Title: Ensemble based Twitter Sentiment Analysis on Big data using Apache Spark.

Title: Framework for Sentiment Analysis on Twitter data using Apache Spark.

**Abstract:**

The development of social media platforms like Twitter and Facebook has been a revolution in the history of mankind. Social media web applications played a great role in getting the world together. There are platforms like twitter where people can share their opinion about an issue which can be seen and heard by the entire world. Tweet count on twitter for a day is about 500 million tweets. By analyzing these tweets on sentiment basis we can understand the public opinion regarding a particular matter. The thesis developed will perform a sentiment analysis on live twitter data using an ensemble algorithm to detect the nature of tweets (tweets are positive or negative). Implementing Sentiment analysis on twitter data is quite a difficult task as it involves operating on a large amount of diverse data. This thesis involves the implementation of the apache spark framework for working with a huge chunk of data, that is big data. Sentiment analysis is a machine learning-based method. This thesis aims to examine various classification algorithm and their impact on live twitter data.

Table of Content:

[Introduction 3](#_Toc46236015)

[Background: 3](#_Toc46236016)

[Sentiment Analysis: 4](#_Toc46236017)

[Big Data Analytics: 4](#_Toc46236018)

[Motivation: 5](#_Toc46236019)

[Purpose: 5](#_Toc46236020)

[Structure of the document: 5](#_Toc46236021)

[Literature Review 7](#_Toc46236022)

[Introduction 7](#_Toc46236023)

[Lexicon Based Sentiment Analysis vs Machine Learning-based Sentiment Analysis: 8](#_Toc46236024)

[Literature Review Discussion: 13](#_Toc46236025)

[Literature review Conclusion: 14](#_Toc46236026)

[Research Question: 14](#_Toc46236027)

[Proposed Methodology: 16](#_Toc46236028)

[Algorithms: 16](#_Toc46236029)

[Why use Decision Trees? 19](#_Toc46236030)

[System Architecture: 21](#_Toc46236031)

[Implementation: 23](#_Toc46236032)

[Libraries Used: 23](#_Toc46236033)

[Flow Diagrams: 24](#_Toc46236034)

[What is Streaming? 25](#_Toc46236035)

[What is Spark Streaming? 25](#_Toc46236036)

[Proposed Evaluation: 27](#_Toc46236037)

[Confusion Matrix: 27](#_Toc46236038)

[AUC-ROC Curve: 28](#_Toc46236039)

[References 31](#_Toc46236040)

# Introduction

## Background:

The development of the Internet has been a revolution in the sector of technology and the world. Some way or another internet has infiltrated the life of every human being. The Internet has brought the world together, anything happening in one part of the world is easily known by the people around the globe. Many social media sites and applications have come into existence, using which people can express and share their views regarding an incident. One of the famous social media applications in the current age is Twitter. Twitter is an internet application where a different person can write their views about something for the whole world to see it. Various powerful and famous people use Twitter to express their views on a particular topic and the public can easily view their tweets. A common man can easily get influenced by the tweets of someone who is a famous person. A place where people can say anything and can be heard by all needs to be free of toxicity. A toxic comment on twitter by a powerful or famous person can create panic among the common people. It becomes very important to keep social media platforms like twitter free of toxic or hatred comments to keep society safe.

(Anon., n.d.) Every sec on average about 6000 tweets are getting twitted on twitter. Tweet count for a day is about 500 million tweets. So it is not possible to view every tweet by a personal and delete the tweet if it is not appropriate. Hence, we require a smart system that automatically detects the tweet by using NLP that analyses each word in the tweet by performing operations of word bagging. By using the concept of Sentiment analysis a sentiment score for each word is generated for a review. If the overall polarity of sentiment score for a review is found to be greater than 0 than the review is termed as positive.

By using sentiment analysis for each comment the system figures out whether the comment is toxic or not. If the sentiment of a comment is found to be negative then it will be deleted automatically.

As we have discussed earlier tweet count for a day on Twitter can go up to 500 million approximately. 500 tweets a day, this is a large amount of data. Processing this much amount of data in Machine Learning is not very efficient.

Operating on live streaming data is a huge challenge due to its size. Due to the continuous growth in data new frameworks came into existence like Apache Spark, Hadoop, Apache Storm, and distributed data storage like HBase and HDFS. These frameworks are designed to operate specially on a huge chunk of data. Libraries like Spark’s MLlib came into existence that can perform machine learning operations on a large amount of data.

This thesis aims at developing a sentiment analysis based machine learning system using Spark’s MLlib library. This system will operate on live twitter data, the streaming and handling of live data will be done with the help of Apache Spark framework. The classifier will classify the tweets as positive or negative. That is, whether a tweet is a hate speech or not. The form of learning used here is Supervised form of Learning.

## Sentiment Analysis:

Sentiment analysis is the analysis of the input text and its classification of emotion like positive, negative, or neutral based on the words used in the sentence using the text analyzing techniques. Sentiment analysis is used by a great number of companies for understanding the feedback for their service or products from the customers. It helps the company to gather useful data regarding their work.

There are various types of sentiment analysis. The type which this project will make use of is Fine-grained Sentiment Analysis. Which will classify a tweet into one of these categories

