International Journal of Communication and Information Technology

**E-ISSN:** 2707-6628
**P-ISSN:** 2707-661X
www.computersciencejournals.com/ijcit
IJCIT 2023; 4(1): 71-83
Received: 20-01-2023
Accepted: 27-02-2023

**Richa Dhanta**
Chandigarh University Engineering Department specialization Artificial Intelligent and machine Learning, Shimla, Himachal Pradesh, India

**Hardwik Sharma**
Chandigarh University Engineering Department Specialization Artificial Intelligence and machine Learning, Punjab, India

**Vivek Kumar**
Chandigarh University Engineering Department Specialization Artificial Intelligence and machine Learning, Haryana, India

**Hari Om Singh**
Chandigarh University Engineering, Department Specialization Artificial Intelligence and machine Learning, Uttar Pradesh, India

**Corresponding Author:**
**Richa Dhanta**
Chandigarh University Engineering Department specialization Artificial Intelligent and machine Learning, Shimla, Himachal Pradesh, India

# Twitter sentimental analysis using machine learning

**Richa Dhanta, Hardwik Sharma, Vivek Kumar and Hari Om Singh**

**DOI:** https://doi.org/10.33545/2707661X.2023.v4.i1a.63

**Abstract**
This research paper aims to explore the effectiveness of machine learning algorithms in analyzing sentiment on Twitter. The study utilizes a dataset of tweets collected from various sources, which were then preprocessed to remove noise and irrelevant data [4, 5]. To categorize the tweets as positive, negative, or neutral, a number of machine learning techniques were used, such as logistic regression and Naive Bayesian [1]. The efficiency of these algorithms is also assessed in the study using a number of criteria, including accuracy, precision, recall, and F1 score. The results indicate that machine learning algorithms are effective in analyzing sentiment on Twitter, with Naive Bayes providing the best performance [18]. The results of this study have significant ramifications for companies and organizations looking to track consumer opinion of their goods or services [7]. This paper examines the problem of analyzing sentiment in Twitter by examining the tweets' expressed sentiments—whether they be favourable, negative, or neutral. Natural language processing methods will be used to analyze the messages that are tweeted.

**Keywords:** Naive bayes, support vector machine, random forest, feature extraction, naive sentiment analysis, linear regression, KNN

## Introduction

Twitter has emerged as a prominent forum for discussing strong emotions, making it a valuable source of information for sentiment analysis [17, 12]. The process of examining text to ascertain the underlying emotional tone is known as sentiment analysis. With the growth of social media platforms like Twitter, sentiment analysis has come an important tool for businesses, associations, and governments to understand public opinion and to make informed opinions. Natural Language Processing (NLP) ways are extensively used for sentiment analysis, as they allow machines to understand and interpret human language. NLP ways can analyze tweets in real- time, identify the sentiment of tweets, and give perceptivity into trends and patterns in public sentiment. Machine literacy algorithms, a subset of NLP, can learn from large datasets and predict the sentiment of new tweets with high delicacy [14].
In this study, we investigate how well machine learning systems do sentiment analysis on Twitter using NLP methods. To categorize tweets as positive, negative, or neutral, we will take a dataset of tweets gathered from multiple sources, preprocess the data to eliminate noise, and then apply machine learning algorithms like logistic regression and Naive Bayes [5, 15]. The effectiveness of these algorithms will be assessed using a variety of criteria, including accuracy, precision, recall, and F1 score. The results of this study have significant ramifications for companies and organizations looking to track consumer opinion of their goods or services. By using sentiment analysis, organizations can make informed decisions about marketing strategies, customer engagement, and crisis management [11]. Additionally, the study advances sentiment analysis NLP techniques, which have applications in a variety of industries, including politics, healthcare, and social sciences.
A few years ago, machine learning algorithms have been applied to twitter sentiment analysis, allowing researchers to extract valuable insights into public opinion on various topics. A variety of industries, including political campaigns, managing brand reputation, and analysing customer feedback, can benefit from Twitter sentiment research. The ability to analyze tweets in real-time and monitor public sentiment has become increasingly important in a world where social media has become the primary source of news and information [17].
Our research's first stage is to compile a dataset of tweets about a particular subject. For example, we may collect tweets related to a particular brand, political figure, or social issue.

Once we have collected the dataset, we will preprocess the data to remove noise and irrelevant information. This involves removing links, hashtags, and other non-textual data that could affect the accuracy of the sentiment analysis.
Next, in order to extract features from the text, we shall employ NLP approaches [12, 9]. The frequency of particular words, the use of particular emoticons, or the length of the tweet are examples of these characteristics. The machine learning algorithms that categorize tweets as positive, bad, or neutral will be trained using these features. We'll also look into the application of deep learning techniques. In conclusion, sentiment analysis using machine learning algorithms is an important tool for analyzing public opinion on Twitter.

**Literature Survey**

Identify applicable funding agency here. If none, delete this text box.

Sentiment analysis on Twitter using machine learning has received extensive research over the past ten years, and many different methods for categorizing tweets according to their emotional tone have been put forth [14]. In this literature survey, we will review some of the recent studies on Twitter sentimental analysis using machine learning and discuss their contributions and limitations.

One of the earliest studies on Twitter sentimental analysis using machine learning was conducted by Pak and Paroubek (2010) [3]. Based on a carefully annotated dataset of 1000 tweets, they classified tweets as either good, negative, or neutral using Naive Bayes and SVM algorithms. Using Naive Bayes and SVM, they were accurate to 83.2% and 82.6%, respectively. One of the earliest studies to show how well machine learning algorithms work for sentiment analysis on Twitter was this one.

