4.2 Report

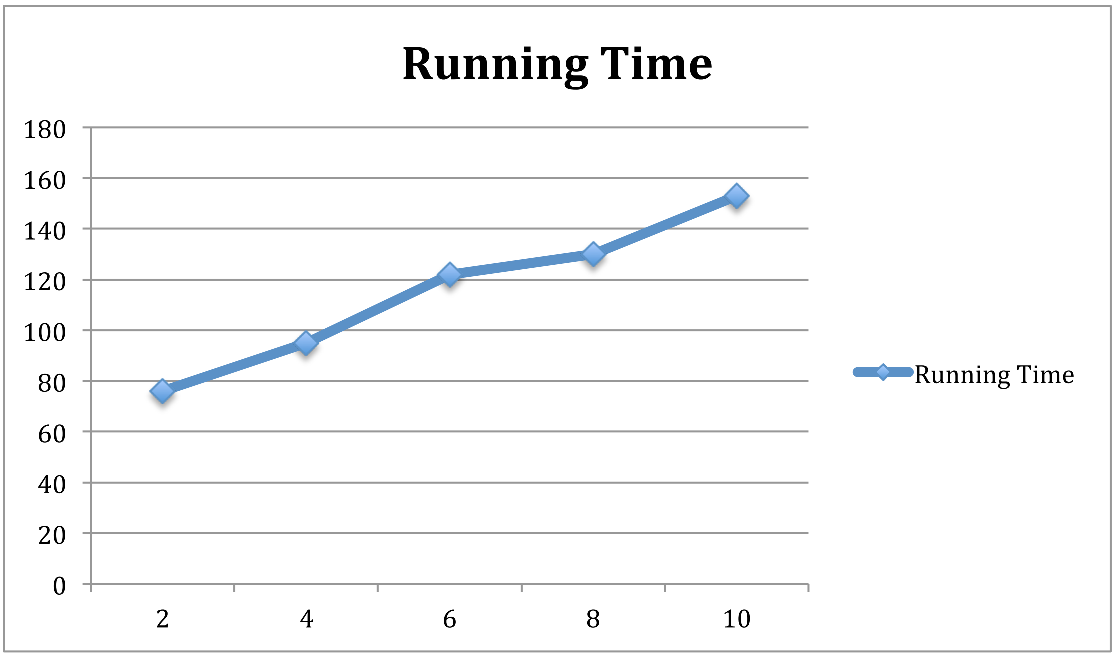
1.

I did not received any help whatsoever from anyone in solving this assignment. But I did receive help from the piazza.

Full detail: "package org.apache.hadoop.conf does not exist. I import the Hadoop packages via maven, and I can write the code without any error. But when I want to compile the java program and package them into jar file, I met this problem. Could you please to help me figure out why.

I did not gave any help whatsoever to anyone in solving this assignment.

2.



The wall time is linear with respect to the number of reducers. With the increase of reducer number, the running time increases. It is because that with the increase of reducer number, the instances need to send more requests through network, which results in more traffic and heavy load to the instances. In this way the running time increase a lot.

3.

Yes. The extreme case of skew that 10% of the words in the document are copies of the word "the" will adversely impact the performance of the MapReuce job, because after the Map job all the "the" word will be assigned to a machine. In this way the machine that is supposed to process "the" word will be on heavy load. Even more the "the" word might also be sent to different Reducers.

A way to fix this problem is to aggregate "the" word in the Mapper in advance. We can create an integer to count the number of appearance of "the" in each Mapper and put the key "the" and the total number of its appearance in a single machine into the context. In this way, reducers will receive much less data, which will highly increase the efficiency.

