

Jomo Kenyatta University of Agriculture and

Technology (JKUAT)

Department of Computing

ICS2406: Computer Systems Project

REF:JKU/2/83/022

**Title**: Predicting Stock Market Movement Using an Enhanced Naïve Bayes Model for Sentiment Analysis Classification.

*Author:*

Name: Mbadi Atieno Sheila Reg. No: CS281-1556/2014

Submission Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Sign: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Course: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*Supervisor 1:*

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Sign: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*Supervisor 2:*

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Sign: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*Supervisor 3:*

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Sign: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Table of Contents

[CHAPTER I: PROJECT PROPOSAL 3](#_Toc506045018)

[Background/ Introduction 3](#_Toc506045019)

[Problem Statement 3](#_Toc506045020)

[Research Questions 3](#_Toc506045021)

[Objectives 4](#_Toc506045022)

[Justification 4](#_Toc506045023)

[Literature Relevant to the Proposal 4](#_Toc506045024)

[Research Methods and Design 5](#_Toc506045025)

[Schedule 6](#_Toc506045026)

[Budget 7](#_Toc506045027)

[Conclusion 7](#_Toc506045028)

[CHAPTER 2: LITERATURE REVIEW 8](#_Toc506045029)

[Introduction 8](#_Toc506045030)

[Challenge in Sentiment Analysis of Twitter Data 9](#_Toc506045031)

[Annotation 10](#_Toc506045032)

[Steps in Sentiment Analysis 11](#_Toc506045033)

[Data Gathering 11](#_Toc506045034)

[Preprocessing 11](#_Toc506045035)

[Tweet Feature Extraction and Sentiment Classification 12](#_Toc506045036)

[Train and build model 12](#_Toc506045037)

[Techniques used in Stock Prediction 12](#_Toc506045038)

[Time Series Analysis 12](#_Toc506045039)

[Moving Average (MA) 13](#_Toc506045040)

[Previous Work 15](#_Toc506045041)

[Twitter Sentiment Analysis to Predict the Stock Market Movement 15](#_Toc506045042)

[Sentiment Classification using an Enhanced Naive Bayes Model. 15](#_Toc506045043)

[Forecasting Stock Market Trend using Exponential Moving Average 16](#_Toc506045044)

[Algorithm 17](#_Toc506045045)

[Naïve Bayes Classifier 17](#_Toc506045046)

[Predicting classifications 20](#_Toc506045047)

[Conclusion 20](#_Toc506045048)

[Reference 22](#_Toc506045049)

# CHAPTER I: PROJECT PROPOSAL

## Background/ Introduction

Social networks like Facebook and twitter have changed the way people communicate. People use such outlets to express their views and opinions on various topics. These information is beneficial for data analysts who mine various opinions from users as feedback which they use to get insight on a product or service offered.

Sentiment analysis is used to extract such remarks of users and then gives them a polarity of positive, negative or neutral. The neutral case will be ignored as it will hold no weight in this study. Sentiment analysis can be described as using Natural Language Processing (NLP), statistics, or machine learning methods to extract, identify, or otherwise characterize the sentiment content of a text unit.

The Twitter API which will be used faces a major limitation in that you can only extract tweets from the last 7 days. Thus it will be prudent to develop a system that will do analytics on a daily basis and stores the result so that there can be enough data to analyze.

The project will focus on the challenges encountered in tweets collection and preprocessing and also in the analysis of these tweets using Naive Bayes Theorem. Naive Bayes is used as it is fast, simple to use and has a high degree of efficiency. The implementation of the project will be done using the Python Natural Language Toolkit (NLTK).

## Problem Statement

The current methods of predicting stock market prices are not as accurate as investors might need them to be. These current methods only incorporate linear regression in there prediction. The Efficient Market Hypothesis (EMH) states that stock market prices are largely driven by new information; thus, there is need to incorporate public sentiments when coming up with a prediction model for stock market prices.

### Research Questions

1. How can Naïve Bayes classifier be enhanced to give better classification accuracy of tweets?
2. How can the enhanced Naïve Bayes classifier be trained to describe the relationship between twitter sentiments and the stock market movement?
3. Is it possible to develop a tweet classification system that can analyze and present tweets for stock market prediction?

