**Twitter Sentimental Analysis using Machine Learning**

Juginder Pal Singh1  
Department of Computer Engineering and Application  
GLA University, Mathura, Uttar Pradesh  
[juginder.singh@gla.ac.in](mailto:juginder.singh@gla.ac.in)

Aman Singh2  
Department of Computer Engineering and Application  
GLA University, Mathura, Uttar Pradesh  
[aman.singh\_cs17@gla.ac.in](mailto:aman.singh_cs17@gla.ac.in)

Ayush Gupta3  
Department of Computer Engineering and Application  
GLA University, Mathura, Uttar Pradesh  
[ayush.gupta\_cs17@gla.ac.in](mailto:ayush.gupta_cs17@gla.ac.in)

**Abstract**

***Nowadays, social media is getting more popularity.  Public and private opinions on a wide range of topics are constantly shared and distributed through a variety of social media platforms. Twitter is considered as a prominent social media platforms.  Twitter provides businesses with a quick and easy way to assess their consumers' views on issues that are vital to their success in the marketplace. Sentimental analysis is a type of natural language processing that is used to track public perception of a product or topic. Sentimental analysis, sometimes known as opinion mining, is the process of developing a framework for capturing and understanding product opinions expressed in blog posts, comments, reviews, or tweets. Sentimental analysis can be beneficial in a variety of situations. Because of its relevance to industry and society as a whole, it has expanded from the computer science field to the fields of management and social sciences. We look at how useful syntactical characteristics are for recognize sentiment in Twitter messages. We assess the utility of current phrasal tools together with features that collect details about the spontaneous and inventive language used on Twitter. We take a supervised method to solving the problem, but we create training data from existing tweets in the Twitter data.***

***Keywords: Machine Learning, python, Logistic Regression, Multinomial naive Bayes algorithm, Stochastic Gradient Descent, Voting classifier, flask***

# INTRODUCTION

## *Machine Learning*

It's a subset of artificial intelligence that aims to explore the meaning of data and refine it into models that people can understand and use. [8]

Machine learning is a branch of computer science that is different from traditional analytical methods. An algorithm is a set of instructions that a computer can use to measure or solve an existing computer problem.[12] Machine learning algorithms, on the other hand, allow computers to learn from data and perform statistical analysis to produce an estimate of a defined range. Machine learning allows computers to easily predict outcomes from data patterns and automate decision-making based on feedback input.[15]

Anyone who accesses technology today has learned from machine learning. Social networking sites are using facial recognition technology and helping users to tag and share pictures to their friends. Optical character recognition (OCR) turns text drawings into moving images. A machine learning-based recommendation system recommends the next movie or TV show to watch based on user preferences. Consumers will soon be able to buy self-driving cars that use machine learning for navigation.[9]

Machine learning is an area which is constantly evolving. As a result, whether interacting machine learning techniques or evaluating the response of machine learning processes, there are a few things that must be considered.

## *Python*

Python is one of the most popular scripting language that can be used for designing software like web development.

Variables, parameters, functions, and methods have no type declarations in source code. This makes the code shorter and more portable, but you lose the source code's form checking at compile time. At runtime, Python maintains track of the types of all values and flags any code that makes no sense.

Source files of python uses the extension “.py” and they are known as modules. The first time a Python file, or "module," is imported somewhere, the furthermost statements in the file, or "module," do its one-time initialization — From top to bottom, such assertions can be found, setting up the module's functions and variables. A Python module can either be used directly or imported and used by another module. Whenever a python file runs directly, the special variable “\_\_name\_\_” has been intended to “\_\_main\_\_”. As a result, when we run a module directly, the boilerplate if \_\_name\_\_ ==... is commonly used to call the main() method, but not when another module imports this module. [11]

## *Logistic Regression*

A Machine Learning algorithm called logistic regression is utilised to address categorization situations. It's a probabilistic-based predictive analytic algorithm. A provision Regression model is nearly same on a rectilinear regression model, except that rather than employing a linear perform, it utilises an additional intensive value perform referred to as the "Sigmoid perform" or "logistic function." [2]

According to the logistic regression hypothesis, the value of the cost function must be between 0 and 1.[8] As a result, because it can have a value larger than 1 or less than 0, linear functions struggle to describe it, According to the logistic regression hypothesis, this is impossible.

