采用分布式 DBSCAN 算法的用电行为分析

赵永彬[1],陈 硕[1],刘 明[1],王佳楠[2,3],贲 驰[4]

[1](国网辽宁省电力有限公司 信息通信调度监控中心,沈阳 110004)
[2](中国科学院 沈阳计算技术研究所,沈阳 110168)
[3](中国科学院大学,北京 100049)
[4](国家电网东北电力调控分中心,沈阳 110180)
E-mail: wangjianan15@mails.ucas.ac.cn

摘 要: 作为描述电网运营状态的重要依据,用户实时用电量数据在各时间段内的均值、方差及离散傅里叶变换(DFT)分量等统计变换结果是用电行为特征提取的重要建模参数. 基于密度聚类的 DBSCAN 算法能够对空间内样本点进行更加准确可靠的类别划分. 层次聚类的子域划分、域内聚类和聚类合并的过程为 DBSCAN 算法的分布式实现提供重要参考. 根据 DBSCAN 算法中的密度参数,保留各子簇的边界特征样本,使子簇合并过程的计算效率进一步提高. 以 Spark 为代表的分布式内存计算系统将数据处理的中间结果存入内存,降低读写开销,为大规模数据的迭代分析提供快捷高效的处理环境. 实验结果证明,在内存计算系统中实现的分布式 DBSCAN 聚类算法能够准确高效的实现大规模用户用电行为分析.

关 键 词: 行为提取; 层次聚类; 簇边界特征; 分布式内存计算
中图分类号: TP391 **文献标识码:** A **文 章 编 号:** 1000-1220(2018)05-1108-05

Analysis of Power Consumption Behavior Using Distributed DBSCAN Algorithm

ZHAO Yong-bin[1], CHEN Shuo[1], LIU Ming[1], WANG Jia-nan[2,3], BEN Chi[4]

[1](Information & Telecommunication Branch, State Grid Liaoning Electric Power Company, Shenyang 110004, China)
[2](Shenyang Institute of Computing Technology, Chinese Academy of Science, Shenyang 110168, China)
[3](University of Chinese Academy of Science, Beijing 100049, China)
[4](State Grid Electric Power Control Northeast Branch Center, Shenyang 110180, China)

Abstract: As an important basis for the description of grid operation state, the statistical transform results of user real-time consumption data in each period of time, including mean value, variance and the discrete Fourier transform (DFT) component, are important parameters of electricity behavior feature extraction. Based on the density clustering, the DBSCAN algorithm can be used to label the sample points more accurately and reliably. The process of hierarchical cluster, such as sub domain clustering and cluster merging, provides important reference for the distributed implementation of DBSCAN algorithm. According to the density parameter of the DBSCAN algorithm, the boundary feature samples of each sub cluster are reserved to improve the computation efficiency of cluster merging. Distributed memory computing system represented by Spark saves the intermediate results of data processing into memory to reduce the cost of reading and writing, which provides the fast and efficient processing environment for the iterative analysis of large-scale data. The experimental results show that the distributed DBSCAN clustering algorithm operating in the memory computing system can accurately and efficiently implement the analysis of large scale user's power consumption.

Key words: behavior extraction; hierarchical clustering; cluster boundary feature; distributed memory computing

1 引 言

作为电力系统的主要参与者,用户个体的实时用电行为是评估电力系统当前所处的运行状态的重要参考依据. 根据用户的实时用电量数据可以实现对用户用电行为特征的提取建模,基于用电行为特征的分析结果可以满足包括异常用电行为的识别、用户类型的评级、电网整体运行状态的评估等多个方面的分析需求[1]. 从而为电力资源调度、售电定价等电力系统运营策略的制定调整提供准确科学的数据支持,进一步提高电力企业的运营水平.

