Studying the Correlation between Tweet Sentiment and Stock Prices

CIS 600: Principals- Social Media and Data Mining
Term Project

Team Members :
Saurabh Kalelkar
Srivatsa Srinivas Rusum
Neeraj Patil
Rachana Fulsundar
Siddhita Nikam
Sai Teja Merla
Avnish Dubey
Brijesh Verma

Table of Contents

1. Abstract

Twitter is a goldmine for organizations and businesses looking to gain insights into the opinions and feedback of their customers. Twitter has grown in popularity as a microblogging medium, with millions of users sending out millions of tweets per day. As a result, it has become a significant source of information for organizations and businesses seeking to get insights into their customers' perceptions of their products or services.

Sentiment analysis is a vital technique for businesses and organizations looking to gain insights into their customers' opinions on social media platforms like Twitter. However, the unstructured nature of Twitter language makes it challenging to extract meaningful information from tweets. Traditional sentiment lexicon have been used to perform sentiment analysis, but their accuracy has been limited, especially in social media contexts. To address this issue, we have implemented a new evaluating tool called VADER, which combines traditional sentiment lexicon with improved ones that have been validated by humans, making it more reliable and of a higher standard. Another advantage of VADER is that it is more sensitive to sentiment expressions in social media contexts and generalizes more favorably to other domains. In addition to VADER, we have leveraged the sentiment analysis tool TextBlob, a Python library that is used for natural language processing tasks. Its sentiment analysis module provides a simple and intuitive interface for analyzing the sentiment of a text. Sentiment analysis tools like VADER and TextBlob are valuable resources for analyzing sentiment on social media platforms and can help organizations and businesses gain a deeper understanding of their customers' opinions. These insights can help them make data-driven decisions to improve their products or services. Researchers can use these tools to study public opinion on various issues and events, making them a valuable addition to the field of sentiment analysis. Additionally, what we aim to explore in this project is how public sentiment derived from tweets affects the stock prices of various companies. Thus, studying the correlation between tweet sentiment and stock prices can provide valuable insights for businesses, investors, and researchers alike, and sentiment analysis tools like VADER and TextBlob can be powerful tools in unlocking this relationship.

2. Introduction

The internet has drastically changed the way we connect and share information, particularly with the emergence of social networks like Twitter. Twitter has become a popular platform for sharing opinions and discussing various topics, with over 100 million users generating 500 million tweets daily. Compared to other social media platforms, Twitter's small message size and instant updates make it ideal for disseminating information about news, opinions, and announcements.

In the 2016 US elections, researchers used Twitter as a source of data to predict the winner. They collected tweets related to both Donald Trump and Hillary Clinton, as both candidates used Twitter to disseminate information about their policies and to attack their opponents. The researchers noted that many people used Twitter to express their views on the presidential candidates and to defend their choice or criticize their opponent. The primary research goal was to determine whether Twitter could be an effective polling method compared to traditional methods such as the IBD/TIPP tracking poll.

More recently, the authors found that Twitter was indeed the most effective polling method compared to the results of three different poll sources. This has led researchers to focus their attention on Twitter as a potential basis for a predictive tool in various areas, through the analysis of people's opinions. The ability to gather large amounts of data in real-time and the ease of access to this data has made Twitter a valuable resource for researchers in fields such as politics, healthcare, and finance. By analyzing people's opinions on Twitter, researchers can gain insights into public sentiment and use this information to make predictions about future events. Moreover, this has made it a valuable source of information for organizations and companies looking to gain insights into their customers' opinions about their products or services.

The stock market is an important part of a country's economy, providing a way for investors to make investments and gain high capital. It involves buying and selling shares, which represent ownership claims on businesses. Trading in the stock market involves transferring money from small individual investors to large trader investors, such as banks and companies. However, investing in the stock market is highly risky due to its unpredictable behavior. Successful prediction of the stock market can be crucial for investors, as it can guide them in making appropriate decisions about buying or selling shares.