### Objectives

1. To research on the enhancement of the Naïve Bayes classifier for sentiment classification of tweets.
2. To train the enhanced Naïve Bayes classifier to do sentiment classification of tweets and use it for sentiment analysis.
3. To develop a tweet classification system that can analyze and present tweets for stock prediction.

## Justification

Prediction of stock prices is a very complex process. There is need for a way to come up with highly accurate predictions so as to yield significant profit by any investor. A lot of research has been conducted on the topic of stock prediction but very few actually take public sentiments into consideration. From previous research work, the incorporation of sentiment analysis in predicting stock prices has resulted in predictions with a higher degree of accuracy when compared to those that only use Linear Regression (Bollen et.al., 2010).

## Literature Relevant to the Proposal

A research conducted by Bollen et al. shows that the collective mood on Twitter, the aggregate of all positive and negative tweets, can predict the Dow Jones Industrial Average (DJIA) index with 87.6% accuracy. This paper will be highly valuable as it shows there is a correlation between twitter feeds and the stock market movement (Bollen et.al, 2010)

Earlier work conducted by Tushar Rao and Saket Srivastava (Rao T. & Srivastava S., 2014) will be used as the paper they published contains valuable information to my research work. Naive Bayes was used in their work therefore it will be a good paper to refer from.

The work of Anshul Mittal and Arpit Goel will also be used in my research as they outlined the steps to be taken when using sentiment analysis to carry out stock prediction in a very orderly manner (Anshul Mittal & Arpit Goel, 2011).

Daniel Jurafsky and James H. Martin book will be used as they go in depth on how Naive Bayes can be used as a machine learning classifier for Sentiment Analysis (Daniel Jurafsky & James H. Martin, 2017).

## Research Methods and Design

An enhanced Naïve Bayes classifier will be used for sentiment classification.

Linear Regression will also be used to correlate twitter sentiment data and stock market movement.

The two methods are explained in detail in chapter 2.

# 

# 

## Schedule

|  |  |  |
| --- | --- | --- |
| **Activity** | **Estimated Start and End Date** | **Deliverables** |
| Project Identification and Proposal Writing | 28th September - 6th October 2017 | Project Proposal |
| Project Research | 10th October - 12th January 2018 | Literature Review Document and Methodology |
| Project Analysis and Design | 5th  February - 16th  February 2018 | Analysis and Design Document |
| System Implementation | 19th February - 30th March 2018 | Working Prototype |
| System Testing and Debugging | 2nd April- 13th April 2018 | Final Prototype |
| Project Submission | May 2018 | Complete System with Documentation |

## Budget

|  |  |  |
| --- | --- | --- |
| **Resource** | **Cost** | **Justification** |
| Printing and Binding | Kshs 1500 | Need for printing and binding of documents |
| Airtime and Bundle Purchase | Kshs 3500 | Researching on the internet and contacting supervisor when there is need |
| Flash Disk | Kshs 2000 | Data transfer when printing and backing up of data |
| **Total** | **Kshs 7000** |  |

## Conclusion

The project can be completed within the allocated time and also within the allocated budget. It will be beneficial to investors and in the long run it may aid in the growth of our nation.

