BloomFilter Join 性能测试报告

修订历史

版本	修订日期	修订描述	作者	备注
Cedar 0.2	2016-06-30	BloomFilter Join 性能测试报告	茅潇潇	无

简洁测试结果

实验证明，基于布隆过滤器的连接算法极大地减少了Cedar对连接操作的处理时间。随着选择率的降低和连接列个数的增加，基于布隆过滤器的连接算法的性能提高的更加明显，并且该算法的性能几乎不受数据分布情况的影响。

1 测试环境

使用4台虚拟机组成的集群作为测试环境，每台虚拟机的配置相同，包括4核1.2 GHz主频CPU、100 GB内存、3000GB磁盘，虚拟机上安装了CentOS release 6.5 系统，相互之间通过千兆以太网连接。集群中的一台虚拟机被配置为RootServer、MergeServer和UpdateServer，另外三台虚拟机被配置为ChunkServer。实验采用的数据是使用数据生成器随机生成的数据。

2 测试方法

为了验证本算法的效率，设计了三组实验，从选择率、连接列和数据分布对性能的影响三个方面，通过观察对相同查询语句的处理时间，分析了基于布隆过滤器（Bloom Filter）的连接算法的性能，并得出了结论。

3 结果分析

3.1 选择率对性能的影响

该实验对比查询语句中连接列的选择率对Cedar中传统的排序归并连接算法与基于布隆过滤器的连接算法的性能影响。测试左表包含10万条记录，右表的数据量从10万到1000万条记录不等，两表连接列均为[1, MAX]的整数，MAX为记录数。