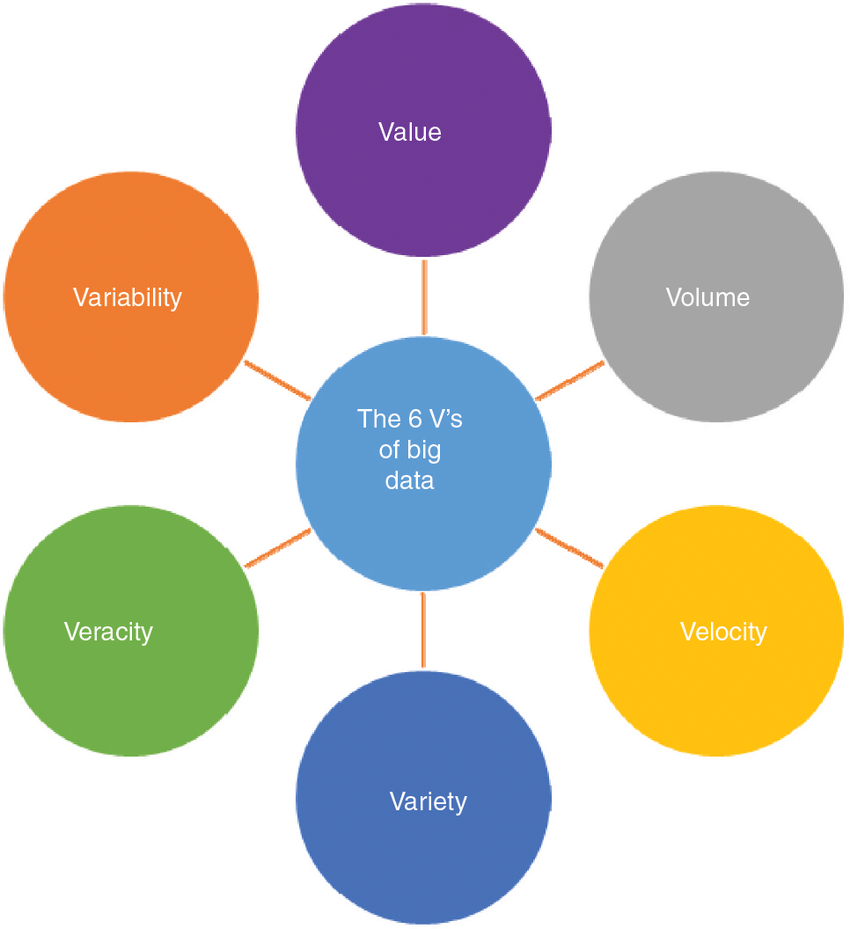
* Positive
* Negative

## Big Data Analytics:

For processing this large amount of data the thesis will make use of Big data concepts. Big data generally refers to a large amount of data in a structured, unstructured, or semi-structured format. This large amount of data is mined and processed using big data concepts. After processing and cleaning of this chunk of data. The resulted data can be used as a dataset for Machine Learning algorithms. Doug Laney was first to identify these characteristics in his article in 2000. The categorization of big data is usually done with the help of 3V concepts. The 3V concept includes.

* The large volume of the data present (Volume)
* The different variety in the present data. (Variety)
* The velocity or speed at which the data is produced and processed. (Velocity)

Overall there 6 foundation pillar for any big-data architecture to be successful:



(Ristevski, 2018)6 v’s of big data

## Motivation:

The Twitter post is very short, each post generally consists of max 140 characters. Hence, the resulting post can be more expressive and emotional. Analyzing tweets and discovering their sentiment is considered as a challenge in micro-blogging area. Analyzing this large amount of data on twitter in the form of tweets can be a great challenge for Machine Learning and Big Data both. This can help me to learn how Big data can go hand in hand with Machine Learning. With the help of sentiment analysis, we can detect any inappropriate language/hate speeches used and the tweet can be deleted.

## Purpose:

The experiments conducted up till now involves the use of both Lexicon based and Machine learning-based methods to perform sentiment analysis on twitter data. In the literature section, I have discussed how machine learning outperforms the lexicon-based method in terms of accuracy. The research conducted up until now have used various machine learning algorithms and came up with different results. But the accuracy obtained up till now is not very promising. Most of the researches done up till now has made use of general algorithms rather than using ensemble-based algorithms. I have discussed the ensemble-based algorithm in the next section. The use of an ensemble-based algorithm should increase the overall accuracy of the system as compared to general-purpose algorithms as multiple homogeneous classifiers are used in the case of an ensemble to reduce the overall error rate. Also, most of the research conducted until now have utilized the Hadoop framework for working with Big data. This project makes use of Apache Spark which is intended to boost the overall performance of the system as compared to the Hadoop framework while streaming live twitter data.

## Structure of the document:

* **Literature Review:**

This section will discuss the research paper studied up until now by me to come up with this project. It will discuss the past methodologies used by different researchers to perform sentiment analysis on big data. An analysis will be done on the advantages and disadvantages of those methodologies. At last, the section will consist of the entire literature review conclusion.

* **Proposed Methodology:**

This section of the proposal will discuss the algorithms that will be used for the development of this project. A step by step implementation of the algorithms will be discussed. At last, the section will also consist of the system architecture of the project.

* **Implementation:**

This section will discuss different libraries that will be used for the implementation of this project. It will consist of a flow diagram that will explain the entire implementation of the project in detail.

* **Proposed Evaluation:**

This section will consist of the evaluation measures using which I am going to evaluate the results obtained by the implementation of the project. Each evaluation measure will be discussed in detail.

* **Conclusion:**

This section will conclude the entire proposal. A concluding statement will be given here regarding the implementation of the project.

* **Reference:**

This section will consist of the list of papers, articles, etc used by me for coming up with this project.

# Literature Review

## Introduction

This section will consist of a detailed discussion of all the literature studied for the development of this thesis. The section will discuss different methods used by different researchers related to sentiment analysis. A discussion will be done on different algorithms used in various research papers based on sentiment analysis.

(Gupta, 2018 ) Sentiment analysis in the field of ML is analyzing the given text and identifying the sentiment of the text from the keywords of the text source. It is based on the contextual mining of the text which is performed on the source text data to extract subjective information from the source provided. Sentiment analysis is used in business for a variety of purposes like chatbot and analyzing the review for a product.

(Khan & Malviya, 2020) Sentiment analysis also known as opinion mining, consists of steps like extracting the sentiment from the source text, classification of the extracted sentiments, finally summarizing it, and detecting any present spams. It tries to understand the emotion and the attitude of the people regarding a product, individual, or company from the review written by the people. With the recent advancements in the sector of deep learning, the ability of the algorithms to analyze text semantics has improved considerably. Sentiment analysis is used by several companies across the world for a better understanding of their product in the market. It has become a great tool for any company to understand the review from the customers to a better extent:

1. Key aspects of a brand’s product and service that customers care about.
2. The underlying user intentions and reactions for a particular service.

Overall there are two ways to go for sentiment analysis(Kolchyna, et al., 2015):

1. Lexicon Based
2. Machine Learning-Based

## Lexicon Based Sentiment Analysis vs Machine Learning-based Sentiment Analysis:

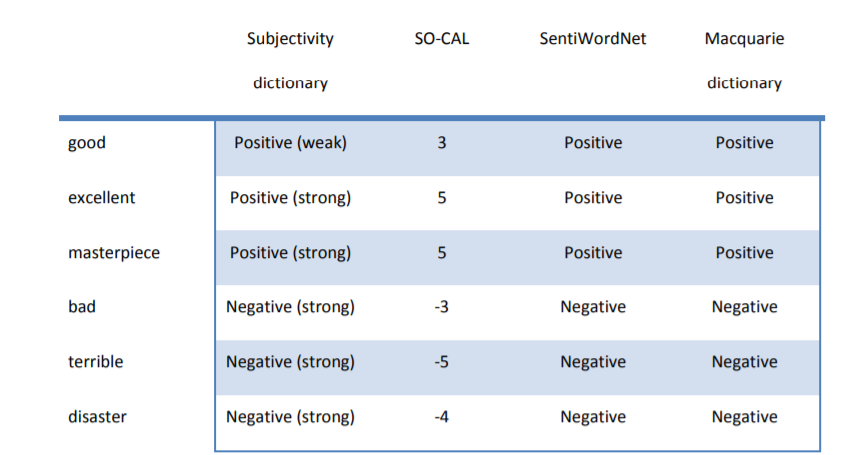
Lexicon based and machine learning are two approaches used for semantic analysis both have their advantages and disadvantages.

