ISSN: 0974-5823 Vol. 7 No. 4 April, 2022

International Journal of Mechanical Engineering

SENTIMENT ANALYSIS FOR TWITTER

Sunjol Singh Paul

Student, School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, Tamil Nadu - 632014, India

Siddarth Anoop Nayar

Student, School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, Tamil Nadu - 632014, India

Dr. Natarajan P

Associate Professor Grade 2, School of Computer Science and Engineering, Vellore Institute of Technology - 632014, Vellore, Tamil Nadu, India

Abstract

Sentiment Analysis is a type of Natural Language Processing (NLP) technique which is used to the feelings associated with a particular text or data. Although there are several models that test for sentiment analysis, each model gives a different result based on how they perceive a tweet. The subjective nature of sentiments renders reliance on a single model to be unsound and results to be deficient. Therefore, to produce accurate results, we propose a model which combines both Machine Learning and lexical-based approaches for better classification of tweets into their respective polarity which yields better accuracy. Existing NLP models like Stanza, Textblob and NLTK are tested and combined with ML classifiers kNN, MultinomialNB, SVC and Random Forest in the process of hybridization to develop a new model which produces the best of accuracy (nearly 91%). This observation suggests that using hybridization is desirable for achieving superior accuracies when compared with standalone models and in achieving finer sentiment analysis.

Keywords: Natural Language Processing; Sentiment Analysis; classification; hybridization.

1. Introduction

Sentiment Analysis is the understanding and order of feelings with the content information utilizing content investigation methods. It is frequently employed on any type of textual data to assist any businesses or social media platforms in monitoring any brand or products that businesses sell, or different types of reactions triggered by various social events posted by people on any respective social media platforms. Machine Learning-based sentiment analysis and Lexical-based sentiment analysis are the two basic types of sentiment analysis approaches. Supervised classification algorithms, where sentiment detection is framed as a binary, are commonly used in Machine Learning-based systems (i.e., positive, or negative). To train classifiers, this method requires labelled data. While learning-based methods have the advantage of being able to adjust and develop trained models for specific purposes and settings, they also have the disadvantage of requiring labelled data, limiting their application to new data. This is due to the fact that labelling data might be costly or even impossible for some jobs. Lexical-based techniques, on the other hand, use a predefined list of words, each of which is associated with a distinct attitude. The lexical approaches used differ depending on the context in which they were developed.

While this is one of the hot fields of discussion over the past decade, the existing research only employs either the Machine Learning based techniques or Lexical-based techniques. This does not help in achieving accurate polarity classification. One of the primary reasons why this approach fails is because sentiments are highly impressionistic. Hence, based on the algorithm employed within the model, the polarities produced and the accuracies achieved highly differ model-to-model.

This paper sets out to address this gap. Using both Machine Learning as well as Lexical-based approaches, we take the positives out of each approach and create a hybrid model which displays an astute accuracy.

2. Literature Survey

The authors in [1] discuss about how social networking platforms, forums, review sites and blogs generate enormous amounts of data in the form of various views, emotions, opinions, and arguments that are presented by different people on various social events, products, businesses and political events. They also talk about how all this unstructured data needs to be analysed so as to provide people with correct views and opinions as it can influence their minds easily. The authors further stated about the challenges to performing Sentiment Analysis, one such challenge being the lack of sufficient labelled data to work with in the Natural Language (NLP). To counter this issue, the authors talked about how Sentiment Analysis and Deep Learning (DL) techniques have come together since deep learning models are effective when it comes to their automatic learning capability. Some models mentioned in this paper are deep neural networks, convolutional neural networks, recursive neural networks, recurrent neural networks, deep belief networks, hybrid neural networks.

Wei Xue and Tao Li talk about how Aspect-based sentiment analysis (ABSA) provides more comprehensive information than a general sentiment analysis in [2] as the former goes on to predict the sentiment polarities of any of the given aspects or entities of a text. They classified the previous approaches into two subtasks which are Aspect Category Sentiment Analysis (ACSA) and Aspect term Sentiment Analysis (ASTA). Furthermore, they explained about the shortcomings of previous approaches that utilize long short-term memory and attention mechanisms to predict the polarity which at the same time is more complex and takes enormous training time. To counter this issue, they proposed a model with a simpler architecture that can selectively output the sentiment features according to a given aspect or entity. Moreover, since convolutional layers have an absence of time dependency, the computations of their model could be parallelised easily during training period.

