第37卷 第17期 计 算 机 工 程 2011年9月
Vol.37 No.17 Computer Engineering September 2011

·软件技术与数据库· 文章编号：1000—3428(2011)17—0023—03 文献标识码：A 中图分类号：TP338.6

二分K均值聚类算法优化及并行化研究

张军伟[1]，王念滨[1]，黄少滨[1]，蔺世明[2]

(1. 哈尔滨工程大学计算机科学与技术学院，哈尔滨 150001；2. 哈尔滨工业大学计算机科学与技术学院，哈尔滨 150001)

摘 要： 二分K均值聚类算法在二分聚类过程中的初始质心选取速度方面存在不足。为此，提出以极大距离点作为二分聚类初始质心的思想，提升算法的运行速度。研究如何在群集系统中进行快速聚类，根据二分K均值聚类算法的特性，采用数据并行的思想和均匀划分的策略，对算法进行并行化处理。实验结果表明，改进后的算法能获得比较理想的加速比和较高的使用效率。

关键词： 数据挖掘；聚类算法；二分K均值；并行化；群集系统

Research on Bisecting K-Means Clustering Algorithm Optimization and Parallelism

ZHANG Jun-wei[1], WANG Nian-bin[1], HUANG Shao-bin[1], MAN Shi-ming[2]

(1. College of Computer Science and Technology, Harbin Engineering University, Harbin 150001, China;
2. College of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China)

【Abstract】 Considering the insufficiency of clustering speed which exists in the selecting the initial centroid of Bisecting K-Means(BKM) clustering algorithm, the idea of selecting the two patterns with distance maximum as the initial cluster centroid is implemented. An in-depth study and analysis is carried out on how to accelerate clustering in clustering system. According to the characteristics of BKM, the parallelism algorithm based on data parallelism and symmetric data-partition is put forward. Experimental results show that the improvement of algorithm gets ideal speedup performance and efficiency.

【Key words】 data mining; clustering algorithm; Bisecting K-Means(BKM); parallelism; clustering system

DOI: 10.3969/j.issn.1000-3428.2011.17.006

1 概述

聚类方法在数据的组织、分析和挖掘中具有重要作用，广泛应用于模式识别、信息检索、图像处理、机器学习等领域[1]。K-Means 算法是基于原型的聚类技术，具有简单、快速并有效处理大规模数据等诸多优点，是应用最广泛的聚类方法之一。缺点是存在过度依赖初始条件，如聚类数目 K 值的确定、初始聚类中心的选取以及数据的输入次序的变化等都会影响聚类结果，制约了其应用范围。二分K均值聚类算法是K-Means算法的变种算法，通过使用基本K-Means算法能够产生划分聚类算法或层次聚类算法，具有不受初始质心选择影响的优点[2]。该算法在文本挖掘领域有着广泛的应用，通过优化算法的二分初始点的选择过程，把多次循环判断最优点改进为一次选择极大值点，提高了算法的整体运行速度，实验表明了算法的高效性。

聚类实际应用处理对象多为海量数据和高维数据，具有很高的时间和空间复杂性。在处理海量 TB 级文本数据时，利用多台主机组成的群集系统，具有强大的并行计算能力。基于群集环境下的二分K均值聚类算法并行化研究，可以极大地提高工作效率，具有一定的现实应用意义。本文通过分析与研究，对二分K均值算法中所存在的不足进行了优化，并在群集环境中实现了并行化改进。

2 相关研究

文献[3]为研究各种文档聚类技术效果，利用层次聚类算法、K-Means聚类算法和二分K-Means聚类算法进行对比实验。结果表明二分 K-Means 聚类算法聚类效果要高于K-Means聚类算法，具有和层次聚类算法相当的聚类质量，但 $O(n)$的时间复杂度优于层次聚类的 $O(n^2)$时间复杂度。文献[4]提出了核二分K-Means聚类算法用以减少SVM训练集样本，以改善 SVM 的可扩展性。该算法通过在核特征空间快速产生相似大小均衡的簇集，能够有效地减少非线性支持向量机训练样本，随着采样时间缩短，保持测试精度的同时，加快了 SVM 训练算法速度。文献[5]在基于整体和局部相似性的序列聚类算法中，利用带剪枝策略的二分K-Means聚类算法对基于整体相似性的序列聚类。通过启发式方法获得二分K-Means聚类算法的质心，可使整个算法关于序列数 t 获得多项式时间复杂度。当序列数据集的自然分类情况较明显时，分别选最长序列和最短序列来作为初始化质心，采取序列数据的概貌向量指导编辑距离的计算次序的剪枝策略，可以有效地加速序列聚类算法的运行。文献[6]利用K-Means算法对索引对象进行聚类分析，构造新的聚类中心使其能处理具有多种形体的索引对象，在 QR-树中引入超结点存储聚类结果，提出 QCR-树空间索引结构来提高查询效率。

