BİL401

Project Status Report

Kerem Elma - 211101079

Mehmet Eski – 221104095

1. Status of Data Collection

As we stated in the proposal report, we had a dataset titled "Air Quality of New York". The website where the dataset was located only had it in json file format.

We downloaded the file in JSON format, found the parts where the data was and converted it to XML, Parquet and Avro formats where we would make the comparison.

During the conversion process, we used Python and used its libraries.

There were some problems while trying to convert the files, these were because hadoop was not added to the path on the local machine but we easily overcame this problem.

1. Status of the Platforms and Systems, Know-how

We installed PySpark in a virtual environment where the project was developed and started running it in local mode. We did not experience any other problems during the installation except for the path addition problem that we mentioned before.

We wrote Python scripts that read the datasets created with Python and saw the running times by using them in PySpark shell.

We used basic libraries such as pandas, fastavro, pyarrow for these file converter and file reader scripts.

In comparison metrics, we can first consider the running times, in the tests we conducted with reader scripts, we easily observed that different formats took different times, and also different file sizes are one of the reasons for this difference. In order to create additional metrics, we can observe the resource usage of the files during their reading times.

In this way, we learned different file types, their conversions and how to process them with Apache Spark.

1. Demo Run Sample

Let's have a demo run where we compare JSON andXML formats. You can see how the JSON file is converted to XML format with the code below:

**file\_converter.py**

import json

import xmltodict

#read json file

with open('datasets/data.json', 'r') as json\_file:

    json\_data = json.load(json\_file)

#converting json format to xml format

xml\_data = xmltodict.unparse({"root": {"item": json\_data}}, pretty=True)

with open('datasets/data.xml', 'w', encoding='utf-8') as xml\_file:

    xml\_file.write(xml\_data)

By executing the above code, “data.xml” is created from “data.json”. We then run the “xml\_reader.py” file.

**xml\_reader.py**

import time

from pyspark.sql import SparkSession

spark = SparkSession.builder.appName("XML Reader").getOrCreate()

start\_time = time.time()

df = spark.read.format('com.databricks.spark.xml').option("root", "item").load("datasets/data.xml")

end\_time = time.time()

reading\_time = end\_time - start\_time

print(f"========================================================================\nReading Time of XML formatted file: {reading\_time} seconds\n========================================================================")

After running it on the command line, we see the output as follows:

**(Part of the output we obtained from the command line and the running time are shown in the middle.)**

25/03/01 02:31:02 INFO TaskSchedulerImpl: Killing all running tasks in stage 0: Stage finished

25/03/01 02:31:02 INFO DAGScheduler: Job 0 finished: fold at InferSchema.scala:95, took 0,321704 s

25/03/01 02:31:02 INFO BlockManagerInfo: Removed broadcast\_0\_piece0 on KEREM.mshome.net:10550 in memory (size: 32.8 KiB, free: 434.4 MiB)

25/03/01 02:31:02 INFO BlockManagerInfo: Removed broadcast\_1\_piece0 on KEREM.mshome.net:10550 in memory (size: 3.5 KiB, free: 434.4 MiB)

**========================================================================**

Reading Time of XML formatted file: 1.8701293468475342 seconds

**========================================================================**

25/03/01 02:31:03 INFO SparkContext: Invoking stop() from shutdown hook

25/03/01 02:31:03 INFO SparkContext: SparkContext is stopping with exitCode 0.

25/03/01 02:31:03 INFO SparkUI: Stopped Spark web UI at http://KEREM.mshome.net:4040

25/03/01 02:31:03 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!

25/03/01 02:31:03 INFO MemoryStore: MemoryStore cleared

25/03/01 02:31:03 INFO BlockManager: BlockManager stopped

1. Conclusion

We compare the efficiency of different data formats (JSON, XML, Avro, Parquet) using Apache Spark. We focus on runtime performance and ease of processing.

In the demo comparisons made above, the basic metric was the speed at which files were read.

No storage method was used for these comparisons at the moment, we will provide a solution to this with HDFS in the continuation of the project. We will also increase the metrics and make more comprehensive comparisons.