**Lexicon based**(Taboada, 2016)**:**Lexicon based analysis consists of a pre-build dictionary that consists of all the words along with their sentiments defined. Words from input text are compared with the dictionary and if the match is found then the sentiment of the word from the dictionary is extracted. This is working on Lexicon based sentiment analysis. Dictionaries use to contain words that are listed as positive and negative. Many dictionaries in lexicon analysis also contain a more descriptive analysis of words like how much positive or negative the word is.

Example of a subjective dictionary of (Wiebe et al 2004) and ( Wilson et al 2009).

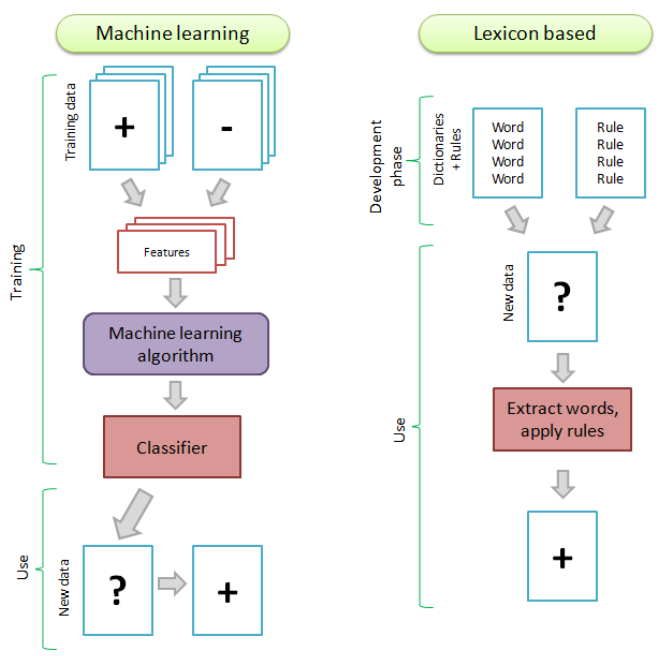
* Positive – strong absolve, accolade, altruistic
* Positive – weak accept, abundance, affluent
* Neutral - accentuate, alliance, alert
* Negative – weak abolish, addiction, alienated
* Negative – strong abuse, abomination, afraid

(T & Hoffman, 2005)has conducted a study on phrase-level for identifying neutrality and polarity of the phrase.



Sample semantic orientation values for different dictionaries (Taboada, 2016)

**Machine Learning-based:**(Anon., n.d.)Here a dataset is first created. The dataset consists of a list of the input text. Consider the example of a product review on a company website. Large numbers of customers write their reviews. Play store application reviews can also be considered an example. A large number of people write reviews every day. The dataset is first created of these reviews. Then a model is build based on some machine learning algorithms. This model is trained on the dataset created. Once the model is trained it is then tested.



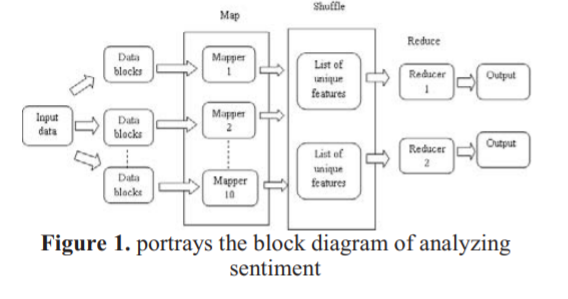
ML vs Lexicon based approach (Taboada, 2016)

(Woldemariam , 2016)performed a comparative study of Lexicon based analysis and machine learning-based analysis. As there is a continuous increase in access to the internet around the world. A huge chunk of data gets generated every minute which is both in structured and unstructured form. To analyze data at such a large scale and figure out the semantics of it is a very challenging task. As a result, it is difficult to retrieve the right type of media to satisfy multimedia content consumers. Test cases for the research were obtained from Zooniverse<https://www.zooniverse.org>. It’s an online platform where the public posts their review regarding the animal pictures. (Woldemariam , 2016)ran sentiment analysis on these reviews using both lexicon and machine learning-based approaches. The Hadoop platform was used for the development of the model. The algorithm used for machine learning was Recursive Neural Tensor Network (RNTN)(Socher, et al., 2013). From the results obtained by (Woldemariam , 2016). He claimed that RNTN outperformed lexicon-based by 9.88% accuracy.

Several researchers like (Vaithyanathan, et al., 2010) and (Vinodhini & Chandrasekaran, 2012) also performed a comparative study between lexicon-based and Machine Learning based analysis and found that in terms of accuracy ML-based analysis gives better results than lexicon-based sentiment analysis.

**Building Twitter-based Sentiment Analysis using Hadoop Framework:**

(Khan & Malviya, 2020)developed a model for twitter-based sentiment analysis.



Block Diagram for the model (Khan & Malviya, 2020)

The sentiment analysis is performed using the Hadoop Framework and deep learning classifier. First, the dataset is divided into multiple clusters (Data blocks). Then important features from all these clusters are extracted. Features like emoticons, all-caps, hashtags, punctuations are extracted. This extraction is performed in the Mapper Phase of the given block diagram. These obtained features are transferred to the shuffle phase for further evaluations. In the shuffle phase list of unique features is created. This list of unique features is then transferred to the reduce phase. The reducer phase than maps the data into positive and negative reviews, using RNN. RNN (Recurrent Neural Network) is a supervised Machine Learning algorithm. A target input pair is provided to the RNN along with a database for training purposes. (Shen & Wang, 2008)RNN (Recurrent Neural Network) comprises three layers of the input layer, recurrent hidden layer, and the output layer. Input is given to the algorithm consist of a sequence vector for a particular period ‘t’.

**Working of RNN for a particular Review:**

For a particular review, the review is first broken down into words and then the stop words are removed. One Hot vector is created for each word, these one-hot vectors are multiplied by the embedding matrix, which results in the creation of multiple embedding vectors. These embedding vectors are fed to the RNN. The job of the RNN classifier is to predict the sentiment for the given review using the embedding vectors fed to it. The results obtained by the RNN classifier with the Hadoop framework has an accuracy of about 90%.

(Rodrigues, et al., 2014)developed a model for analyzing sentiment reviews for twitter. They used Map-reduce techniques for extracting the sentiments of tweets. The approach of (Rodrigues, et al., 2014) was completely focused on obtaining the speed of performance rather than the accuracy. They used open nlp for various purposes like removing stop words, converting unstructured data to structured data, converting emoticon symbols to actual words.

