具有容错特性的两层数据流聚类方法

由育阳 朱纪洪 杨志宏

(清华大学 计算机系, 北京 100084) (中国医学科学院 药用植物研究所, 北京 100193)

摘 要: 提出一种具有容错能力的进化数据流聚类算法 FTGDStream (Fault-Tolerant Grid-Density Clustering over Data Stream),通过在聚类过程中引入适当的松弛条件,从而在含有噪声的真实世界数据中获取更加泛化的有用知识. 首先利用基于相似性度量和小波技术的 HLSFTS (Hierarchical Lifting Scheme Fault-Tolerant Synopses)层次概要数据结构实现在线微聚类过程,然后采用基于网格密度的聚类算法实现离线宏聚类过程. 在线算法所构造的小波概要数据结构对原始数据的高压缩率降低了离线网格密度聚类算法的计算负载,提高了二层数据流聚类算法的效率. 在 UCI 数据集上的仿真实验结果表明,FTGDStream 算法可以聚类任意空间形状的数据并且适用于高维数据流环境,是一种具有容错能力的高效数据流聚类算法.

关 键 词: 进化数据流聚类; 容错; 概要结构; 网格密度

中图分类号: TP 311.13

文献标识码: A **文 章 编 号**: 1001-5965(2012)05-0665-05

Two-layer clustering over data stream with fault-tolerance

You Yuyang Zhu Jihong

(Department of Computer, Tsinghua University, Beijing 100084, China)

Yang Zhihong

(Institute of Medicinal Plant Development, CAMS, Beijing 100193, China)

Abstract: A new envolving data stream clustering algorithm with fault-tolerance characteristic was proposed named FTGDStream (fault-tolerant grid-density clustering over data stream). It introduces appropriate relaxation of conditions for discover generalised knowledge in real world data polluted by noise. First, FTGDStream uses similarity measure technology and lifting wavelet to construct synopsis HLSFTS (hierarchical lifting scheme fault-tolerant synopses) to realize online micro-cluster phase. Second, FTGDStream uses grid-density clustering technology to realize offline macro-cluster phase. High compression ratio of HLSFTS in micro-cluster reduces the computation load of grid-density clustering algorithm in macro-cluster and improves the efficiency of two-layer algorithm. Simulation in UCI data set proves that FTGDStream is able to clustering any shape in data space and suitable for dealing with high-dimensional data streams. FTGDStream is an efficient clustering algorithm with fault-tolerance.

Key words: evolving data stream clustering; fault-tolerance; synopses; grid density

近年来数据流环境下的聚类研究工作广泛展开,具有代表性的相关研究包括基于 k-median 聚类思想的 LOCAL SERACH 算法和在此基础上进行改进的 STREAM 聚类算法[1]. 文献[2]将密度聚类与网格聚类联合应用,提出了基于密度和网格技术的数据流实时聚类算法 D-Stream,实现了数据流环境下的高效密度聚类,相关研究被认为是近年来数据流聚类技术的重要发展. 文献[3]首次提出了进化数据流聚类框架 CluStream,其基本思想是将聚类过程分为在线的微观聚类(mi-

收稿日期: 2011-02-07; **网络出版时间**: 2012-03-20 10:38

网络出版地址: www.cnki.net/kcms/detail/11.2625.V.20120320.1038.011.html

基金项目: 国家自然科学基金资助项目(81102879)

作者简介: 由育阳(1977 -),男,黑龙江哈尔滨人,博士生,arthurwy@163.com.

cro-cluster)过程和离线的宏观聚类(macro-cluster)过程,CluStream 算法是一种通用的数据流聚类算法,用户可以根据不同处理需求在算法基础上进行多种方式的扩展分析,该算法灵活的扩展性是其受到广泛关注的主要原因之一. CluStream 同样存在一些缺点:首先更适合对球形数域空间进行聚类,然而在真实世界数据流中聚簇往往具有非凸的空间形状;其次对于含有大量噪声的原始数据适应性较差;最后需要预先指定聚簇数量,极大的影响了原始数据聚簇的形态分布. 针对以上问题在 CluStream 算法的基础上提出了 HP-Stream 算法[4],采用高维投影和衰减函数来适应对高维数据流的挖掘,但是仍然需要用户预先指定平均聚类数目.