对于省级电力公司而言,其业务所涉及的用户数量已达到千万户的数量级,每小时将产生上百 GB 的实时用电量数据. 以 Strom、Spark 为代表的分布式计算系统凭借着其高效性、高可靠性、高可拓展性的优势[2],在满足系统计算资源需求的同时,提供了高效且易于开发的分布式数据处理框架,为大规模数据的集中处理和快速分析提供了平台保证.

收稿日期: 2017-04-28 收修改稿日期: 2017-06-20 基金项目: 辽宁电力公司科技项目(SGLNXT00DKJS1600242)资助. 作者简介: 赵永彬,男,1975 年生,硕士,高级工程师,研究方向为智能电网、Web 工程、信息集成; 陈 硕,男,1983 年生,博士,高级工程师,研究方向为智能电网、Web 工程、信息集成; 刘 明,男,1979 年生,硕士,高级会计师,研究方向为电力信息; 王佳楠,男,1993 年生,硕士研究生,研究方向为智能电网、工业大数据; 贲 驰,女,1965 年生,高级工程师,研究方向为电量采集与计费统计.

对于用电行为等无法进行明确类别划分的样本数据，适用于聚类等无监督学习的分析方式，根据样本点在整个样本空间的分布情况，实现对各样本点所属的类别的划分. 相较于如 k-means 等基于划分的聚类算法，DBSCAN 等基于密度的聚类算法能够克服局部不收敛、聚类结果易受初始设定影响等局限性[3]. 将聚类算法的分析思想与分布式计算框架的处理流程相结合，进一步提高算法的处理效率，为大规模数据的处理分析提供了重要解决途径.

本文采用基于密度聚类的 DBSCAN 算法实现对用户用电行为类型的标注，根据标注的离群点识别异常用电行为. 将 DBSCAN 密度可达的搜索合并思想与区域划分、聚类合并等层次聚类的策略相结合[4]，使用 Spark 分布式内存计算框架所提供的处理架构实现聚类算法的并行化，提高算法的处理规模. 将各子簇中的边界样本作为本簇的特征点，降低聚类合并过程中的计算开销，进一步提高分布式 DBSCAN 算法的效率. 最后，基于实际的用电量数据验证分布式 DBSCAN 算法在数据处理规模、算法执行效率及准确性上的优势.

2　行为特征提取与分析平台选择

为保证电网运营状态的准确实时监控，电力公司以秒级的时间粒度读取每一个用户的实时用电量数据，而单纯的用电量数值数据无法准确直观的反应用户真实的用电行为. 考虑到省级电力公司千万级的用户规模，在对用户用电行为特征进行提取分析时，应选用能够满足大数据量和高实时性处理性能需求的数据处理分析平台.

2.1　用电行为特征的提取构建

作为一种阶段性的状态描述，用电行为特征可以由用户一段时间内的实时用电量数据进行提取构建. 因此，本文选择以 5 min 为一个时间窗，根据当前时间窗内的实时用电量数据完成用户本时间窗内用电行为特征的抽取建模，见表 1.

表 1　用电行为特征参数

Table 1　Characteristic parameters of power consumption

特征参数	参数含义
Avg_Consumption	用电量平均值
Max_Consumption	用电量最大值
Min_Consumption	用电量最小值
Variance	用电量方差
Freq_Feat_i	用电量频域特征分量
Sample_Value_j	瞬时用电量采样值
Sample_Change_k	瞬时变化率采样值

为实现对用户用电行为的全面描述，本文采用各时间窗内实时用电量的平均值、方差、最大值、最小值 4 项统计指标，各时间窗内以 1 分钟为采样间隔的瞬时用电量及变化率各 5 条样本数据以及描述用电量数据变化波动情况的 10 个频域特征，构造出包含 24 维特征的用户用电行为特征向量实现对用户单个时间窗内用电行为的描述. 其中，描述用电量变化情况的频域特征由时间窗内的实时用电量经过离散傅里叶变换(DFT)[5]后的结果合并提取后获得.