In the paper [3], Amrita Shelar and Ching-yu-Huang demonstrated Sentiment Analysis on Twitter with their data as tweets regarding donations, fundraisers, or charities. Their primary aim was to capture sentiments of people towards donating for a cause. For their model they used Natural Language Processing Toolkit (NLTK) wherein they used to NLTK VADER analyser which identifies and categorises text into three sentiments that are positive, neutral, or negative. They concluded in their work that the word “charity” had the highest polarity.

Zhao Jianqiang explains in [4] that the majority of current research focuses on extracting sentiment features from lexical and syntactic features. Here, sentiment words, emoticons, exclamation marks, and other symbols are used to express these characteristics. The paper introduces a word embeddings technique based on big twitter corpora that uses latent contextual semantic associations and co-occurrence statistical properties between words in tweets. To create a sentiment feature set of tweets, these word embeddings are merged with n-grams features and word sentiment polarity score features. The feature set was used to train and predict sentiment classification labels using a deep convolutional neural network. The model was compared with the baseline model (n-grams model) on five Twitter data sets and the results produced showed that the model performed better on accuracy and F1 measure.

In the paper [5], Ali Hasan and others say that despite the usage of a variety of machine-learning algorithms and tools for sentiment analysis during elections, a state-of-the-art methodology is urgently needed. To address these issues, their paper's contribution is the use of a hybrid technique that comprises a sentiment analyser as well as machine learning. The paper also provides a comparison of techniques by applying supervised machine learning algorithms such as Naïve Bayes and Support Vector Machines (SVM). They used Textblob, SentiWordNet and Word Sense Disambiguation (WSD) sentiment analysers and in order to validate the results they used Waikato Environment for Knowledge Analysis (Weka). They achieved results with Textblob producing the highest accuracy (62.67%) and concluded that Textblob and WSD (62.33%) are better than SentiWordNet for making more accurate predictions.

In the paper [6], Lei Wang and other authors talk about how most present Twitter sentiment analysis tools simply consider textual information from tweets and fail to perform effectively when dealing with short and ambiguous tweets. Furthermore, they say that recent research has found that sentiment dissemination patterns on Twitter are linked to the polarities of Twitter posts. In this paper, they look into ways to combine textual information from Twitter tweets with sentiment diffusion patterns to improve sentiment analysis results on Twitter data. They investigate a phenomenon called Sentiment Reversal wherein a tweet and its retweet have different sentiment polarities. We analyse the properties of sentiment reversals and proposed a sentiment reversal prediction model. In the model, they explored the interplay between textual information in Twitter messages and sentiment diffusion patterns, and introduced SentiDiff, an iterative algorithm for predicting sentiment polarities in Twitter communications. The results showed PR-AUC improvements between 5.09% and 8.38% with experiments on real-world datasets on Twitter classification tasks.

The authors in [7] talk about how the usefulness of various sentiment categorization strategies, ranging from simple rule-based and lexicon-based approaches to more advanced machine learning algorithms, highlighted in recent studies. The paper also mentions the shortcomings of lexicon-based approaches that are lack of dictionaries and labelled data, and machine learning approaches which suffered in terms of accuracy. Moreover, to bridge the gap between lexicon-based and machine learning approaches, the paper suggests an integrated genetic algorithm (GA) based feature reduction technique. Using this hybrid, the authors were able to reduce feature-set size by 42% without compromising accuracy. Lastly, the comparison with principal component analysis (PCA) and latent semantic analysis (LSA) based feature reduction techniques displayed 15.4% increased accuracy and 40.2% increased accuracy with PCA and LSA respectively.

In the paper [8], the authors Sahar A. El_Rahman and others how companies can benefit from the data collected from microblogging platforms such as Twitter where people express their views, opinions, and concerns openly. However, since the data available on Twitter highly unstructured and unorganised, it makes it difficult to analyse. Hence, the authors proposed a model which combined the aspects of supervised and unsupervised machine learning algorithms. They used unsupervised learning algorithms to label data as -1, 0, +1 (negative, neutral, and positive respectively) and also used multiple supervised learning algorithms for the purpose of training like Naive Bayes, support vector machine (SVM), maximum entropy, decision tree, random forest, and bagging. Various testing criteria, such as cross validation and f-score, were used to test the results of these models.