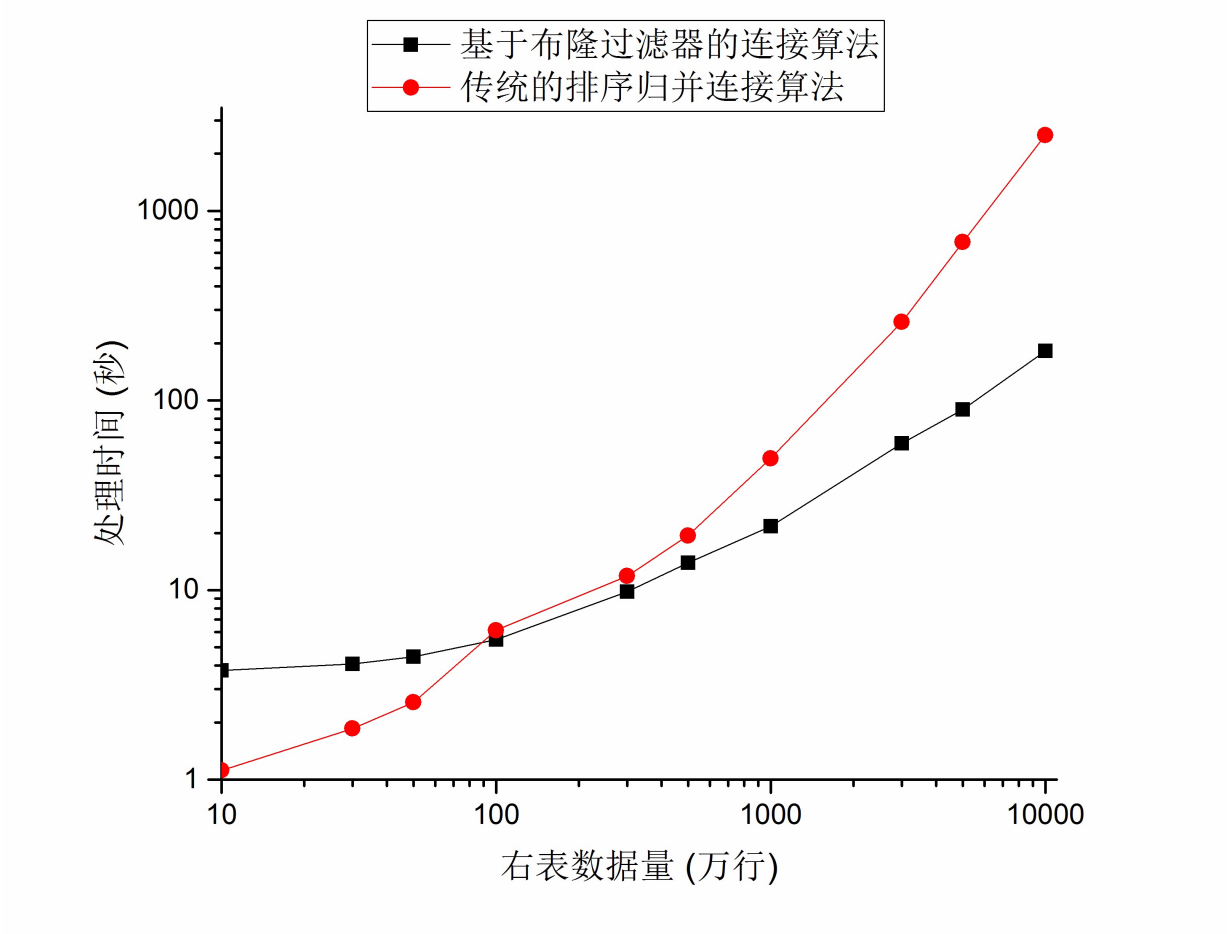

图3.1 选择率对性能的影响结果

从图3.1中可以看出，当右表的数据量较小，即左表对右表的选择率较高时，布隆过滤器的构建和查找增加了计算开销，右表数据传输的网络开销降低地并不明显，基于布隆过滤器的连接算法的处理时间比传统的排序归并连接算法的处理时间要多。但是当右表的数据量达到100万行以上，即左表对右表的选择率越来越低时，传统的排序归并连接算法的处理时间大大增加，而使用布隆过滤器的连接算法的处理时间则呈近似线性增长。这是因为在选择率较小时，数据传输的网络代价将会占查询处理时间的主要部分。布隆过滤器以极低的计算代价，极大地降低了网络开销，提高了连接操作的性能。

3.2 连接列对性能的影响

该实验对比查询语句中连接列的个数对Cedar中传统的排序归并连接算法和基于布隆过滤器的连接算法的性能影响。测试左表包含100万条记录，右表包含1000万条记录，连接列的个数从1到7个不等，两表连接列均为[1, MAX]的整数，MAX为记录数。

图3.2表明，查询语句中的连接列个数增多时，基于布隆过滤器的连接算法优势更加明显。布隆过滤器将多个连接列映射到一组对应的位数组，随着连接列的增多，布隆过滤器对数据的描述更加准确，对数据的过滤也更加有效。

