Big Data Systems and Architectures - Spark Assignment 2021

Exploring International Flights in 2017 Data – Task 3

The requested task is to investigate if it is possible to train a linear regression model that could predict the departure delay a flight may have by using, as input, its origin (column “ORIGIN”), its airways (column “CARRIER”), and its departure time (column “DEP_TIME”). Before starting the implementation, it is required to install numpy, spark, findspark and pyspark. Some packages such as SparkContext, SparkSession, SQLContext, DataFrameReader, StringType, Window, RegressionEvaluator, LinearRegression, DenseVector, OneHotEncoder, VectorAssembler should also be imported. It is important to run findspark.init(), so as to initialize the findspark.

A new entry point to Spark SQL is created by the use of SparkSession.builder with appName("FlightsAssignment"), as shown below:

After creating the temporary view, the data are loaded to a variable named `flights_data` by the use of spark.read() function with different options, such as that the dataset contains header and it is in comma delimited format. From this dataset we keep only the requested columns DEP_DELAY, ORIGIN, CARRIER and DEP_TIME.

Firstly, we have to transform the DEP_TIME column, as its type is integer and cannot be properly handled. We have to convert it to string and padded with zeros in the beginning as there are values with 3 digits. Each time should have length 4 digits. After that substring is used to keep only the first two digits, which indicate the hour. The outcome is stored in a new column

named DEP_TIME_HOUR. One final transformation in time is the replacement of DEP_TIME_HOUR=24 to 00, as it is indicating the midnight.

The same transformations we implemented in Task 2 about the outliers, also implemented here, in order to omit the airports/airways belonging in the lowest 1% percentile.

One hot encoding is used to prepare the feature columns for the pipeline. For each column ORIGIN, CARRIER and DEP_TIME_HOUR is created one string indexer and one hot encoder named ORIGIN_ENCODED, CARRIER_ENCODED, DEP_TIME_HOUR_ENCODED respectively. The output is stored in FEATURES. All the above string indexers, one hot encoders and the FEATURES are embedded in the same pipeline. This pipeline is used to fit and transform the dataset.

Subsequently, we split the dataset in training and test with ration 70-30. The Linear Regression is used to fit the model by using the training dataset. We use the FEATURES and the DEP_DELAY as our response. After fitting the model, the summary and the RMSE are calculated.

Finally, having prepared our model with the training dataset, we move on making the predictions on the test dataset. Our final output containing the predictions, and the feature vectors are calculated and the first 10 records are shown. Apart from the above predictions, the accuracy of the model is shown through the metrics R2 and RMSE.