Maryem Rhanoui and the other authors in their paper [9], explains how sentiment analysis at the document level is difficult due to the vast size of the text, which results in a high number of words and opinions, some of which are conflicting, in the same document. They mention some of the benefits for studying press articles and blog posts on a specific product or company, as it necessitates a high level of focus, especially when the subject under discussion is sensitive. Nonetheless, the majority of existing models and methodologies are built to handle short text from social media and collaborative platforms due to which they cannot perform document level sentiment analysis. In this paper, the authors proposed a model for opinion analysis in long texts that combines Convolutional Neural Networks (CNN) and Bidirectional Long Short-Term Memory (BiLSTM) models with Doc2vec embedding. With Word2vec/Doc2vec embeddings, the CNN-BiLSTM model was compared to CNN, LSTM, BiLSTM, and CNN-LSTM models. On French newspaper articles, the Doc2vec with CNN-BiLSTM model outperformed the other models with a 90.66% accuracy.

In the paper [10], László Nemes and Attila Kiss explain how in today's society, social media is ubiquitous, and everyone comes into contact with it on a daily basis and perform a lot of research and statistics using social media data. Within the scope of this paper, the authors conclude and analyse the sentiments and manifestations (comments, hashtags, posts, tweets) of Twitter users using Natural Language Processing and Sentiment Classification using Recurrent Neural Networks, based on the main trends (by keyword, which in this article is mostly the 'covid' and coronavirus theme) and analyse, assemble, visualise, and summarise statistics for further processing. the trained model operates considerably more accurately, with a reduced margin of error, in predicting emotional polarity especially when it comes to RNN.

According to Li Yang and others [13] explain how online shopping has become a popular way for people to buy and consume goods, and user satisfaction can be improved by analysing the sentiment of a huge number of user reviews on e-commerce sites. For the same, the authors of this paper have proposed SLCABG, a new sentiment analysis model based on the sentiment lexicon that blends Convolutional Neural Networks (CNN) and attention-based Bidirectional Gated Recurrent Units (BiGRU). In terms of methodology, the SLCABG model combines the benefits of sentiment lexicon with deep learning technology to solve the inadequacies of previous product review sentiment analysis models. The benefits of sentiment lexicons and deep learning techniques are combined in the SLCABG model. Explaining the model, firstly, the sentiment lexicon is applied to the reviews to improve the sentiment features. Next, The CNN and the Gated Recurrent Unit (GRU) network are then used to extract the major sentiment and context elements from the reviews, which are then weighted using the attention method. Finally, classify the sentiment features that are weighted. The model was trained and tested on a Chinese website, and the scale of data reached 1,00,000 orders of magnitude which proved to be helpful in Chinese sentiment analysis.

In the paper [14], Mohammad Ehsan Basiri and other authors mention that in recent times Deep neural network (DNN) models are being used to do sentiment analysis tasks, and their findings are encouraging. Moreover, Long short-term memory (LSTM) models and its derivatives, such as gated recurrent unit (GRU), have gotten a lot of attention among the many neural architectures used for sentiment analysis. However, although these models can analyse sequences of any length, utilising them in the feature extraction layer of a DNN increases the dimensionality of the feature space. Another disadvantage of such models is that they value diverse characteristics equally. To tackle these problems, the authors proposed an Attention-based Bidirectional CNN-RNN Deep Model (ABCDM). Explaining the model, ABCDM will extract both past and future contexts by evaluating temporal information flow in both directions using two independent bidirectional LSTM and GRU layers. In addition, the attention mechanism is used on the outputs of ABCDM's bidirectional layers to place more or less emphasis on various words. ABCDM employs convolution and pooling algorithms to reduce feature dimensionality and extract position-invariant local features. Five review and three Twitter datasets were used in the experiments. ABCDM achieves state-of-the-art performance on both long review and short tweet polarity classification when compared to six previously suggested DNNs for sentiment analysis.

Ranjan Kumar Behera and others in [15] propose a hybrid strategy combining two deep learning architectures, the Convolutional Neural Network (CNN) and the Long Short-Term Memory (LSTM) (RNN with memory), is for sentiment categorization of reviews from various fields. Moreover, Deep convolutional networks have shown to be very successful in local feature selection, whereas recurrent networks (LSTM) have shown to be very effective in the sequential processing of large texts. The Co-LSTM model is primarily focused on two sentiment analysis goals. First, it is very adaptive in analysing large amounts of social data while keeping scalability in mind, and second, unlike traditional machine learning approaches, it is domain-agnostic. The experiment was conducted on four review datasets from various disciplines in order to train a model that can handle all types of dependencies that can occur in a post. In terms of accuracy and other characteristics, the suggested ensemble model surpasses previous machine learning algorithms, according to the results.