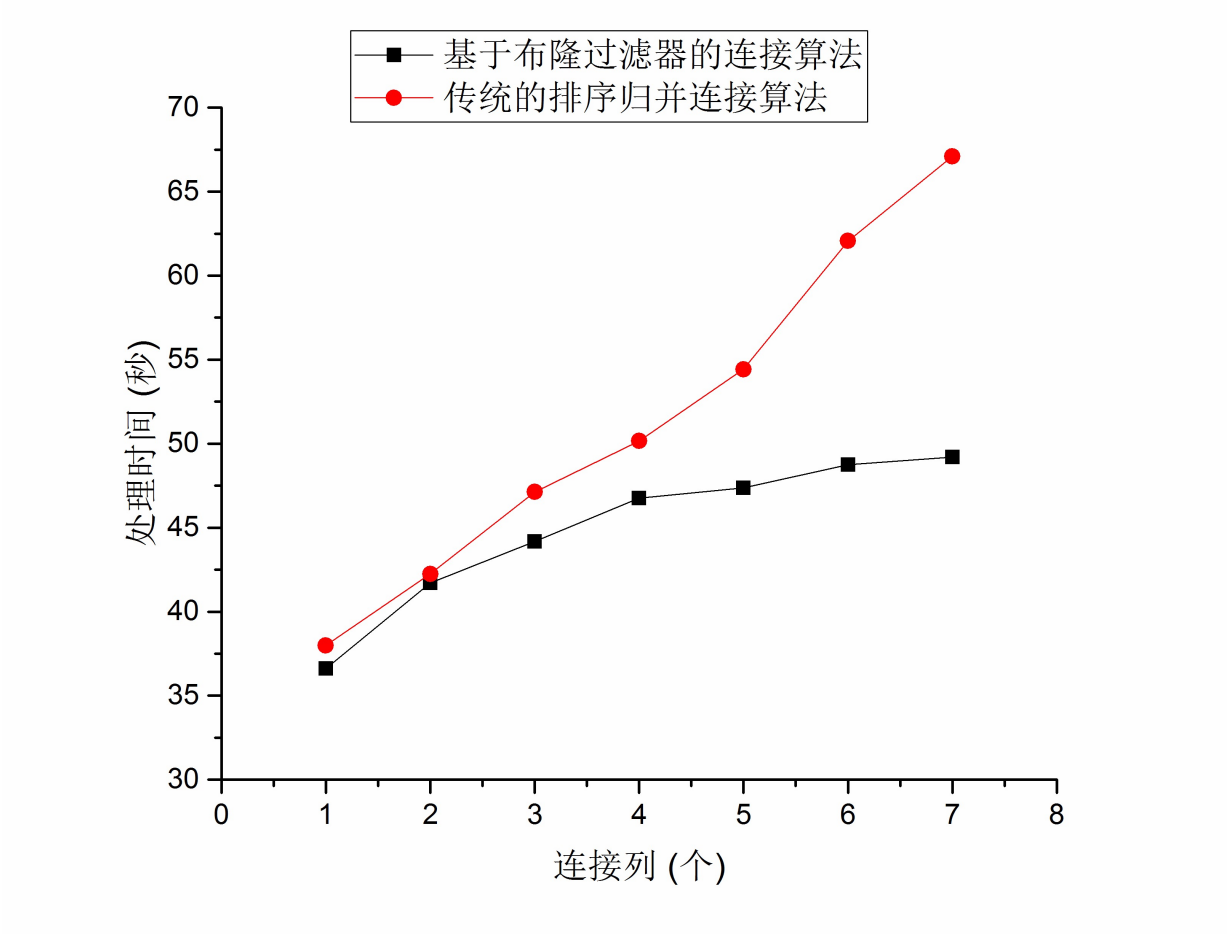

图3.2 连接列对性能的影响结果

3.3 不同数据分布对性能的影响

该实验对比连接列中数据不同的分布情况对基于布隆过滤器的连接算法的性能影响。测试左表包含10万条记录，右表的数据量从10万到1000万条记录不等。为避免数据分布情况对两表连接结果的大小产生影响，控制三种数据分布下两表连接的结果集大小相等。

如图3.3所示，数据的分布情况对布隆过滤器的性能几乎没有影响。布隆过滤器构建和查找的性能主要由前表和后表的元组数决定，因此在不同的数据分布下，布隆过滤器的性能是较为稳定的。

综上，改进后的连接算法极大地减少了Cedar对连接操作的处理时间，并且随着选择率的降低和连接列个数的增加，性能的提高也更加明显。

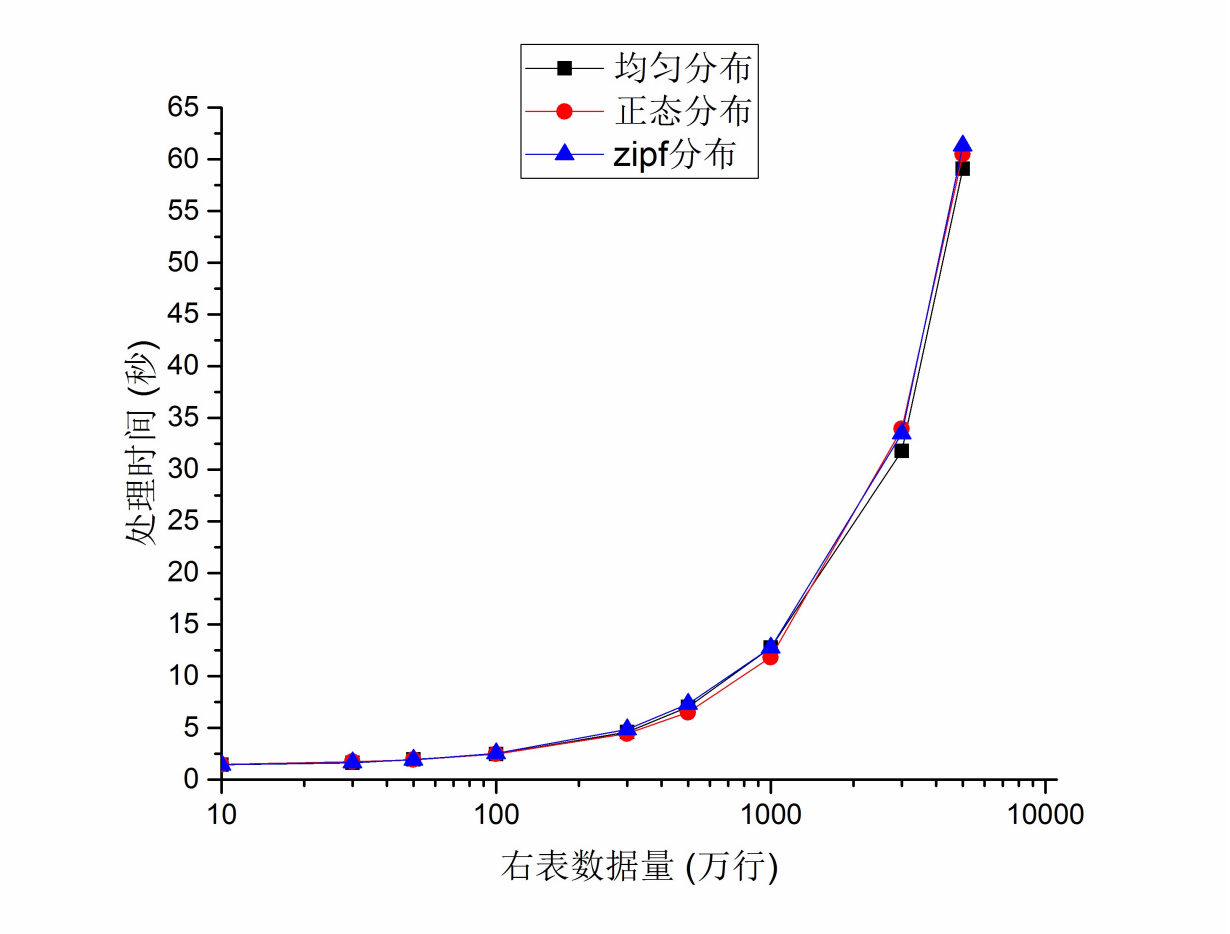

图3.3 不同数据分布对性能的影响结果