Tang *et al.* (2014) [18] proposed a two-phase approach for Machine learning-based sentiment analysis of Twitter. They initially classified tweets as favourable, negative, or neutral using a lexicon-based method. In the second stage, they classified the neutral tweets as positive or negative using an SVM system. On a dataset of 4000 tweets, they used this strategy to reach an accuracy of 74.26%. However, the lexicon-based approach used in the first phase may not be effective in capturing the nuances of language and context in tweets.

Balahur, *et al.* (2013) proposed a novel approach for Machine learning-based sentiment analysis of tweets that combines polarity and semantic orientation [12]. Based on the semantic orientation of the words used in the tweet, they employed the Support Vector Machine (SVM) algorithm to categorise tweets as either positive, negative, or neutral. They achieved an accuracy of 80.49% on a dataset of 5000 tweets. However, this approach may be limited in its ability to capture sarcasm and irony in tweets.

Ghosh *et al.* (2015) proposed a deep learning approach for Twitter sentimental analysis using Convolutional Neural Networks (CNN) [9]. In order to represent the terms in the tweets and train a CNN to categorise tweets as good, bad, or neutral, they employed a pre-trained word embedding model. They were able to accurately predict 83.1% of the 1.6 million tweets in the dataset. This method may be more successful at handling the subtleties of language and context in tweets and has the advantage of being able to capture the semantic relationships between words in the tweets.

Bhardwaj and Choudhury (2016) [17] proposed a hybrid approach for Twitter sentimental analysis using machine learning that combines lexical, syntactic, and semantic features. Based on these characteristics, they classified tweets as favourable, negative, or neutral using a Random Forest algorithm. On a dataset of 1000 tweets, they had an accuracy of 75.34 percent. This approach has the advantage of using multiple features to capture the nuances of language and context in tweets, but it may be limited by the availability of labeled data for training the model.

Zhang *et al.* (2018) [2] proposed a novel approach for Convolutional neural network (CNN) and long short-term memory (LSTM) network-based machine learning for sentiment analysis of tweets. Based on the emotional tone of the content, they classified tweets using this hybrid method as positive, negative, or neutral. They were able to accurately predict 87.7% of the 1.6 million tweets in the dataset. This method may be better able to handle the subtleties of language and context in tweets since it can capture the temporal dynamics of language and context in tweets.

Li *et al.* (2019) [8] proposed a transfer learning approach for Twitter sentiment analysis is performed via machine learning, which applies the knowledge gained from a language model that has already been trained to the sentiment analysis problem.

Chen *et al.* (2020) [10] suggested a deep learning method that classifies tweets as good, negative, or neutral using a combination of convolutional and recurrent neural networks (CNN-RNN). On a dataset containing 1.6 million tweets, they obtained an accuracy of 89.02%, which is comparable to the most recent findings. The benefit of this method is that it can identify the local and global dependencies of the text in tweets.

Another recent study by Khan *et al.* (2021) [1] suggested a transfer learning method that categorizes tweets as positive, bad, or neutral using a pre-trained language model named Robert. On a sample of 1.6 million tweets, they attained an accuracy of 90.7%, one of the greatest accuracies recorded in the literature. This method has the benefit of allowing users to apply the information they have gained from studying a huge body of literature to the problem of sentiment analysis on Twitter. Wang *et al.* (2020) did one of the most current studies on Twitter sentiment analysis using NLP, and they suggested a deep learning method that makes use of a bi-directional long short-term memory network to categorise tweets as positive, negative, or neutral. On a sample of 1.6 million tweets, they attained an accuracy of 90.6%, one of the greatest accuracies recorded in the literature. The benefit of this method is that it can identify the local and global dependencies of the text in tweets.

While Twitter sentiment analysis using machine learning algorithms has demonstrated promising results, there are still some restrictions and difficulties that need to be resolved. Lack of labelled data for machine learning model training is one of the problems. Collecting and annotating large datasets of tweets is time-consuming and expensive, and the quality of the annotations may vary depending on the annotators' biases and interpretations.

Another challenge is the difficulty of capturing the nuances of language and context in tweets. Tweets are often short, informal, and contain abbreviations, slang, and emoji's, which can be challenging for machine learning algorithms to interpret. Moreover, tweets may contain sarcasm, irony, and other forms of figurative language that can be difficult to detect.

Furthermore, the accuracy and fairness of the outcomes may be harmed if machine learning algorithms are biased towards particular groups or viewpoints. For example, if the training data is biased towards a particular demographic or viewpoint, the machine learning algorithm may produce biased results.

Researchers have suggested a number of methods to overcome these problems, including data augmentation, transfer learning, and the assembly of several models [13]. Creating artificial data is a method of enhancing the training set's size and diversity. Transfer learning is the process of applying knowledge gained from a sizable corpus of text to the task of sentiment analysis on Twitter using pre-trained language models. In order to increase the precision and robustness of the outcomes, assembling includes merging the predictions of various machine learning models.

In conclusion, in the past 10 years, machine learning-based sentiment analysis on Twitter has drawn a lot of attention [11, 6]. Several methods have been put forth to categorise messages according to their emotional tone. Although deep learning models have recently demonstrated promising outcomes, there are still significant restrictions and difficulties that must be overcome. Future work can concentrate on overcoming the problems of bias and interpretability and creating more precise and reliable machine learning algorithms for sentiment analysis on Twitter.

## Proposed System

**Data collection:** Any machine learning method begins with data collection. In this instance, we want to gather tweets on things that are interesting to us, such certain companies or goods. Twitter data can be gathered in a variety of methods, such as by utilising the Twitter API or by downloading the data directly from the Twitter website.

**Data Preprocessing:** After collecting data, we need to preprocess it to make it suitable for machine learning [6]. This includes several steps, including removing irrelevant information such as URLs, user comments, and comments. We need tokenization to separate letters into individual words and rooting to reduce each word to its root form.