3. Sentiment Analysis and Twitter Background

3.1. What is Sentiment Analysis

Sentiment Analysis is the study of feelings in relation to content information using content investigation techniques. We employ Natural Language Processing (NLP) techniques to extract any subjective information that aids in determining whether a data set is positive, negative, or neutral. Sentiment Analysis is frequently performed on any type of textual data (which may or may not include emojis) to assist any businesses or social media platforms in monitoring any brand or products that businesses sell, or different types of reactions triggered by various social events posted by people on any respective social media platforms.

3.2. Types of Sentiment Analysis methods

While the primary aim of Sentiment Analysis lies on understanding the polarity of a text (positive, negative, or neutral), it can also detect specific moods and emotions (anger, content, sorrow), urgency (whether urgent or not) and even intents (interested or not). Depending on how a person interprets any tweets or data, categories can be tailored to meet your sentiment analysis requirements. Ad interim, some of the popular types of sentiment analysis methods are:

- Graded Sentiment Analysis

If fidelity in polarity is important for your understanding in tweets or data, one such method is expanding the polarity categories to include different levels of positive or negative, i.e., Very Positive, Positive, Neutral, Negative, Very Negative. One way of achieving this may be to incorporate a 5-star rating to classify a tweet into any of the aforementioned categories. This can help your model produce more precise polarity.

- Emotion Detection

Emotion detection sentiment analysis is a type of sentiment analysis that allows you to detect emotions such as Happiness, Frustration, Anger, Sadness etc. To achieve this, normally lexicons (a list of words and the respective emotions conveyed) are used along with some Machine Learning algorithms such as Support Vector Machine (SVM), Random Forest (RF) and Nearest Neighbour Algorithm (kNN). However, one of the downsides of using lexicons is that emotions can be expressed differently by different people. Some words that usually express a particular emotion may express something else when phrased differently.

- Aspect-based Sentiment Analysis

While analysing tweets for sentiments, you would like to know which aspects people are mentioning in a positive, negative, or neutral manner. When an aspect-based sentiment analysis is applied, it will be able to classify whether a particular feature of a tweet expresses positive or negative or neutrality of tweet.

There are about two main types of sentiment analysis methods – Machine Learning-based and Lexical-based. Machine Learning based methods often rely on supervised classification approaches, where sentiment detection is framed as a binary (i.e., positive or negative). This approach requires labelled data to train classifiers [12]. While one advantage of learning-based methods is their ability to adapt and create trained models for specific purposes and contexts, their drawback is the availability of labelled data and hence the low applicability of the method on new data. This is because labelling data might be costly or even prohibitive for some tasks [11]. On the other hand, lexical-based methods make use of a predefined list of words, where each word is associated with a specific sentiment. The lexical methods vary according to the context in which they were created [11].

3.3. Twitter

Twitter is a microblogging and social networking website based in the United States that allows users to send and receive messages known as "tweets." Registered users can write, like, and retweet tweets, but unregistered individuals can only read those that are publicly viewable. It is one of the most popular social media platforms with over 100 million daily active users and 500 million tweets sent daily. People use twitter to express their opinions, emotions and express their views about certain topics which are part of everyday life. It is also a platform where people provide their respective reviews about a certain service provided by a company which can influence a customer whether or not to go for the service or not after checking such reviews.

4. Overview

4.1. Introduction and Related Concepts

Random Forest is a well-known machine learning algorithm that uses the supervised learning method. It can be used to solve difficulties in both classification and regression. It is based on ensemble learning, which is a method of integrating several classifiers to solve a complex problem and increase the model's performance. Random Forest is a classifier that combines a number of decision trees on different subsets of a dataset and averages the results to increase the dataset's predicted accuracy. Instead, then relying on a single decision tree, the random forest collects the forecasts from each tree and predicts the final output based on the majority votes of predictions.

KNN (K Nearest Neighbours) is a simple and commonly used classification technique that classifies a new data point based on its resemblance to a group of nearby data points. This produces a competitive outcome. The algorithms calculate the distances between a given data point in the set and any other K numbers of data points in the dataset that are close to the initial point, then vote for the category with the highest frequency. Typically, Euclidean distance is used as a distance measurement. As a result, the final model is just labelled data in a space.

