Spark Streaming For Machine Learning Project Report

Dataset chosen - Spam Classification
Each record consists of 3 features - the subject, the email content and the label
Each email is one of 2 classes, spam or ham
30k examples in train and 3k in test

Design details -

Data processing is sorting through massive knowledge sets to spot patterns and establish relationships to unravel issues through data analysis is data processing. There are a variety of major data processing techniques that are developing and victimization in data mining comes recently as well as association, classification, clustering, prediction, serial patterns and call tree.

The Model pipeline includes Tokenization(with regular expression), Remove Stop Words, Count vectors, (“document-term vectors”),

String Indexing - encodes a string column of labels to a column of label indices.

After partitioning the data into training and testing sets,
For model training and evaluation we have used:

1) **Logistic Regression** - It is a statistical analysis method used to predict a data value based on prior observations of a data set. We are performing this regression using Count Vector Features and TF-IDF Features.
2) **Naive Bayes** - These classifiers are a collection of classification algorithms based on Bayes’ Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.

Surface level implementation details about each unit -

1. **Streaming Data -**
   We used spark streaming to receive data streamed from stream.py file.
   Using socketTextStream we take in the data as Dstreams and convert it to RDD.

2. **Preprocessing -**
   As a preprocessing step we computed the length of the message to find if any outliers exist and dropped them.
   Then removing data messages with special characters and removing stop words.
   Then converting the label column that is spam/ham to categorical

3. **Building and testing the model -**
   To build the model we used classifiers like Linear regression, logistic regression and Naive Bayes.
   These classifiers are trained on features computed in preprocessing step with 70-30 split.
   Out of all three naive bayes gave better accuracy.

Reason behind design decisions -

We decided on the mentioned designs as they helped us provide the appropriate results from preprocessing the features and really good accuracy with the models used on the testing data which shows improved performance.

Takeaway from the project -

We learnt how real world big data projects are structured and built.

We learnt how to preprocess live data and classify them into spam or ham.