Stop Words: Stop words consist of words like a, an, of, for, etc. These kinds of words do not portray any kind of emotion, hence has no use in feature extraction

Unstructured to Structured Conversion: Twitter reviews mostly consist of words that generally short forms or unconstructed. For example, ‘awesome’ would be written as ‘awsm’ etc

Emoticons: Emoticons can include smiley faces, or crying face different kinds of emoji’s, etc. These emoji’s need to be converted to proper words for extracting sentiment details.

**Increasing the speed and real-time data processing:** Twitter API was used for gathering reviews data from twitter. A Sentiment dictionary was created which consists of words with all possible usage in the language. These words were obtained from sentiment wordnet. For example, the word “good” can be used in various manners in different forms of language. For example, someone can use the word “better”, “nice” etc. All these words have positive equal sentiment. So overall sentiment is stored so time is not wasted on searching for these words.

The map-reduce algorithm is used for faster real-time processing of data. The data is re-organized in the cluster format of Hadoop architecture and all the operations are performed in a parallel manner for faster execution of the code. The accuracy obtained for this model was about 72%.

(Kamal, et al., 2017)developed a model for sentiment analysis of twitter data using Hadoop and Spring XD. (Kamal, et al., 2017)te research was based on the research of (Rodrigues, et al., 2014) and (Pang & Lee, 2008) who performed sentiment analysis using Hadoop and Machine learning. (Rodrigues, et al., 2014)te approach was completely focused on obtaining a higher speed. The real-time gathering of data from twitter was done using Spring XD. Spring XD is one of the best open-source tools. Live data is extracted from the twitter using Spring XD in real-time. A particular limit is set. This limit consist of the number of tweets Hadoop can handle for a particular operation. The system is continuously receiving data using Spring XD from twitter. Once the limit is reached the receiving of the data stops and the Hadoop begins its operations. The data is transferred to the HDFS system. The result obtained from this research contains an accuracy of about 75% which was better than (Rodrigues, et al., 2014).

**Using Apache Spark for Sentiment Analysis:**

(Baltas, et al., 2017)developed a system for sentiment analysis on twitter data using Apache Spark. (S, et al., 2002)made use of machine learning algorithms like Naïve Bayes, SVM, and maximum entropy to analyze the sentiment for movie review. Based on their research (Baltas, et al., 2017) developed their system using algorithms like Naïve Bayes and Logistic regression for twitter data analysis. Instead of going with Hadoop like (S, et al., 2002), (Baltas, et al., 2017) went with the Apache-spark framework for their model. The algorithms used for implementation were Logistic regression, Decision trees, Naïve Bayes. One of the major advantages of using the Spark framework is its ability to deal with multiple algorithms.MLlib library was made use for the implementation of machine learning.

**Classification Steps:**

Binary Classification: For the training phase, the dataset is first classified as Positive or Negative data. Multiple segments are generated from the dataset. Now for each of these segments following classification was done:

Unigrams**:** Frequency of words occurring in the source text.

Bigram: Frequency of two words occurring together in the number of different tweets.

Trigrams: Consist of frequency if three words.

Username: A flag which will indicate the presence of any username.

HashTag: A flag which will indicate the presence of any hash.

(Baltas, et al., 2017)from their research concluded that Naïve Bayes gives better results as compared to Logistic regression and Decision tree in terms of performance. From their results, they conclude that the size of the dataset plays a vital role in the accuracy of Naïve Bayes. The size of the dataset change generates a change in the F-measure value of the model. While they also observed that Logistic regression and Decision tree are not heavily affected by the data size.

## Literature Review Discussion:

### Discussion for ML vs Lexicon approach:

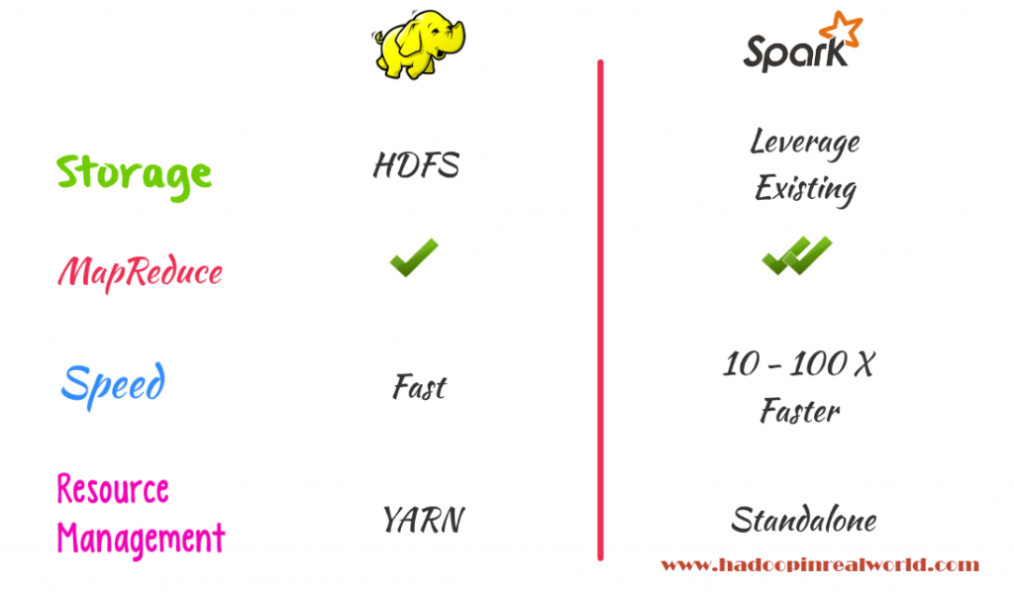
Based on results obtained by (Taboada, 2016), I decided to go with Machine Learning for this thesis. As in the case of (Taboada, 2016) Machine Learning produced the best accuracy as compared to Lexicon based analysis.

### Discussion for Map-based data cluster:

(Woldemariam , 2016)did a fine job by using the Hadoop data cluster-based method which divided the large dataset into multiple small parts and then executing them parallelly. This thesis follows a similar approach of dividing the dataset into clusters and performing a parallel execution with it.

### Hadoop vs Spark framework discussion:

As spark architecture as a great amount of advantage over Hadoop in terms of performance speed, this thesis will be using Apache-Spark and pyspark for dataset processing.



Hadoop vs spark (Anon., 2017)

Spark framework was build considering the base level framework for Hadoop following a similar principle. Hadoop is considered best for processing large amounts of data. But has a great drawback in terms of performance speed in some conditions. For example, Hadoop is not good while dealing with iterative or graph-based algorithms. Another major problem with Hadoop is that it does not store cache for intermediate operations which could generally result in faster processing. After each intermediate step Hadoop deletes the metadata, hence reducing its performance speed. Spark is built with keeping these drawbacks in mind hence overcoming these drawbacks. Spark is a very dynamic language-oriented framework with the support of multiple languages like Scala, Java Python, and R. It can operate with Hadoop or can be used as a separate platform.