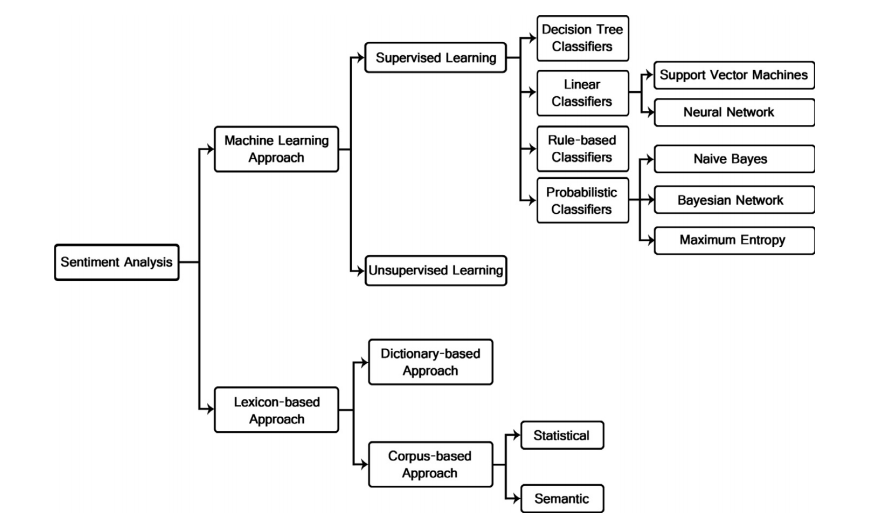
# CHAPTER 2: LITERATURE REVIEW

## Introduction

Sentiment Analysis is an area of study within Natural Language Processing that is concerned with identifying the mood or opinion of subjective elements within a text towards an entity (Bhadane, Dalal & Doshic, 2015). The entity can represent individuals, events, topics or a product/service offered. It is becoming a [popular](http://www.readwriteweb.com/archives/sentiment_analysis_is_ramping_up_in_2009.php) area of [research](http://www.semanticweb.com/news/everything_i_need_to_know_about_sentiment_analysis_158453.asp) and [social media analysis](http://mashable.com/2010/04/19/sentiment-analysis/), especially around [user reviews](http://www.nytimes.com/2009/08/24/technology/internet/24emotion.html?_r=1) and [tweets](http://smallbiztrends.com/2010/03/tracking-twitter-sentiment.html). It is a special case of [text mining](http://en.wikipedia.org/wiki/Text_mining) generally focused on identifying [opinion polarity](http://www.sevendayworkweek.com/web-analytics-tools/what-is-automated-sentiment-analysis-good-for/), and while it’s often [not very accurate](http://www.marketingpilgrim.com/2009/08/why-sentiment-analysis-is-about-as-reliable-as-a-canary-in-a-coal-mine.html), it can still be [useful](http://priyankmohan.blogspot.com/2010/04/sentiment-analysis-can-you-get-it-right.html) (Text Classification for Sentiment Analysis – Naive Bayes Classifier, 2010).

The tasks involved in sentiment analysis include finding opinions, identify the sentiments they express, and then classify their polarity.

There are two main sentiment classification techniques, these are lexical-based approach and machine learning approach. These two have subgroups which are shown in Figure 1 (Medhat, Hassan, & Korashy, 2014).

**Figure** **1**. Sentiment Classification Techniques

Lexical and machine learning approaches are sentiment classification techniques, they are used to classify a given piece of natural language text according to the opinions expressed in it.

For the dictionary-based lexical approach, a dictionary is prepared to store the polarity values of lexicons. For calculating polarity of a text, polarity score of each word of the text, if present in the dictionary, is added to get an ‘overall polarity score’. If the overall polarity score of a text is positive, then that text is classified as positive, otherwise it is classified as negative. Though this approach seems very basic, variants of this lexical approach have been reported to have considerably high accuracy.

Since the polarity of the text depends on the score given to each lexicon, there has been a large volume of work dedicated to discovering which lexical information is most efficient. A target word along with the word ‘good’, and a second with the target word with the word ‘bad’ was used by Turney in the AltaVista search engine queries. The polarity of the target word was determined by the search result that returned the most hits (Turney, 2002). This approach resulted in accuracy of 65%. In another research, Turney & Littman, 2003, mapped the semantic association between the target word and each word from the selected set of positive and negative words to a real number. By subtracting a word's association strength to a set of negative words from its association strength to a set of positive words, an accuracy rate of 82% was achieved.

For the machine learning approach, a series of feature vectors are chosen and a collection of tagged corpora is used to prepare a model. The model can then be used to classify an untagged corpus of text. In machine learning approach, the selection of features is crucial to the success rate of the classification. Most commonly, a variety of unigrams, which are, single words from a document, or n-grams, which are, two or more words from a document in sequential order are chosen as feature vectors. Most commonly employed classification techniques of machine learning are Support Vector Machines (SVMs), Naive Bayes and Maximum Entropy algorithm. Other techniques include Neural Networks, Bayesian Networks and Decision Tree Classifiers. The accuracy results for these algorithms greatly depends on the features selected.

## Challenge in Sentiment Analysis of Twitter Data

Twitter data costs a lot of money, and if it has not been possible to retrieve or set up a system to retrieve Twitter data within 7 days on a topic of interest, then it becomes difficult to obtain the data. This is because using the free Twitter public API ecosystem it is only possible to retrieve Twitter data going back in time 7 days. However, it is possible to obtain this data through other means such as obtaining them from Twitter at a fee, by using a licensed reseller of Twitter data or by using an existing dataset. Historical Twitter data can range from not that expensive, to very expensive depending on both the query and time of retrieval (Ahmet, 2015).