本文提出了一种具有容错特性的数据流网格密度聚类算法 FTGDStream(Fault-Tolerant Grid-Density Clustering over Data Stream),设计了一种新的概要数据结构 HLSFTS(Hierarchical Lifting Scheme Fault-Tolerant Synopses),利用概要结构的单次扫描、高压缩和容错程度可控的特点对新数据点和聚类中心点进行容错度量. 在 FTGDStream 二层框架聚类算法的第 1 过程中,在线微观聚类要求算法对原始数据流进行单次扫描就可以高效的构造小波概要数据结构. 在第 2 过程中,聚类算法在离线状态下进行,无需等待数据积累到希望的规模才进行一次性处理.

1 相关概念

现有的相关研究已经证明任意的传统小波都可以采用惰性提升的方法经过有限次的提升和对偶提升后乘以一个常数得到[5].

定义 1 若(h,g)构成补,则 h 的任何其他补 $g'(z)$都可以表示为 $g'(z)=g(z)+h(z)s(z^2)$,其中 $s(z)$为 Laurent 多项式.

定义 2 若(h,g)构成补,则 g 的任何其他补 $h'(z)$都可以表示为 $h'(z)=h(z)-g(z)t(z^2)$,其中 $t(z)$为 Laurent 多项式.

惰性提升的基本思想是首先将输入信号分成奇偶 2 部分,由偶数项组成信号的低频部分,奇数项组成原始信号的高频部分,然后对低频和高频 2 个数据序列做有限次提升和对偶提升后乘以某常数.

定义 3 存在 2 个数据流片段 $D_x=\{d_{x,1},d_{x,2},\cdots,d_{x,n}\}$和 $D_{x+1}=\{d_{x+1,1},d_{x+1,2},\cdots,d_{x+1,n}\}$,定义相似性函数形如 $S(D_x,D_{x+1})$,如果 $S(D_x,D_{x+1})\leqslant\delta$ 成立,其中 δ 为由用户设定的正常数,则称 2 个数据序列 D_x 和 D_{x+1}在以 δ 为边界的情况下具有相似性.

相似性函数 $S(D_x,D_{x+1})$满足如下关系[6]:

1) 正定性,$S(D_x,D_{x+1})\leqslant 0$;

2) 对称性,$S(D_x,D_{x+1})=S(D_{x+1},D_x)$;

3) 三角不等关系,$S(D_x,D_{x+1})\leqslant S(D_x,D_{x+2})+S(D_{x+1},D_{x+2})$.

用户设定的 δ 通常在$(0,1]$内取值,当取 $\delta=0$ 时表示 D_x 和 D_{x+1}完全相同,当 $\delta\to 1$ 时表示 D_x 和 D_{x+1}相似性越小. 相似性度量通常与度量相似模式之间的距离联系在一起,最常用的距离度量方法为 Minkowski 距离,定义如下:

$$L_p(D_x,D_{x+1})=\left(\sum_{i=1}^{n}|d_{x,j}-d_{x+1,i}|^p\right)^{\frac{1}{p}} \quad (1)$$

Minkowski 距离是对欧几里得距离的推广,当取 $p=2$ 时即为欧几里得距离,当取 $p=1$ 时为 Manhattan 距离,当取 $p=\infty$ 时为最大距离度量. 基于距离的度量方法计算量小且算法效率相对较高,经过简单的加权处理就可以实现对平行移动、拉伸或压缩、振幅放大或压缩和有限趋势叠加的数据序列进行相似性度量.

2 二层容错聚类

2.1 HLSFTS 概要构造

设概要结构分为 l 层,每层概要结构代表长度为 L 的原始序列则存在 $L_l=L_{l-1}=\cdots=L_1$,每一层都保留了数目远小于次级层次的小波系数,l 越大 HLSFTS 对原始序列的压缩水平越高.

定义 4 当前窗口中的数据序列为 P,定义提升小波容错概要的分辨率为 $r=P/2^i$,i 为小波误差树的层数.

显然提升小波容错概要结构的分辨率决定了如何对原始数据流进行子序列划分.

定义 5 提升小波容错概要结构为一个四元组(t,n,f_t,l_s),其中 t 是概要结构的时间戳,n 是原始数据流中的数据个数,f_t 是概要的相似性度量,l_s 是提升小波概要数据结构.