In Natural Language Processing, the Multinomial Naive Bayes method is a common Bayesian learning approach (NLP). Using the Bayes theorem, the programme estimates the tag of a text, such as an email or a newspaper piece. It assesses the likelihood of each tag for a given sample and returns the tag with the highest probability. The multinomial Naive Bayes classifier is good for discrete feature classification (e.g., word counts for text classification). Normally, integer feature counts are required for the multinomial distribution. Fractional counts, such as tf-idf, may also function in practise.

4.2. Framework

Some of the important frameworks that we will be using in this project are:

- TextBlob is a text processing package for Python 2 and 3. It offers a basic API for doing standard natural language processing (NLP) activities like part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, and translation, among others.
- The Natural Language Toolkit (NLTK) is a popular Python environment for working with human language data. It includes a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum, as well as easy-to-use interfaces to over 50 corpora and lexical resources like WordNet.

Tokenizing, part-of-speech tagging, stemming, sentiment analysis, topic segmentation, and named entity recognition are among the algorithms included in NLTK. NLTK aids the computer's analysis, pre-processing, and comprehension of written language. The initial step in text analytics is to tokenize the data. Tokenization is the process of breaking down a text paragraph into smaller parts such as words or phrases. A token is a single element that forms the foundation of a sentence or paragraph. The paragraph is broken down into sentences, which are then broken down into words, and so on. The basic goal of Part-of-Speech (POS) tagging is to determine a word's grammatical category. Depending on the context, it could be a noun, pronoun, adjective, verb, or adverb. POS Tagging looks for connections between words in a sentence and assigns a tag to each one.

- Stanza is a Python library for natural language processing. It includes tools for converting a string of human language text into lists of sentences and words, generating basic forms of those words, parts of speech, and morphological features, performing a syntactic structure dependency parse, and recognising named entities in a pipeline. Using the Universal Dependencies formalism, the toolkit is designed to run in parallel across more than 70 languages. Stanza is a neural network-based system that looks for sentiment in a dataset.

5. Methodology

5.1. Proposed System Model

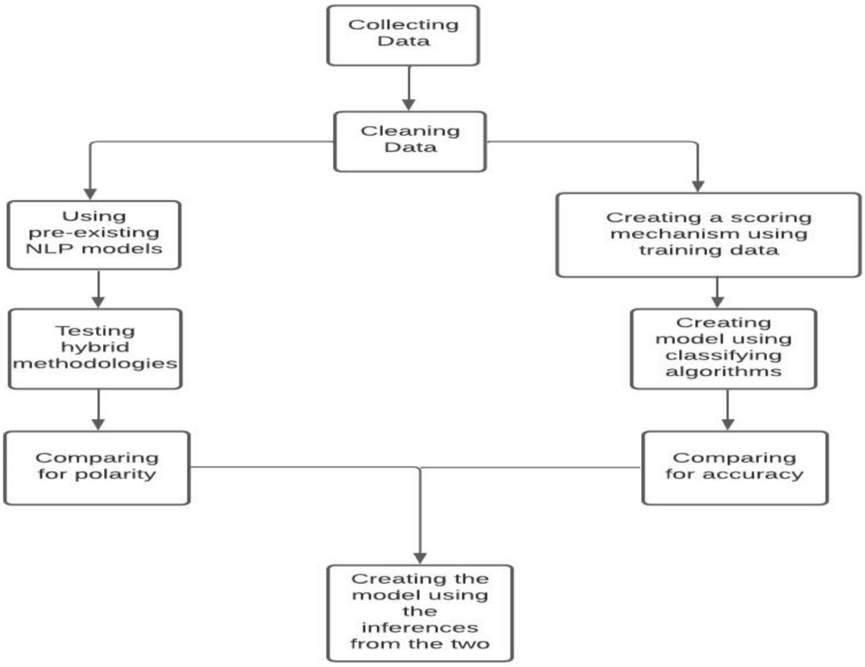

Fig. 1. Module Diagram

The above figure shows the process of doing sentiment analysis. We first collect the data following which the data is cleaned by removing any retweets and repeated words. Next the process is split into two phases; in the first phase, we use the pre-existing NLP models such as Stanza, Textblob and NLTK and test for hybrid methodologies. On completion of this, we go ahead and compare the polarities of the hybrid methodologies. We go ahead with the methodology having a more balanced polarity. In the second phase, we create a scoring mechanism using the training data and create models using classifier algorithms like MultinomialNB, Random Forest, kNN and SVC. Following this, we compare the accuracy of these respective classifiers and use the ones with the highest accuracy.