The solution to this is opting for the free data from the Twitter public API. Thus, it will be prudent to develop a system that will do analytics on a daily basis and store the results so that there can be enough data to analyze.

## Annotation

To check the accuracy of any emotion detecting algorithm, the results need to be compared to a human-labeled text. The process in which humans manually label a text is called annotation. Annotation can be done on multiple levels: word, sentence, paragraph, section, or even the entire document.

Annotation can be based on polarity, emotion, and intensity. When annotating on polarity, text is labeled with positive, negative, or neutral emotion. Text annotated on emotion is labeled based on some predefined list. The most common lists of emotions used are those suggested by Ekman, Izard and Plutchik (Mulcrone, 2012). Additionally, some studies annotate the text by labeling the intensity of the emotion. Intensity is based on a numeric scale, but there are no standards for this type of annotation. The first two categories are the ones that are applicable for this study, this is because emotions are closely related to the polarity of texts.

In general, studies either use pre-annotated datasets to test an algorithm or undergo a small annotation process. In the latter case, annotators who are qualified to label emotion in text, such as psychologists are used. The text is annotated to the given specification and level of analysis. Sometimes annotators are given an additional word list that consists of words from the original text. These lists help determine which words are attached to a specific emotion and which vary by context. When the annotation process is complete the agreement among the annotators is calculated using a method such as the Kappa Value.

## Steps in Sentiment Analysis

### Data Gathering

This is the first step in sentiment analysis. The tweets will be obtained from the Twitter public API and Tweepy Python library will be used to access the API. The desired features to be analyzed are then obtained from the tweets.

### Preprocessing

A tweet is composed of a maximum of 280 characters. To start with, the text is split into individual words each word becoming a feature in the feature vector which is stored in a bag-of-words. This breakdown of a sentence into individual words is known as tokenization.

Although many tokenizers are geared towards throwing punctuation marks away, for sentiment analysis a lot of valuable information could be deduced from them. “!” puts extra emphasis on the negative/positive sentiment of the sentence, while “?” can mean uncertainty (no sentiment).

“, ‘, [], () can mean that the words belong together and should be treated as a separate sentence. Same goes for words which are bold, *italic* or underlined. But symbols such as the “@” symbol, and links can be removed as they are regarded as noise generating elements.

Word Normalization should be applied. This is the reduction of each word to its base/stem form (by chopping of the affixes). This is known as stemming or lemmatizing. An example is walking to walk. Capital letters should be normalized to lowercase, unless it occurs in the middle of a sentence; this could indicate the name of a writer, place, brand etc. Words with an apostrophe should also be handled. “George’s phone” should obviously be tokenized as “George” and “phone”, but I’m, we’re, they’re should be translated as I am, we are and they are. To make it even more difficult, it can also be used as a quotation mark.

Other preprocessing steps include the elimination of stop words. These are the un-informative words in tweets which include “so”, “and”, “or” and “the”. A stop-word list can be created or searched against a language-specific stop word dictionary. Further, Parts of Speech taggers (PoS) are used to classify words into the English 8-parts of speech. Nouns and pronouns do not contain any sentiment according to (Fang and Zhan, 2015). As such these words are exempted from the classification process.

### Tweet Feature Extraction and Sentiment Classification

After the text has been segmented into sentences, each sentence has been segmented into words, the words have been tokenized and normalized. We can make a simple bag-of-words model of the text. In this bag-of-words representation you only take individual words into account and give each word a specific subjectivity score. This subjectivity score can be looked up in a sentiment lexicon. If the total score is negative the text will be classified as negative and if it is positive the text will be classified as positive.

The sentiment lexicon can be created using some simple statistics of the training set. To do this the class probability of each word present in the bag-of-words will be determined. This will be done by using Pandas (Python) dataframe as a data container, but it can just as easily be done with dictionaries or other data structures.

The sentiment lexicon is simple to make, but is less accurate because it does not take the word order of the grammar into account. A simple improvement on using unigrams would be to use bigrams and trigrams. That is, not to split a sentence after words like “not”, ”no”, ”very”, “just” etc. It is easy to implement but can give significant improvement to the accuracy.