对于时间窗内 N 个($0 \leqslant n \leqslant N-1$)实时用电量数据构成的有限长序列 $x(n)$，它的离散傅里叶变换 $x(k)$ 仍为一个长度为 $N(0 \leqslant k \leqslant N-1)$ 的频域有限长序列. 则有:

$$x(k) = \mathrm{DFT}[x(n)] = \sum_{n=0}^{N-1} x(n) e^{-j\frac{2k\pi}{N}n} \quad (1)$$

将经过离散傅里叶变换后的序列中每个频域分量 w_i 对应的幅值记作 a_i. 将各频域分量进行排序后等距划分为 10 个频域区间，则描述实时用电量变化情况的 10 个频域特征值由各频域区间内所有频域分量的幅值进行求和后得到.

为避免噪声数据和缺失值的影响，对每个用户各时间窗内的实时用电量数据进行等距分箱，在分箱内对数据进行抽样平滑等预处理操作. 最终，对于每个时间窗内保留 50 个数据点，用以进行特征的提取和构建.

2.2　基于流计算的特征提取平台

作为典型的 Master-Worker 架构的分布式流计算系统，Apache Storm 大吞吐量、高可扩展性、高容错性、高可靠性和易操作性的性能优势[6]，使其能够高效的完成对大规模用户高时间密度的实时用电量数据进行的整合、清洗及特征构建等一系列操作.

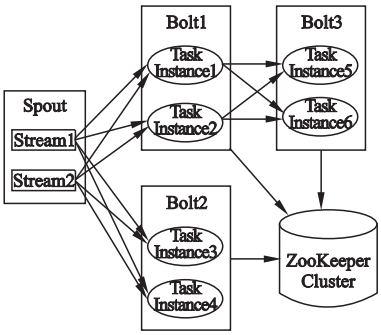

图 1　Storm 内部处理逻辑图

Fig. 1　Storm internal processing logic diagram

在如图 1 所示的 Storm 流作业处理逻辑中，Storm 将数据处理操作划分为用以进行数据接入的 Spout 和用以进行数据处理的 Bolt 两大类. 结合 Kafka 分布式消息队列[7]，将持续产生的实时用电量数据根据其所对应的用户来源划分入对应的 Topic 中，实现数据的并行化接入.

为特征提取过程的数据划分、分箱平滑、抽样、均值方差统计、频域特征构建、特征归一化等一系列操作定义对应的 Bolt 逻辑. 同时，设置 Bolt 之间的数据传递方向，在提高数据并行化处理效率的同时，实现处理逻辑的高效复用.

2.3　基于内存计算的行为分析平台

Spark 是由加州大学伯克利分校 AMP 实验室开发的分布式内存计算系统，凭借与 Hadoop 的 HDFS 和 YARN 具有良好的兼容特性，使其拥有能够高效可靠的处理大规模数据的性能优势. 基于弹性分布式数据集(Resilient Distributed Datasets，RDD)的抽象概念实现大规模数据在集群内存中的统一管理和处理分析[8]，解决了传统 MapReduce 分布式数据处理框架将中间结果数据保留入磁盘，不适合处理机器学习算法中大规模迭代运算的性能短板.

在 Spark 中，RDD 被定义为只读的、分区记录集合，可以通过程序中的容器对象、文件系统中的序列化文件以及其他 RDD 等多种来源进行构造. 通过定义合理的 RDD 分区策略，提高对 RDD 转换(Transformation)和动作(Action)两类基本

操作的处理效率.

3 DBSCAN 聚类算法的分布式实现

基于密度的聚类算法具有能够挖掘出任意形状的聚类簇、避免噪声数据对聚类结果和收敛效率产生影响的性能优势. 为适应大规模数据集的分析处理需求,需要采用分布式计算的策略提升算法对大规模数据的处理能力. 为进一步提高算法的性能,通过对聚类簇生成过程中样本点的搜索合并策略进行优化,降低计算过程中的时间和存储开销.