## *Multinomial naive Bayes algorithm*

According to the logistic regression hypothesis, the value of the cost function must be between 0 and 1.[8] As a result, the linear function is terminated. In Natural Language Processing, the Multinomial Naive Bayes calculation is a popular probabilistic learning approach (NLP). The computation predicts the tag of a document, for example, an email or a paper post, utilizing the Bayes hypothesis. It computes each label's likelihood for a given example and yields the tag with the most noteworthy likelihood. It is addressed by the letter l since it might have a worth more noteworthy than 1 or under 0, which is unimaginable as per the strategic relapse speculation. [4]

The Naive Bayes classifier is made up of a number of algorithms that all have one thing in common: each feature being categorised is unrelated to any other feature. A feature's presence or absence has no bearing on the presence or absence of another feature. [5]

The Naive Bayes algorithm is a powerful tool for analysing text data and solving problems with multiple groups. Since the Naive Bayes theorem is based on the Bayes theorem, it is necessary to first understand the Bayes theorem principle. The Bayes theorem, which was developed by Thomas Bayes, measures the probability of an occurrence occurring based on previous knowledge of the event's conditions. [2]

## *Stochastic Gradient Descent*

On very large datasets, gradient descent can be slow. When there are several millions of instances in the training dataset, one cycle of the gradient descent algorithm includes a forecast for each case in the preparation dataset. When you have a lot of data, you can use stochastic gradient descent, which is a variant of gradient descent.[7] The gradient descent technique is used in this variant, but instead of updating the coefficients at the end of the batch of instances, it is done for each training instance.[10]

The training dataset must be randomised in order for the technique to work. This is done to change the order in which the coefficients are modified. Since the coefficients are modified for each training case, the updates, like the corresponding cost function, will be noisy and jump all over the place.[14] It harnesses this random walk by varying the order in which the coefficients are modified, preventing it from being disturbed or trapped.

## *Voting Classifier*

Perhaps the least demanding approaches to consolidate forecasts from a few AI calculations is by voting classifier. Casting a voting classifier is a covering for a bunch of various classifiers that are prepared and assessed in equal to exploit every calculation's interesting highlights. We can utilize various calculations and outfits to prepare informational collections and afterward foresee the last exhibition. The ultimate result of a figure is controlled by a majority vote dependent on hard and delicate democratic.[13]

## *Flask*

Flask is a Python-based web staging in tiny. It is referred to as a small system because it does not necessitate the use of specific assets or libraries. It lacks a data set reflection layer, structure approval, and any other parts that rely on third-party libraries to do basic tasks. Augmentations, then again, might be utilized to join application usefulness as though they were executed in Flask itself. There have been enhancements to article social mappers, structure approval, transfer the board, various open validation advances, and other system-related apparatuses.[10]

# RELATED WORK

Sentimental analysis is a quickly extending field of Natural Language Processing, with considers going from record characterization to understanding the extremity of words and expressions. Since tweets have a set number of characters, characterizing the feeling of Twitter messages is generally tantamount to sentence-level sentiment analysis. Twitter sentiment analysis, on the other hand, is a somewhat different challenge due to the informal and technical terminology used in tweets, as well as the existence of the microblogging domain. How well the features and techniques used on more well-formed data can translate to the microblogging domain is an open question. [1]

To detect sentiments in text, there are two simple methodologies. Symbolic techniques and Machine Learning strategies are the two sorts. These strategies are talked about in the accompanying two pages. [2]

1. *Symbolic Techniques*

The use of available lexical tools is used in a lot of unsupervised sentiment classification research using symbolic techniques. Relationships between individual words are ignored in this approach, and a text is interpreted as a set of words. The general conclusion is determined by deciding the sentiments of each word and joining those qualities with some conglomeration functions. He found a survey's extremity by computing the normal semantic direction of tuples separated from the audit, where tuples are phrases with descriptive words or modifiers. Using the search engine Altavista, he discovered the connotation alignment of tuples. Due to the need for a large lexical database, the knowledge base solution has proven to be challenging. Sentiment analysis has become repetitive and inaccurate as social networks produce massive amounts of data continuously, often greater than the size of the accessible lexical dataset. [2]