Next, we combine the hybrid methodologies obtained in the first phase along with the classifiers to produce a model which has a balanced polarity as well as high accuracy.

5.2. Steps for the process

This section explains the steps that were followed for the entire process

(1) Collection of data

We use Python Twitter API Tweepy from which we can get tweets in an unlabelled raw form. This helps recreate the real-time environment. The data acquired here is used to test the hybrid methods. However, for the classifier algorithms, we use the dataset obtained from Kaggle.

(2) Cleaning of Data

In the next step, we move ahead and clean the data by removing any unnecessary words and retweets which seem redundant, using the NLTK package. Moreover, this helps in making the word bag more efficient and saves memory.

After step 2, the process is split into two sub sections. In the first sub section we are trying to select a stable hybridisation method while in the second section we are trying to select an accurate classifier algorithm.

(3) Selecting the Hybrid Model

(i)Initialization of Models

Here, we have selected three existing NLP models which are NLTK VADER, Textblob Analyzer, and Stanza. These models already have a pre-existing standardisation of scores for each word. The main reason we use these models to assess the hybridisation methods is because sentiments are subjective in nature and can cause outliers.

(ii)Training the Model

After the initialization and cleaning process of data, the models are training using the cleaned data and their polarities are compared.

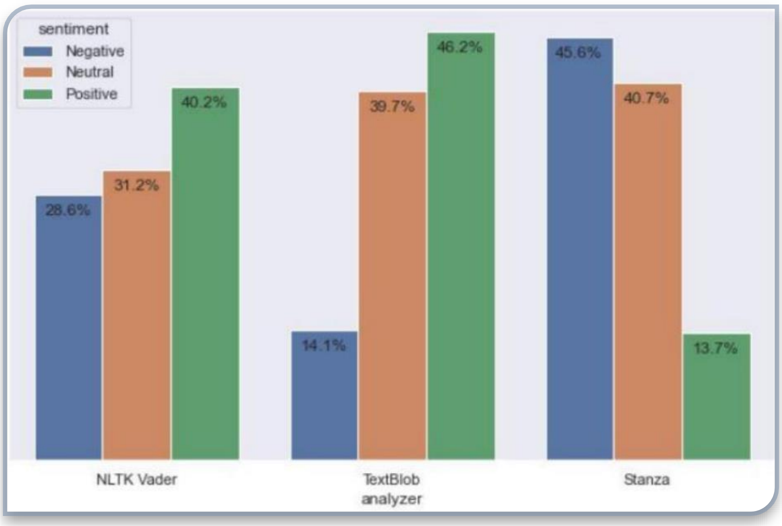

Figure 2. Polarity of existing NLP Models

(4) Selecting Classifier Algorithm

(i)Word Bag Creation

We maintain a count of number of times a word occurs in the statement of an expected result and score it accordingly.

(ii)Training the Classifier Models

After the creation of the scoring methodology, the classifier algorithms are used to train the model.

kNN Classifier

We load the data and next we select a K value. Now we calculate the distance between test data and each row using Euclidean distance method.

$$d = \sqrt{[(x2 - x1)2 + (y2 - y1)2]} \qquad (1)$$

where, (x1, y1) are the coordinates of one point and (x2, y2) are the coordinates of the other point. d is the distance between (x1, y1) and (x2, y2).

Based on the distance, we sort them in ascending order. Next, we select the top K rows from the sorted array. Next, it will assign a class to the test point based on most frequent class of these rows.

SVC Classifier

The objective is to fit to the data provided, returning a “best fit” hyperplane that divides, or categorizes the data. After getting the hyperplane, we feed some features into the classifier and verify what the “predicted” class is.

MultinomialNB Classifier

Here we use the Naïve Bayes algorithm which is used for text data analysis. The classifier mainly employs the Bayes Theorem and calculates the probability of an event occurring based on prior knowledge of conditions related to the event. The formula is given below,

$$P(A|B) = P(A) * P(B|A)/P(B) \qquad (2)$$

Here, P(B) is prior probability of B, P(A) is prior probability of class A, and P(B|A) is occurrence of predictor B given class A probability.

