BIG DATA PROJECT REPORT

Machine Learning with Spark Streaming

Sentimental Analysis Dataset
BD_036_186_425_462

Aishwarya B S	PES1UG19CS036
Harshitha C	PES1UG19CS186
Samhitha Harish	PES1UG19CS425
Shreya Srinivas	PES1UG19CS462

Our dataset is the sentimental analysis. Which basically involves preprocessing the streamed data in batches and classifying using models we have used in our context. We have used decision trees and random forest classifiers as our 2 classification algorithms.

Starting from the streaming part of it we created a streamterminal.py file that takes in the argument which takes in the given batchsize and streams it according to the number given. Argparse() module helps in taking the values and parsing through. Time.sleep(2) indicates that it takes a timestamp of 2 seconds to process every batch

Preprocessing: The initial part of the data analysis starts with preprocessing. This part focuses to create basis for building our model. We have used 2 transformations here by importing tokenize and IF-TDF modules from the pyspark MLLIB features. We have localized the names “tweetcol” and “sentimentcol” for easier preprocessing. We tokenize the words then use idf for inverse column matrix which makes it easier to calculate the transpose of the words for it to be generated according to our criteria to convert it into numerical values as models work well with numerical values.

CLASSIFIER ALGORITHMS:

1) Random Forest:

The Random forest or Random Decision Forest is a supervised Machine learning algorithm used for classification, regression, and other tasks using decision trees. The Random forest classifier creates a set of decision trees from a randomly selected subset of the training set. It is basically a set of decision trees (DT) from a randomly selected subset of the training set and then collects the votes from different decision trees to decide the final prediction. The libraries used are sklearn.metricsv , accuracy score sklearn.model selection ,(train_test_split) sklearn.preprocessing (OneHotEncoder).

2) Decision Tree:

Decision tree is a type of supervised learning algorithm that can be used for both regression and classification problems. The algorithm uses training data to create rules that can be represented by a tree structure. Like any other tree representation, it has a root node, internal nodes, and leaf nodes. The internal node represents condition on attributes, the branches represent the results of the condition and the leaf node represents the class label. The libraries used are numpy, itertools combinations sklearn.metrics accuracy_score.

DATA BEFORE AND AFTER PREPROCESSING FOR 1 STREAMED BATCH