The best words to put in a bag-of-words include salient words that give domain specific information and discriminatory words that help to clear distinctions of polarities (Mulcrone, 2012). The bag-of-words model simply uses a statistical approach to classify polarities.

### Train and build model

The above mentioned steps will be carried out in the training set. A test set will then be used to do classifications and to tell the efficiency of the classifier. Overfitting can be checked here.

A Python program will be used to extract tweets, analyze them, and store relevant information in appropriate files that can be retrieved to show relation of public sentiments and stock market movement.

## Techniques used in Stock Prediction

### Time Series Analysis

A time series is an ordered sequence of values of a variable at equally spaced time intervals. It can be taken on any variable that changes over time.

In investing, it is common to use a time series to track the price of a security over time. This can be tracked over the short term, such as the price of a security on the hour over the course of a business day, or the long term, such as the price of a security at close on the last day of every month over the course of five years.

Time series analysis can be useful to see how a given asset, security or economic variable changes over time. It can also be used to examine how the changes associated with the chosen data point compare to shifts in other variables over the same time period. It can therefore be used to determine the relationship between public sentiments of a stock and the movement in price of the mentioned stock.

Time series can be used for quantitative forecasting by using information regarding historical values and associated patterns to predict future activity. Most often, this relates to trend analysis, cyclical fluctuation analysis and issues of seasonality.

Inherent in the collection of data taken over time is some form of random variation. There exist methods for reducing or canceling the effect due to random variation. An often-used technique is "smoothing". This technique, when properly applied, reveals more clearly the underlying trend, seasonal and cyclic components. Therefore, in this research work there will be need for **smoothing functions** that react quickly to changes in the signal, hence the need for **moving averages**.

### Moving Average (MA)

A moving average (MA) is a widely used indicator in technical analysis that helps smooth out price data by filtering out the “noise” from random price fluctuations and form a trend following indicator. They do not predict price direction, but rather define the current direction with a lag. Moving averages lag because they are based on past prices. Despite this lag, moving averages help smooth price action and filter out the noise.

The two basic and commonly used moving averages are the:

1. **Simple** **moving average (SMA).**
2. **Exponential moving average (EMA)**.

#### **Simple Moving Average**

**A Simple Moving Average is formed by computing the average price of a security over a specific number of periods.** Most moving averages are based on closing prices. A 5-day simple moving average is the five-day sum of closing prices divided by five. As its name implies, a moving average is an average that moves. Old data is dropped as new data comes available. This causes the average to move along the time scale.

#### Exponential Moving averages (EMA)

An exponential moving average (EMA) is a type of moving average that is similar to a simple moving average, except that more weight is given to the latest data. It's also known as the exponentially weighted moving average. This type of moving average reacts faster to recent price changes than a simple moving average.

It is often used where latency is critical, such as in real time financial analysis. In this average, the weights decrease exponentially. Each sample is valued some percent smaller than the next most recent sample. With this constraint the moving average can be calculated very efficiently.

The formula is:

avgt = (alpha \* samplet ) + (( 1 – alpha ) \* avgt-1 )

Where alpha is a constant that describes how the simple weights decrease over time. For example if each sample was to be weighted at 80% of the value of the previous sample, you would set alpha = 0.2.

Each new sample needs to be average with the value of the previous average. So computation is very fast. In theory all previous samples contribute to the current average, but their contribution becomes exponentially smaller over time.

This is a very powerful technique, and probably the best in getting a moving average that responds quickly to new samples, has good smoothing properties and is fast to compute.

It is therefore beneficial to apply the Exponential Moving Average in our Time Series Analysis to get better results in predicting the direction of the stock market movement.

## Previous Work

### Twitter Sentiment Analysis to Predict the Stock Market Movement

Earlier work done by Bollen, Mao, & Zeng, (2010) shows how collective mood on Twitter (aggregate of all positive and negative tweets) is reflected in the Dow Jones Industrial Average (DJIA) index movements. They investigated whether public sentiment, as expressed in large-scale collections of daily Twitter posts, could be used to predict the stock market.