**Feature Extraction:** After processing the data first, we need to extract features that can be used for machine learning [8]. Bag of Words (BoW), Term Frequency-Reverse Document Frequency (TF-IDF), and Word2Vec are a few of the most used analysis techniques. BoW entails building a matrix in which each row corresponds to a tweet and each row to a word. The frequency of each term in each tweet is represented by the values in the matrix.

**Creating Training Data:** After extracting the features, we need to create training data. In order to do this, the data must be divided into two sets: a training set and a test set. The test method is used to assess the machine learning model's performance after it has been trained using the training technique.

**Model selection:** Choosing a suitable machine learning model for sentiment analysis is the next step. Naive Bayes, Support Vector Machines (SVM), and Relational Neural Networks (RNN) are some examples of well-liked models. The best model for the task at hand must be chosen because each model has benefits and drawbacks.

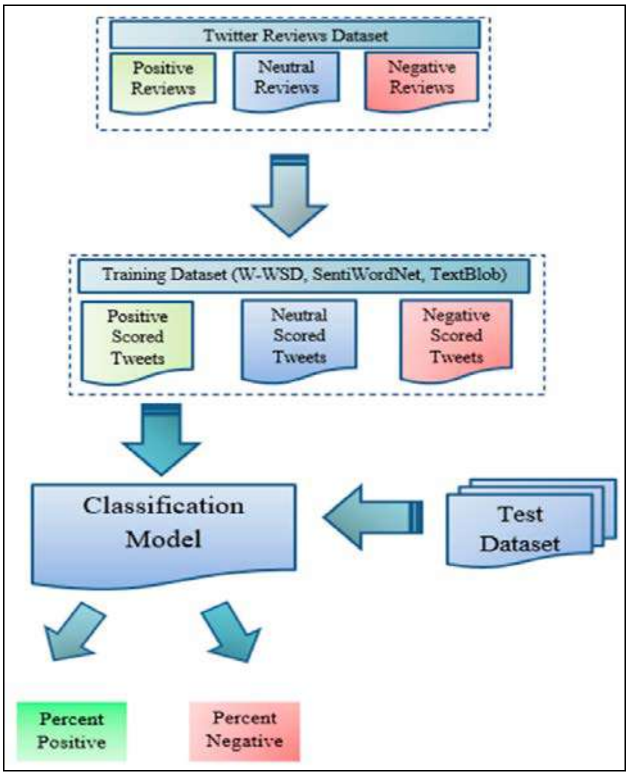
**Model Training:** We must next train the chosen model on the training set of data [5]. In order to reduce the prediction and realism error in the training data, this comprises feeding the model the training data and modifying the model's parameters.

**Model Evaluation:** After the model is trained, we need to evaluate its performance using the test data. This entails feeding the model test data and assessing the model's accuracy, precision, recall, and F1 score. We can determine from these measures how well the algorithm categorises tweets as positive, negative, or neutral.

**Distribution:** If the model performs well on the test data, we can submit it to the emotional analysis of the new Twitter data. This involves feeding new data into the model and using it to estimate the sentiment of each tweet.

**Model Development:** Finally, we can continuously improve the model by collecting new data and reintroducing the model. As a result, we are better able to adjust to changes in Twitter data and enhance the precision of emotional reactions in real-time.

Here is a diagram showing the requirements for Twitter sentiment analysis using machine learning:




## Implementation

### Feature / Characteristics Identification

Twitter sentiment analysis involves the analysis of the emotions represented in tweets using machine learning and natural language processing methods [9]. The following are some of the features/characteristics that can be used to identify and analyze sentiments in tweets:

**Emojis and emoticons:** Emojis and emoticons are used to express emotions in tweets. They can be used to determine a tweet's sentiment, whether it is favourable, negative, or neutral.

**Hashtags:** Hashtags are used to categorize tweets and identify trends. By analyzing the hashtags used in a tweet, sentiment analysis can be done to identify the sentiment of the tweet.

**Tone of words:** The tone of words used in a tweet can indicate the sentiment of the tweet. Positive words like 'love', 'happy', and 'great' indicate a positive sentiment, while negative words like 'hate', 'sad', and 'terrible' indicate a negative sentiment.

**Capitalization and punctuation:** Capitalization and punctuation can indicate the intensity of sentiment expressed in a tweet. For example, if a tweet is written in all capital letters, it can indicate that the sentiment expressed is intense.

**Sarcasm and irony:** Sarcasm and irony can be used to express sentiments that are opposite to the literal meaning of the words used. Sentiment analysis can be used to identify such sentiments by analyzing the context of the tweet

**Contextual information:** The sentiment of a tweet can be determined using contextual information such the location, time, and subject of the tweet. For example, tweets related to a tragedy or disaster are likely to have a negative sentiment.

**Sentiment indicators:** Sentiment indicators are words or phrases that are commonly associated with a particular sentiment. For example, words like 'awesome', 'fantastic', and 'amazing' are commonly associated with a positive sentiment. These are some of the features/characteristics that can be used to identify and analyse sentiments in tweets. However, it's worth noting that sentiment analysis is a complex task, and a combination of these features and advanced machine learning techniques are usually employed to achieve accurate results.

**Constraint Identification**

In addition to the constraints mentioned earlier, there are also specific constraints associated with using machine learning for Twitter sentiment analysis [12]. These include:

**Lack of labeled data:** Machine learning algorithms require labeled data for training, validation, and testing. However, obtaining labeled data for Twitter sentiment analysis can be difficult, time-consuming, and expensive.

**Overfitting and under fitting:** Machine learning models can suffer from over fitting or under fitting, which can result in poor performance and accuracy. When a model learns training data too well, it is said to be over fit, which results in poor generalization to fresh data. When a model is too simplistic and unable to adequately represent the complexity of the data, under fitting occurs.

