基于 Spark 的聚类算法的优化

李晨曦[1]　鲍正德[2*]
四川大学锦城学院计算机与软件学院　成都　611731

【摘要】对于聚类算法在整个过程的最初数据选择的随机性问题,在非均匀采样的基础上对聚类算法进行优化。与此同时,出于要进行优化聚类算法这一问题,以 Spark 为基础让算法有所改观并进行优化。数据集采样阶段,聚类阶段以及算法的并行优化,这最主要的三个阶段极大的实现了聚类算法的优化,在存在大量数据时,都会有一定的精准性。同时,在 Spark 的基础之上,实行聚类算法速度更快,扩展性更好,由此说明聚类算法的优化可以处理更高要求的数据。
【关键词】聚类算法　Spark　优化　大数据

1　Spark 相关概述

1.1　Spark 大数据的计算框架

对于 Spark 这一概念而言,数据的处理和运行都是 MapReduce 这一框架为载体下进行的,在计算的时候会采用思想分布式,与 MapReduce 在磁盘上运行的有所不同,相对而言,Spark 时在内存为背景下的计算,并且运用 Spark 是一种基于内存的计算,且通过 DAG 的算法,在程序执行命令时会节省出大量的有效利用的时间。在高效的运行数据时,Spark 无数次运用 MapReduce 算法,能更好的对数据进行挖掘。Spark 计算模型流程图如图 1 所显示:

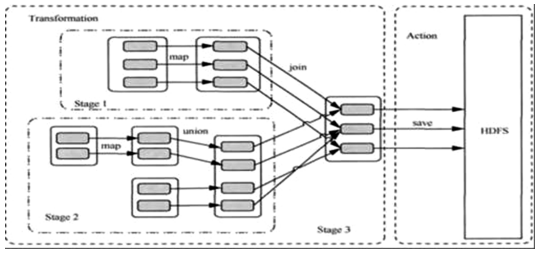

图 1　Spark 计算模型

1.2　Spark 计算模型的两个重要概念

Spark 计算模型的两个重要的概念是弹性分布式数据集(RDD)和 DAG。谈到逻辑这个概念,RDD 是典型的代表。它能够实现在在很多机器上分区。换句话说,实现多台器械上对不同的 RDD 的控制就可以明显看到机器与机器之间的数据更清晰。在图 1 所示的框图是一个 RDD。它的小框图是数据的不同分区。其原因是,Spark 提供了许多运营商,如 map、reduce、union、filter、reduceByKey、partitionBy、join、count 等等一系列运营商优化了转换,所以在 RAM 中缓存 RDD 更自由 RDD 的,所以它加快了数据的访问。转换 RDD 是一个是一个从读取数据到储存结果形成的一个过程,每一个特定的箭头都会有一个指定概念,即 DAG。[2] 在程序处理数据时,DAG 会带动着数据,从它的方向流动,这一过程集群并不会在中间算出结果,会在 action 的算子之后计算出。这一过程有效地对磁盘进行操控,有效的节省了磁盘的空间。所以,Spark 计算框架更加的适合对数据的挖掘处理工作。

2　聚类算法概述

2.1　聚类算法

聚类算法是基于 Spark 挖掘处理工作之后所有的方法,并被称为十大数据算法之一。聚类算法是将 n 个数据点区分到 m 个聚类之中,而每一个数据点所代表的意义都不同,属于离他最近的数据平均值代表的聚类。给定一组的 n 维数据 X =(X1,X2,...,XN),假设初始聚类中心,使用聚类算法的,每个数据点被分成 M 个组。还有另一组概念聚类中心,其被表示为 C=(C1,c2,,ck),对于每个点计算其到各个聚类中心的路程||x−c||,根据式(1)将该点分配到距离最近的聚类中:d(x,C)=minx∈X,c∈C||x−c||。在结束了数据的聚类计算后,还需要整合新的数据再次计算它们的聚类中心,具体而言,执行以下操作:CI=1|NCi|xi∈CiΣxi(2),其特征在于,CII 代表簇数据集,C 还 X,NCI 代表包含在该子集的数据的数目的。该聚类算法可以知道是否由相应的数据的误差的平方之和,以停止迭代和标准如下:SSE=CΣk=1nΣi=1|XI−CI|2(3)当 SSE 是比初始设定值。然后越小,迭代停止,具体步骤如下:1C←随机采样来自数据集 Xk 个点以形成初始聚类中心;②分别计算每个点 xi 到 k 个聚类中心 Ci 的距离,将 xi 划入最近的 Ci 中;③计算新的聚类中心 Ci;④计算 SSE,判断是否满足收敛条件,满足则结束迭代,否则重复②、③步骤。[3]

2.2　根据不均匀采取样品的聚类算法

从我们清楚知道的普遍计算方法出发得出。第一步就是想我们所知道的随机采样的办法相同,选取其中某些随机量,这些就可以做为类聚算法的样品。因此,得出的类聚算法的数据会和选取的这些值有着关系。要是选用的另一种方式来计算,随机