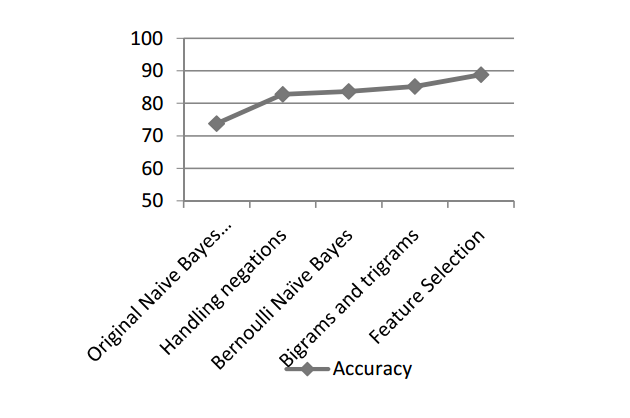
They used two tools to measure variations in the public mood from tweets submitted to the Twitter service from February 28, 2008 to December 19, 2008. The first tool, OpinionFinder, analyzed the text content of tweets submitted on a given day to provide a positive vs. negative daily time series of public mood. The second tool, Google-Profile of Mood States (GPOMS), similarly analyzed the text content of tweets to generate a six-dimensional daily time series of public mood to provide a more detailed view of changes in public along a variety of different mood dimensions.

The resulting public mood time series were correlated to the DJIA to assess their ability to predict changes in the DJIA over time. The results indicated that the prediction accuracy of a standard stock market prediction models is significantly improved when certain mood dimensions are included, but not others. In particular variations along the public mood dimensions of Calm and Happiness as measured by GPOMS seem to have had a predictive effect, but not general happiness as measured by the OpinionFinder tool.

The research found an accuracy of 87.6% in predicting the daily up and down changes in the closing values of the DJIA and a reduction of the Mean Average Percentage Error by more than 6%. This therefore shows that there is a strong correlation between twitter data and stock market movement.

### Sentiment Classification using an Enhanced Naive Bayes Model.

Narayanan, Arora & Bhatia showed that a simple Naive Bayes classifier can be enhanced to match the classification accuracy of more complicated models for sentiment analysis by choosing the right type of features and removing noise by appropriate feature selection. Naive Bayes classifiers were thought to be less accurate than their more sophisticated counterparts like support vector machines and logistic regression. The enhanced Naïve Bayes classifier was able to achieve an 88.8% accuracy from the 73.77% accuracy when using the original Naive Bayes algorithm with Laplacian smoothing. The improvements from applying the various processes are illustrated in Figure 2.

**Figure 2.** Evolution of classification accuracy.

### Forecasting Stock Market Trend using Exponential Moving Average

From the research, Alexander Decker concluded that the curve of the moving average shows the market/ stock trend. If the curve is in an upward direction, the market heads towards Bull Run. If the moving average is going down the market is in Bear phase and a flat moving average shows consolidation.

He further states that there are 3 steps in calculating the EMA:

1. Calculate the simple moving average for the initial EMA value. An exponential moving average (EMA) has to start somewhere, so a simple moving average is used as the previous period's EMA in the first calculation.
2. Calculate the weighting multiplier.
3. Calculate the exponential moving average for each day between the initial EMA value and today, using the price, the multiplier, and the previous period's EMA value.

The formula below is for a 10-day EMA.

*Initial SMA: 10-period sum / 10*

*Multiplier: (2 / (Time periods + 1)) = (2 / (10 + 1)) = 0.1818 (18.18%)*

*EMA: {Close – EMA (previous day)} x multiplier + EMA (previous day).*

By using this technical analysis tool, he was able to find out the trends in different stocks; whether they are going upwards, downwards or stagnant. This increases the chance of investors to predict the prices more accurately by prediction the trend reversals in advance and hence increased profit in the share markets.

In short, EMA when combined with market behavioral analysis, such as Twitter sentiments about a company, then the probability of it being correct for forecasting market trends will increase.

## Algorithm

### Naïve Bayes Classifier

A Naive Bayes classifier is a simple probabilistic model based on the Bayes rule along with a strong independence assumption.

The Naïve Bayes model involves a simplifying conditional independence assumption. That is, given a class (positive or negative), the words are conditionally independent of each other. Due to this simplifying assumption the model is termed as “naïve”. This assumption does not affect the accuracy in text classification by much but makes really fast classification algorithms applicable for the problem.

The maximum likelihood probability of a word belonging to a particular class is given by the equation:



The frequency counts of the words will be stored in hash tables during the training phase.

