# **UE19CS322: BIG DATA PROJECT**

MACHINE LEARNING USING SPARK STREAMING

|  |  |
| --- | --- |
| **NAME:** | **SRN:** |
| Rishab K.S | PES1UG19CS384 |
| Pranav M.R | PES1UG19CS340 |
| K K Tarun Kumar | PES1UG19CS222 |
| Rakeshgowda D S | PES1UG19CS376 |

**Design Details:**

* This project is aimed at understanding and analysing machine learning tasks on large data streams. We have chosen “Sentiment140” dataset.
* The csv dataset is encoded into json and then it is streamed via TCP socket. Then the stream is received using readstream function with the source as socket.
* After receiving a batch of data, it is Pre-Processed.
* After Pre-Processing we are building models for clustering and classification

Classification models – SGD Classifier, PassiveAggresive Classifier

Clustering Model – Mini-batch K-Means

* The aforementioned models are trained on each batch of data.
* After training the models on the whole dataset we use them to predict the test batches.
* F1 score, precision, recall and accuracy are calculated.

**Surface Level Implementation Details:**

* The Streaming Dataframe is converted to Static Dataframe using foreachBatch() operation which allows us to use any functions on the dataframe.
* Pre-Processing: The batch of data is filtered to remove special characters, digits and twitter handles.
* For Classification Hashingvectorizer is used which maps the string to feature integer index.
* For Classification we are using SGD Classifier and PassiveAggressive classifier from the scikit learn library as these classifiers support incremental learning. We are performing incremental learning by using partial\_fit() function on the training dataframe.
* For clustering TF-IDF Vectorizer is used which converts string to a matrix TF-IDF features.
* The feature decomposition is performed using Truncated SVD to obtain the principle features.
* The output of the Truncated SVD is normalized and used to incrementally train the Mini-batch K-Means model on each batch of data.

