}(t_3)=\lambda^{2l_2}\sum_{i=t_1+1}^{t_2} x_{ki}(t_2)y_{ji}(t_2)+\sum_{i=t_2+1}^{t_3} x_{ki}(t_3)y_{ji}(t_3)$$

3　算法框架和动态 k-means 算法

算法 CORREL-cluster 的总体框架如下:

输入：n 条数据流 $x_1, x_2, \cdots, x_n$ 输出：在各个时间片的聚类结果；
begin
1) $t = 0$；
2) While 数据流没有结束 do
3) 同时读入各条数据流上的一个数据 x_{ki}，构成 $x_{ki}(t)$，$t = t + 1$；
4) if $t \bmod l = 0$ then
5) {计算该数据段的各 $\sum x_{ki}(t)$，$\sum x_{ki}^2(t)$ $\sum x_{ki}x_{ji}(t)$。
6) 将该数据段加入数据段组中，作为最新的一个数据段。
7) 如果数据段个数大于 m，则去掉最“旧”的一个数据段。
8) 使用动态 k-means 聚类算法 Dynamic-k-means 进行聚类。
9) 使用聚类调整算法 adjust 进行聚类调整，更新 k 的值；
10) 输出聚类结果 }
11) Endwhile
end

在算法的第 8 行，本文提出一种动态的 k-means 的聚类算法。该算法首先用 k-means 方法产生初始聚类。在以后的各次聚类操作中，由于流数据的变化是逐渐的，相邻两次的聚类结果之间有大部分是重叠的。因而每次聚类时，仅需在前一次聚类的基础上，用少量的几次 k-means 迭代就可以得到结果。

在聚类过程中，以数据流之间相关系数 ρ 的倒数作为距离，使用目标函数 $G = \sum_{i=1}^{k} \sum_{x_j \in c_i} \rho_{x_j c_i'}$，评价聚类质量，$G$ 值越大，说明聚类的质量越高。其中 c_i' 是类 c_i 的中心，$\rho_{x_j c_i'}$ 是数据流 x_j 与相应聚类中心 c_i' 之间的相关系数。设 g 为迭代次数，n 为数据流的条数。动态 k-means 聚类算法框架如下：

Algorithm Dynamic-k-means (k, $Center_k$, R_k)
输入：类的个数 k，中心点的集合 $Center_k$，当前聚类方案 R_k；
输出：更新后的聚类方案 R_k 及其目标函数 G_K；
begin
1) for $i = 1$ to g do
for $j = 1$ to n do
计算数据流 X_j 到 k 个中心点的距离，将 X_j 归入与其最近的中心点所在的类中；
endfor j
2) 对各个类计算出新的中心，更新中心点集合 $Center_k$；
3) endfor i
4) 计算目标函数 G_K。
end

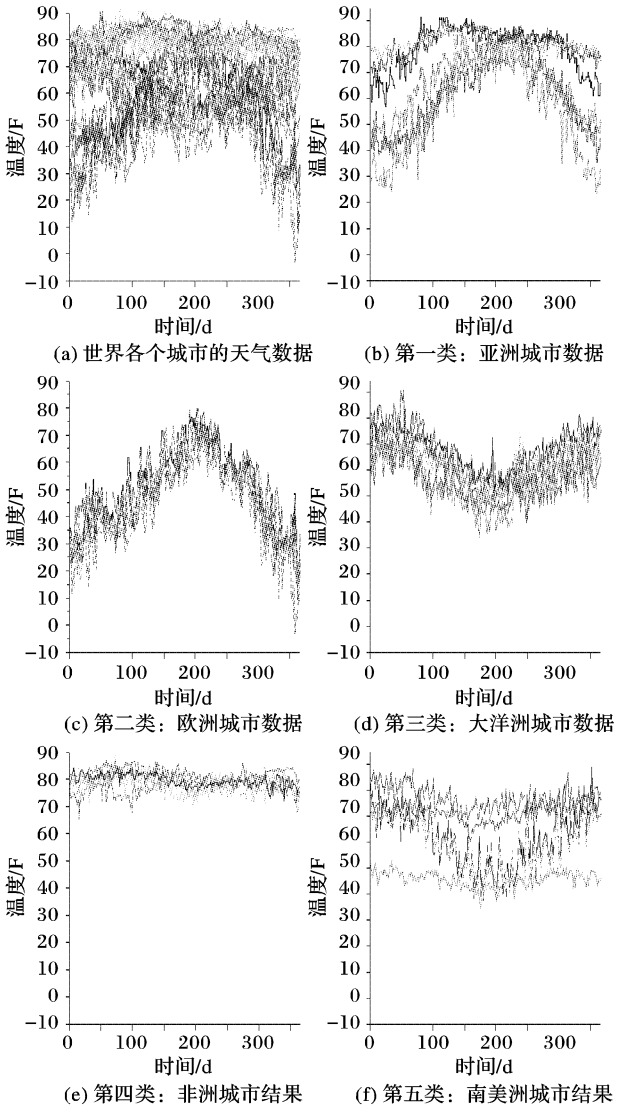
因为在实际应用中，聚类的个数 k 不会固定在某一个常数上，算法 CORREL-cluster 在第 9 行使用聚类调整算法 Adjust 进行聚类调整，更新 k 的值。调整 k 的大小时，只需考虑 k 加 1 或减 1 的情况。设目前具有 k 个类的聚类结果为 R_k，若将其调整为具有 $k-1$、$k+1$ 个类的聚类结果分别为 R_{k-1}、R_{k+1}，算法在三者中取质量最高的作为新的聚类结果，同时相应地更新聚类的个数 k。聚类调整算法 Adjust 的框架如下：