From (Baltas, et al., 2017) selection of algorithm and their result section, it can be seen that Naïve Bayes performs well in terms of accuracy, while logistic regression and Decision tree had good f-measure values. For this thesis, I will use Naïve Bayes, Logistic regression, and Decision tree along with other ensemble algorithms to perform a comparative study of the algorithm.

## Literature review Conclusion:

This will help me to understand how much more effective can Naïve Bayes be when working with my system as it had the best performance (Baltas, et al., 2017).

(Baltas, et al., 2017)used spark framework to perform sentiment analysis. They used the map-reduce method which divided their dataset into multiple clusters for faster execution. Naïve Bayes algorithm produced the best results for them. I am also implementing spark framework for this thesis

I will also use an ensemble algorithm (Stacking), which will help me to understand can an ensemble algorithm perform better than traditional machine learning algorithm like Naïve Bayes.

If the ensemble algorithms produce better accuracy than the traditional Machine learning algorithm, will there be a performance increase when used with apache-spark rather than Hadoop?

## Research Question:

**To perform a comparative study of machine learning algorithms to find out which algorithm can produce the highest accuracy.**

After the application of the algorithms, the results will be compared based on various outcomes like accuracy, precision, auc-roc value, recall. All these variables measure different sections of the result. For example, auc-roc will denote the separability degree. Precision is the measure of the value of correctly classified samples from all the value which are classified positive by the algorithm. Recall is the measure of the value of positive classified samples from all the samples which are present in the dataset. These values will help perform a comparative study among algorithms.

**To see how effective can an ensemble algorithm be in sentiment analysis with Apache spark framework.**

In the case with (Baltas, et al., 2017) Naïve Bayes produced the best results compared to the other algorithms. Features based on the unigrams and large dataset proved to be a boost in the accuracy of the Naïve Bayes algorithm. A large dataset will also be used for this research along with an ensemble algorithm and spark framework architecture.

With this, the objective of the research is to improve the overall performance of the sentiment analysis done on twitter data.

**How efficiently and accurately we can understand the sentiment of a Twitter tweet using MLlib library and spark.**

Performing sentiment analysis on a dataset and performing a sentiment analysis on a huge dataset streaming live are two different tasks having separate difficulty levels. For this thesis mllib library of apache spark is used for performing sentiment analysis using an ensemble algorithm. As apache spark is faster than Hadoop, this project will make it clear whether the ensemble used from mllib library can outperform regular machine learning algorithms when used on big data using spark architecture.

**To handle Large Scale of tweet data in order to address the nature of BigData :**

There is a large volume of data generated on twitter every minute. Developing a scalable & robust sentiment prediction for this large amount of data with a better accuracy rate using the Apache Spark framework.

# Proposed Methodology:

This section will consist of a brief discussion on the algorithms that will be used for the development of this thesis. The overall architecture of the system will also be discussed in this section.

## Algorithms:

The algorithms I will be using for this thesis are:

1. Naïve Bayes
2. Logistic regression
3. Decision Tree
4. Gradient Boosting

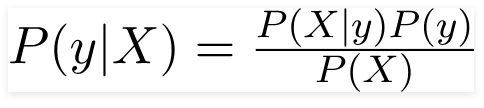
### Naïve Bayes:

Naïve Bayes algorithm follows a simple assumption of all the available predictors of being independent of one another. Naïve Bayes considers that the result of a feature for a particular model does not affect the outcome of the other features for the given model. Every feature is considered independent of one another. Hence, the algorithm is termed as Naïve.

For example, a fruit that is round, red, and has a small size can be considered as an apple. Each of these features may be dependent on one another. One feature may depend on the other, but in the case of naïve Bayes, each of these features independently contributesto the probability of fruit being an apple.

For calculating the probability for features. Bayes theorem uses a theorem called Posterior probability.

Posterior probability can be written as:

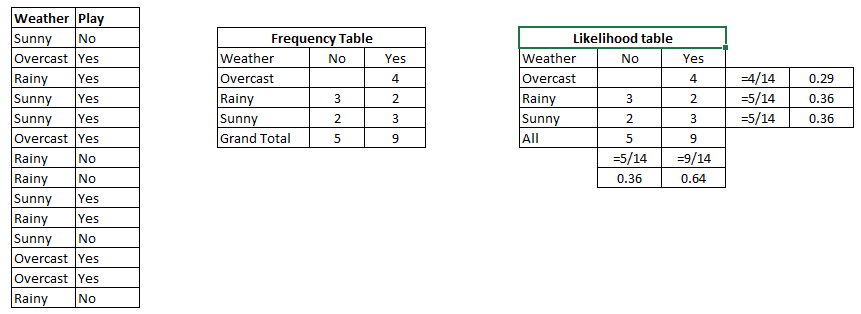


Here ‘y’ is the class variable. ‘x’ represents the set of parameter/feature.

Let’s understand it using an example. From the training data given the target is whether a game of golf will be played or not. Based on weather conditions.

Step 1: The dataset is first converted to a frequency table.

Step 2: A Likelihood table will be created by finding the probabilities of features like sunny = 0.70 and the probability of playing is 0.64.



Likelihood tables (RAY, 2017)

Step 3: Naïve Bayesian theorem will be used to calculate the posterior probability for all the existing classes. The class which will result in higher posterior probability will be considered as the best outcome.

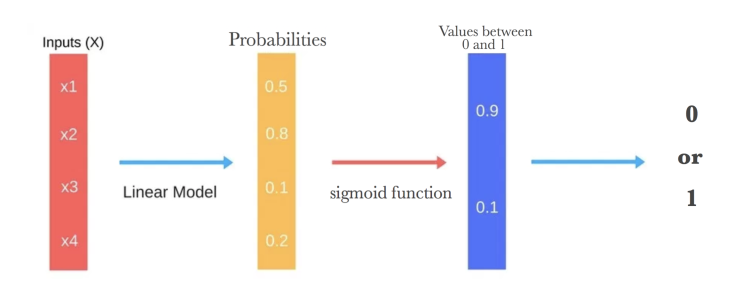
### Logistic regression:

Logistic regression is very much like linear regression. In linear regression, a threshold is used for making a classification. While in the case of logistic regression a sigmoidal function is used. Logistic regression is used when the output required is categorical data.

**How it works:**

In the case of logistic regression, the output is generally obtained using the relationship of the dependent and independent variable present. Dependent variables are variables that depend upon other variables for its value. In simple terms, dependent variables are output values.