3 算法优化及并行化

3.1 二分K均值聚类算法

二分K均值聚类算法是基本K均值算法的变种，它基于以下思想：为得到 K 个簇，首先将所有数据集作为一个簇 V，放到簇集 S 中。然后，循环从簇集 S 中取出一个簇，用基本

基金项目： 国家自然科学基金资助项目(60973028)；国家科技支撑计划基金资助项目(2009BAH42B02)

作者简介： 张军伟(1971－)，男，硕士研究生，主研方向：并行数据处理；王念滨、黄少滨，教授、博士；蔺世明，硕士

收稿日期： 2011-03-18 **E-mail：** zjw_1997@yahoo.com.cn

K 均值聚类算法，通过 I 次对选定的簇做二分聚类，选择具有最小总 *sse* 的 2 个簇，把这 2 个簇放回簇集 S 中。如此往复，直到产生 K 个簇。其中，I 表示二分试验的次数；误差的平方和(sse)称为散布。对于多次运行 K 均值产生的簇集，选择具有误差的平方和最小的那一个，这意味着聚类的质心是簇的最好代表。

sse 定义如下：

$$sse=\sum_{i=1}^{K}\sum_{x\in c_i} dist(c_i,x)^2$$

在二分时，使用多次 K 均值聚类算法以找到 *sse* 最小的聚类结果，但这是局部的，不是全局的。所以最后使用结果质心作为 K 均值聚类算法的初始质心，进行全局优化。相对于 K 均值聚类算法，该算法最后优化时采用的质心是多次二分产生的，因此，避免了随机产生质心而得到局部最优化结果。

3.2 算法优化的思想

通过分析二分 K 均值聚类算法的描述可知，其考虑到了随机选取初始质心的 K 均值聚类算法的不确定性，容易使聚类结果走向局部最优化，采用多次使用 K 均值聚类算法进行聚类，选择具有最小 *sse* 的簇集作为二分结果。但在实际的信息检索、图像处理等领域，所需处理的对象多为海量、高维的数据，多次重复进行 K 均值聚类大大增加了时间复杂度。通过以上分析，需要找到一种方法，只进行一次二分聚类就可以找到具有最小 *sse* 的簇集。

定义 1 极大距离，从簇集中任选一点，寻找距该点最远点，把最远点作为初始点，接着寻找该点最远点，如此下去，直到找到簇中的 2 个极大距离点，这 2 个点之间的距离就是簇的极大距离。

由多个尺寸相似簇组成的簇集，如果把该簇集分为两部分，那么沿着簇分布最长的方向划分为两部分较为合理。以最远 2 个样本点为初始点，各个样本以中心线划分，能获得较高的收敛速度，同时划分后样本点距中心点较近，能产生较高的聚类质量。

通过以上分析，以下给出优化的二分 K 均值聚类算法：

输入 簇的数目 K，数据集 $X=\{x_1,x_2,\cdots,x_n\}$

输出 K 个簇 $S=\{S_1,S_2,\cdots,S_k\}$

初始化：簇 V=X，S={ }

Begin

Repeat

从簇集 S 中取出一个具有最大 sse 的簇 V

用极大距离法从簇 V 中选取 2 个中心点 c_1，c_2

使用基本 K 均值，二分选定的簇 V 为 V_1，V_2

$$\begin{cases} x_i\in V_1 & \text{if} \quad \|x_i-c_1\|\leqslant\|x_i-c_2\| \\ x_i\in V_2 & \text{if} \quad \|x_i-c_2\|\leqslant\|x_i-c_1\| \end{cases}$$

将这 2 个簇添加到簇集 S 中

Until 簇集 S 中包含 K 个簇

以簇集 S 中 K 个簇的中心为初始质心对所有点进行 K 均值聚类，得到最终结果

End

3.3 算法的并行化

3.3.1 数据并行思想

由 PC 组成的群集并行系统，通常采用网络联接，由于其节点间通信一般采用的是 TCP/IP 等协议，因此通信过程产生的时间延迟相对于节点的计算能力来说是相当大的。如果在算法设计中，要求节点间有过多的通信，那么往往会阻碍进一步提高并行系统的运行性能。因此，算法在设计时不能要求过多的通信，应采用数据并行的思想。

数据并行的思想就是通过将整个数据集划分成若干数据子集，在每一个节点执行同一聚类算法，通过节点间的信息交换，完成最终聚类。各节点划定数据子集后同时处理数据，彼此间进行少量通信或不通信，直到最后合并聚类结果时才进行通信。充分地发挥了群集的优势，得到高效的聚类结果。