Algorithm Adjust (k, $Center_k$, R_k)
输入：类的个数 k，中心点的集合 $Center_k$，当前聚类方案 R_k；
输出：更新后的聚类个数 k，聚类方案 R_k 及其目标函数 G_K；
begin
1) 试探性计算 R_{k+1}：
(1) 在每个类中，选取一个离类中心点最远的点 X 作为新增的类中心；
(2) $Center_{k+1} = Center_k \cup \{X\}$；
(3) Dynamic-k-means ($k+1$, $Center_{k+1}$, R_{k+1})；
2) 试探性计算 R_{k-1}：
(1) 选取两个类中心的距离最小的类，设其中心点为 C_1、C_2；
(2) 合并这两个类，求这个新类的中心 C_3；
(3) $Center_{k-1} = Center_k \cup \{C_3\} - \{C_1, C_2\}$；
(4) Dynamic-k-means ($k-1$, $Center_{k-1}$, R_{k-1})；
3) 在 R_{k-1}、R_k、R_{k+1} 三者中取质量最高者，设为 $R_{k'}$；
则 $k = k'$，$R_k = R_{k'}$，$G_K = G_{K'}$
end

4 实验结果及分析

为了测试算法 CORREL-cluster 的性能，我们对世界气象数据集做了一系列实验，实验运行环境为 Windows XP 操作系统，内存为 512 MB，用VC++ 6.0 编程。

使用的实际数据集为世界 169 个城市 1995 年 1 月至 2006 年 10 月每天的气温数据。把每个城市看成一条数据流，每条数据流中有 3416 个数据。取聚类长度 L 为 360，每个片段长 m 为 30，测试实验得到的聚类结果如图 1 所示。图 1(a)为算法的输入数据，即世界部分城市的天气数据。算法 CORREL-cluster 根据气温的变化趋势，得到了如图 1(b ~ f)所示的五个类，每个类中的城市均属于在同一洲，属于同一气温带，证明了聚类结果的有效性。



(a) 世界各个城市的天气数据 (b) 第一类：亚洲城市数据
(c) 第二类：欧洲城市数据 (d) 第三类：大洋洲城市数据
(e) 第四类：非洲城市结果 (f) 第五类：南美洲城市结果
图 1 对世界部分城市天气数据的聚类结果

在上述数据集上分别运行 CORREL-cluster 算法与 DFT-cluster 算法[17]（30 个 DFT 系数），比较两者的正确率，其测试

结果如图 2 所示。

图 2 表明,CORREL-cluster 算法在各种片段数下的正确率均比 DFT-cluster 算法高。这是因为,DFT-cluster 算法对系数个数有较大的依赖性,只有增加系数的个数,它的聚类质量才能有所提高。此外,由于 CORREL-cluster 算法采用相关系数来进行聚类,充分考虑了不同数据流上的数据间在时间上的对应关系。而 DFT-cluster 算法不能考虑数据的变化趋势,影响了聚类的质量。

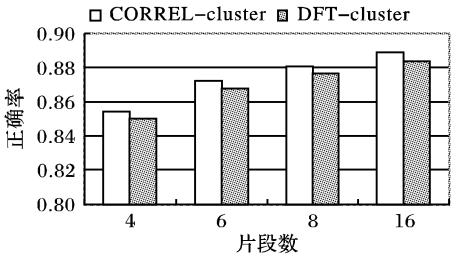


图 2　不同片段数下聚类质量的比较

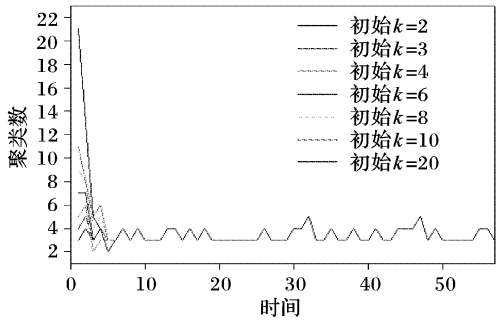


图 3　不同初始聚类个数下聚类数的变化

为了分析 CORREL-cluster 算法的稳定性,我们设置不同的初始聚类个数 k 对相同的数据集进行聚类,分析其聚类结果的变化。图 3 示出了不同初始聚类个数下,聚类数的变化情况。从图中可以看出,设置不同的初始聚类个数 $k(k = 2, 3, \cdots, 20)$,经过一段时间后,聚类个数很快就会变得相同,说明了算法对参数 k 不敏感,有较好的稳定性。这主要是因为 CORREL-cluster 算法采用的是动态 k-means 方法,聚类结构可以随数据流变化被动态地调整,以反映真实的变化情况。