To make predictions a sigmoidal function is used by logistic regression also known as a logistic function. This sigmoidal function transforms the probabilities into binary values for generating the results. The sigmoidal function is denoted by an S-shaped curve graph. It takes any real number as input and can classify or map it into a value ranging between 0 and 1 excluding 0 and 1. A threshold classifier is then used for transforming the results of the sigmoidal function into 1 or 0.



Steps involve in Logistic regression (Donges, 2019)

### Decision Tree:

The decision tree algorithm falls under a supervised learning algorithm. It can perform both classification and regression. Decision tree as names suggest has a tree-based structure where the internal nodes represent features, decision rules are represented using branches and finally, the leaf node represents the result or the final prediction.

Working:

The decision tree divides the given dataset into multiple smaller sets based on the features available. Each set is solved independently and the final result is the outcome of all the decisions made on these divided sets.



(Anon., n.d.) Decision tree

Algorithm:

* **Step-1:** The dataset is considered as a root node. Begin the tree from the root node. Consider this root node as (R).
* **Step-2:** With the help of **(ASM) Attribute Selection Measure** explores the best attribute available.
* **Step-3:** R is divided into multiple subsets holding the value of the best available attributes.
* **Step-4:** Generate the decision tree node, with the best possible attribute.
* **Step-5:** Iteratively keep on generating new decision trees with the help of the subsets created in step-3. This process must be continued until a final stage is obtained at which the nodes cannot be further classified. This final node should be called a leaf node.

## Why use Decision Trees?

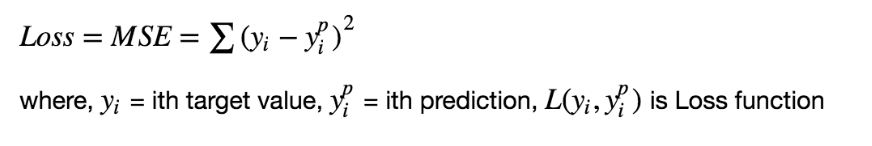
Two basic reasons are:

* Decision trees usually tend to mimic the core concept of human thinking which involves multiple if and else, which makes it easier to understand.
* As the decision tree tends to have a tree-like structure the logic behind them could be understood very easily.

### Gradient Boosting:

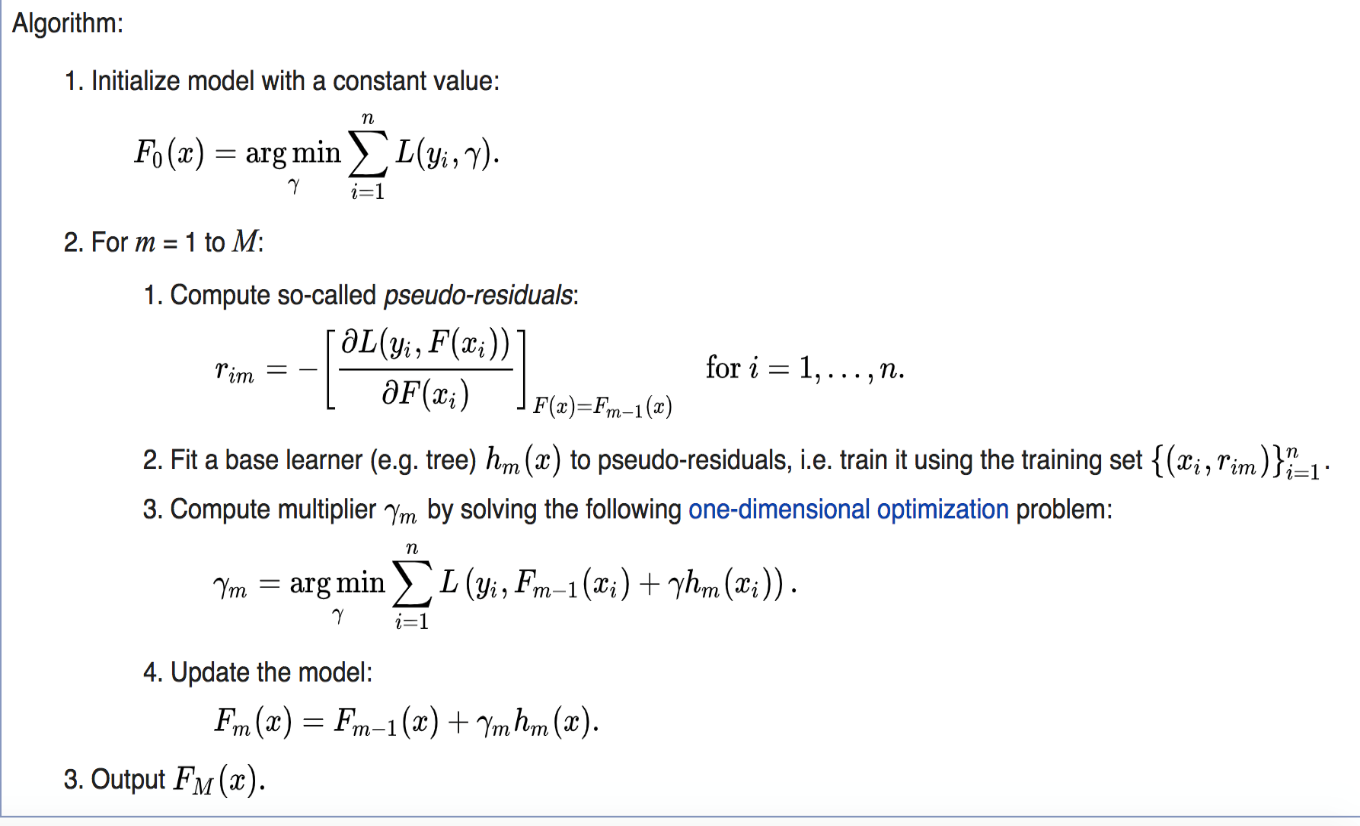
Gradient boosting is a machine learning technique that involves using multiple decision trees one after another to boost the accuracy of the system by minimizing the error rate during each iteration.

As said the main objective or concept behind gradient boosting is to reduce the loss function.

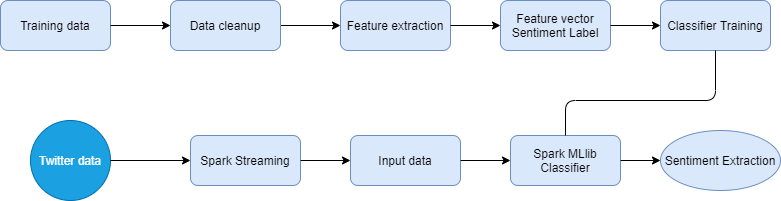


Boosting in general means transforming a weak classifier into a strong one. Initially, a classifier is selected. When a model is developed using this classifier it will contain some error rate. This error rate is reduced by using a loss function iteratively. The loss function is a measure that indicates how well the coefficient of a model is at fitting the underlying data.

