Final project

Objectives:

1. Having a good grasp on the concepts taught in this course.
2. Revision and practical applications of using Big data technologies.

Basic instructions:

1. Each team will need to submit one report along with the code in a zipped folder (name: **datatset_<dataset-number>_<team_name>.zip**) in the moodle and show a demo of their project. The deadline for the Moodle submission is **9th May, 23:59 p.m**.
2. Once the report is submitted, based on your and TAs availability, you need to show a live demo of your project, which will take around 45 minutes.

General instructions:

Batch computation:

For this task, your team needs to perform the pre-processing and other necessary tasks in a DataProc Cluster using Spark. Your team needs to generate the model and then save the best model for evaluation. You will need to write code for evaluation that can access this model and make predictions for real-time evaluation. You need to run this evaluation code in the demo/presentation. **Note: We don't require you to perform the training during the demo/presentation**.

Overview:

1. Use a DataProc Cluster and submit a Spark job for data pre-processing and model training.
   Note: For pre-processing, you can find this link helpful to get a better understanding of data. It is about working with Jupyter Notebook on Google Cloud Platform.
2. Store the model in your GCS Bucket.

Real-time computation:

This task will require your team to show us a working demo of how you can use the trained model to perform real-time predictions of test data streaming to Kafka. The basic idea is that one of the members will be feeding the data into Kafka cluster, another member needs to use the data in a Spark cluster to generate real time predictions. You also need to print out the accuracy and F1-score obtained for each batch.

Overview:

1. Stream the test data stored on the GCS bucket into Kafka.
2. Use Spark Streaming to read the data and make real-time predictions using your stored model.

Note: The test data will be provided to you as a GCP bucket address during the demo. So, write your code in a way that can take in a GCP bucket address containing the test data and produce real-time predictions along with accuracy and F1-score for each batch. You can assume that test data will exactly have the same structure as training data (will contain the label as well, however we will verify that your code doesn't simply use that label as the prediction) .

Task-specific instructions:

Dataset-specific instructions will be sent to all the team-members by mail.

Evaluation:

Depending on the presentation, demo and report, your team will be evaluated and you will receive a score for the same. The parameters for the evaluation are:

a. Batch computation:
   i. Data exploration and pre-processing
   ii. Feature Engineering
b. Real-time computation:
   i. Publish/Subscribe code
   ii. Message parsing
   iii. Relative* Real-time accuracy/F1-score

*Relative means with respect to other teams which have been assigned the same dataset

Score distribution:

- Batch computation - 7.5 marks
- Real-time computation - 7.5 marks
- Demo - 5 marks

Report:

The basic outline of the report should be as follows:

1. Objective
2. Pre-processing (with screenshots)
3. Performance of the best model (with visualizations) and Inferences
4. Real-time computation (with screenshots) and latency of processing each window
5. Conclusion