5　结语

提出了基于相关系数的多条数据流的实时聚类算法。算法引入衰减系数来突出新数据比旧数据在聚类结构中有更大的重要性,采用更新时间片段来反映聚类结构的变化过程。用相关系数度量流数据的相似度,相比于欧氏距离等方法,能更准确的刻画趋势变化的相似性。算法有较高的正确率,效率和稳定性。

参考文献:

[1] HAN J, KAMBER M. 数据挖掘: 概念与技术(英文版)[M]. 2 版. 北京: 机械工业出版社, 2006: 467 - 489.

[2] BABCOCK B, BABU S, DATAR M. Models and issues in data stream systems[C]// Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems. New York: ACM Press, 2002: 1 - 16.

[3] AGGARWAL C, HAN J, WANG J, *et al.* On demand classification of data streams[C]// Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. New York: ACM Press, 2004: 503 - 508.

[4] WANG H, FAN W, YU PS, *et al.* Mining concept-drifting data streams using ensemble classifiers[C]// Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2003: 226 - 235.

[5] MANKU G, MOTWANI R. Approximate frequency counts over data streams[C]// Proceedings of the 28th Intl Conference on Very Large Data Bases. San Francisco: Morgan Kaufmann Publishers, 2002: 346 - 357.

[6] GIANNELLA C, HAN J, PEI J, *et al.* Mining frequent patterns in data streams at multiple time granularities[C]// Next Generation Data Mining. Cambridge, Massachusetts: MIT Press, 2003: 191 - 212.

[7] METWALLY A, AGRAWAL D, ABBADI A E. Efficient computation of frequent and top-k elements in data streams[C]// Proceedings of the 10th ICDT International Conference on Database Theory. Berlin: Springer, 2005: 398 - 412.

[8] GUHA S, MISHRA N, MOTWANI R, *et al*. Clustering data streams [C]// Proceedings of 41st Annual Symposium on Foundations of Computer Science. [S.l.]: IEEE Computer Society, 2000: 359 - 366.

[9] GUHA S, MEYERSON A, MISHRA N, *et al.* Clustering datastreams: theory and practice[J]. IEEE Transaction on knowledge and data engineering, 2003, 15(3): 515 - 528.

[10] AGGARWAL C, HAN J W, WANG J Y, *et al.* A framework for clustering evolving data streams[C]// Proceedings of 2003 International Conference on Very Large Data Bases. Berlin: Morgan Kaufmann Publishers, 2003: 81 - 92.

[11] AGGARWAL C, HAN J W, WANG J Y, *et al.* A framework for projected clustering of high dimensional data streams[C]// Proceedings of 2004 International Conference on Very Large Data Bases. Toronto: Morgan Kaufmann Publishers, 2004: 852 - 863.

[12] GUHA S, RASTOGI R, SHIM K. Cure: an efficient clustering algorithm for large databases[C]// Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data. New York: ACM Press, 1998: 73 - 84.

[13] BARBAR D. Requirements for clustering data streams[J]. ACM SIGKDD Explorations, 2003, 3(2): 23 - 27.

[14] O'CALLAGHAN L, MISHRA N, MEYERSON A, *et al.* Streaming-data algorithms for high-quality clustering[C]// International Conference on Data Engineering. Washington: IEEE Computer Society, 2002: 685 - 704.

[15] DATAR M, GIONIS A, INDYK P, *et al.* Maintaining stream statistics over sliding windows[C]// Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms. Philadelphia: Society for Industrial and Applied Mathematics, 2002: 635 - 644.

[16] YANG J. Dynamic clustering of evolving streams with a single pass [C]// Proceedings of IEEE International Conference Data Mining (ICDE '03). Washington: IEEE Computer Society, 2003: 695 - 697.

[17] BERINGER J R, LLERMEIER E. Online clustering of parallel data streams[J]. Data & Knowledge Engineering, 2006, 58(2): 180 - 204.

[18] DAI B R, HUANG J W, YEH M Y, *et al.* Adaptive clustering for multiple evolving streams[J]. IEEE Transaction on Knowledge and Data Engineering, 2006, 18(9): 1166 - 1180.