3.1 DBSCAN 聚类算法的核心思想

DBSCAN 聚类算法通过评估各样本点之间的密度可达性,将所有密度相连的样本点构成一个独立的聚类簇[9],并将每个聚类簇的大小与算法设置的参数阈值 MinPts 进行比较,将样本个数小于 MinPts 的簇标记为噪声簇.

对于每一个样本点,其 E-邻域内即距离小于邻域半径 Eps 的所有样本点都是密度可达的. 在样本空间中常用的距离衡量标准为欧氏距离,但也可以根据样本分布特征选用其他的距离衡量标准. 同时,密度可达性具备可传递性,即对于图 2 中的样本点 p,与其 E-邻域内存在的样本点 m 是密度可达的. 同理,样本点 q 与 m 也为密度可达的,则样本点 p 与样本点 q 之间也为密度可达的.

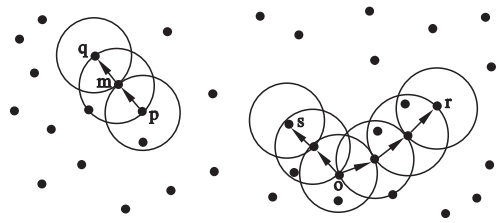

图 2 密度可达性原理图

Fig. 2 Density-reachable diagram

由于密度可达为单向的传递概念,图 2 中样本点 o 到样本点 s 和 r 均为密度可达的,则认为样本点 s 与 r 为密度相连的. 因此,DBSCAN 聚类算法在选定未进行类别标注的样本点后,根据算法选用的距离衡量标准,将 E-邻域内所有密度可达的样本点加入聚类簇后,根据密度可达性的传递性特点将新加入点的 E-邻域内所有的样本加入聚类簇,直至找不到新的样本点加入,最终令所有的样本点与所属簇中的所有样本点均为密度相连的.

通过对每一个簇中新加入样本点的 E-邻域进行搜索拓展,DBSCAN 算法能够挖掘出任意形状的聚类簇. 由于 DBSCAN 算法所生成聚类簇的样本点个数至少为 1,因此其能确保每一个样本点都拥有自己对应的类别标注,通过设置合理的 MinPts 参数,将密度不符合要求的聚类簇标记为噪声,使算法能够更好的识别噪声数据.

3.2 DBSCAN 算法的分布式计算过程

当单台计算机的存储资源远远无法满足大规模数据集的处理需求时,传统的 DBSCAN 算法无法保证对所有新加入样本点的 E-邻域进行全面的搜索. 层次聚类基于分治策略的算法思想将样本空间细分为多个子空间,在子空间内完成聚类分析后,再对各个子空间中获得的聚类簇进行合并获得最终的聚类结果[10],为实现大规模数据的聚类分析提供了有效的解决思路.

对于拥有 N 维特征的无标注样本集,将其每一维特征进行 2^k 等分后可以获得 2^{kN} 个样本子空间. 在每一个样本子空间中,使所有密度相连的样本构成一个聚类簇,并将各个样本子空间中的聚类子簇依次进行合并得到最终的聚类结果. 因此,DBSCAN 算法的分布式实现过程如下所示:

1) 设置合适的算法参数: Eps 和 MinPts,并定义样本空间中的距离衡量标准 Distance(p,q)

2) 将每个样本点 i 的初始簇类别标注记为 c_i,并对 N 维样本特征的值域分别进行 2^k 等分,获得最初的样本子空间集合 $S=\{s_1,s_2,\cdots,s_t\}$ 并保证初始样本子空间各维度的宽度大于所设置的 Eps.

3) 将各聚类簇根据其所包含样本点所属的样本子空间进行分组,分别在各样本子空间内进行聚类簇的合并.

4) 对于每一个样本子空间 s_t,若 s_t 中存在两个分属于聚类簇 c_a 和 c_b 的样本点是密度相连的,则将两个聚类簇的类别标注统一为 c_a.

5) 记录各样本子空间中所有聚类簇的合并操作,对于经过合并操作的聚类簇,将其所有样本点的类别标签由原先的 c_i 更新为 c'_i.