3.3.2 并行聚类算法

并行聚类技术数据的划分通常有 2 种方式。第 1 种是对整个数据集按节点数进行均匀划分，各节点数据量相同，理论上会大致同时完成各自的计算，几乎同时向主节点发送信息，这种方式是同步通信。第 2 种是对整个数据集按增量递增划分不同大小，各节点在聚类计算过程中就会形成一定时间差。利用时间差，主节点与其他节点分别完成计算和通信，不致发生通信冲突，这就是异步通信方式。

本文主要采用均匀分配数据方法，这是根据二分 K 均值聚类算法的特点决定的。二分 K 均值聚类算法要对全局数据计算才能得出准确结果，而把数据按节点数均匀划分到各个节点上，这样子数据集才能够真实反映数据的全局特性，得出比较准确结果。通过实验发现，在各个节点上，即使数据量是相等的，但是由于数据点本身的差异性，实际完成的计算时间还是不同的，在最后的通信中并不会造成太大的冲突。为提高并行的效率，减少通信交换次数，首先在数据子集聚类，得到靠近中心的点，用这些点做质心进行全局数据聚类，这样就会大大减少全局计算的迭代及通信次数。

图 1 显示了该并行算法的处理流程。

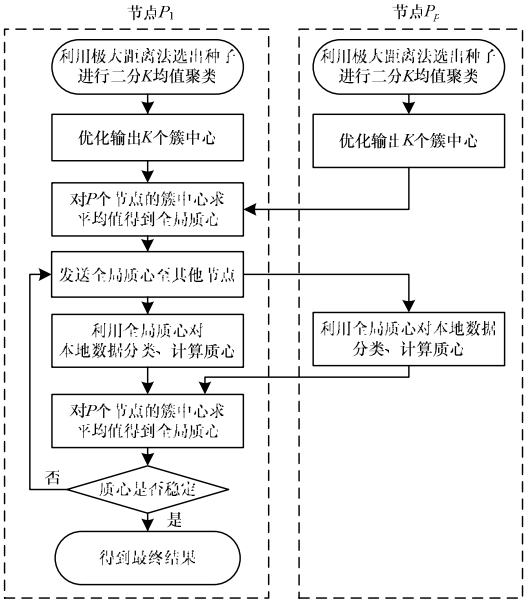

图 1 并行算法的处理流程

具体并行算法步骤如下：

(1)把数据集按节点数进行划分，把数据子集分配到 P 个节点。

(2)各节点同时对各自的数据子集进行初始化，输入欲聚类簇的数量 K，按照二分 K 均值聚类算法进行聚类，直到得出目标簇数和 K 个簇中心。

(3)各节点分别用各自得出的 K 个聚类中心作初始质心，对二分 K 均值聚类结果进行优化。

(4)各节点把优化后的聚类中心传送到主节点，主节点把各节点传来的聚类中心点进行合并取平均值，得到新的质心。

(5)主节点把新的质心传回到各个节点，各节点按新的质

心把子集数据进行分类，然后计算质心。

(6)各节点把计算完的均值传回主节点，主节点把这些均值合并，得到新的全局质心。

(7)主节点判断全局质心是否稳定，如果稳定执行(8)；如果不稳定，则执行(5)。

(8)得到最终聚类结果。

4　实验结果与分析

4.1　算法优化实验

数据集:随机产生 100 000 个二维数据,大小为 1 256 KB，形态为 4 个呈圆形、相互分离的簇，左右簇间距离大于上下簇间距离。实验平台配置：CPU 为 Intel 双核 2.7 GB，2 GB 内存， Visual C++编程实现。实验方案：对数据集进行 100 次 K 均值聚类，取最后平均值作为结果，设 K=2。结果如表 1 所示。

表 1　二分聚类结果

结果	平均时间/s	聚类结果中心点	耗用时间/s
随机选初始质心	4.08	2 种，需计算选取具有最小 *sse* 的结果	2~10
选最远距离质心	4.39	1 种，具有最小 *sse*	3~6
选极大距离质心	4.00	1 种，具有最小 *sse*	4

定义 2 最远距离点，从簇中任选一点 x_1，寻找距该点最远点 x_x，$\max(dist(x_1,x_x))$，x_x 为 x_1 点的最远距离点。

实验分析：在把一个簇分为 2 个簇的过程中，采用随机选初始质心的 K 均值聚类算法，出现了 2 种结果，并且花费时间从 2 s~10 s 不等，必须经多次运行，计算结果的 *sse* 才能获得最优方案。采用选极大距离 2 个点做质心的 K 均值聚类算法具有聚类结果稳定，并且经实验验证具有最小的 *sse*，而平均时间花费与随机选初始质心的算法相比略小。这样在二分 K 均值聚类算法中的一次二分中，多次运行随机选初始质心的 K 均值算法要比一次运行采用极大距离的 2 个点做质心 K 均值聚类算法的时间代价要大得多。