Random Forest Classifier

This is a process that operates among multiple decision trees to get the optimum result by choosing the majority among them as the best value. Here, we select random K data points from the training set. Next, we build the decision trees associated with the selected data points (subsets). Choose the number N for decision trees that you want to build and repeat the steps. Finally, for new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes.

After the training of models, we compare the models for their accuracy.

6. Results and Discussions

After the first phase of the project, we were able to come up with two models namely Average Voting Model (AVM) and Maximum Voting Model (MVM). We had used the standardization of the existing NLP models of NLTK, Textblob Analyzer and Stanza. The models gave the respective results as shown below

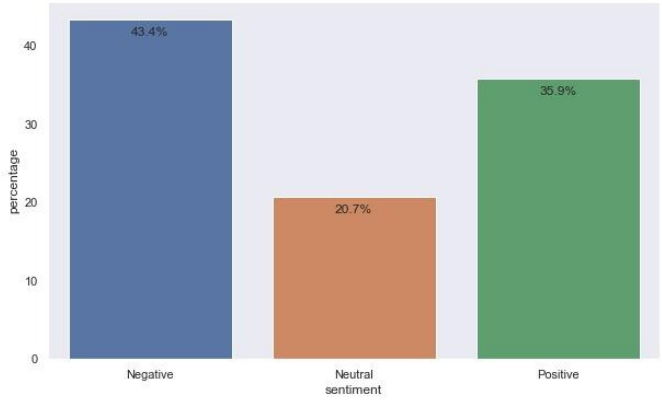

Figure 3. Polarity of Average Voting Model

Figure 3 represents the results obtained from the AVM. Here, we can see that the polarity of the tweets is more stable. We see that about 43.4% of the tweets are classified as Negative, 20.7% are Neutral and 35.9% as Positive.

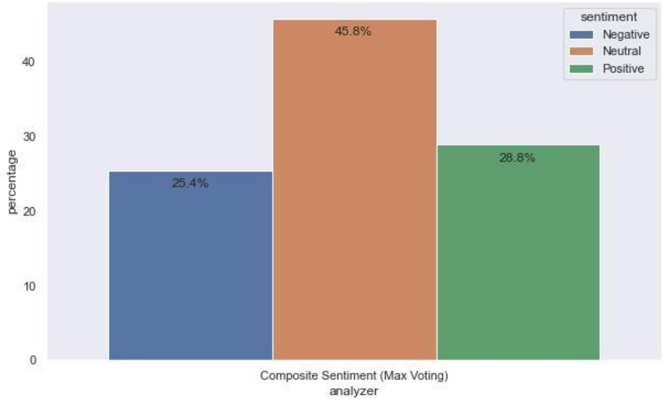

Figure 4. Polarity of Maximum Voting Model

Figure 4 shows that in contrast to AVM, MVM displays majority of the tweets as Neutral (45.8%). We see that only 25.4% and 28.8% tweets are classified as Negative and Positive respectively.

In the phase two of the process, we had tested various classifier algorithms using the data obtained from Kaggle dataset. The models were then compared based on their accuracies and the results are shown below

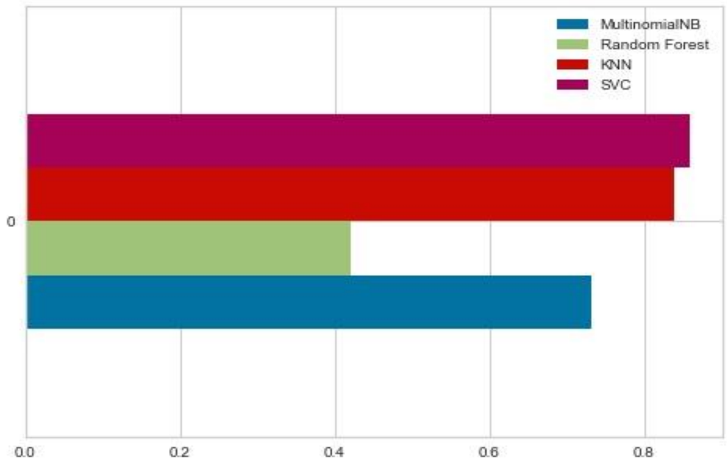

Figure 5. Accuracies of classifiers

Here, we observe that kNN and SVC classifiers outperformed the others with accuracies greater than 0.84 and 0.86 (out of 1.0) respectively. Hence, we move ahead with the kNN and SVC classifiers and Textblob to create a hybrid model along with Maximum Voting Model (MVM) while standardizing the scores.