1. *Machine Learning Techniques*

For characterization, machine learning techniques procedures utilize a preparation set and a test set. The info include vectors and their relating class marks are contained in the preparation set. An classification model is made utilizing this preparation set, which endeavours to group the info include vectors into comparing class names. The model is then approved by foreseeing the class marks of concealed capacity vectors utilizing a test set. [2]

To classify reviews, a variety of machine learning techniques such as Naive Bayes (NB), Maximum Entropy (ME), and Support Vector Machines (SVM) are utilized. Term Presence, Term Frequency, refutation, n-grams, and Part-of-Speech are a portion of the highlights that can be utilized for feeling order. The semantic direction of words, expressions, sentences, and archives can be resolved utilizing these highlights. The polarity of semantic orientation may be either positive or negative.[2]

Opinion mining is an expansive term that incorporates normal language processing, text mining, and computational etymology, and it involves the computational examination of assumptions, perspectives, and feelings communicated in text. Albeit a conclusion is a term used to portray a perspective or mentality that depends on feeling instead of rationale. Subsequently, lending to a comparable for notion investigation or opinion mining. Opinion mining has a wide range of applications, including accounting, law, science, entertainment, education, technology, politics, and marketing, according to the author. Many social media platforms used to provide web users with an outlet to express and share their thoughts and opinions.[3]

They explored new methods and techniques for sentimental analysis of unstructured data on the web in this study. Sentiment Analysis currently focuses on subjective statements or subjectivity, ignoring factual statements that carry sentiment (s). As a result, they suggested a new method for classifying and handling subjective and objective claims in sentimental analysis.[4]

N-gram features are used in all of the techniques listed above. It's uncertain whether or not using Part-of-Speech tagging is beneficial. Some use various technique of attribute preference or leverage knowledge of microblogging to improve accuracy. Utilizing more fundamental procedures utilized in Sentiment Analysis, for example, stemming, two-venture gathering, nullification identification, and extent of refutation, we fortify our exhibition. [5]

In sentiment analysis, negative identification is a method that has been extensively researched. Negative words such as "not," "never," "no," and "never" can dramatically alter the context of a sentence and thus the emotion contained in it. The sense of nearby words changes as a result of the existence of those words. The spectrum of negation is said to include such terms. Many studies have been conducted to determine the scope of negation.[5]

From that word to the next punctuation, a cue's scope of negation can be determined. There was a discussion on a method for determining the scope of negation cues in a sentence. They search the text for explicit negation clues and for each word in the scope. Then, at that point they figure the contrast among it and the closest bad signal on the left and right sides. [5]

The creators utilized emoji’s (e.g., ":)", ":(", and so on) to categorise as positive and negative tweets in a regulated characterization investigation on tweets in English. This methodology was utilized to assemble a collection of positive tweets including positive emoticons ":)" and negative tweets including negative emoticons ":(". Then, at that point utilizing different regulated and different arrangements of highlights to arrive at the resolution that basically utilizing unigrams creates great outcomes, however that consolidating unigrams and bigrams improves results barely. [6]

The creators utilize a crossover approach, joining managed learning with assumption bearing word data gathered from the DAL assessment word reference (Whissell, 1989). Retweets are taken out, truncations are converted into unique words and connections are erased, a tokenization cycle is performed, and grammatical feature labelling is performed. They use n-gram highlights with SVM and syntactic highlights including Partial Tree Kernels, just as information on the extremity of the terms showing up in the tweets, to arrange tweets into positive and negative classes. [6]

# PROPOSED WORK

Sentimental analysis is also termed as opinion mining, relates to a type of machine learning activity in which we try to figure out what a document's overall sentiment is. We may extract subjective information from a text and try to categorize it according to its duality, such as positive, neutral, or negative, with the assistance of machine learning and natural language processing. It is a very useful study because it allows us to decide the overall opinion about a selling object or forecast stock markets for a given business, for example, if the majority of people think it is a good idea, its stock prices will rise. Sentimental analysis is nowhere near being solved due to the complexity of the language, but this is also a reason why it is so fascinating to work on.

