Assignment 2.2 - Big Data Analytics

Katerina Chinnappan, Rohit Shaw, Kjell Zijlemaker

Experience with Hadoop/Spark

Configuration

The first step for setting up Hadoop was to create three VMs; one master and two slaves, and contextualize them with persistence and a designated IP. Because this was also in the previous assignment, this was not too difficult.

However, the configuration of Hadoop was proven to be more difficult because of the amount of XML files which needed configuration, as well as connecting the slaves to the master. The first nodes was required to be designated as a namenode and the other nodes as datanodes. Configuring these nodes took a longer time then expected, but thankfully, because of the following source (1), we were able to install Hadoop without any problems.

Finally, it was also required to give YARN the correct values for the total amount of RAM and CPU resources. When this is configured incorrectly, not all resources will be used.

Spark was easier to configure, because it runs on Hadoop and YARN which were already configured previously. We followed the following source (2) to install Spark. Using Spark out of the box is a good experience. You can submit jobs easily from the command line and get enough information about the progress of the job.

Using Spark

Spark was used for the PageRanking algorithm, therefore we will not discuss Hadoop here.
To use Spark with a python script, only one command has to be given to the namenode, namely: 'spark-submit pagerank.py'. If configured correctly, Spark will request resources which are given by the user in the application. Thus, in the application parameters will have to be given of the amount of resources which the user wants to use. In our case we used approximately 16GB of RAM, distributed accross two VMs using one CPU per VM.

In total, the amount of time it took to run the application to completion was approximately 10 to 15 minutes with this setup, which is relatively fast given the usage of only two nodes in the cluster.

Finally, when the application was done running, files were uploaded on the HDFS cluster in parts. We used the Hadoop -getmerge command to merge all small files into one text file. Then we sorted the file on the second column (score) in an descending order and took the top 10. Please see Appendix A for the full table of rankings.

Performance

The performance of the PageRank algorithm of course varies when performed on a single VM or multiple VMs. When only one VM is used, resources which are required are used with the possibility of the VM is under-or-overfitted with respect to these resources. When using multiple VMs, only the resources which are required are really used for the PageRank job. Other resources within the cluster are then available for other jobs to be used by other users.

Another aspect is that the job is distributed across multiple VMs and is therefore also load balanced, which increases the performance of the job and the total amount of resources which can be used for the job.

Appendices

Top 10 websites PageRank

ID	Score
272919	6531.324623752435
438238	4335.323158564435
571448	2383.8976074118855
601656	2195.3940755967296
316792	1855.6908757901426
319209	1632.8193684975693
184094	1532.2842374483223
571447	1492.9301630938778
401873	1436.1600933469197
66244	1261.5783958673337

References

[1] Hadoop. Source: https://www.linode.com/docs/databases/hadoop/how-to-install-and-set-up-hadoop-cluster/. accessed: 25-03-2019

[2] Spark. Source: https://www.linode.com/docs/databases/hadoop/install-configure-run-spark-on-top-of-hadoop-yarn-cluster/. accessed: 25-03-2019