6) 若样本子空间集合 $S=\{s_1,s_2,\cdots,s_t\}$ 中的元素个数不为 1,将 S 中的样本子空间进行合并得到新的样本子空间集合 $S'=\{s'_1,s'_2,\cdots,s'_r\}$,返回步骤 3)

7) 将样本点个数小于 MinPts 的聚类簇的类别标签标注为噪声数据类别,获得最终的聚类结果.

3.3 基于边界特征提高聚类合并效率

在分布式 DBSCAN 聚类算法中,两个不同类别标注的聚类簇合并依据为存在两个类别不同的样本点是密度可达的. 在已有的分布式 DBSCAN 聚类算法的实现方式中,通常采用增量合并的方式[11],即在合并各样本子空间中的聚类簇时,令某单个样本子空间的聚类簇作为合并基准,再加入其他样本子空间中的聚类簇,实现聚类簇的合并. 在聚类簇合并检测时,需要计算待加入聚类簇中所有的样本点与基准聚类簇样本点的密度连通性,会产生较大的计算开销.

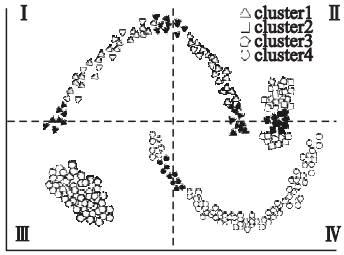

图 3 分布式 DBSCAN 聚类簇合并原理图

Fig. 3 Cluster merging diagram of distributed DBSCAN

在图 3 所示的聚类簇合并过程中,检测来自于不同样本子空间的两个聚类簇能否进行合并时,对于每一个簇 C 只需保留到样本子空间边界 $E=\{e_1,e_2,\cdots,e_s\}$ 的距离小于 Eps 的样本点,即 $P=\{p_1 \mid \text{distance}(p_i,e_s) < \text{Eps}, p_i \in C, e_s \in E\}$ 作为本聚类簇的特征点集合. 在各聚类簇中所有到达样本子空间边界的距离大于 Eps 的样本点,其 E-邻域中不可能包含属于

其他样本子空间的样本点. 同样的,对于由 P_1,P_2,…,P_t 共计 t 个聚类簇合并而得到的新聚类簇 P′,其特征点集合中的样本点仅来源于原先 t 个聚类簇的特征点集合.

尽管层次聚类分治合并的算法思想与分布式计算框架相结合能够实现对大规模数据的分析处理,但在聚类簇合并过程中会产生巨大的存储开销,即由单个节点完成对来源于多个节点数据的合并汇总. 由于两个聚类簇的合并依据为其特征点集合之间是否存在密度相连的样本点,因此在聚类合并时,只需将各聚类簇的编号及其特征样本集合作为输入,获得聚类簇原编号与新编号间的对应关系及新聚类簇的特征点集合. 在全局中只需维护样本点 ID 与其所属聚类簇编号的对应关系,在每次聚类合并过程后,更新各样本所属的类别,仅保留各聚类簇特征点集合内样本点的特征值.

相较于原有分布式 DBSCAN 算法所采用的增量合并策略,在聚类合并时仅比对特征点集合中样本间的密度连通性,减少了不必要的计算操作. 同时,也降低了层次聚类策略在聚类合并阶段的存储需求,避免层次聚类的性能瓶颈. 在簇内样本点较为分散或对更大范围样本子空间进行合并等非特征点比重较大的情况下,对聚类合并过程的优化更为明显.

3.4　基于 Spark 实现高效的迭代运算

通过将数据处理过程抽象为对 RDD 的操作,Spark 在实现对数据分布式处理的同时,将数据处理过程中的中间结果存放在内存中,降低了对数据进行分布式迭代分析时的数据读写开销,使计算分析过程能够高效进行.