We built a model based on probabilities to classify tweets from Twitter into "positive" or "poor" sentiment. Twitter is a microblogging platform where users can express themselves easily and randomly by sending 140-character tweets. To categories Twitter tweets into "good" or "bad" sentiments, we developed a prototype based on prospects. Twitter is a chatroom/forum that allows users to quickly and spontaneously express themselves by sending 140-character tweets.

## *Data*

First of all, we segment the data set into two parts: the training and test sets. To do so, we first shuffle the data set to remove any order added to the data, then take 3/4 of the tweets from each set and combine them together to create the training set. The remainder will be used to construct the test set. Finally, the training set contains 1183958 tweets, while the test set contains 394654 tweets. It's worth noting that they're evenly distributed and follow the same pattern as the original data collection.

Additionally, in order to obtain a corpus of tweets, we must first ensure that we have a balanced data set, which means that we must have an equal number of positive and negative tweets, as well as a large enough data set. Indeed, the more data we have, the more we can train our classifier and the higher the accuracy.

Finally, because we'll be dealing with Twitter terms in the tweets, let's review the Twitter terminology:

* Hashtag: Every word or expression followed by the # symbol is referred to as a hashtag. You'll see other Tweets containing the same keyword or subject if you click on a hashtag.
* @username: On Twitter, you're known by your username, which is often followed by the @ symbol. For example, Shah Rukh Khan is @iamsrk.
* MT: Equivalent to the abbreviation RT (Retweet), which stands for "Modified Tweet. At the point when clients physically retweet a message with changes, such as minimizing a Tweet, this text is placed before the Retweeted text.
* Retweet:  RT, A Retweet is a Message that you forward to your followers. Retweets are often used on Twitter to share news or other useful discoveries, and they still keep the original attribution.
* Emoticons: They're made up of punctuation and letters and are used to convey feelings succinctly, ";) :)...”

## *Pre-processing*

We can pre-process the tweets now that we have the corpus of tweets and all of the tools that might be useful. It's critical because all of the changes we make during this phase would have a direct effect on the classifier's efficiency. Cleaning, normalisation, transformation, function extraction and selection, and so on are all part of the pre-processing. The effect of pre-processing would be consistent and uniform data that can be used to improve the efficiency of the classifier.

## *Emoticons*

Using the emoticon dictionary, we substitute all emoticons with their sentiment polarity ||pos|| and ||neg||. To perform the replacement, we go through each tweet and use a regex to see whether it contains emoticons; if it does, the emoticons are substituted with their polarity.

## *URL*

In their tweets, users often use hyperlinks. Twitter minimizes the url size using its own URL minimizing tool, such as http://t.co/FCWXoUd8; this allows Twitter to notify individuals if the connection prompts outside from its domain.  A specific URL is completely irrelevant from the perspectives of text classification.  The existence of a URL, on the other hand, can be a useful function. Because of the various types of URLs that can exist, the regular expression used to detect a URL is quite intricate, but according to Twitter's minimizing service, a comparably simple regular algorithm can be used, (http|https|ftp)://[a-zA-Z0-9./]+, which can be replaced by a simple term, URL.

## *Unicode*

We choose to exclude any unicode character that could be inaccurate for the classifier for the sake of simplicity and because the ASCII table should fit.

## *HTML entities*

HTML entities are reserved characters in HTML. To make them understandable, we need to decode them into characters entities.

## *Case*

The case may seem to be insignificant, but it is critical for distinguishing between proper nouns and other types of words. Indeed, "General Motor" and "General Motor", or “MSc” and “msc” are the same thing. As a result, converting all letters to lowercase should be done with caution. We won't worry about it for now because we think it won't have a significant effect on the classifier's results.

## *Targets*

The goal corresponds to Twitter usernames followed by “@” symbol. Its use is to address a tweet from a particular person or simply to draw attention to something. The tag ||target|| is used to override all usernames/targets.

## *Test Driven Calculations*

After we've generated the training and testing datasets, we'll need a third set of data called the validation set. It's extremely useful since it'll be used to test our model against unknown data and tune the learning algorithm's potential parameters to prevent underfitting and overfitting, for example. This validation set is needed since our test set can only be used to check how well the model generalises. If we use the test set instead of the validation set, our model will be too positive, causing the results to be skewed.

