基于 Hadoop 的 C4.5 决策树分类算法并行化*

林树地，吴扬扬
（华侨大学 计算机科学与技术学院，福建 厦门 361000

摘 要：通过研究各种决策树分类算法的并行方案后，并行设计 C4.5 算法。同时根据 Hadoop 云平台的 MapReduce 编程模型，详细描述 C4.5 并行算法在 MapReduce 编程模型下的实现及其执行流程。最后，对输入的海量文本数据进行分类，验证了算法的高效性和扩展性。

关键词：云计算；Hadoop；MapReduce；数据分类；C4.5 算法；并行

中图分类号：TP301.6 **文献标识码**：A **文章编号**：1674-7720(2013)12-0085-03

The parallelization of C4.5 algorithm based on Hadoop

Lin Shudi, Wu Yangyang
(College of Computer Science and Technology, Huaqiao University, Xiamen 361000, China)

Abstract: In this paper, a parallel C4.5 algorithm is put forward by the study of a variety of decision tree classification algorithm parallel programs and the MapReduce programming model of the Hadoop cloud computing platform. At the same time, the execution flow of the C4.5 parallel algorithms in the MapReduce programming model is introduced. Finally, the input of mass text data is classificated to verify the efficiency and scalability of the algorithm.

Key words: cloud computing; Hadoop; MapReduce; data classification; C4.5 algorithm; parallel

随着信息技术的高速发展，人们积累的数据量急剧增长，如何从海量数据中提取有用的知识成为当务之急。数据挖掘应运而生，它是一个从大量的、不完全的、有噪声的、模糊的、随机的实际应用数据中，提取隐含在其中的人们事先不知道的但又是潜在有用的信息和知识的过程[1]。然而随着数据量的增加，数据挖掘处理海量数据的能力成为了不可忽视的问题。

云计算是解决这个问题的有效途径，它把大量高度虚拟化的资源管理起来，组成一个大的资源池，用来统一提供服务。云计算是最近几年出现的一门新兴技术，是并行计算、分布式计算、网格计算的发展[2]，具有广泛的应用前景。IBM、Google、微软等众多公司都很重视云计算技术，都快速推出了自己的云计算平台。目前比较热门的开源云计算平台有：Abiquo 公司的 abiCloud、Amazon 公司的 Eucalyptus、MongoDB、Enomalism、Nimbus、Hadoop。其中 Hadoop 平台是完全模仿 Google 体系架构做的一个开源项目，是现在应用最广、最成熟的平台。

决策树分类算法作为一个经典的数据挖掘方法，通过对大量数据的属性值进行分析，构造决策树，来发现数据中蕴涵的分类规则。然而，在数据增长大爆炸的时代，这些算法处理海量数据的性能总有些差强人意。云计算作为一个处理海量数据的良好途径，将算法布置在云计算平台中进行分布式计算是一个行之有效的方法。

本文采用 Hadoop 开源云平台，对数据集进行数据横向和纵向划分，分布到不同的节点对不同的属性进行并行处理，对海量文本数据进行分类。

1 Hadoop 开源云平台

Hadoop 是 Apache 软件基金会旗下的一个开源分布式平台，以 Hadoop 分布式文件系统 HDFS 和 MapReduce（Google MapReduce 的开源实现）为核心，为用户提供了系统底层细节透明的分布式基础架构[3]。HDFS 的高容错性、高伸缩性等优点允许用户将 Hadoop 部署在低廉的硬件上，MapReduce 分布式编程模型允许用户在不了解分布式系统底层细节的情况下开发并行应用程序。因此用户可以充分利用集群的计算和存储能力，完成海量数据的处理。

1.1 分布式文件系统 HDFS

HDFS 采用了主从(Master/Slave)结构模型，一个 HDFS

* 基金项目：福建省科技计划重点项目(2011H0028)

集群由一个 NameNode 和若干个 DataNode 组成。其中 NameNode 作为主服务器，管理文件系统的命名空间和客户端对文件的访问操作；集群中的 DataNode 管理存储的数据。HDFS 允许用户以文件形式存储数据。从内部来看，文件被分成若干个数据块，而且这若干个数据块存放在一组 DataNode 上[3]。NameNode 执行文件系统的命名空间操作，如打开、关闭、重命名文件或目录等，也负责数据块到具体 DataNode 的映射。DataNode 负责处理文件系统客户端的文件读写请求，并在 NameNode 的统一调度下进行数据块的创建、删除和复制工作。图 1 所示为 HDFS 的体系结构。

图 1 HDFS 体系结构图

1.2 并行编程框架 MapReduce

Hadoop 上的并行应用程序开发基于 MapReduce 编程框架，框架由一个单独运行在主节点上的 JobTracker 和运行在每个集群从节点的 TaskTracker 共同组成。它的原理是：利用一个输入的 key/value 对集合来产生一个输出的 key/value 对集合。用户用 Map 和 Reduce 两个函数来表达计算[3]。