Also, some researchers suggest that by analyzing users' opinions on Twitter, it is possible to gain valuable information about the stock market and predict price movements. This is not a new concept, as investors have always sought ways to gain an advantage through scarce and valuable information. For example, in the past, people were employed specifically to read newspapers and extract sentiment and relevant information. The financier Rothschild also used his network of carrier pigeons to gain an advantage in the market. As investing in the financial market is complex and risky, techniques are being developed to minimize losses and maximize profits.

However, due to the unstructured nature of Twitter language, sentiment analysis is necessary to extract meaningful information from tweets. Sentiment analysis involves analyzing the sentiment reflected in tweets and presenting the results as a percentage on a particular scale. This information can be used to monitor changes in sentiment during events, analyze public

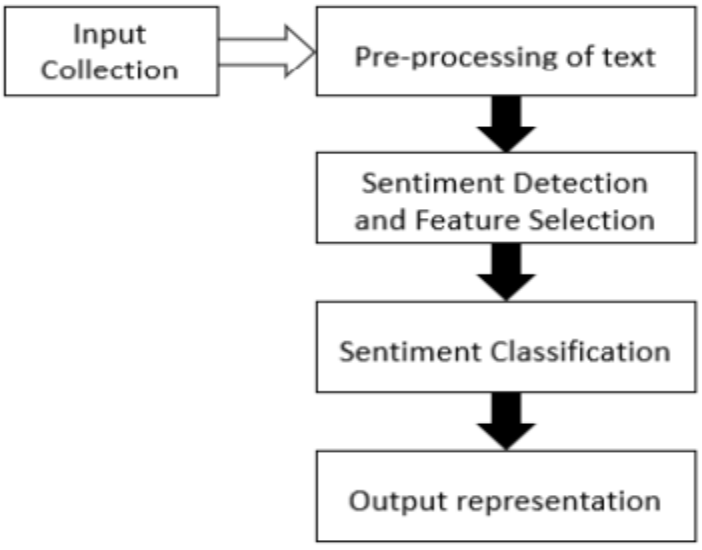
views on government policies, and evaluate the sentiment towards a particular brand or product release.

To improve the accuracy of sentiment analysis, a new evaluating tool called VADER (Valence Aware Dictionary and sEntiment Reasoner) has been developed. VADER combines traditional sentiment lexicon with improved ones that have been validated by humans. This makes the VADER sentiment lexicon more reliable and of a higher standard. Additionally, VADER is more sensitive to sentiment expressions in social media contexts and generalizes more favorably to other domains. Overall, VADER is a valuable tool for sentiment analysis on Twitter and other social media platforms.

Furthermore, with VADER, organizations and businesses can gain a deeper understanding of their customers' opinions, which can help them make data-driven decisions to improve their products or services. This tool can also be used by researchers to study public opinion on various issues and events, making it a valuable addition to the field of sentiment analysis.

Apart from VADER, TextBlob is a Python library used for natural language processing (NLP) tasks, such as sentiment analysis, part-of-speech tagging, and noun phrase extraction. It is built on top of the Natural Language Toolkit (NLTK) library and uses the Pattern library for its text processing tasks.TextBlob's sentiment analysis module provides a simple and intuitive interface for analyzing the sentiment of a text. The library analyzes the polarity (positive or negative) and subjectivity (objective or subjective) of a piece of text and assigns a sentiment score based on these values.

The workflow for Sentiment Analysis using both these libraries is as follows:



The goal of this analysis is to study the state of the art concerning stock market prediction using sentiment analysis.The impact of Twitter on financial markets is the focus of this dissertation. The research will particularly examine how tweets about a company may influence its share price trends.

3. Literature Review

Previous studies have contributed to the field of sentiment analysis by developing general sentiment classification systems for use in target domains where no labeled data is available. These systems use labeled data from different domains to predict the polarity of sentiments expressed in people's opinions. However, traditional classification algorithms require manually labeled text data, which can be expensive and time-consuming.