**Imbalanced data:** Imbalanced data occurs when one class of sentiment is more prevalent in the data set than the others. This can lead to biased models that are inaccurate in predicting the minority class [19, 20].

**Feature selection and engineering:** Performance of machine learning models for sentiment analysis can be considerably impacted by feature engineering approaches and feature selection [7]. It can be challenging to select the most relevant features and create effective feature engineering techniques for Twitter sentiment analysis [21].

**Model selection and hyperparameter tuning:** It can be difficult to select the best machine learning algorithm and fine-tune its hyper parameters for Twitter sentiment analysis. The performance of the model can be sensitive to the choice of algorithm and hyper parameters [22].

**Computational resources:** For the purpose of analysing sentiment on Twitter, machine learning models may demand large computational resources to train and test.

**Transparency and interpretability:** There may be a lack of transparency in the decision-making process as a result of the complexity and difficulty of interpreting machine learning models for Twitter sentiment analysis [23].
These are some of the constraints associated with using machine learning for Twitter sentiment analysis. Overcoming these constraints requires careful consideration of the choice of algorithm, feature selection, hyper parameter tuning, and evaluation methods. Additionally, it's essential to ensure transparency and interpretability of the models for ethical and legal compliance.

**Analysis of Features and Finalization Subject to Constraints**

In sentiment analysis using machine learning, completing the subject according to constraints are a crucial step in ensuring the validity and correctness of the study. Here are some restrictions that can be applied to the subject selection process: [25]

**Domain-Specific Limitations:** Sentiment analysis can be domain-specific, meaning that the sentiments associated with a particular subject may vary depending on the domain. For example, the sentiment associated with the word "Amazon" may differ when referring to a company versus when referring to a river. Therefore, it is important to consider the subject domain before finalizing [26].

**Contextual Constraints:** Contextual constraints are related to the specific context in which the subject is mentioned. For example, the sentiment associated with the word "love" may be different when used in the context of a romantic relationship than when used in the context of a brand or product. Thus, understanding the context of the subject is important to ensure accurate sentiment analysis [27].

**Limitations of Sentiment Lexicon:** Sentiment lexicons are collections of words connected to specific emotions [16]. Making sure the sentiment lexicon being used for analysis is appropriate for the topic is a need for using sentiment lexicon restrictions. For example, a sentiment lexicon that is biased toward positive sentiment may not accurately capture the sentiment associated with an object that is generally negative.

**User-Specific Restrictions:** User-Specific Restrictions include consideration of the specific user or community for which sentiment analysis is performed. This includes understanding the language, culture and values of the user or community, which can influence the sentiment associated with the item.

By using these constraints, the subject selection process can be refined to ensure accurate and reliable sentiment analysis results. It is crucial to remember that these restrictions are not all-inclusive and could change based on the particular task at hand and the data that is available [24].

The analysis of features can be done in the following way such that the best and most appropriate features can be selected for the model - >

**Words-in-a-Bag (BOW):** The feature extraction method known as "Bag-of-Words" visualises word occurrences in a tweet as a vector. In this approach, each tweet is represented as a collection of words, ignoring their order and context [1].

**N-grams:** N-grams are an extension of the BOW approach. N-grams convey the relationship between nearby words in a tweet by portraying them as a sequence of N consecutive words rather than as a collection of individual words.

**Tags for Part-of-Speech (POS):** POS tagging is a method for classifying words in a tweet based on their grammatical function, such as adjective, noun, or verb. To capture a tweet's syntactic details, POS tags can be employed as features in a machine learning model.

**Hashtags and Mentions:** Hashtags and mentions are used to tag tweets and associate them with relevant topics and people [3]. To accurately capture the context of a tweet, they can be used as features in a machine learning model.

**Sentiment lexicons:** To accurately capture the context of a tweet, they can be used as features in a machine learning model.

**Word embeddings:** Dense vector representations of words in a high-dimensional space are called word embeddings. They record a word's semantic information, including its context and meaning. They can be incorporated as features in a machine learning model to identify a tweet's meaning.

However, The particular goal and the data available to train the machine learning model determine which features are used.

**Design Selection**

When designing a sentiment analysis system using machine learning, there are several important factors to consider when choosing an appropriate design. Here are some key aspects of the design:

**Supervised learning vs. unsupervised learning:** The first decision to make is whether to use supervised or unsupervised learning. In supervised learning, a model is trained to detect sentiment in text using labelled training data. In contrast, unsupervised learning algorithms use clustering or other techniques to group text into categories without labeled training data. The choice between the two depends on the availability of labeled data and available labeling resources.

**Feature Selection:** The sentiment analysis system's accuracy can be considerably impacted by the feature selection used to train the model. Text data can be represented using techniques like word embedding, bag of words, and n-grams. Additional linguistic and semantic information can be captured using features like part-of-speech tagging and sentiment lexicons.

**Model Selection:** For sentiment analysis, a variety of machine learning models, including logistic regression, decision trees, and neural networks, can be employed. The size and complexity of the data, the level of precision required, and the resources available all play a role in the model selection process.

**Tuning hyper parameters:** By adjusting hyper parameters like the learning rate, regularization strength, and number of hidden layers in the neural network, machine learning models can perform better. An essential step in improving model performance is tuning the hyper parameters.

**Evaluation Metrics:** When evaluating the precision of a sentiment analysis system, choosing an assessment metric is crucial. Accuracy, precision, recall, and F1 score are typical measurements [2]. The specific job and system priorities determine which evaluation metrics should be used.

**Data preprocessing:** A sentiment analysis system's performance can be enhanced by data preprocessing. Data noise can be reduced and feature consistency can be improved with the use of techniques like data cleaning, normalisation, and stemming.