用户自定义的 Map 函数接收一个输入的 key/value 对，然后产生一个中间 key/value 对的集合。MapReduce 把所有具有相同 key 值的 value 集合在一起，然后传递给 Reduce 函数。自定义的 Reduce 函数接收 key 和相关的 value 集合，合并这些 value 值，形成一个较小的 value 集合。

图 2 为 MapReduce 的数据流图，这个过程简而言之就是将大数据集分集为成百上千个小数据集，每个或若干个小数据集分别由集群中的一个节点进行处理并生成中间结果，然后这些中间结果又由大量的节点合并，形成最终结果。此框架下并行程序中需要 3 个主要函数：Map、Reduce、Main。在这个结构中，需要用户完成的工作仅仅是根据任务编写 Map 和 Reduce 两个函数。

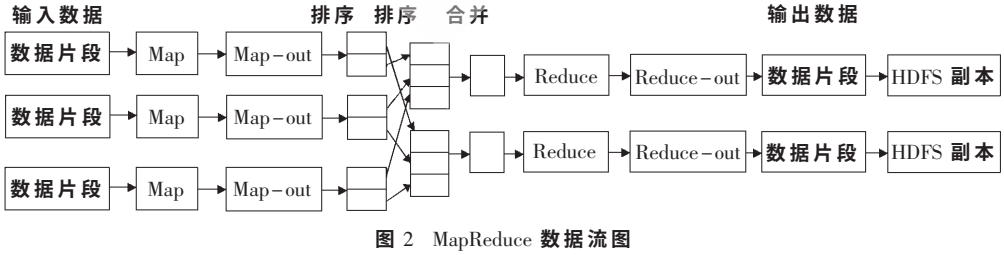

图 2 MapReduce 数据流图

2 C4.5 决策树分类算法

在决策树分类算法中，最常用、最经典的是 C4.5 算法，它继承了 ID3 算法的优点并对 ID3 算法进行了改进和补充。此算法描述如下[4]：

(1)预处理样本数据集。若存在连续取值的属性变量，则将其进行离散化，形成决策树的训练集；若不存在则忽略此步。

①根据原始数据，分别找到该连续型属性的最小取值 a_0 和最大取值 a_{n+1}；

②在区间 $[a_0, a_{n+1}]$ 内插入 n 个数值，将其等分为 $n+1$ 个小区间；

③分别以 $a_i(i=1,2,\cdots,n)$ 为分段点，将区间 $[a_0, a_{n+1}]$ 划分为两个子区间：$[a_0, a_i]$，$[a_{i+1}, a_{n+1}]$，对应该连续型属性变量的两类取值，有 n 种划分方式。

(2)计算每个属性的信息增益和信息增益率。

①计算属性 A 的信息增益 Gain(A)；

②计算属性 A 的信息增益率 GainRatio(A)。对于取值连续的属性，以 $a_i(i=1,2,\cdots,n)$ 为分割点计算相应分类的信息增益率，选择最大信息增益率对应的 a_i 作为该属性分类的分割点。而后选择信息增益率最大的属性作为当前的属性节点，得到决策树的根节点。

(3)根节点属性的每一个取值对应一个子集，对样本子集递归执行步骤(2)，直到划分的每个子集中的数据在分类属性上取值都相同，或者没有剩余属性可以进一步划分数据，或者给定的分支中没有数据，便停止划分，生成决策树。

(4)根据生成的决策树提取分类规则，对新的数据集进行分类。

3 C4.5 算法并行化

本文结合数据横向和纵向划分方法，以提高并行效率。具体设计思想如下：

Map 阶段：主要任务是处理输入的 1/M 训练样本集，扫描每条记录，将其格式化为<key，value>对。具体格式为 key=属性 S，value=<对应属性 S 的值 s，所属类别 c，原表中此记录的 id>。格式化后即可进行 Map 操作，每个 Map 任务为划分归类具有相同 key 的键值对，写到相应文件，由 partition 用模计算将各个文件分配到指定的 Reduce 上，即将相同的 key 分配到同一个 Reduce 节点上，如图 3 所示。

Reduce 阶段：主要任务是处理 Map 输出的<key，value>键值对。对于具有连续值的属性，先从小到大排序属性值，生成直方图，用来记录该属性对应记录的类分布，初始化为 0，每个 Reduce 任务都计算分割点的信息增益率，并及时更新直方图。对于离散的属性，不需要排序，也无需更新直方图，第一次扫描数据 Map 的输出记录时，生成相对应的直方图，计算

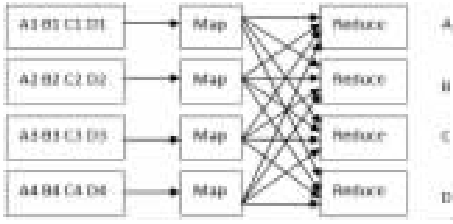

图 3 将相同 key 分配到同一个 Reduce

各子节点的信息增益率。每个 Reduce 节点都将计算得到各自属性列的信息增益率和分裂点，根据分裂点分割属性列表，用列表的索引号生成记录所在节点的哈希表，存储分裂点两侧的数据记录。哈希表格式为<key，value>键值对，value=<树节点号 NodeID，当前树节点号的子节点号 SubNodeID>，其中 SubNodeID 为 0 表示左节点，1 表示右节点。哈希表中的第 N 条记录表示原数据中第 N 条记录所划分到的树节点号。最后根据哈希表各分节点对其他属性列表进行分割，并生成属性直方图。