每层小波误差树的分辨率不同,原始序列数据个数 n 也不同. 相似性度量 f_t 可以定义多种度量方式. l_s 代表提升小波概要数据结构. 提升小波概要数据结构的小波系数选择可以采用不同的误差度量准则,小波系数的正规化方法采用 $c_i^*=c_i/\sqrt{2^{(c_k)_{\text{level}}}}$. 重构误差采用 $(P,P')_{\text{sse}}=\sum_{i=1}^{n}(p_i-p'_i)^2$ 衡量.

HLSFTS 构造流程如图 1 所示.

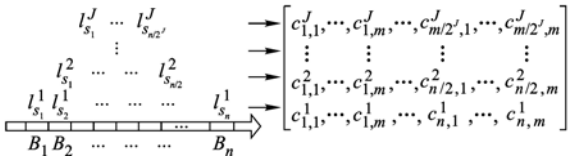


图 1　HLSFTS 构造流程

以 Haar 小波为参照,每个惰性提升小波概要结构的构造算法计算时间复杂度至多不会超过 $O(N(\log N)\log m)$,算法的空间复杂度至多不会超过 $O(N)$,HLSFTS 概要数据结构由至少 1 个最多 $|B|$ 个提升小波概要结构构成,重复进行最少 $|B|$ 次最多 $|B| + |B|/2 + \cdots + |B|/2^J$ 次的提升小波变换,所以 HLSFTS 空间复杂度最大为 $O(|B| \times N)$. 时间复杂度最大为

$$O((N(\log N)\log m) \times |B| + (N(\log N)\log m) \times \frac{|B|}{2} + \cdots + (N(\log N)\log m) \times \frac{|B|}{2^J}) \rightarrow O(\mathrm{poly}(N(\log N)\log m))$$

当 $\delta = 0$ 时 HLSFTS 概要数据结构不具有容错能力,对原始数据流进行无容错压缩,图 2 为无容错 HLSFTS 结构示意图.

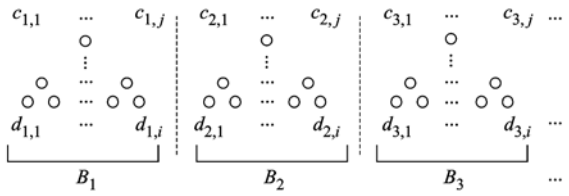


图 2　无容错 HLSFTS 结构图

当概要结构不具有容错特性时,只输出一层提升小波系数,出于简化考虑假设将当前窗口划分为 3 个批次 B_1,B_2 和 B_3. 对 B_1 和 B_2 中的数据进行惰性提升,得到小波系数序列 $\{c_{1,1},\cdots,c_{1,l},c_{2,1},\cdots,c_{2,l}\}$,当窗口滑动,$B_1$ 中的数据过期而 B_3 中的数据插入,删除由 B_1 中的数据产生的小波系数序列,对 B_3 中的数据做惰性提升得到新的小波系数序列,与由 B_2 中数据产生的小波系数序列合并输出 $\{c_{2,1},\cdots,c_{2,l},c_{3,1},\cdots,c_{3,l}\}$,完成 HLSFTS 概要数据结构的重构.

2.2　FTGDStream 聚类算法

二层聚类框架下的宏观聚类阶段采用基于网格密度的聚类算法更为适合. HLSFTS 算法输出小波系数的数据规模很小,在少量小波系数输出的前提下加快了宏观聚类过程中密度网格的划分,算法结构如图 3 所示.

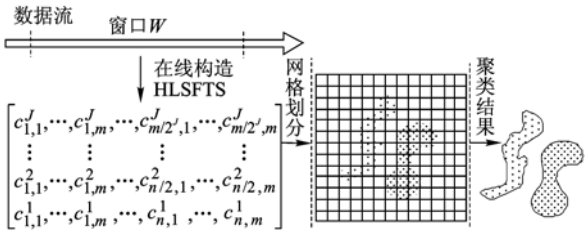


图 3　容错网格密度聚类示意图

空间网格的划分是 FTGDStream 算法的重要过程,基础网格划分定义不同算法的复杂度和聚类结果也不同[7].

定义 6　设 HLSFTS 概要结构随着数据流的流动输出 d 维提升小波系数 C 集,将 C 按维数划分为 $C = C_1 \times C_2 \times \cdots \times C_d$,在此基础上进一步对任意维小波系数 C_i 进行等长划分为 $C_i = C_{i1} \times C_{i2} \times \cdots \times C_{ip}$,则整个提升小波系数集 C 被划分为 $\prod d_i = {}_1P_i$ 个单元格. 对于任意单元格 g 存在 $\prod(C_{1,j_1} \times C_{2,j_2} \times \cdots \times C_{d,j_d}) \in g$,其中 $j_i = 1,2,\cdots,p_i$ 则 $g = (j_1,j_2,\cdots,j_d)$.