According to the Bayes Rule, the probability of a particular text belonging to some particular class is given by:



If the simplifying conditional independence assumption is used, that is, given a class (positive or negative) the words are conditionally independent of each other. The following equation will be used.



Here the xi’s are the individual words of the text. The classifier outputs the class with the maximum posterior probability.

Naive Bayes classifiers are thought to be less accurate than their more sophisticated counterparts like support vector machines and logistic regression classifier. A simple Naive Bayes classifier can be enhanced to match the classification accuracy of these more complicated models for sentiment analysis. The advantages of using Naive Bayes as our classifier are:

* Naive Bayes classifiers due to their conditional independence assumptions are extremely fast to train and can scale over large datasets.
* They are robust to noise and less prone to over-fitting.
* Ease of implementation is also a major advantage of Naive Bayes.

A significantly high accuracy can be achieved by applying the following processes to the simple Naive Bayes classifier:

1. **Bernoulli Naïve Bayes**

Duplicate words are removed from the text as they don’t add any additional information, this type of Naïve Bayes algorithm is called Bernoulli Naïve Bayes. Including just the presence of a word instead of the count has been found to improve performance marginally, when there is a large number of training examples.

1. **Laplacian Smoothing**

If the classifier encounters a word that has not been seen in the training set, the probability of both the classes would become zero and there won’t be anything to compare between. This problem can be solved by Laplacian smoothing



Usually, k is chosen as 1. This way, there is equal probability for the new word to be in either class. Since Bernoulli Naïve Bayes is used, the total number of words in a class is computed differently. For the purpose of this calculation, each text is reduced to a set of unique words with no duplicates.

1. **Negation Handling**

Negation handling is one of the factors that contribute significantly to the accuracy of this classifier. A major problem faced during the task of sentiment classification is that of handling negations. Since we are using each word as a feature, the word “good” in the phrase “not good” will be contributing to positive sentiment rather that negative sentiment as the presence of “not” before it is not taken into account.

To solve this problem a simple algorithm for handling negations using state variables and bootstrapping will be devised. This builds on the idea of using an alternate representation of negated forms. This algorithm uses a state variable to store the negation state. It transforms a word followed by a not into “not\_” + word. Whenever the negation state variable is set, the words read are treated as “not\_” + word. The state variable is reset when a punctuation mark is encountered or when there is double negation.

Since the number of negated forms might not be adequate for correct classifications. It is possible that many words with strong sentiment occur only in their normal forms in the training set. But their negated forms would be of strong polarity.

This problem is addressed by adding negated forms to the opposite class along with normal forms of all the features during the training phase. That is to say if we encounter the word “good” in a positive document during the training phase, we increment the count of “good” in the positive class and also increment the count of “not\_good” for the negative class. This is to ensure that the number of “not\_” forms are sufficient for classification. This modification will result in a significant improvement in classification accuracy (about 1%) due to bootstrapping of negated forms during training. This form of negation handling can be applied to a variety of text related applications

1. **n-grams**

Generally, information about sentiment is conveyed by adjectives or more specifically by certain combinations of adjectives with other parts of speech.

This information can be captured by adding features like consecutive pairs of words (bigrams), or even triplets of words (trigrams). Words like "very" or "definitely" don't provide much sentiment information on their own, but phrases like "very bad" or "definitely recommended" increase the probability of a document being negatively or positively biased. By including bigrams and trigrams, we will be able to capture this information about adjectives and adverbs. Bigrams will be used as the 280 characters limit will hinder the use of more n-grams. The counts of the n-grams will be stored in a hash table along with the counts of unigrams.

1. **Feature Selection**

Feature selection is the process of removing redundant features, while retaining those features that have high disambiguation capabilities.

The use of higher dimensional features like bigrams and trigrams presents a problem, that of the number of features increasing. Most of these features are redundant and noisy in nature. Including them would affect both efficiency and accuracy. A basic filtering step of removing the features/terms which occur only once will be performed. The features can then be further filtered on the basis of mutual information.

### Predicting classifications

To predict a classification we simply look at which of the two classes give a higher Naïve Bayes probability: That is given a sample X and two classification categories ω1 and ω2, if P (ω1| x)>P (ω2| x) classify sample X as ω1 else classify as ω2. That is the classification opts for the class that gets a high probability with the Naïve Bayes formula.