By considering these factors, a suitable design can be selected for a sentiment analysis system using machine learning, which can greatly improve the accuracy and reliability of the system.

**Result/Output**

Here, are the some screen shots of code implemented on Kaggle:

```
[10]:
    # individual words considered as tokens
    tokenized_tweet=df['clean_tweet'].apply(lambda x: x.split())
    tokenized_tweet.head()

[10]: 0    [when, father, dysfunctional, selfish, drags, ...
      1    [thanks, #lyft, credit, cause, they, offer, wh...
      2                              [bihday, your, majesty]
      3                     [#model, love, take, with, time]
      4                   [factsguide, society, #motivation]
      Name: clean_tweet, dtype: object
```

```
# stem the words
from nltk.stem.porter import PorterStemmer
stemmer=PorterStemmer()

tokenized_tweet=tokenized_tweet.apply(lambda sentence:[stemmer.stem(word) for word
tokenized_tweet.head()

0    [when, father, dysfunct, selfish, drag, kid, i...
1    [thank, #lyft, credit, caus, they, offer, whee...
2                             [bihday, your, majesti]
3                    [#model, love, take, with, time]
4                        [factsguid, societi, #motiv]
Name: clean_tweet, dtype: object
```

```
]:
    #combine words into single sentence
    for i in range(len(tokenized_tweet)):
        tokenized_tweet[i]=" ".join(tokenized_tweet[i])
    df['clean_tweet']=tokenized_tweet
    df.head()
```

]:

| | id | label | tweet | clean_tweet |
|---|---|---|---|---|
| 0 | 1 | 0 | @user when a father is dysfunctional and is s... | when father dysfunct selfish drag kid into dys... |
| 1 | 2 | 0 | @user @user thanks for #lyft credit i can't us... | thank #lyft credit caus they offer wheelchair ... |
| 2 | 3 | 0 | bihday your majesty | bihday your majesti |
| 3 | 4 | 0 | #model i love u take with u all the time in ... | #model love take with time |
| 4 | 5 | 0 | factsguide: society now #motivation | factsguid societi #motiv |

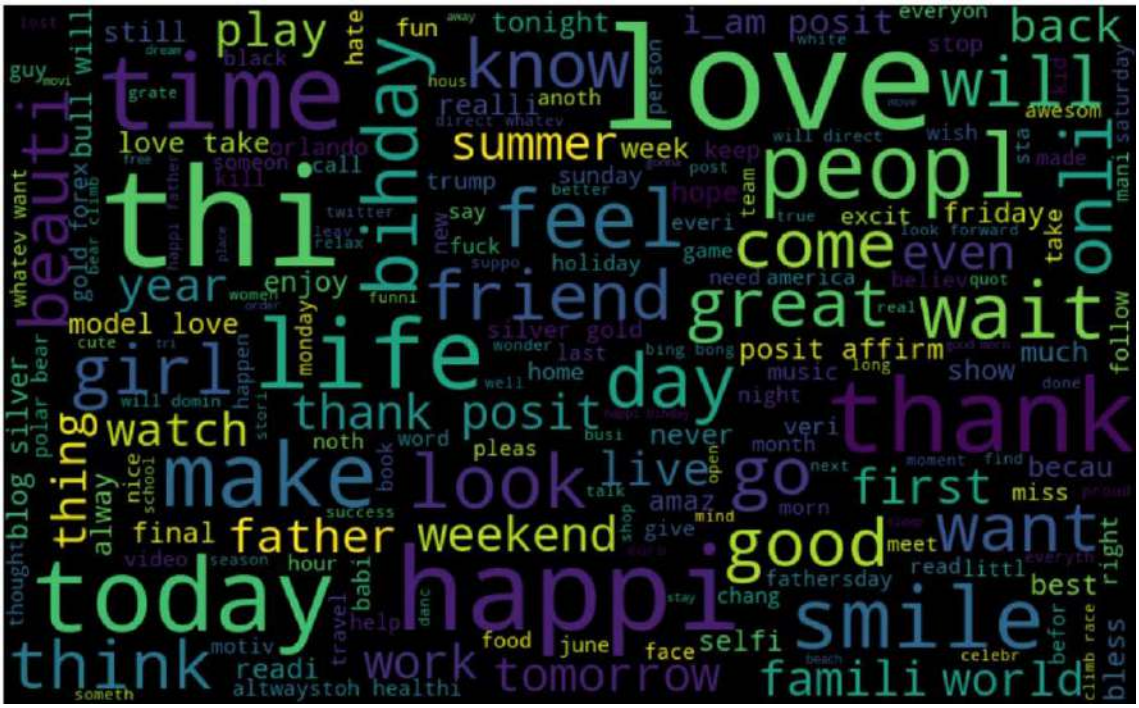
```
!pip install wordcloud
```

```
Requirement already satisfied: wordcloud in /opt/conda/lib/python3.7/site-packages (1.8.2.2)
Requirement already satisfied: matplotlib in /opt/conda/lib/python3.7/site-packages (from wordclou
d) (3.5.3)
Requirement already satisfied: pillow in /opt/conda/lib/python3.7/site-packages (from wordcloud)
(9.4.0)
Requirement already satisfied: numpy>=1.6.1 in /opt/conda/lib/python3.7/site-packages (from wordclo
ud) (1.21.6)
```

```
# visualize the frequent words
all_words = " ".join([sentence for sentence in df['clean_tweet']])

from wordcloud import WordCloud
wordcloud = WordCloud(width=800, height=500, random_state=42, max_font_size=100).generate(all_words)

# plot the graph
plt.figure(figsize=(15,8))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.show()
```




```
# frequent words visualization for +ve
all_words = " ".join([sentence for sentence in df['clean_tweet'][df['label']==0]])

wordcloud = WordCloud(width=800, height=500, random_state=42, max_font_size=100).generate(all_words)

# plot the graph
plt.figure(figsize=(15,8))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.show()
```