定义 7　假设任意时间戳内的任意数据 x 在时间点 t_c 到达,则其密度是与时间 t 相关的函数,定义为 $D(x,t) = \lambda^{t-T(x)} = \lambda^{t-t_c}$,其中 λ 称为衰减因子,且 $\lambda \in (0,1)$.

定义 8　对任意空间格 g 和给定时间 t,令 $E(g,t)$ 为在时间 t 或之前的空间格 g 的数据映射,其密度 $D(g,t)$ 被定义为 $D(g,t) = \sum_{x \in E(g,t)} D(x,t)$.

定理 1　假设单元格 g 在 t_l 时刻接收数据,在 t_n 时刻接收新数据,并且存在 $t_n > t_l$,则新的密度计算公式为 $D(g,t_n) = \lambda^{t_n - t_l} D(g,t_l) + 1$.

证明　令单元格 g 在 t_l 时刻之前接收的数据为 $C = \{c_1,c_2,\cdots,c_m\}$,根据定义 8 可进一步得到 $D(g,t_l) = \sum_{i=1}^{m} D(c_i,t_l)$,且根据定义 7 可得到 $D(x_i,t_n) = \lambda^{t_n - T(x_i)} = \lambda^{t_n - t_l}\lambda^{t_n - T(x_i)} = \lambda^{t_n - t_l} D(x_i,t_l)$ 其中 $i = 1,2,\cdots,m$,因此有

$$D(g,t_n) = \sum_{i=1}^{m} D(x_i,t_n) + 1 = \sum_{i=1}^{m} \lambda^{t_n - t_l} D(x_i,t_l) + 1 = \lambda^{t_n - t_l} \sum_{i=1}^{m} D(x_i,t_l) + 1 = \lambda^{t_n - t_l} D(g,t_l) + 1$$

证毕

定义 9　定义任意空间容错格 g 的特征向量为三元组 (t_l, D, L),其中 t_l 为单元格 g 最后更新数据的时间,D 为数据更新后的格密度,L 为格的

类标签.

定理 2 令 $C(t)$ 为时间戳 t 内到达的所有数据,则存在

1) $\sum_{c\in C(t)} D(c,t) \leqslant \frac{1}{1-\lambda}$,其中 $t = 1,2,\cdots$;

2) $\lim_{t\to\infty} \sum_{c\in C(t)} D(c,t) = \frac{1}{1-\lambda}$.

定义 10 对 2 个单元格 $g_1 = (j_1^1, j_2^1, \cdots, j_d^1)$ 和 $g_2 = (j_1^2, j_2^2, \cdots, j_d^2)$,如果存在 k 且 $1 \leqslant k \leqslant d$ 满足:$j_1^1 = j_i^2, i = 1,2,\cdots,k-1,k+1,\cdots,d$,且 $|j_k^1 - j_k^2| = 1$,则称 g_1 和 g_2 为 k 维上的相邻格,记为 $g_1 \sim g_2$.

定义 11 设存在单元格集合 G 和任意单元格 $g \in G$,若 $g = (j_1, j_2, \cdots, j_d)$ 在任意维上都有相邻单元格,那么 g 为单元格集合 G 的内部单元格,否则 g 为 G 的边界单元格.

定义 12 令 $G = (g_1, g_2, \cdots, g_m)$ 是一个单元格集合,若 G 的每一个内部单元格都是稠密的,每一个边界单元格都是过渡单元格,则称 G 为单元格聚簇.

定义 13 t 时刻对于单元格 g 存在 $D(g,t) \geqslant C_m/N(1-\lambda) = D_m$,其中 C_m 为取值大于 1 的控制阈值,则称 g 为稠密单元格.

若在 t 时刻对于单元格 g 存在 $D(g,t) \geqslant C_l/N(1-\lambda) = D_l$,其中 $0 < C_l < 1$,则称 g 为稀疏单元格.

若在 t 时刻对于单元格 g 存在 $C_l/N(1-\lambda) \leqslant D(g,t) \leqslant C_m/N(1-\lambda)$,则 g 为过渡单元格.