### *Logistic Regression*

After applying this algorithm, accuracy is 80.29%

### *Multinomial naive Bayes algorithm*

After applying this algorithm, accuracy is 77.50%

### *Stochastic Gradient Descent*

After applying this algorithm, accuracy is 77.53%

### *Voting Classifier*

After using Logistic Regression, Multinomial Naïve bayes Theorem and Stochastic Gradient Descent altogether, accuracy is 79.95%

# RESULT & DISCUSSION

Following table shows the result of each algorithm and expected model.

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Algorithms** | **Accuracy** |
| 1. | Logistic Regression | 80.29% |
| 2. | Multinomial Naïve Bayes Algorithm | 77.50% |
| 3. | Stochastic Gradient Descent | 77.53% |
| 4. | Voting Classifier | 79.95% |

# CONCLUSION

We tried to demonstrate how to use Naive Bayes as a baseline for classifying tweets whether they are positive or negative message category, as well as how language models are related to Naive Bayes and can provide us with better results. We might enhance our classifier by extracting more features from the tweets, experimenting with various types of features, the fine-tuning of parameters of the naive Bayes classifier, or using a different extractor altogether.

# ACKNOWLEDGEMENT

I thank my guide/mentor, Mr. Juginder Pal Singh (Assistant Professor) and especially my friends for their everlasting support which helped me in learning the different technologies’ working. They also helped me in completing the various tasks of the project in the most apt way possible.

# References

|  |  |
| --- | --- |
| [1] | T. J. M. Efthymios Kouloumpis, "Twitter Sentiment Analysis:," in *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*, Barcelona, 2011. |
| [2] | R. R. Neethu M S, "Sentiment Analysis in Twitter using Machine," in *4th ICCCNT*, Tiruchengode, 2013. |
| [3] | C. N. S. B. Aliza Sarlan, "Twitter Sentiment Analysis," in *2014 International Conference on Information Technology and Multimedia (ICIMU)*, Putrajaya, Malaysia, 2014. |
| [4] | M. M. M. Onam Bharti, "SENTIMENT ANALYSIS ON TWITTER DATA," *International Journal of Computer Science and Mobile Computing,* vol. 5, no. 6, pp. 601-609, 2016. |
| [5] | Y. G. a. N. Chatterjee, "Sentiment Analysis of Twitter Feeds," in *Springer International Publishing*, Switzerland, 2014. |
| [6] | A. Balahur, "Sentiment Analysis in Social Media Texts," in *Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, Atlanta, Georgia, 2013. |
| [7] | B. X. I. V. O. R. a. R. P. Apoorv Agarwal, "Sentiment Analysis of Twitter Data," in *Proceedings of the Workshop on Language in Social Media (LSM 2011)*, Portland, Oregon, 2011. |
| [8] | M. T. Rudy Prabowo, "Sentiment analysis: A combined approach," *Journal of Informetrics,* pp. 143-175, 2009. |
| [9] | E. R. H. ,. E. R. H. J. Nádia F.F. da Silva, "Tweet sentiment analysis with classifier ensembles," *Decision Support Systems,* p. 10, 2014. |
| [10] | R. I. S. A. M. A. Hanan Aljuaid, "Important citation Identification using Sentiment Analysis of In-text citations," *Telematics and Informatics,* 2020. |
| [11] | N. S. G. S. Priyavrat Chauhan, "The emergence of social media data and sentiment analysis in election," *Journal of Ambient Intelligence and Humanized Computing,* 2020. |
| [12] | P. C.-M. A. B. V. L. B. M. Alberto Tonon, "ArmaTweet: Detecting Events by semantic tweet analysis," in *Springer International Publishing AG 2017*, Vienna, 2017. |
| [13] | S. G. a. K. V. Kumar, "Tweet Analysis Based on Distinct Opinion of Social Media Users’," in *2018 International Conference on Soft-computing and Network Security (ICSNS)*, Coimbatore, India, 2018. |
| [14] | J. Y. Richard D.Watersa, "Tweet, tweet, tweet: A content analysis of nonprofit organizations’ Twitter updates," *Public Relations Review 37 (2011),* vol. 37, no. 3, pp. 321-324, 2011. |
| [15] | S. K. D. A. Hikmat Ullah Khan, "Twitter trends: A ranking algorithm analysis on real time data," *Expert Systems With Applications 164 (2021),* vol. 164, 2021. |