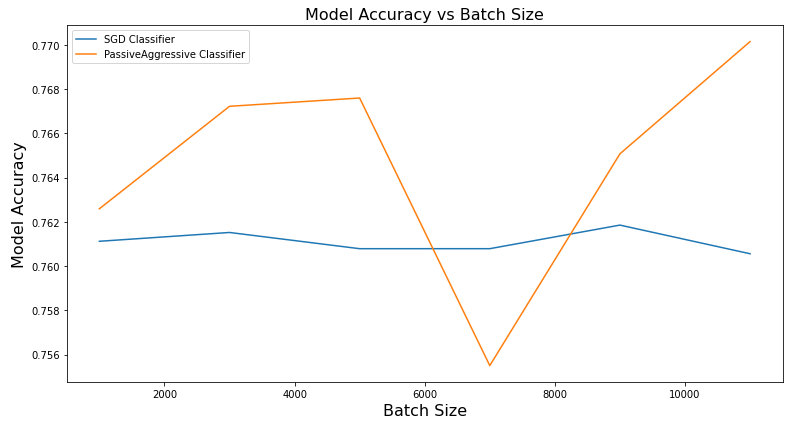
**Motive behind the Design Decisions:**

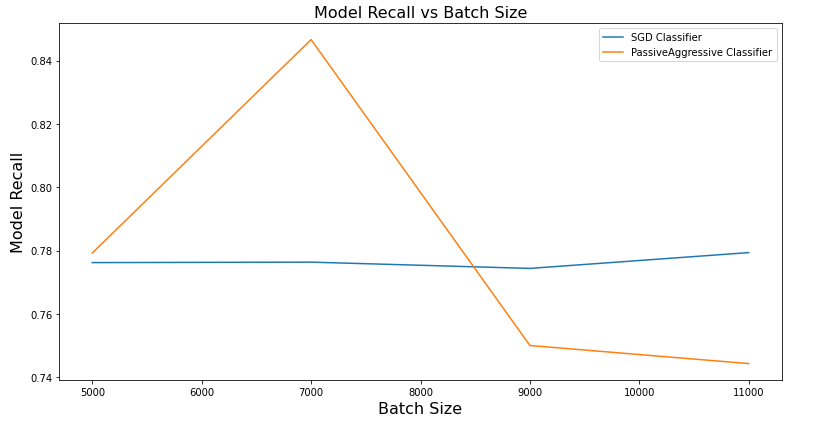
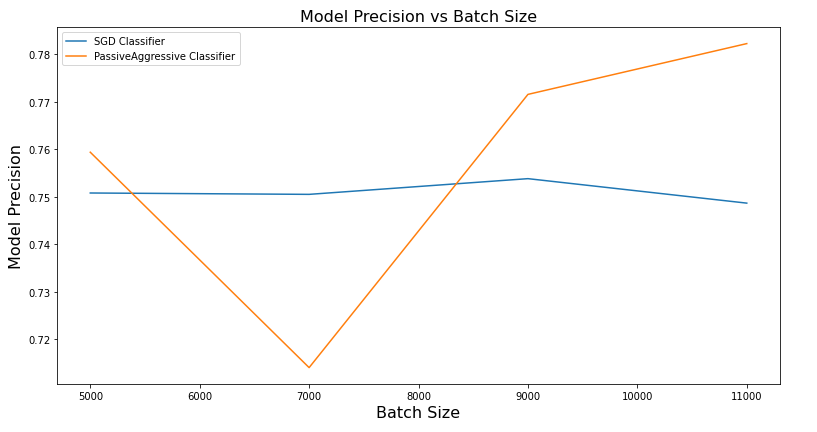
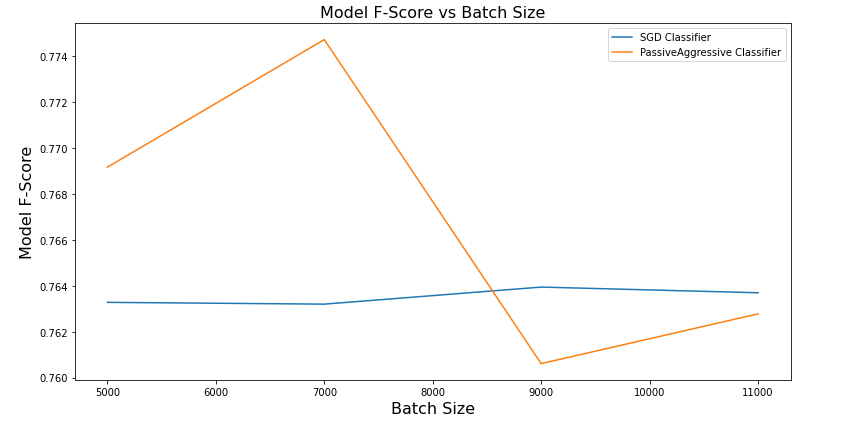
* SGD and PassiveAggressive models are used since they provide higher accuracy compared to Naïve Bayes learners.
* Mini-Batch K-means supports incremental clustering.
* Hashing Vectorizer is used since the complete dataset is unavailable.

**Knowledge acquired from the project:**

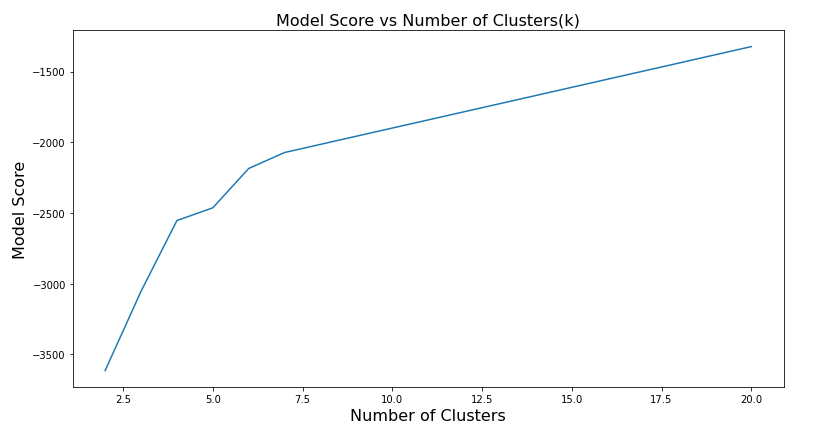
* Working with streaming data using pyspark
* Incremental Machine learning models
* Effect of varying batch sizes on model metrics

**Analysis:**





* Accuracy, Precision, Recall and F1 Score have less variation for SGD Classifier for all batch sizes.
* The PassiveAggessive Classifier:
  + While Accuracy and precision dips to an all-time low at batch size = 7000 and later increases as the batch sizes increase.
  + F1 Score and Recall peaks at batch\_size = 7000 and later reduces as the batch sizes increase



* Elbow Method is used to find the optimal K-Value (5-7).