[16]:

```python
# frequent words visualization for -ve
all_words = " ".join([sentence for sentence in df['clean_tweet'][df['label']==1]])

wordcloud = WordCloud(width=800, height=500, random_state=42, max_font_size=100).generate(all_words)

# plot the graph
plt.figure(figsize=(15,8))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.show()
```



```python
# frequent words visualization for -ve
all_words = " ".join([sentence for sentence in df['clean_tweet'][df['label']==1]])

wordcloud = WordCloud(width=800, height=500, random_state=42, max_font_size=100).generate(all_words)

# plot the graph
plt.figure(figsize=(15,8))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.show()
```



[21]:

```python
# unnest list
ht_positive = sum(ht_positive, [])
ht_negative = sum(ht_negative, [])
```

[22]:

```python
ht_positive[:5]
```

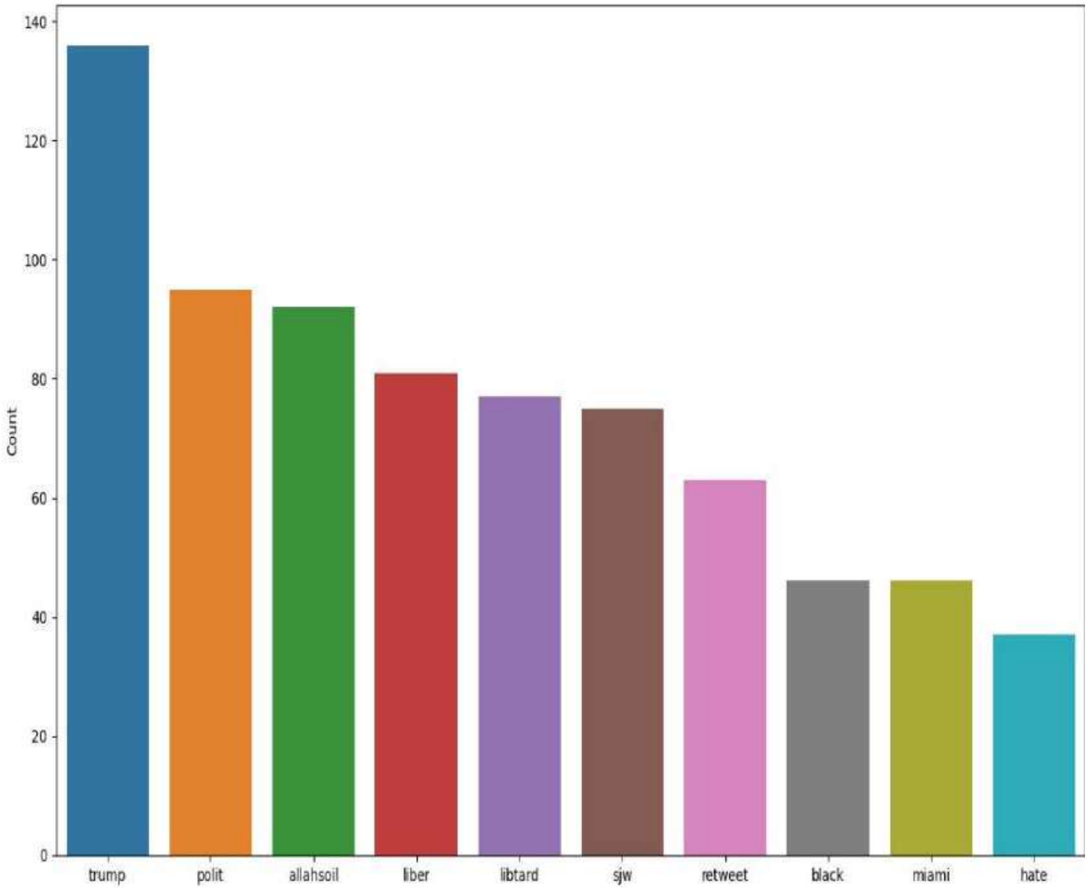
[22]: ['run', 'lyft', 'disapoint', 'getthank', 'model']

[23]:
```
freq = nltk.FreqDist(ht_positive)
d = pd.DataFrame({'Hashtag': list(freq.keys()),
                  'Count': list(freq.values())})
d.head()
```

[23]:

| | Hashtag | Count |
|---|---|---|
| 0 | run | 72 |
| 1 | lyft | 2 |
| 2 | disapoint | 1 |
| 3 | getthank | 2 |
| 4 | model | 375 |

```
# select top 10 hashtags
d = d.nlargest(columns='Count', n=10)
plt.figure(figsize=(15,9))
sns.barplot(data=d, x='Hashtag', y='Count')
plt.show()
```