以同样的实验平台对 10 万、20 万、30 万的二维数据进行对比实验，设原二分 K 均值聚类算法中的 I 为 3 次。图 2 显示了优化后聚类算法和原算法的对比结果。

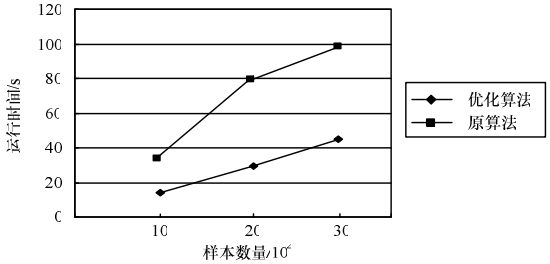

图 2　优化前后算法的运行时间对比

从实验结果的分析可以得出，优化后的算法要节省很多时间，并且随着数据量的增加，时间呈线性增长。而原算法时间代价较高，如果增加 I 的次数，那么原算法的时间代价会更大。

4.2　并行化实验

实验平台：CPU 为 Intel 双核 2.7 GB，2 GB 内存的多台 PC 机组成的群集系统，Visual C++编程。数据样本：分布形态为 6 个近似圆形,相互分离的簇,有 1 000 000 个二维样本，大小 12 564 KB。从表 2 中可见，在总数据量不变的情况下，随着群集中节点数的增加，加速比逐渐变大，运算时间逐步减少。当节点数为 2 时，加速比大于 P，属于超加速比。而随着节点数的增加，加速比逐渐小于 P。出现这种情况是由于并行算法的时间复杂度和节点数目有直接关系。随着节点的增多，节点间的通信次数在增加，总的时间延迟在增加，故加速比逐步小于 P。

表 2　并行聚类结果

节点数	I/O 时间/s	总运行时间/s	加速比
1	27	389	1.000
2	13	185	2.100
4	6	101	3.858
8	3	52	7.680

图 3 显示，当节点数为 2 时，时间线下降比较陡，随着节点数的增多，时间线下降变得缓慢，这说明当节点数增加到一定时，时间效益和计算成本代价将达到一个平衡点，在实际应用时需综合考虑。总体上讲，所作的优化显著地提高了运算性能，同时其并行化效果达到了预期目标。

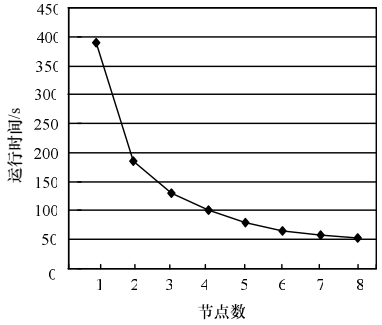

图 3　运行时间和节点数关系

另外，从表 2 中还可以看出，I/O 时间占总运行时间 7% 左右，在对海量数据进行处理时，将成为影响其完成时间长短的瓶颈问题。采用足够多的高速 I/O 设备、快速的文件系统、优化 API 的研究将是解决这一问题的必要方法。

5　结束语

串行二分 K 均值聚类算法在聚类过程中，需要人为设置实验次数，本文对其初始质心选择算法进行了改进，提高其聚类效率，利用群集计算机系统，在各节点对数据子集聚类完成后，通过节点间通信优化聚类结果，从理论上研究了串行算法并行化的可行性，并通过实验进行了验证。由于离群点对划分聚类结果影响很大，如何减少离群点对二分 K 均值聚类算法的影响需要进一步研究。

参考文献

[1] Han Jiawei, Kamber M. 数据挖掘概念与技术[M]. 范　明, 孟小峰, 等, 译. 北京: 机械工业出版社, 2006: 223-254.

[2] Savaresi S M, Boley D. On the Performance of Bisecting K-Means and PDDP[C]//Proc. of the 1st SIAM International Conference on Data Mining. Chicago, USA: [s. n.], 2001: 1-14.

[3] Steinbach M, Karypis G, Kumar V. A Comparison of Document Clustering Techniques[C]//Proc. of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Boston, USA: [s. n.], 2000: 525-526.

[4] Liu Xiaozhang, Feng Guocan. Kernel Bisecting K-Means Clustering for SVM Training Sample Reduction[C]//Proc. of the 19th International Conference on Pattern Recognition. Tampa, USA: [s. n.], 2008: 1-4.

[5] 戴东波, 汤春蕾, 熊　赟. 基于整体和局部相似性的序列聚类算法[J]. 软件学报, 2010, 21(4): 702-717.

[6] 高　云, 侯贵宾, 张　辉, 等. 基于 QCR-树的空间索引方法[J]. 计算机工程, 2010, 36(12): 80-82.

编辑　任吉慧