在如图 4 所示的算法实现过程中,RDD1 和 RDD4 由存放在 HDFS 上的原始数据集文件转换而成. 其中,RDD4 中的每个元素以 < Point_ID,Cluster_ID > 的格式存放各样本点所属的类别编号. RDD1 中则以 < Cluster_ID,Set < Point_n > > 的形式存放各个聚类簇所对应的特征点. 在进行算法的初始化时,每个样本点被分配单独的聚类簇编号,每个聚类簇的特征点为与之对应的样本点.

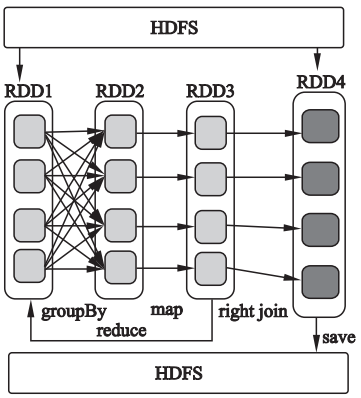

图 4　分布式 DBSCAN 算法在 Spark 上的实现过程

Fig. 4　Process of distributed DBSCAN on Spark

在设置算法参数 Eps 和 MinPts 的同时,确定样本空间的距离衡量标准 Distance(p,q) 和特征划分粒度 2^k. 对 RDD1 进行 groupBy 操作,将各聚类簇根据其特征点的位置分入对应的样本子空间后生成 RDD2. 其中,RDD2 中的各个元素以 < Parition_ID,Set < Cluster_ID,Set < Point >>> 的形式存放每个样本子空间中各聚类簇的编号及其特征点.

对 RDD2 中的各元素进行 map 操作,即分别对每个样本子空间内的聚类簇进行合并,获得聚类合并后的结果. 并以 < Parition_ID,Set < Old_ID,New_ID,Set < Point >>> 的元素格式生成 RDD3,存储各样本子空间内的聚类簇经过聚类合并后的类别编号及原始编号.

将 RDD3 与 RDD4 进行以原始类别编号进行 right join 操作,更新类别编号发生改动的样本点,并以 RDD4 的原有元素形式保留操作结果作为新的 RDD4. 对 RDD3 按照新的类别编号进行 reduce 操作,并按照当前样本子空间的边界更新簇的特征点,仅保留拥有特征点的聚类子簇,以 RDD1 的元素形式保留待合并的聚类子簇作为新的 RDD1.

4　实验结果分析与算法性能评估

本文采用的实验环境为由 5 台 PC 机构成的小型分布式集群. 其中,每台 PC 机均配备 Intel Core I5 6500 处理器和 8G 内存的存储计算资源. 集群中部署了包括 Kafka、Storm、Hadoop、Spark 等完成分布式存储计算任务的系统组件.

为检验文中提出的分布式 DBSCAN 聚类算法在较大规模数据集上的分析效果和处理能力. 本文选用某省电力公司 10 万户用户在 2016 年 6 月 10 日的用电高峰时段内 1 小时的实时用电量数据,并将其按时间顺序依次写入 Kafka 中相应的话题. 以 Kafka 为数据源调用 Storm 流处理作业,以 5 min 为时间窗对每个用户的实时用电量数据进行预处理及特征提取操作,获得共计 120 万条用电行为特征记录,作为验证实验结果的原始数据集.

4.1　用电行为分析的实验结果

使用本文提出的分布式 DBSCAN 聚类算法在 Spark 中对原始数据集进行聚类分析,从异常用电行为识别和用户用电等级评估两个方面的分析结果对算法结果的准确性进行评估. 当算法的 Eps 参数和 MinPts 参数分别设为 0.57 和 5 时,原始数据集被划分为 5 个聚类簇,以实现对 5 个用户用电等级的对应. 对于不属于任何一个聚类簇的样本,则统一被标注为噪声数据.