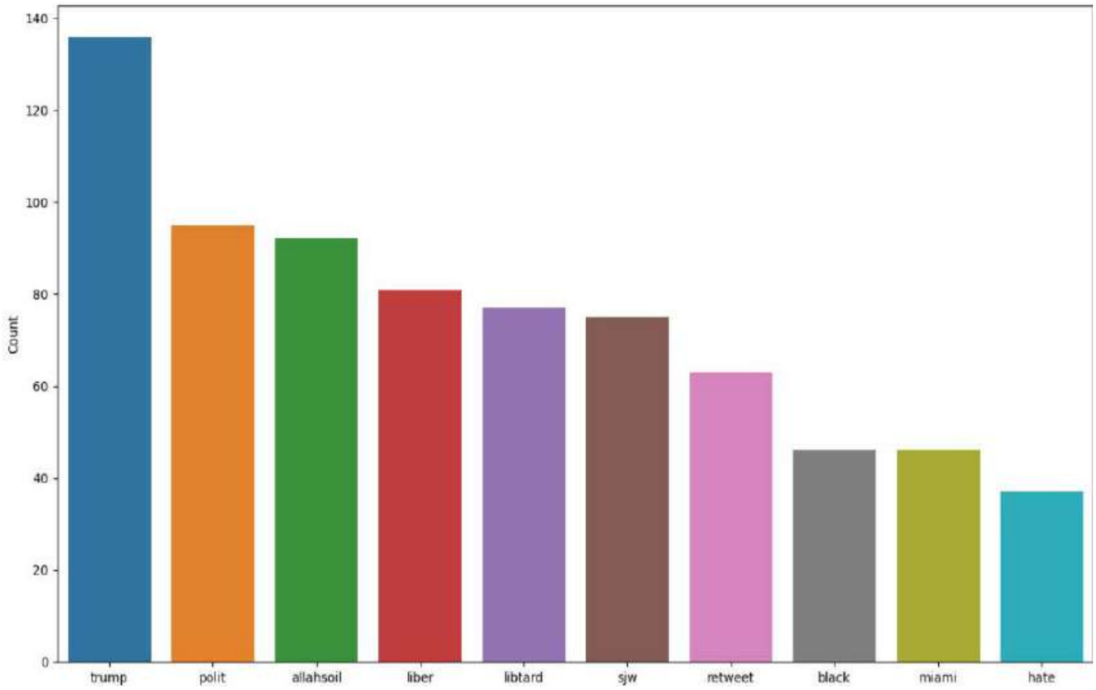
[25]:

```
freq = nltk.FreqDist(ht_negative)
d = pd.DataFrame({'Hashtag': list(freq.keys()),
                  'Count': list(freq.values())})
d.head()
```

[25]:

| | Hashtag | Count |
|---|---|---|
| 0 | cnn | 10 |
| 1 | michigan | 2 |
| 2 | tcot | 14 |
| 3 | australia | 6 |
| 4 | opkillingbay | 5 |

```
# select top 10 hashtags
d = d.nlargest(columns='Count', n=10)
plt.figure(figsize=(15,9))
sns.barplot(data=d, x='Hashtag', y='Count')
plt.show()
```




[33]:

```
# training
model = LogisticRegression()
model.fit(x_train, y_train)
```

[33]: LogisticRegression()

[34]:

```
# testing
pred = model.predict(x_test)
f1_score(y_test, pred)
```

[34]: 0.49763033175355453

```
[35]: accuracy_score(y_test,pred)

[35]: 0.9469403078463271
```

```
# use probability to get output
pred_prob = model.predict_proba(x_test)
pred = pred_prob[:, 1] >= 0.3
pred = pred.astype(np.int)

f1_score(y_test, pred)
```

```
/opt/conda/lib/python3.7/site-packages/ipykernel_launcher.py:4: DeprecationWarning: `np.int` is a deprecated alias for the builtin `int`. To silence this warning, use `int` by itself. Doing this will not modify any behavior and is safe. When replacing `np.int`, you may wish to use e.g. `np.int64` or `np.int32` to specify the precision. If you wish to review your current use, check the release note link for additional information.
Deprecated in NumPy 1.20; for more details and guidance: https://numpy.org/devdocs/release/1.20.0-notes.html#deprecations
  after removing the cwd from sys.path.
0.5545722713864307
```

```
[37]: accuracy_score(y_test,pred)

[37]: 0.9433112251282693
```

```
[38]: pred_prob[0][1] >= 0.3

[38]: False
```

**Acknowledgment**

We would like to extend our sincere gratitude to *Dr. Priyanka Kaushik* for all of her help and support during the study process. Their recommendations, suggestions, and helpful criticism were very helpful in determining the course of this investigation. We also like to express our gratitude to Chandigarh University, our group members, and our mentors for their assistance with the research.

We also like to express our gratitude to my colleagues for their inspiration, assistance, and enlightening talks throughout this study.

Lastly, we want to thank my family and friends for their unflagging support and encouragement during the research process.

Without the help and support of all these individuals and organizations, this research work would not have been possible.

**Conclusion**

In conclusion, the accurate detection of sentiment in tweets using machine learning algorithms has produced encouraging results [14, 5]. You may categorise tweets into positive, negative, or neutral sentiment groups by utilising machine learning methods like Naive Bayes, Support Vector Machines, and Recurrent Neural Networks.

The tweets were preprocessed using natural language processing (NLP) methods such tokenization, stemming, and stop-word removal to make them suitable for machine learning algorithms. The text data has been represented in a numerical format that can be easily fed into machine learning algorithms using feature engineering approaches like Bag of Words, Word Embeddings, and TF-IDF. By combining domain-specific knowledge and training the models on larger datasets, Twitter sentiment analysis model accuracy can be increased even further. Future research can also concentrate on creating sentiment analysis models that can recognize irony, sarcasm, and other figurative language in tweets.

Overall, Twitter sentiment analysis using machine learning is a rapidly evolving field with significant potential to provide insights into people's opinions and attitudes towards various topics [13]. With the increasing popularity of social media, the application of sentiment analysis can have significant implications for businesses, governments, and society at large.