FTGDStream 算法采用微观在线聚类和宏观离线聚类的双层聚类技术实现. 当窗口滑动时批次内的原始数据做提升小波变换得到第 1 层小波系数,相邻批次的小波系数做容错度量,对满足容错程度的相邻批次小波系数做提升小波变换生成第 2 层小波系数,不满足容错条件的小波系数保留在第 1 层. 这一过程反复迭代直至不再存在相邻批次小波系数满足容错条件,生成 d 维容错小波系数且 $d \geqslant 1$. 采用倾斜时间框架以快照的形式对在线微观聚类进行保存[8]. FTGDStream 算法首先对 d 维小波系数进行单元格划分,然后计算单元格的特征向量,并根据定义划分所有单元格的类型,根据稠密单元格和过渡单元格对整个单元格空间进行聚类,得到最终的聚类簇. 以上聚类过程反复执行实现了数据流环境下的容错聚类.

3 实验结果分析与比较

为验证 FTGDStream 算法的效率,在实验中与 HPStream 算法和 D-Stream 算法进行了比较,实验结果如图 4 ~ 图 8 所示. 实验在主频为 3.0 GHz Intel Celeron 处理器及 512 MB 内存的 windows XP 系统平台下采用 Visual C ++ 6.0 与 Matlab 混合编程实现.

实验数据采用 UCI 数据库中的真实数据集 Bag of Word Data Set, US Census Data (1990) Data Set 和 Forest Cover Type Data Set. 其中 Bag of Word Data Set 数据集包含 8 000 000 个词汇实例. US Census Data (1990) Data Set 数据集为 1990 年美国人口普查数据集,包含 2 458 285 个数据实例和 68 个属性. Forest Cover Type Data Set 数据集来源于美国林务局的资源信息系统数据,数据集中包含 581 012 个数据实例和 54 个属性值.

首先对 FTGDStream 算法的聚类准确性进行验证,其中取 $C_m = 3.0$, $C_l = 0.8$, $\lambda = 0.998$,取数据集 54 维数据中的 44 维属性作为实验对象,对多次试验取平均值. 图 4 显示了宏聚类过程中对每一维的数据属性划分长度的不同对聚类正确率的影响,在微观聚类过程中取容错等级为零,FTGDStream 算法的平均聚类准确率保持在 96% 以上.

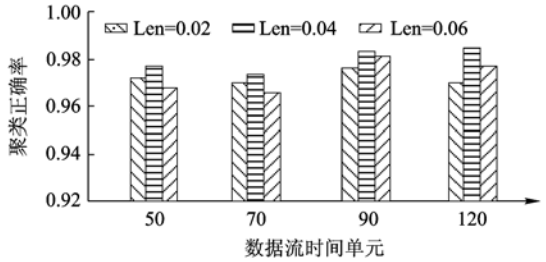


图 4 FTGDStream 算法聚类准确率

图 5 显示了微观聚类取不同的容错等级时 FT-GDStream 算法的最终聚类准确率,划分长度对聚类正确率的影响显著小于容错等级对聚类准确率的影响,容错程度的影响起主导作用.

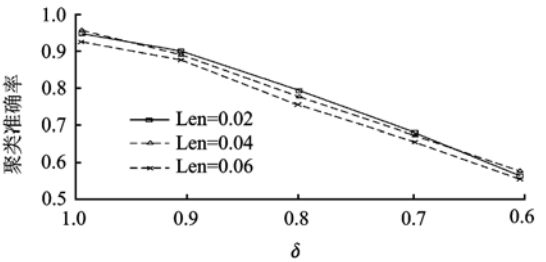


图 5 容错水平对聚类准确率的影响

采用 Bag of Word Data Set, US Census Data (1990) 数据集对 FTGDStream 算法、HPStream 算法和 D-Stream 算法计算效率进行了比较,数据维数取值 10,FTGDStream 算法和 D-Stream 算法的维数划分取值 0.02,取 $C_m = 3.0$, $C_l = 0.8$, $\lambda =$

0.998.

图 6 显示 FTGDStream 算法和 D-Stream 算法的时间效率约为 HPStream 算法 1/3 左右,FTGD-Stream 算法比 D-Stream 算法的时间效率有略微差距,主要是 2 种算法的微观聚类阶段运行效率的差异造成的,宏观聚类阶段由于 HLSFTS 概要结构输出的小波系数远远小于 D-Stream 算法参加网格划分的数据规模,因此宏观聚类阶段 FT-GDStream 算法的计算效率比 D-Stream 算法高.

