Analysis of Map Reduce Performance using Prefetching Mechanism

Abhilasha Singh  
Chetali Parve  
Priyanka Tripathi

ABSTRACT
In this massive technological atmosphere the number of information generated is increasing at an awfully high rate. Distributed storage system will be used for storing this immense quantity of information. Massive information will be handled by victimization hadoop and map reduces. The multiple node clusters will discover victimization the hadoop framework. Hadoop comes with default distributed file system that is hadoop distributed file system. The mapped is enforced by map technique and that we can see the performance of map reduce task on the bases of bytes written. The amount of bytes written by map task doesn't increase with the amount of files increasing. The explanation is that once the reduce function perform reduces the map output it simply combines the output of the map function. Thus so as to enhance the performance of map reduce task we are going to setup the cluster of nodes in heterogeneous atmosphere and analyze the behavior of map reduce task victimization the perfecting mechanism.

Keywords
Hadoop, hadoop distributed file system, map reduce, prefetching mechanism.

1. INTRODUCTION
Big data is that the term used for a group of knowledge sets that are so large that they are impractical to process using hands on management tools or ancient processing applications. huge knowledge suggests that heaps and much of knowledge i.e. knowledge from terabytes to petabytes. The challenges embrace with huge knowledge is how we are able to capture, create, store, search and share this immense quantity of knowledge. The massive knowledge characteristics are volume, variety, veracity and rate.

Table 1: Characteristics of Big Data

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>volume</td>
<td>Huge amount of data</td>
</tr>
<tr>
<td>variety</td>
<td>Different forms of data</td>
</tr>
<tr>
<td>velocity</td>
<td>Rate at which data flows</td>
</tr>
<tr>
<td>veracity</td>
<td>Inconsistency data due to unavailability of data</td>
</tr>
</tbody>
</table>

Big data client situation includes net and e-tailing, adhere card, health care and natural science, banking and money services and retail. Hadoop is the best resolution for big data problem [1]. Hadoop could be a framework that leave distributed process of enormous large sets across clusters of trade goods computers employing an easy programming model.

Hadoop is an open supply knowledge management with scale out storage and distributed process. Hadoop core parts area unit are HDFS (hadoop distributed file system) and reduce. Hadoop relies on master slave design. Characteristics of hadoop area unit Reliable as a result of fault tolerance, Economical as a result of trade goods hardware, scalable and versatile.

2. RELATED WORK
To increase the performance degradation caused by data transmission, some work is done. Data transmission overhead can be removing by using data prefetching mechanism. To avoid directly modifying the native Hadoop, a bi-directional processing approach is proposed in HPMR [2]: the processing fetches the data from the beginning of the input split while prefetching fetches the data from the end. While the proposal in this paper fetches data from the beginning of input data to reduce the overhead of data transmission at the maximum.

Some researchers focus on optimizing task scheduling algorithms or data replication policies to improve data locality in Map Reduce [3]. These proposals only improve the probability of data locality in MapReduce and may increase the complexity of achieving load balance. The LATE scheduling algorithm is proposed for Map Reduce in heterogeneous environments [4]. M. Zaharia et al., have proposed a delay scheduling algorithm, which addresses the conflict between locality and fairness in shared Map Reduce cluster [5]. In MTSD [6], computing nodes are classified by computing capability and a modified task scheduling algorithm is studied. X. Zhang et al. have studied scheduling with consideration about data locality in homogeneous cluster [7]. DARE is a distributed adaptive data replication algorithm that is sensitive to the heterogeneity of computing nodes, and the more powerful nodes get more data replications [8].

3. AREA OF STUDY
Big data are often handled by exploitation the Hadoop and Map reduce. The multiple node clusters are often got wind of exploitation the Hadoop and map reduce. Our space of interest during this paper is how map reduce works . The environmental execution of Map Reduce task will be outlined. Map reduce may be a programming model and an associated implementation for process and generating massive knowledge sets with a parallel, distributed algorithmic program on a cluster.

A Map reduce program consists of a Map () procedure that performs filtering and sorting (such as sorting students by name into queues, one queue for every name) and a Reduce () procedure that performs an outline operation (such as enumeration the quantity of scholars in each queue, yielding name frequencies). The "Map reduce engine" (also known as "infrastructure" or "framework") orchestrates the process by marshalling the distributed servers, running the assorted tasks in parallel, managing all communications and knowledge transfers between the assorted components of the system, and providing for redundancy and fault tolerance. The model is impressed by the map and reduces functions normally utilized in useful programming though their purpose within the Map reduce framework isn't an equivalent as in their original forms. The key contributions of the Map reduce framework don't seem to be the particular map and reduce functions, however the measurability and fault-tolerance is achieved for a range of applications by optimizing the execution engine once.
Map reduce libraries are written in several programming languages, with completely different levels of improvement. A well-liked ASCII text file implementation is Apache Hadoop. The name Map reduce originally spoken the proprietary Google technology, however has since been generalized.

Prefetching theme

In prefetching mechanism the corresponding computing node that is process [the knowledge|the info|the information] as shown in figure two can prefetch following knowledge before hand on his cache with the assistance of prefetching thread as a result of as that the name node provides data to the information nodes so data nodes know from wherever it’s to browse the information ,what is the situation of and from wherever it’s to require the information for process therefore before finishing the present process data the information node can fetch the information beforehand on its cache. As presently as double current knowledge task is finished knowledge node can take following task from its native cache and begin process grasp the information node don’t got to look forward to data that it want from master node .Prefetching mechanism improves the information process tasks and overhead of underutilized mainframe area unit diminished and overall performance of mapped and reducer tasks area unit exaggerated.

5. CONCLUSION

Map Reduce is an effective programming model for large-scale data-intensive computing applications. Hadoop, an open-source implementation of Map Reduce, has been widely used. The communication overhead from the big data sets' transmission affects the performance of Hadoop greatly. To improve the performance of Map Reduce in heterogeneous or shared environments, a data prefetching mechanism is proposed in this paper, which can fetch the data to corresponding compute nodes in advance. We have seen that when prefetching mechanism is applied on mapper function the overall performance of hadoop cluster is increased so in order to further increase the performance of map reduce we will apply the same prefetching mechanism in reduce function.

6. REFERENCES