## Conclusion

In conclusion, I will use the Yahoo Finance website to get the stock market data to analyze. Python Panda’s Data-Reader will be used to grab the data from this website as a Panda DataFrame. The stock data that I will analyze for this study will be data for Google and Amazon companies as they are among the most tweeted companies on Twitter.

The sentiment data that I will analyze will come from the free Twitter public API. The Tweepy Python library will be used to access the API data. Ekman’s emotions will be used as a reference point when creating the annotated data set to be used during the study. The MongoDB Database will be used to store the Twitter data for later analysis.

The Enhanced Naïve Bayes classifier that I will use has already been implemented in Python and it has been hosted on GitHub. I will add any other needed functionalities that have not yet been implemented in it.

After classifying tweets and obtaining the stock data, a time series graph of the two will be constructed. The Exponential Moving Average will be used to make the stock predictions more accurate. MatPlotLib Python library will then be used to correlate the two graphs by plotting the axes of stock prices on the left and that of the sentiment score on the right. Trends in the two (twitter sentiments and stock prices) will then be studied to establish the meaning of their movements and concluding remarks will be made.

The website to visualize this study will be created using Django Python Framework.

## Reference

1. Hong Miao, Sanjay Ramchander, and J. K. Zumwalt (2011). Information driven price jumps and trading strategy: Evidence from stock index futures. SSRN eLibrary.
2. Burton G. Malkie (2003). The efficient market hypothesis and its critics. *Journal of Economic Perspectives*, *17*(1), 59–82.
3. Johan Bollen, Huina Mao, and Xiao-Jun Zeng (2010). Twitter mood predicts the stock market. *1*(1), 1–8.
4. Rao T., Srivastava S. (2014). Twitter sentiment analysis: How to hedge your bets in the stock markets. In: Can F., Özyer T., Polat F. (Eds.) State of the Art Applications of Social Network Analysis. Lecture Notes in Social Networks. Springer, Cham
5. Anshul Mittal, Arpit Goel (2011). Stock prediction using twitter sentiment analysis.
6. Daniel Jurafsky and James H. Martin (2017). Speech and language processing An Introduction to Natural Language processing. *Computational Linguistics, and Speech Recognition*.
7. Bhadane, C., Dalal, H., & Doshic H., (2015). Sentiment analysis: Measuring opinions*. International Conference on Advanced Computing Technologies and Applications (ICACTA-2015)*. 808-814. Retrieved from <http://www.sciencedirect.com/science/article/pii/S1877050915003956>
8. *Text Classification for Sentiment Analysis – Naive Bayes Classifier*. (2010). Retrieved from <https://streamhacker.com/2010/05/10/text-classification-sentiment-analysis-naive-bayes-classifier/>
9. Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal, 5,* 1093-1113
10. Turney, P. (2002). Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews*. Proceedings of ACL*. 417-424.
11. Turney, P.D., Littman. 2003. M.L.Measuring Praise and Criticism: Inference of Semantic Orientation from Association. *ACM Transactions on Information Systems*. 315-346.
12. Ahmet, T. (2015). *Text classification and sentiment analysis.* Retrieved from <http://ataspinar.com/2015/11/16/text-classification-and-sentiment-analysis/>
13. Ahmet, T. (2016). *Sentiment analysis with bag-of-words.* Retrieved from<http://ataspinar.com/2016/01/21/sentiment-analysis-with-bag-of-words/>
14. Mulcrone, K. (2012). Detecting Emotion in Text. University of Minnesota.
15. Fang, X. & Zhan, J. (2015). Sentiment analysis using product review data.
16. Narayanan, V., Arora, I., & Bhatia A. (). Fast and accurate sentiment classification using enhanced Naive Bayes model.
17. Time Series https://www.investopedia.com/terms/t/timeseries.asp#ixzz55mTuzkK8
18. Moving Average (MA), (n.d). Retrieved from. <https://www.investopedia.com/terms/m/movingaverage.asp#ixzz55mXdcgu6>
19. Moving Averages- Simple and Exponential, (n.d). Retrieved from. <http://stockcharts.com/school/doku.php?id=chart_school:technical_indicators:moving_averages>
20. Exponential Moving Averages For Irregular Time Series, (2013) <https://oroboro.com/irregular-ema/>