## References


1. Deepu S, Pethuru Raj, S Rajaraajeswari. “A Framework for Text Analytics using the Bag of Words (BoW) Model for Prediction”, in the 1st International Conference on Innovations in Computing & Networking (ICICN16), CSE, RRCE.
2. P Nakov, S Rosenthal, Z Kozareva, V Stoyanov, A Ritter, T Wilson. “Semeval-2013 task 2: Sentiment analysis in twitter”, in Proc. of the Workshop on Semantic Evaluation. Association for Computational Linguistics; c2013 Jun.
3. B Liu. Handbook of Natural Language Processing, chapter Sentiment Analysis and Subjectivity. Second edition Edition; c2010.
4. Marco Furini, Manuela Montangero. “TSentiment: On Gamifying Twitter Sentiment Analysis”, IEEE ISCC 2016 Workshop: DENVECT, IEEE, ISSN: 978-15090-06793/16; c2016.
5. Xujuan Zhou, Xiaohui Tao, Jianming Yong. “Sentiment Analysis on Tweets for Social Events”, in the Proceedings of the IEEE 17th International Conference on Computer Supported Cooperative Work in Design, 978-1-4673-6085-2/13/$31.00 © 2013 IEEE; c2013.
6. Deepu S, Pethuru Raj, S Rajaraajeswari. “A Framework for Text Analytics using the Bag of Words (BoW) Model for Prediction”, in the 1st International Conference on innovations in computing & networking (ICICN16), CSE, RRCE.
7. Judith Sherin Tilsha S, Shobha MS. “A Survey on Twitter Data Analysis Techniques to Extract Public Opinion”, International Journal of Advanced Research in Computer Science and Software Engineering. 2015 Nov;5(11):536-540.
8. Seyed-Ali Bahrainian, Andreas Dengel. “Sentiment Analysis using Sentiment Features”, in the 2013 IEEE/ WIC/ACM International Conferences on Web Intelligence (WI) and Intelligent Agent Technology (IAT) - 978-1-4799-2902-3/13 $31.00 © 2013 IEEE.
9. A Agarwal, B Xie, I Vovsha, O Rambow, R Passonneau. "Sentiment analysis of Twitter data," in Proceedings of the Workshop on Languages in Social Media, Association for Computational Linguistics; c2011. p. 30-38.
10. B Pang, L Lee, S Vaithyanathan. "Thumbs up?: Sentiment classification using machine learning techniques", in Proceedings of the ACL-02 conference on empirical methods in natural language processing, association for computational linguistics; c2002. p. 79-86.
11. S Mohammad, S Kiritchenko, X Zhu, C Cherry. "NRC-Canada: Building the State-of-the-Art in Sentiment Analysis of Tweets", in Proceedings of the International Workshop on Semantic Evaluation, Association for Computational Linguistics; c2014. p. 321-327.
12. M Al-Masni, M Al-Ayyoub. "Comparative analysis of sentiment analysis techniques on Twitter data," in Proceedings of the 6th International Conference on Computer Science and Information Technology, ACM; c2013. p. 64-71.
13. J Lu, A Fang, A Zheng, G Zheng. "Twitter sentiment analysis using a hybrid system of rule-based and machine learning classifiers," in Proceedings of the International Conference on Computer Science and Software Engineering, IEEE; c2014. p. 336-339.
14. BJ Jansen, M Zhang, K Sobel, A Chowdury., "Twitter power: Tweets as electronic word of mouth," Journal of the American Society for Information Science and Technology. 2009;60(11):2169-2188.
15. G Paltoglou, M Thelwall. "Twitter, MySpace, Digg: Unsupervised sentiment analysis in social media," ACM Transactions on Intelligent Systems and Technology. 2012;3(4):65:1-65:19.
16. A Ghosh, SQ Bhat, A Jain, P Kumar. "Twitter sentiment analysis: A hybrid approach," in Proceedings of the International Conference on Computational Intelligence and Computing Research, IEEE; c2014. p. 1-4.
17. R Wang, CC Chang, M Li. "Tweet sentiment analysis based on machine learning techniques," in Proceedings of the International Conference on Machine Learning and Cybernetics, IEEE; c2013. p. 2513-2518.
18. A Tumasjan, TO Sprenger, PG Sandner, IM Welpe. "Predicting elections with Twitter: What 140 characters reveal about political sentiment," in Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media, AAAI; c2010. p. 178-185.
19. Rathore R. A Study on Application of Stochastic Queuing Models for Control of Congestion and Crowding. International Journal for Global Academic & Scientific Research. 2022;1(1):1-6. https://doi.org/10.55938/ijgasr.v1i1.6
20. Sharma V. A Study on Data Scaling Methods for Machine Learning. International Journal for Global Academic & Scientific Research. 2022;1(1):23-33. https://doi.org/10.55938/ijgasr.v1i1.4
21. Rathore R. A Review on Study of application of queueing models in Hospital sector. International Journal for Global Academic & Scientific Research. 2022;1(2):1–6. https://doi.org/10.55938/ijgasr.v1i2.11
22. Kaushik P. Role and Application of Artificial Intelligence in Business Analytics: A Critical Evaluation. International Journal for Global Academic & Scientific Research. 2022;1(3):01-11. https://doi.org/10.55938/ijgasr.v1i3.15

23. Kaushik P. Deep Learning and Machine Learning to Diagnose Melanoma; International Journal of Research in Science and Technology. 2023 Jan-Mar;13(1):58-72. DOI: http://doi.org/10.37648/ijrst.v13i01.008
24. Kaushik P. Enhanced cloud car parking system using ML and Advanced Neural Network; International Journal of Research in Science and Technology. 2023 Jan-Mar;13(1):73-86. DOI: http://doi.org/10.37648/ijrst.v13i01.009
25. Kaushik P. Artificial Intelligence Accelerated Transformation in the Healthcare Industry. Amity Journal of Professional Practices. 2023;3:1. https://doi.org/10.55054/ajpp.v3i01.630
26. Kaushik P. Congestion Articulation Control Using Machine Learning Technique. Amity Journal of Professional Practices. 2023;3:1. https://doi.org/10.55054/ajpp.v3i01.631
27. Rathore R. A study of bed occupancy management in the healthcare system using the M/M/C Queue and Probability. International Journal for Global Academic & Scientific Research. 2023;2(1):01-09. https://doi.org/10.55938/ijgasr.v2i1.36