主程序算法设计描述如下：

```
Main()
{
  输入训练样本集 T;
  生成有序的属性列表 A 和直方图 G;
  创建节点队列 Q，首节点 N 为训练样本集所有数据记录;
  while(队列不为空)
    {
      取出队列首节点;
      while(节点数据样本不属于同一类 && 还有属性可操作 && 样本数据不是太少)
        {
        将节点中的训练样本集进行水平划分，分割为 M 份，由 InputFormat 完成，将数据划分为 InputSplit 发送到各个 Map 节点进行处理，这里同时也进行垂直分割;
        Map 操作;
        Reduce 操作，以 Map 节点的输出中间结果作为输入;
        根据各个 Reduce 节点返回的输出结果，选择最大信息增益的属性，以其分裂点和哈希表分裂原始数据集，生成子节点 N1 和 N2，放入队列 Q;
        }
    }
}
```

例如，以 ALLElectronic 顾客数据为训练集，Hadoop 默认参数进行分片，其中水平和垂直分割过程如图 4 所示。

对 ALLElectronic 顾客数据集进行分类，顾客数据集的属性分别为 ID、年龄、收入水平、学生身份标志、信用卡水平。根据这些属性对顾客消费潜力进行评估，将顾客分为会消费顾客和不会消费顾客，训练产生的决策树模型如图 5 所示，用此模型对数据进行分类。

图 5 训练 ALLElectronic 数据集得到的决策树模型

4 实验

本实验主要验证算法的高效性和扩展性。实验数据为 UCI 数据集 Bank Marketing，用来预测用户是否会定期存款。该数据集属于商业领域，具有多变量的特征，包括客户的年龄、工作、婚姻情况、学历、年均收入、房贷、支出等 17 个属性，而且是实数，有 45 211 个元组，没有缺省值，经常用来完成分类的任务。

实验环境：软件方面：采用 Hadoop-0.20.2 版本，Ubuntu Linux 9.04 系统，Eclipse3.3.2 开发工具，Jdk 7.0；

图 4 水平分割和垂直分割过程

硬件方面：7 台 PC(其中 1 台作为主机，其他 6 台作为从机)，每台 PC 的配置为：CPU i3，内存 1 GB，网卡 100 Mb/s。

实验内容：采用复制的手段将 Bank Marketing 扩大，产生分别为 100 MB、200 MB、400 MB、700 MB、1 GB 大小的数据集。测试各个数据集在不同数量的集群中算法的运行时间，从机集群数量分别设为 1、2、4、6 台。运行结果统计如图 6 所示。

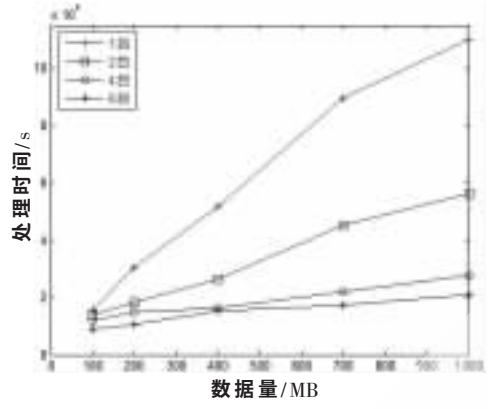

图 6 不同数据量处理运行结果

数据的处理时间主要花费在数据的分割和记录的格式化过程，由图 6 可见，随着集群数量的增大，处理时间有效地缩短了。具体原因分析如下：MapReduce 对数据的分割一般以 64 MB 为一基本单位。例如，700 MB 大小的数据可分割为 11 个数据块，如果用 1 台机器去处理，需要计算 11 次；用 2 台处理，需要计算 6 次；4 台处理则只要计算 3 次；6 台则只要计算 2 次。因此可以得出此算法有很高的高效性和扩展性。

决策树分类算法有广泛的应用领域，如客户关系管理、专家系统、语音识别、模式识别等。C4.5 并行化决策树分类算法与传统决策树分类算法相比，有较大优势，可以支持海量数据的处理。同时将其运行在 Hadoop 云计算平台上，能够高效地对大规模海量数据进行分类。

参考文献

[1] 房祥飞.基于决策树的分类算法的并行化研究及应用[D].济南：山东师范大学，2007.

[2] 刘鹏.云计算[M].北京：电子工业出版社，2010.

[3] 陆嘉恒.Hadoop 实战[M].北京：机械工业出版社，2011.

[4] 田金兰，赵庆玉.并行决策树算法的研究[J].计算机工程与应用，2001，16(5)：17-20.

(收稿日期：2013-03-19)

作者简介：

林树地，男，1988 年生，硕士研究生，主要研究方向：数据库技术。

吴扬扬，女，1957年生，教授，主要研究方向：数据库技术。