**Algorithm:**



## System Architecture:

****

**Proposed System architecture**

Initially, a training dataset is created. This training dataset is in CSV file format. The dataset will consist of the input features and the output result.

### Data Cleanup:

Here the input dataset is cleaned i.e the unwanted words which do not contribute towards calculating sentiments will be removed. For example, consider the sentence “wow ….I loved this place”. In this sentence ‘….’, this, a place will not affect sentiment analysis, so words like these will be removed. These words are called as stopwords. Python provides nltk library to identify these stop words. Using nltk library the stopwords will be removed. All punctuations and commas will also be removed and the words will be converted to the smaller case.

### Feature extraction:

After the dataset is cleaned we will be left with only sentiment calculative relevant words. A sparse matrix will be created based on these words where the frequency of words for each sentence will be set.

### Classifier Training:

This is the step where the algorithms will be applied on the given sparse matrix and a final model will be created which will predict input data (Tweets).

### System specification:

The system will be developed on a computer with 4GB RAM, 10GB HDD, Intel 1.66 GHz Processor Pentium 4 processor, windows 7. Python 3.6.3 will be used as the base language. VScode will be used as the text editor.

### Apache Spark Framework:

Apache spark is a framework very similar to the Hadoop but an advanced version of it. It can perform processing of data on a large scale. It can distribute data on multiple systems for parallel execution.

# Implementation:

This section will discuss the libraries used for writing the code for implementation purposes. A discussion will be on the complete implementation steps of the overall project.

## Libraries Used:

### Mlliblibrary :

Mllib is a library used by a spark for performing machine learning algorithms on big data. It consists of various machine learning classification and regression algorithms along with clustering, dimensionality reduction based methods, etc.

### Tweepy:

No sentiment analysis or detection of any hate tweet can be done if there is no data for analyzing. This problem is solved using tweepy. Tweepy is a library used for getting live data from twitter and then we can use that data as we want.

* A simple command was used for installing Tweepy on the system (pip install tweepy).
* Then create a developer account on twitter for accessing data.

### OAuthHandler:

It is Tweepy function. It is used for verifying the credential of the user for further processing of data. It uses some keys for user verification.

consumer\_key = ''

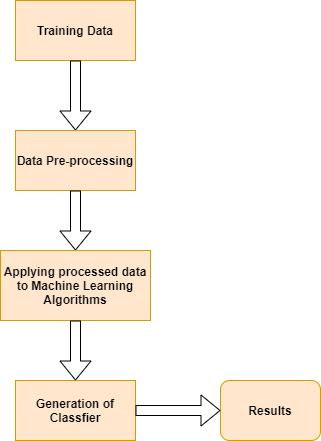
consumer\_secret = ''

access\_token = ' '

access\_secret = ''

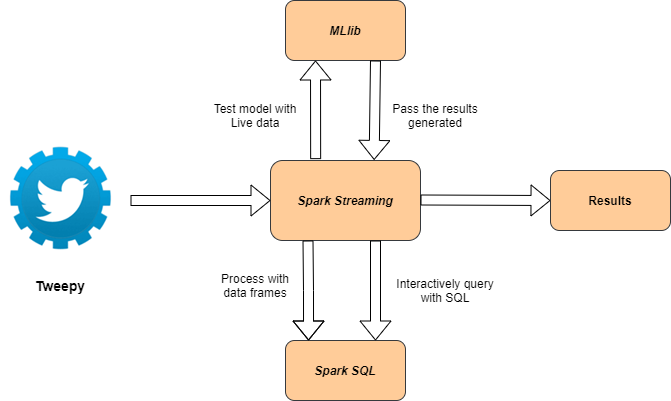
These keys are used for user verification.

## Flow Diagrams:



Training Phase Flow Diagram

The Dataset created is first divided into 80:20 parts. The first half for training purposes and the second half for testing purposes. The first half data that is the training is subjected to some pre-processing steps. This pre-processed data is then fed to the Machine learning algorithms to the generated classification model. The Test data is passed into this classification model for generating results like accuracy, precession, recall, etc.



Testing Phase Flow Diagram

Live data is obtained from twitter using Tweepy Library. This real-time data is handled by a spark streaming library. A similar kind of pre-processing happens in Spark real-time data testing as in the training phase. The input tweets are processed and the resulting output data is transferred to the Machine Learning model created during the training phase.

Spark SQL is used for making changes in the dataset, like deleting a row, changing column name, adding rows or columns, etc. After all necessary pre-processing and SQL operations the final data is passed to the model created during the training phase.

## What is Streaming?

Streaming is a way used to transfer large amounts of data continuously in the form of a stream.

## What is Spark Streaming?

Input data flowing in is divided into multiple small segments in the streaming case. These multiple segments are analyzed and processed individually. The processing and analysis of these segments are called stream processing. In 2013 Apache spark was added to data streaming which leads to the evaluation of multiple new methods like a window, reduce, join and map.

**Pre-Processing in training and testing phase:**

For the training Phase, the dataset consists of tweets and a dependent variable which indicates whether the tweet is hate speech or not. As the machine learning approach for this project follows a supervised form of learning, the dependent variable is used for training the model.

* The data type of this dependent variable is changed to an integer using the ‘StringIndexer’ function which takes in a column name and changes the datatype of that column.
* After the first step, the tweets are tokenized that is the tweets are split based on white spaces into words.
* ‘StopWordsRemover’ library is used to remove any stop word in the tokenized sets of words obtained from the above step. Stop words are words that do not make any contribution in identifying the sentiment of the tweet. For example ‘a’, ‘an’, ‘but’, ‘house’ etc does not have any sentimental value, so these words are removed from the tokenizing set of words.
* ‘CountVectorizer’ function will be used to count each word and store them in a dictionary format with the word and its count.
* ‘IDF’ function is used to calculate the tf-idf value for a tweet.

After this pre-processing for both the training and testing phase, the data is fed to the machine learning model for further evaluation and result generation.

# Proposed Evaluation:

For each algorithm, the result will consist of a confusion matrix, AUC-ROC Curve, precision, recall, and f1-score value. All these measures will help to gain a proper analysis of results obtained by each algorithm in detail.

Confusion Matrix:  
A confusion matrix is a table that summarizes the result or the output obtained from the training or testing data set on the machine learning model. The table displays the count of all the predictions made. That is, a count is stored for all the true positives, False Positives, True negatives, false negatives.This is the key concept of the confusion matrix, it displays the actual state our model is in after working on the dataset.