Table 1. Accuracies of Hybrid Model and classifiers

Models	Accuracy (out of 1.0)
Hybrid Model	0.9088
MultinomialNB	0.7326
Random Forest	0.4200
kNN	0.8393
SVC	0.8586
NLTK	0.4242
Textblob	0.4292

Table 1 shows the models and their respective accuracies. Here, we can clearly see that the accuracy of the Hybrid model is almost 91%, and this shows the best accuracy compared to other models.

7. Conclusion

Sentimental analysis is the process of understanding and organizing feelings in connection to content information using content research methods. Despite the fact that there are a variety of sentiment analysis models accessible, each model yields a different result depending on how they read a tweet. Because feelings are subjective, we can't rely on a single model to give accurate results. To address this, we presented a hybrid model that incorporates existing NLP methods. To create the hybrid models, we used Textblob standardization scores and the Maximum Voting Model, as well as kNN and SVC classifiers. The classifiers kNN and SVC provided accuracy of 83.9 percent and 85.8%, respectively. The hybrid model, on the other hand, produced a 90.8 percent accuracy. As a result, we can conclude that employing a hybrid model is beneficial to standalone models, and that this will benefit the discipline of opinion mining.

References

[1] Ain, Qurat Tul, et al. "Sentiment analysis using deep learning techniques: a review." Int J Adv Comput Sci Appl 8.6 (2017): 424.

[2] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In ACL Conference on Empirical Methods in Natural Language Processing, pages 79–86, 2002.

[3] Basiri, Mohammad Ehsan, et al. "ABCDM: An attention-based bidirectional CNN-RNN deep model for sentiment analysis." Future Generation Computer Systems 115 (2021): 279-294.

[4] Behera, Ranjan Kumar, et al. "Co-LSTM: Convolutional LSTM model for sentiment analysis in social big data." Information Processing & Management 58.1 (2021): 102435.

[5] El Rahman, Sahar A., Feddah Alhumaidi AlOtaibi, and Wejdan Abdullah AlShehri. "Sentiment analysis of twitter data." 2019 international conference on computer and information sciences (ICCIS). IEEE, 2019.

[6] Gonçalves, Pollyanna, et al. "Comparing and combining sentiment analysis methods." Proceedings of the first ACM conference on Online social networks. 2013.

[7] Hasan, Ali, et al. "Machine learning-based sentiment analysis for twitter accounts." Mathematical and Computational Applications 23.1 (2018): 11.

[8] Iqbal, Farkhund, et al. "A hybrid framework for sentiment analysis using genetic algorithm based feature reduction." IEEE Access 7 (2019): 14637-14652.

[9] Jianqiang, Zhao, Gui Xiaolin, and Zhang Xuejun. "Deep convolution neural networks for twitter sentiment analysis." IEEE Access 6 (2018): 23253-23260.

[10] Nemes, László, and Attila Kiss. "Social media sentiment analysis based on COVID-19." Journal of Information and Telecommunication 5.1 (2021): 1-15.

[11] Rhanoui, Maryem, et al. "A CNN-BiLSTM model for document-level sentiment analysis." Machine Learning and Knowledge Extraction 1.3 (2019): 832-847.

[12] Rustam, Furqan, et al. "A performance comparison of supervised machine learning models for Covid-19 tweets sentiment analysis." Plos one 16.2 (2021): e0245909.

[13] Shelar, Amrita, and Ching-Yu Huang. "Sentiment analysis of twitter data." 2018 International Conference on Computational Science and Computational Intelligence (CSCI). IEEE, 2018.

[14] Wang, Lei, Jianwei Niu, and Shui Yu. "SentiDiff: combining textual information and sentiment diffusion patterns for Twitter sentiment analysis." IEEE Transactions on Knowledge and Data Engineering 32.10 (2019): 2026-2039.

[15] Xue, Wei, and Tao Li. "Aspect based sentiment analysis with gated convolutional networks." arXiv preprint arXiv:1805.07043 (2018).

[16] Yang, Li, et al. "Sentiment analysis for E-commerce product reviews in Chinese based on sentiment lexicon and deep learning." IEEE access 8 (2020): 23522-23530.