将每个用户的 12 个时间窗对应样本中占比最大的样本类别作为该用户的类别标注. 分别统计各个聚类类别所对应的用户数,与原有的用户用电等级标注进行对比. 由表 2 中的

表 2　聚类标注与实际用电等级对比

Table 2　Comparison of cluster label and actual level

各等级实际用户数	各类对应样本数	误划分样本数	漏划分样本数
13355	14761	1689	283
37208	35652	1391	2947
22481	24478	3065	1068
18362	19045	2243	1560
8594	7376	683	1 901

对比结果可知,DBSCAN 聚类算法对用户类型的划分结果与实际对应的用电等级分布基本相同.

异常用电行为识别的准确性则使用正确率(Precision) 和召回率(Recall) 两个指标进行衡量. 将聚类结果中标注的 463 个噪声样本点与各自对应时间窗中是否出现异常用电行为的

警告记录进行比对,可以得到本算法在异常用电行为识别上的正确率为 87.57%,召回率为 94.81%.

由实验结果可知,本文中的分布式 DBSCAN 聚类算法在用户用电行为的分析上具有较高的准确性. 凭借基于密度的聚类策略,DBSCAN 算法能够根据样本点的分布特性实现聚类,同时不易受到噪声数据的影响,但结果中的类别数量由算法参数决定,需要调整参数才能获得所需的类别数目.

4.2 改进分布式 DBSCAN 算法的性能评估

文中提出的分布式 DBSCAN 算法采用仅保留聚类簇边界特征样本点作为聚类合并依据的策略,降低不必要的计算开销,从而提高聚类合并过程的效率. 为检验该策略对算法性能的提升效果,本文将原有采用增量合并策略的分布式 DBSCAN 聚类算法与 Spark MLlib 库中提供的分布式 k-means 算法[13] 作为对比. 分别保留原始数据集中 30 万、60 万、90 万及 120 万条用电行为特征记录构成不同规模的数据集用以验证算法的性能.

在参数设定上,对于分布式 k-means 算法,将算法中对应的类别参数 K 设为 10,迭代轮次 n 设为 1000,收敛阈值 α 设为 0.05. 对于分布式 DBSCAN 算法,将 Eps 设为 0.57,MinPts 设为 5,对于每一维特征采取 16 等分,两类算法的距离衡量标准均采用欧氏距离.

表 3 各算法在不同规模数据集上的时间开销

Table 3 Time cost of each algorithm on different data sets

数据规模	分布式 k-means	原有分布式 DBSCAN	改进分布式 DBSCAN
30 万	14.71s	20.69s	18.22s
60 万	29.63s	72.86s	31.57s
90 万	61.05s	181.18s	49.36s
120 万	134.88s	341.27s	88.59s

由表 3 中的实验结果可知,尽管 DBSCAN 聚类算法在结果准确性方面存在优势,但原有的分布式实现方式计算开销较大. 在处理中等规模的数据集时,分布式 k-means 算法具有较为明显的性能优势. 随着数据集规模的进一步增大,改进的分布式 DBSCAN 聚类算法相较于原有实现方式的时间开销增长幅度较小. 因此,基于边界特征的聚类簇合并优化策略能够有效地提高分布式 DBSCAN 算法的计算效率.

5 结束语

作为一种直接有效的数据分析手段,基于用户的实时用电量数据提取出用户的用电行为特征能够为后续的行为分析提供更加准确的数据支持. 将 DBSCAN 聚类算法成熟的分析思想与分布式计算框架的性能优势相结合,提高算法对大规模数据集的处理能力. 在算法分布式执行的过程中制定合理有效的计算策略,省去不必要的对比计算,降低聚类合并过程中所需要的存储开销,进一步提高算法效率.

采用分布式 DBSCAN 算法能够实现对大规模用户的用电行为类型进行较为准确的划分,达到对异常用电行为的识别和用户等级的评估的目的. 由于本文仅从实时用电量数据的统计特征、采样特征和频域特征三个方面进行行为特征构建,今后的研究工作中,可以更进一步的拓展特征构建的数据来源和特征指标,结合特征选择算法保留最优特征子集,使算法的分析结果更加准确.