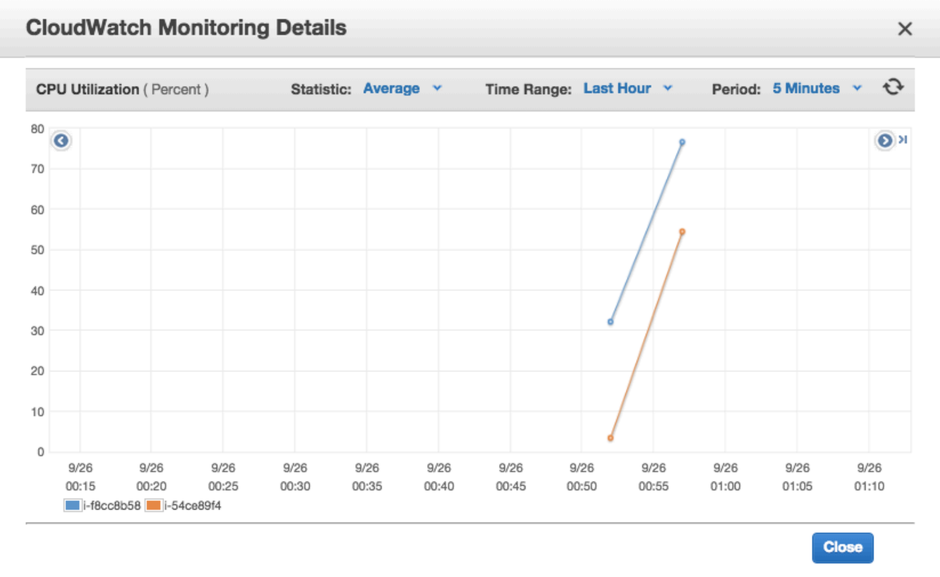
4.

First of all, HDFS is supposed to store large datasets. If a file size excesses the block size, another block will be assigned to the file. So if the block size is too small, say 4Kb, a large file will send requests through network for many times, which will result in a lot of traffic.

On the other hand, the total time to find a information from file system equals to the seek time adds the read time. The smaller the block size, the more the seek time. And for map reduce job, we need to traverse data through the file system all the time, it makes more sense to decrease the seek time by decreasing the block number.

5.

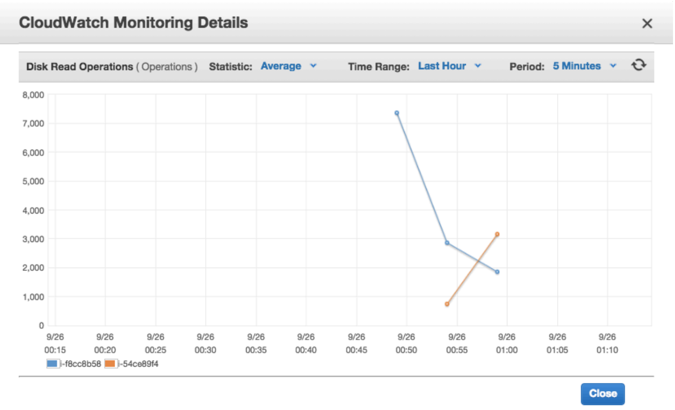
CPU Utilization:



CloudWatch Monitoring Details

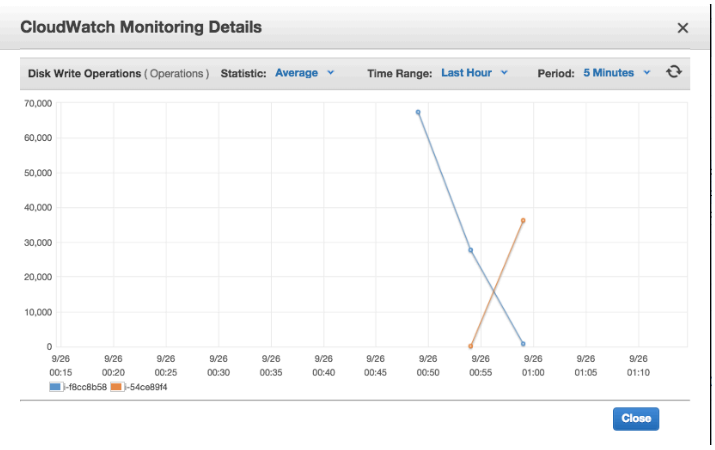
CPU Utilization (Percent)   Statistic: Average   Time Range: Last Hour   Period: 5 Minutes

Disk read/write operation:



CloudWatch Monitoring Details

Disk Read Operations (Operations) Statistic: Average Time Range: Last Hour Period: 5 Minutes



CloudWatch Monitoring Details

Disk Write Operations (Operations) Statistic: Average Time Range: Last Hour Period: 5 Minutes

When start a new MapReduce job, the CPU Utilization of the two job instances increased from 0 to 55 and from 30 to 78. So the CPU Utilization is not high. However when the CPU Utilization is relatively low the disk read/write operation is frequent , and the CPU Utilization reached the peek when the disk read/write operation numbers began to decrease. It means the CPU working efficency did not match the work load pattern. So to decrease the running time, we need to increase the machines' working efficency by warming up the instances before running the program.