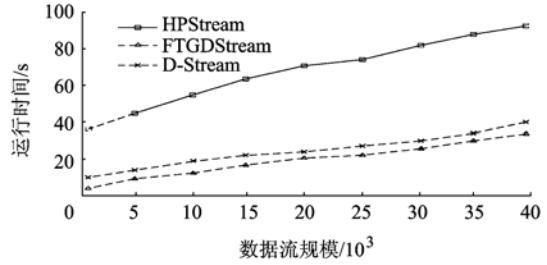


图 6　数据规模对算法效率的影响

根据以上分析可以适当的调整 HLSFTS 概要结构的误差度量准则,使 HLSFTS 概要结构输出更多的小波系数,提高对原始数据流的重构误差的同时也有助于提高聚类的准确性.

图 7 显示了当原始数据流的维数发生变化时对 FTGDStream 算法、HPStream 算法和 D-Stream 算法计算效率的影响,数据维数增加对 HPStream 算法影响最大. 对 Bag of Word Data Set 数据集进行预处理截取其中 1 000 000 个实例,当数据维数增加时 HPStream 算法计算时间消耗急剧增加,FTGDStream 算法与 D-Stream 算法的时间效率始终相差不多,当维数超过 60 维时 FTGDStream 算法的时间效率渐渐超过 D-Stream 算法,这主要是因为 HLSFTS 概要结构对高维数据的压缩能力要好于基于网格的划分方法,使得参加宏观聚类的小波系数数量要少于 D-Stream 算法,因此造成了 FTGDStream 算法对高维数据的处理能力略好一些,总体看来时间效率差距不大.

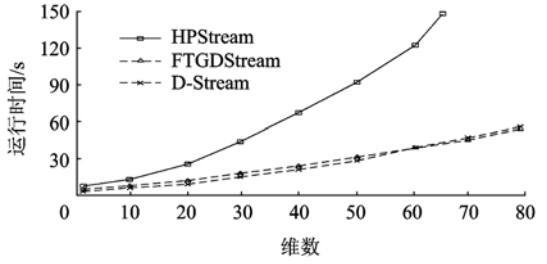


图 7　数据维数对算法效率的影响

图 8 显示了 Bag of Word Data Set, US Census Data(1990) Data Set, Forest Cover Type DataSet 3 种不同数据规模的数据集对 FTGDStream 算法时间效率的影响. 取 3 个数据集中的 500 000 个数据实例,其中 Bag of Word Data Set 数据集取 100 维属性值,US Census Data (1990) Data Set 数据集取 68 维属性值,Forest Cover Type Data Set 数据集取 44 维属性值. 随着不同数据集维数的增加产生相同聚簇数量的时间也随之增加,可以看出数据集的维数对 FTGDStream 算法的时间效率影响比较大,虽然提升小波变换具有相对较好的多维数据处理性能,当数据量较大且维数较高时算法效率有所下降.

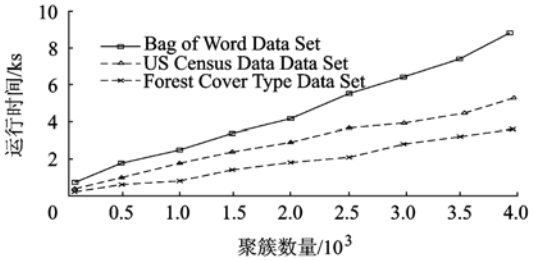


图 8　聚簇数量对算法效率的影响

4　结　论

本文首先介绍了二层数据流挖掘框架的基本思想,以及将其应用于数据流容错聚类的优势. 将二层框架聚类思想与层次小波容错概要数据结构结合在一起,提出一种新的基于二层框架的数据流容错聚类算法,利用容错小波概要数据结构实现在线微观聚类,然后在此基础上对小波容错概要数据结构输出的小波系数进行密度划分,给出了网格的划分方法和基于网格密度聚类的离线宏观聚类算法思想. 在 UCI 数据集上的仿真实验结果表明,FTGDStream 算法利用小波概要对原始数据的高压缩率减小网格密度聚类算法的计算负载,增强算法对高维数据的处理能力,进而提高了基于不同容错等级的数据流聚类算法的效率,是一种有效的数据流容错聚类方法

参考文献 (References)

[1] Callaghan O L, Mishra N, Meyerson A. Streaming data algorithms for high-quality clustering[C]//San Jose. Proceedings of International Conference on Data Engineering. California: IEEE Computer Society, 2002: 685 - 699