均匀采样的缺点就是样品的选出几率是一样的。然而在现实中我们得出的数据肯定是不一样的,应该思考下我们在选取中样品的几率是否符合,依据这些缘由,我们可以得出来的不均匀采取样品的类聚算法的好处与其他方式的类聚。

第一必须规定其中代数的函数:ΦX(C)=ΣxϵXd2(x,C)=kΣi=1min||x-Ci||2(4)依据其中的概率公式:px=d2(x,C)ΦX(C)(5)可以明确这个函数公式的准确意义是根据不断的验算中得出的数据平方和,以尽可能小的误差得出最后的理想数据。重复上面。首个样品 Φ 与多的采取数据系数 α,它的根本原理就是依据 α 值得客观的采取数据。[4]得出计算数据的方式为:1C←在基本母体 X 里随意选出 1 点;2φ←得出原始数据;3forlogφtimesdo;4C'←以本来的数据概率 Px 在数据 X 里选出 α 点,平均率 px=d2(x,C)ΦX(C);5C'←C∪C';6end;7 根据这样的计算清楚母体 C 的 k 个聚类中心。从首要的数据计算里,不管任何的数据点里,采取随机点进行的处理,接下来的得出基本值 φ,然后进行验算,这样的公式算法导论是要确定它的基本数据与得出结果的迭代次数,从这样的全程计算中,任何步骤必须选出 α 个点,迭代的计算概率例(4)。距离所得值得中间距离很远时,选出的样品数据的概率很大时,目地是此数据的聚类中心与数据的聚类中心最远。[5]首要的完成迭代时,就可得出新的 ΦX(C),更容易得出第二次的采取样品的概率准确性。

聚类算法的优化会依据采用阶段的过程进行,具体的采样阶段运用 RDD 实现。通过你上一代迭代聚类中心和这一代集群中心的叠加,形成一个完整的数据集。进行过 logφ 迭代之后,产生的新数据集会生成 C,和一个总 αlogφ 数据是在数据组 C. 最后,将进行平常的聚类算法,把所有的数据划分成 k 个聚类中心。经过对算法的初级处理,所得到的数据集 C 比开始的数据集 X 小了不止一倍,所以在产生新数据集之后聚类算法会进行的更加快。所以在优化算法的时候,通过运用 logφ 次迭代而代替传统的算法。不仅可以极大的减少每一过程的迭代次数,[6]而且对于海量大数据的运算会更高效的完成,更能大量的缩短时间。

3 聚类算法的并行优化

3.1 数据采样阶段

首先把所有的数据按照他们的特点进行排布,并且把它的结果输出来。这其中的每一个结果会代表一种聚类中心,而每一个中心属于一聚类。然后我们就进行算法,从 HDFS 文件一次读取初始的数据,确定其数据中心,把形成的 RDD1 作为一个内存来储存 cache 算子。根据每个算子的大小,并以相应的单位去划分他们。分区是一个很普遍的概念,比如说:

RDD1 是在整个数据上会有相应的算子 map,这一系列的 map 算子由 master 作为主要的节点去把它们分配到相应的节点 worker 上完成,这些步骤是对数据进行一个大体估计处理,那么最开始的数据则是 String(x1,x2,,xn)这一类型的,并且等量变换成 DenseVector 矢量(x1,x2,,xn)。[7]下一步是算子又生成了新的 RDD2,接下来启动另一种类型的算子 takeSample 算子。上述的步骤相当从这所有的聚类中随机的选择一个聚类中心,作为他最初的中心,生成 RDD3。RDD2 启动 map 算子,根据式(4)计算它的初始值 φ,之后算法会进入一个程序中进行迭代。在整个迭代的进程中,首先根据式(1)和式(4)不断的更新每个数据中心背选用的几率,而后运用 takeSample 算子,一具备选择几率的大小来作为新的聚类中心,这个时候把采用的数据点 1+α 个,作为 RDD4。然后运用 RDD3 中的 union 算子,把 RDD3 和 RDD4 放在一起形成 RDD5,之后通过 logφ 次迭代,RDD5 其中的数据个数,count≤1+αlogφ。每完成一次完整的迭代后,RDD5 会开始运用 saveAsTextFile 算子,然后把它的计算结果保存到 HDFS 文件当中,后来就开始下一过程的聚类计算,最后的结果会以数据集中的聚类中心 k 为主。采样阶段流程如图 2 所示:

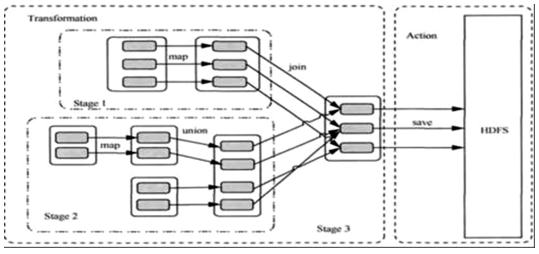

图 2 采样阶段 RDD 流程

3.2 聚类阶段

通过一系列的采样数据完成之后,整个数据的内存会大大的缩小。在运用采样聚类中心时,传统的聚类方式是根据 k 个聚类中心确定 k 个聚类点,那么这些点又可以形成新的聚类中心。[8]聚类阶段的过程示于图 3 中所示:

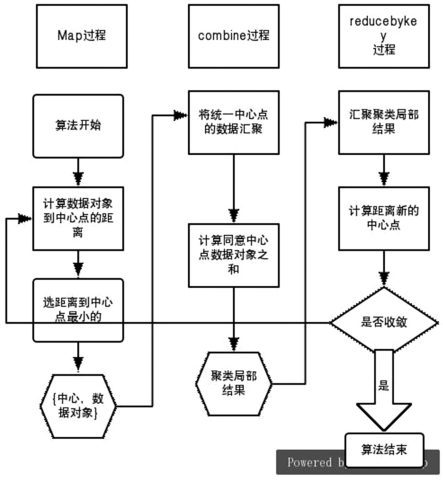

图 3 K-means 并行化流程

3.3 聚类算法的优化

首次把 RDD1 计算出来的聚类结果记载下来，并且运用 RDD1 开启 takeSample 算子。[9]由此可以得出，聚类中心的选取以 k 个点为主。表 kpoint(RDD2)。重新设置这个函数 map，细致算出每一个点的精确长度及其他数据中心点的 kpoint(RDD2)，下来选择距离聚类中心距离最小的点来形成 RDD3 的新的聚类中心。RDD3 再一次运用 collectAsMap，把他们放在同一个聚集中心点 RDD4 上，并且是与他相同的数据。RDD4 运用 reduceByKey，根据式(2)作用新的数据集群来计算其他的数据中心，根据计算所产生的误差(3)。[10]菱形 SSE，当 SSE 不满足计算条件时，那么它总的迭代次数会增加，否则聚类中心计算的输出结果会产生误差。RDD 过程的 K 均值阶段如图 4 所示：

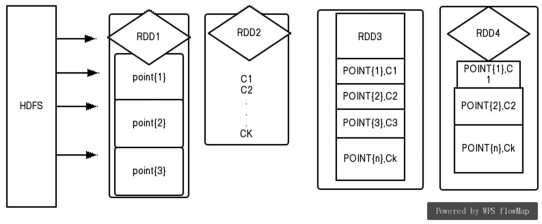

图 4　K－means 聚类算法 RDD 流程

4 总结

本论文主要研究了以 Spark 为基础的聚类算法的优化，并且采用文献法，采样法等多种方法进行研究。其中采用法式最主要的基于聚类中心被选取的几率大小来确定初始数值，聚类结果能够很好地反映开始的数据集的聚类优化，这样一来，数据算法的误差被平方和取代，大大的减少了误差，它在一定程度上改进了聚类算法。通过在集群上进行算子的运行，算子的加速度随着时间的进行也加快，计算结果的高效性以及准确性多次被实验所证实。一系列的数据表明，在海量数据的处理和运算过程当中，串行计算确实满足不了时代对大数据的需求。所以在未来的研究领域中，聚类算法的优化会发挥越来越重要的作用。

【参考文献】

[1]HAN J W, KAMBER M. Data mining: concepts and techniques[M]. San Francisco, CA, itd: Morgan Kaufmann Publishers, 2000.

[2]WU X D, KUMAR V, QUINLAN J R, et al. Top 10 algorithms in data mining[J]. Knowledge and Information Systems, 2008, 14(1): 1－37.

[3]ZHANG T, RAMAKRISHNAN R, LIVNY M. BIRCH: an efficient data clustering method for very large databases[C]//ACM Sigmod Record. 1996:103－114.

[4]毛典辉. 基于 MapReduce 的 Canopy－Kmeans 改进算法[J]. 计算机工程与应用, 2012,48(27): 22－26.

[5]XU Y J, QU W, LI Z, et al. Efficient k－means＋＋ Approximation with MapReduce [J]. IEEE Computer Society, 2014,25(12):3135－3144.

[6]ZIMICHEV E A, KAZANSKIY N L, SERAFIMOVICH P G. Spectralspatial classification with k－means ＋ ＋ particional clustering [J]. Computer Optics, 2014, 38(2): 281－286.

[7]张刚红. Hadoop 下并行遗传算法研究及在应急设施选址中的应用[J]. 互联网天地, 2013(8): 11－18.

[8]DEAN J, GHEMAWAT S. MapReduce: simplified data proce－ssing on large clusters [J]. Communications of the ACM,2008, 51 (1):107－113.

[9]ZAHARIA M, CHOWDHURY M, FRANKLIN M J, et al. Spark:cluster computing with working sets[C]//Book of Extremes. 2010:1765－1773.

作者简介：

第一作者：李晨曦(1998—7)，男，汉，贵州贵阳，本科，四川大学锦城学院，研究方向：大数据技术开发。

第二作者(通讯作者)：鲍正德(1989—7)，男，汉，黑龙江哈尔滨，研究生，讲师，四川大学锦城学院，研究方向：电子商务。