References:

[1] Jiang Ling, Wang Xu-dong, Yu Jian-cheng, et al. Research on power usage Behavior analysis based on distributed computing [J]. Computer Technology and Development, 2016, 26(12): 176-181.

[2] Cheng Xue-qi, Jin Xiao-long, Wang Yuan-zhuo, et al. Survey on big data system and analytic technology [J]. Journal of Software, 2014, 25(9): 1889-1908.

[3] Jin Jian-guo. Review of clustering method [J]. Computer Science, 2014, 41(b11): 288-293.

[4] Yu Xiao-shan, Wu Yang-yang. Parallel text hierarchical clustering based on MapReduce [J]. Journal of Computer Applications, 2014, 34(6): 1595-1599.

[5] Xiong Yuan-xin, Chen Yun-ping. Research on definition of discrete fourier transform [J]. Engineering Journal of Wuhan University, 2006, 39(1): 89-91.

[6] Sun Da-wei, Zhang Guang-yan, Zheng Wei-min. Big data stream computing: technologies and instances [J]. Journal of Software, 2014, 25(4): 839-862.

[7] Niu Mu. A distributed cache and analysis platform for large scale streaming data based on Kafka [D]. Changchun: Jilin University, 2016.

[8] Wang Tao, Yang Yan, Teng Fei, et al. Distributed clustering ensemble based on RDDs [J]. Journal of Chinese Computer Systems, 2016, 37(7): 1434-1439.

[9] Li Shuang-qing, Mu Sheng-di. Improved DBSCAN algorithm and its application [J]. Computer Engineering and Applications, 2014, 50(8): 72-76.

[10] Hai Mo, Zhang Shu-yun, Ma Yan-lin. Algorithm review of distributed clustering problem in distributed environments [J]. Application Research of Computers, 2013, 30(9): 2561-2564.

[11] Tian Lu-qiang. Research and application on distributed clustering and incremental clustering based on DBSCAN [D]. Beijing: Beijing University of Technology, 2016.

[12] Likas Aristidis, Vlassis Nikos, J. Verbeek Jakob. The global K-means clustering algorithm [J]. Pattern Recognition, 2003, 36(2): 451-461.

附中文参考文献:

[1] 蒋 菱,王旭东,于建成,等. 基于分布式计算的海量用电数据分析技术研究 [J]. 计算机技术与发展,2016,26(12):176-181.

[2] 程学旗,靳小龙,王元卓,等. 大数据系统和分析技术综述 [J]. 软件学报,2014,25(9):1889-1908.

[3] 金建国. 聚类方法综述 [J]. 计算机科学,2014,41(b11):288-293.

[4] 余晓山,吴扬扬. 基于 MapReduce 的文本层次聚类并行化 [J]. 计算机应用,2014,34(6):1595-1599.

[5] 熊元新,陈允平. 离散傅里叶变换的定义研究 [J]. 武汉大学学报(工学版),2006,39(1):89-91.

[6] 孙大为,张广艳,郑纬民. 大数据流式计算:关键技术及系统实例 [J]. 软件学报,2014,25(4):839-862.

[7] 牛 牧. 基于 Kafka 的大规模流数据分布式缓存与分析平台 [D]. 长春:吉林大学,2016.

[8] 王 韬,杨 燕,滕 飞,等. 基于 RDDs 的分布式聚类集成算法 [J]. 小型微型计算机系统,2016,37(7):1434-1439.

[9] 李双庆,慕升弟. 一种改进的 DBSCAN 算法及其应用 [J]. 计算机工程与应用,2014,50(8):72-76.

[10] 海 沫,张书云,马燕林. 分布式环境中聚类问题算法研究综述 [J]. 计算机应用研究,2013,30(9):2561-2564.

[11] 田路强. 基于 DBSCAN 的分布式聚类及增量聚类的研究与应用 [D]. 北京:北京工业大学,2016.