Confusion Matrix

* Class 1: Positive
* Class 2: Negative

**Definition of the Terms:**

* Positive indicates that the observation is positive (for example: is an apple).
* Negative indicates that the observation is not positive (for example: is not an apple).
* True Positive (TP) = It means that the input is positive and is correctly classified by the classifier.
* False Negative (FN) = It means that the input is positive and is in-correctly classified by the classifier.
* True Negative (TN) = It means that the input is negative and is correctly classified by the classifier.
* False Positive (FP) = It means that the input is negative and is in-correctly classified by the classifier.

With the help of the confusion matrix, the algorithms results can be compared easily. Confusion matrix will make it possible to visualize the TP, TN, FP, and FN for the various algorithm.

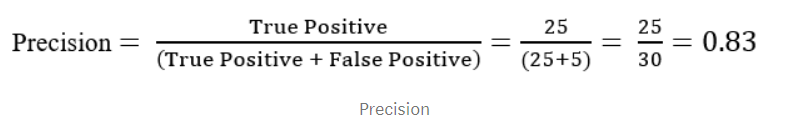
## AUC-ROC Curve:



AUC-ROC Curve is also known as the Seperatibility curve. It represents the degree of separability between the given data points. The higher its value the better is the classification.

**Precision:**

It is the measure of the value of correctly classified samples from all the value which are classified positive by the algorithm.



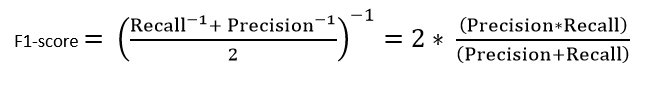
**Recall:**

It is the measure of the value of positive classified samples from all the samples which are present in the dataset.



**F1-score:**

F1-score is a harmonic mean of precision and recall. It displays a balance between precision and recall.



All these values will help me to perform a comparative study of algorithms on the dataset. These values will help to analyze which algorithm performs well in which department. For example which algorithm has better accuracy, which algorithm consists of low false positive and false negative. We will also see whether the ensemble algorithm can have better results than the other algorithms. We will find out that whether using spark framework over Hadoop generates any performance increase of the overall processing.

The overall aim of the project is to improve the accuracy of the system. The Spark framework used in place of Hadoop should boost the performance of the system. The accuracy should be increased with the help of an ensemble algorithm. From the literature review, I learned that the decision tree and naïve Bayes gave promising performance and accuracy increase on sentiment analysis. The increase in the dataset increased the performance of the decision tree and naïve Bayes drastically. For this project, a large dataset will be used so that the algorithms perform up to their potential. In the case of Gradient boosting, the algorithm should increase the accuracy over decision tree and naïve Bayes as it reduces the error rate by continuous application of multiple decision trees.

**Conclusion:**

In the above-discussed system, we propose a tool for analyzing tweets based on their sentiments using machine learning techniques on the Apache Spark framework.

Considering the nature of the data generated, we use the Spark framework to stream the data in realtime and perform the sentiment analysis model on it. Apache Spark was preferred over Hadoop for Big Data processing, considering its performance in regards to time and scalability.

A big data of tweets will be classified as positive or negative using different machine learning algorithms from the Apache Spark’s Machine Learning Library, entitled MLlib.

From the system proposed and evaluation metrics used we can conclude that the ensemble algorithm should have an increase in the accuracy of the system designed. The spark framework should increase the overall performance of the system, due to its caching abilities after each processing step.

# References

* Anon., n.d. *Decision Tree Classification Algorithm.* [Online]   
  Available at: https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm
* Anon., n.d. Machine Learning-Based Sentiment Analysis for Twitter Accounts.
* Anon., n.d. *Twitter Usage Statistics.* [Online]   
  Available at: https://www.internetlivestats.com/twitter-statistics/#:~:text=Every%20second%2C%20on%20average%2C%20around,200%20billion%20tweets%20per%20year.
* Baltas, A., Kanavos, A. & Tsakalidis, A. K., 2017. An Apache Spark Implementation for Sentiment Analysis on Twitter Data. *Springer.*
* Donges, N., 2019. *The Logistic Regression Algorithm.* [Online]   
  Available at: https://www.experfy.com/blog/the-logistic-regression-algorithm/#:~:text=Logistic%20Regression%20is%20one%20of,well%20enough%20in%20many%20tasks.
* Dutta, A., 2019. *Stacking in Machine Learning.* [Online]   
  Available at: https://www.geeksforgeeks.org/stacking-in-machine-learning/
* Gupta, S., 2018 . *Sentiment Analysis: Concept, Analysis and Applications.* [Online]   
  Available at: https://towardsdatascience.com/sentiment-analysis-concept-analysis-and-applications-6c94d6f58c17
* Kamal, R., Shah, M. A., Hanif, A. & Ahmad, J., 2017. Real-time Opinion Mining of Twitter Data using Spring XD and Hadoop. *IEEE.*
* Khan, M. & Malviya, A., 2020. Big data approach for sentiment analysis of twitter data using Hadoop framework and deep learning. *IEEE.*
* Kolchyna, O., Souza, T. & Treleav, . P., 2015. Twitter Sentiment Analysis: Lexicon Method, Machine Learning Method and Their Combination. *ARXIV.*
* Pang, B. & Lee, L., 2008. Opinion Mining and Sentimental Analysis Foundations and Trends in Information Retrieva. *IEEE.*
* RAY, S., 2017. *6 Easy Steps to Learn Naive Bayes Algorithm with codes in Python and R.* [Online]   
  Available at: https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/
* Ristevski, B., 2018. Big Data Analytics in Medicine and Healthcare. *ResearchGate.*
* Rodrigues, A. P., Robnik, M. & Chiplunkar, N. N., 2014. Real-time Twitter data analysis using Hadoop ecosystem. *IEEE.*
* Shen, Y. & Wang, J., 2008. An Improved Algebraic Criterion for Global Exponential Stability of Recurrent Neural Networks With Time-Varying Delays. *IEEE.*
* Socher, R. et al., 2013. Recursive deep models for semantic compositionality over a sentiment treebank. *IEEE.*
* S, V., L. & P., 2002. Thumbs up? Sentiment classification usingmachine learning techniques.
* Taboada, M., 2016. Sentiment Analysis: An overview from Linguistics. *ResearchGate.*
* T, W. & Hoffman, J., 2005. Recognizing contextual polarity in phraselevel sentiment analysis.
* Vaithyanathan, S., Lee, L. & Pang, B., 2010. Thumbs up sentiment classification using machine learning. *IEEE.*
* Vinodhini , G. & Chandrasekaran, R., 2012. Sentiment analysis and opinion mining: A survey. *IEEE.*
* Woldemariam , Y., 2016. Sentiment Analysis in A Cross-Media Analysis Framework. *IEEE.*