[2] Chen Y X, Tu L. Density-based clustering for real-time stream data[C]//San Jose. Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. California: ACM, 2007: 133 - 142

(下转第 674 页)

Hu Shiqiang, Zhang Tianqiao. An on line adaptive weighted fusion algorithm for multi-sensor tracking [J]. Journal of Beijing Institute of Technology, 2002, 22(1): 117 – 120(in Chinese)

[8] 宋建勋,张进,吴钦章. 基于 D-S 证据理论的多特征数据融合算法[J]. 火力与指挥控制,2010,35(7):96 – 98
Song Jianxun, Zhang Jin, Wu Qinzhang. The algorithm of multi-feature data fusion based on D-S theory of evidence[J]. Fire Control & Command Control, 2010, 35(7): 96 – 98(in Chinese)

[9] 翟翌立,戴逸松. 多传感器数据自适应加权融合估计算法的研究[J]. 计量学报,1998,19(1):69 – 75
Zhai Yili, Dai Yisong. Study of adaptive weighted fusion estimated algorithm of multisensor data [J]. Acta Metrologica Sinica, 1998, 19(1): 69 – 75(in Chinese)

[10] 吴成东,刘宗富. 多传感器数据最优融合[J]. 传感器技术,1994,13(3):21 – 24
Wu Chengdong, Liu Zongfu. The optimization of multi-sensor data [J]. J Sensor & Transducer Techno, 1994, 13(3): 21 – 24(in Chinese)

[11] Richardson J M, Kenneth A M. Fusion of multisensor data [J]. The International Journal of Robotics Research, 1988, 7(6): 78 – 96

[12] 杨位钦,顾岚. 时间序列分析与动态数据建模[M]. 北京:北京工业学院出版社,1986:37 – 89
Yang Weiqing, Gu Lan. Time series analyzing and dynamic data modeling[M]. Beijing: Beijing Institute of Industrial Press, 1986: 37 – 89(in Chinese)

[13] 翁思义. 自适应控制系统[M]. 北京:水利电力出版社,1995:43 – 46
Weng Siyi. Adaptive control system[M]. Beijing: China Water-Power Press, 1995: 43 – 46(in Chinese)

[14] 胡广书. 数字信号处理 – 理论、算法与实现[M]. 北京:清华大学出版社,1997:320 – 327
Hu Guangshu. Digital signal processing-theory, algorithm and implement [M]. Beijing: Tsinghua University Press, 1997: 320 – 327(in Chinese)

[15] 赵晋,张建秋,高羽. 迭代异方差估计及其在多传感器数据融合中的应用[J]. 电子学报,2008,36(10):1938 – 1943
Zhao Jin, Zhang Jianqiu, Gao Yu. Iterative heteroscedastic variance estimation with its application for multisensor data fusion [J]. Acta Electronica Sinica, 2008, 36(10): 1938 – 1943(in Chinese)

(编 辑:张 嵘)

(上接第 669 页)

[3] Aggarwal C C, Han Jiawei, Wang Jianyong, et al. A framework for clustering evolving data streams[C]//Proceeding of the 29th International Conference on Very Large Data Bases. Berlin: Morgan Kaufmann, 2003: 81 – 92

[4] Aggarwal C C, Han Jiawei, Wang Jianyong, et al. A framework for projected clustering of high dimensional data streams[C]//Proceeding of the 29th International Conference on Very Large Data Bases. Toronto, Canada: Morgan Kaufmann, 2004: 852 – 863

[5] Chen Mingsheng, Wu Xianliang, Wei Sha, et al. Fast multipole method accelerated by lifting wavelet transform scheme[J]. Applied Computational Electromagnetics Society Journal, 2009, 24(2): 109 – 115

[6] Shahin M, Badawi A, Kamel M. Biometric authentication using fast correlation of near infrared hand vein patterns [J]. International Journal of Biometrical Sciences, 2007, 2(3): 141 – 148

[7] Cao Feng, Ester M, Qian Weining, et al. Density-based clustering over an evolving data stream with noise[C]//Proceedings of the 6th SIAM International Conference on Data Mining. Bethesda, MD: SIAM, 2006: 326 – 337

[8] Han J, Kamber M. Data mining: concepts and techniques[M]. 2nd ed. Morgan Kaufmann: Elsevier Inc, 2006: 467 – 589

(编 辑:文丽芳)