In-class Lab 2 - HiveQL Queries

Introduction

Apache Hive is a Big Data tool that allows analyst to query large datasets using an easy to understand query language HiveQL which in many ways mimics SQL queries. When a HiveQL query runs, behind the scenes it is converted to a Mapreduce proram and thus able to use the computational resources of a cluster.

The aim of this lab is to demonstrate your understanding and skills in big data analysis using Apache Hive. You are required to get a dataset from Kaggle, load it into Hive, write queries showing useful insights.

1 Tasks

1. Review the videos on Apache Hive posted on the course shell and make sure to go over the hands-on exercises.

2. Retrieve a dataset from Kaggle that interests you and that you consider suitable for this lab.

3. Load your chosen dataset into Linux and review the dataset to get familar with its schema (the column names and types). Make sure that the data is what you expect (do a head command for example). Note that the datasets must be downloaded and then uploaded into GCP/Cloudera. There is generally no direct link where you can use the wget command as we did in class.

4. Load the dataset into a location in HDFS. Note that this location should only have the dataset and no other files and folders.

5. Start Hive and create a database. Then create a table in the database using the dataset schema and load it using the LOCATION attribute as discussed in class.

6. Based on the dataset, write some HiveQL queries that provide useful insights into the dataset. Be sure to avoid generic queries such as selecting all rows or simple filters, the query should be insightful (usually insightful queries use grouping and aggregate functions).

2 Deliverables

1. A video recording where you run the Hive queries. During the video, explain your actions, thought process, and any interesting observations. Note that silent demonstrations or simply going over the report in the video will not be acceptable.

2. A report which discusses what dataset and the insights that you gained based on your queries. Include in the report all screenshots, showing the steps you followed to load the dataset, the queries you ran, the results you obtained, and the insights you gained. The report should be clear, concise, and easy to read.

3 Marking Scheme

Your submission will be evaluated based on the following criteria:

- Video (40%): Emphasis will be on the clarity of your explanations and the effectiveness of your live demos.

  There will be no marks for the assignment without a video recording. Video recordings where there is no audio (speaking in English) will not be accepted.

- Code (40%): You will be judged on the quality, correctness, and efficiency of your Hive queries and Spark RDD MapReduce programs and your explanations of the code.

- Report (20%): Importance will be given to how well-organized and understandable your report is, and how effectively it communicates your insights.

Pitfalls

Avoid the following pitfalls.

- If something doesn't work or you can't get everything working it's not a big deal, it happens in life, just state it in your video and explain what you think is wrong. Do not try to hide details. An honest effort gets most of the marks.

- "Mimicking" where you just ask for someone's assignment and just mimic their results is considered an academic misconduct and will result in a mark

of zero (for both the mimicker and supplier). Everyone must work on assignments independently. You can discuss solutions in general, but just running code that you have obtained from someone else is not considered discussion.

- Requirements for assignment must be followed as clearly stated. Changing requirements will result in a mark of zero.
- Late penalty is 10% per day, please respect course policies on late penalties and do not ask for extensions outside the scope discussed in first class.