Recently, scholars have shown a growing interest in sentiment analysis, particularly with reference to Twitter data. Previous works that have contributed to the field of sentiment analysis in recent years are included below. Wagh et al. [7] created a broad sentiment categorization method for usage when there is no label data in the target domain. Labeled data from a separate domain are used in this system. This technique was also used to compute the frequency of each term in a tweet. A dataset including four million tweets made public by Stanford University was evaluated in this study. The polarity of feelings expressed in people's opinions was predicted using this dataset.Traditional classification techniques can be used to train sentiment classifiers using manually labeled text data, but this is an expensive and time-consuming process. The study discovered that when a classifier learned in one domain is applied immediately to another, its performance is exceedingly low. The paper demonstrated the accuracy of various algorithms for varying quantities of tweets, including Naive Bayes, Multi-nominal NB, Linear SVC, Bernoulli NB classifier, Logistic Regression, and the SGD classifier. The results demonstrated that the suggested system outperformed the existing systems in terms of efficiency.

Gilbert created VADER, a simple rule-based model for general sentiment analysis, and compared its performance to 11 industry benchmarks, including Affective Norms for English Words (ANEW), Linguistic Inquiry and Word Count (LIWC), the General Inquirer, Senti WordNet, and machine learning-oriented techniques based on the Naive Bayes, Maximum Entropy, and Support Vector Machine (SVM) algorithms. The research described the creation, validation, and testing of VADER. The researcher employed a mix of qualitative and quantitative methodologies to create and validate a sentiment lexicon for usage in social media.To assess the sentiment of tweets, VADER employs a simple rule-based algorithm. The study found that VADER enhanced the benefits of classic sentiment lexicons like LIWC.

Mane et al. described a sentiment analysis that made use of Hadoop, which processes massive volumes of data in real time on a Hadoop cluster. The researchers' goal was to determine whether the users' opinions were good or negative. This approach emphasized the speed with which sentiment analysis of real-time Twitter data could be performed using Hadoop. The Hadoop platform was created to handle problems involving massive amounts of unstructured and complicated data. It processed such data using the divide and rule method.The time taken to access various modules determined the overall correctness of the project. The code performed admirably in the analysis. The study rated the statements in multiclasses using a numbering approach, which assigned an acceptable range of distinct moods. Furthermore, the method might be used to other social media platforms, such as movie reviews (for example, IMDB reviews) and personal blogs. Along the same lines, Bouazizi and Ohtsuki [9] established SENTA, which assists users in selecting the attributes that are best suited for the application used to execute the categorization. The researchers employed SENTA to perform multi-class sentiment analysis on Twitter texts. The investigation was restricted to seven distinct sentiment classifications.

4. Fundamental idea

The proposed project aims to investigate the impact of social media sentiment on stock prices using Twitter data and sentiment analysis. Twitter is a popular social media platform that provides real-time information about user reactions and opinions towards various topics, including stocks and companies. By analyzing these tweets using sentiment analysis techniques, we can classify them as positive, negative, or neutral and determine the overall sentiment towards a particular stock or company.

To conduct sentiment analysis on Twitter data, we will be using two popular Python libraries, VADER and TextBlob. VADER (Valence Aware Dictionary and sEntiment Reasoner) is a rule-based sentiment analysis tool that is specifically designed to analyze sentiment in social media texts. It uses a lexicon-based approach to analyze the sentiment of each tweet and provides a score for the positivity, negativity, and neutrality of each text.

On the other hand, TextBlob is a Python library that uses machine learning techniques to conduct sentiment analysis. It is based on the Natural Language Toolkit (NLTK) and provides various tools for text processing and sentiment analysis. It uses a trained Naive Bayes classifier to classify text into positive, negative, or neutral.

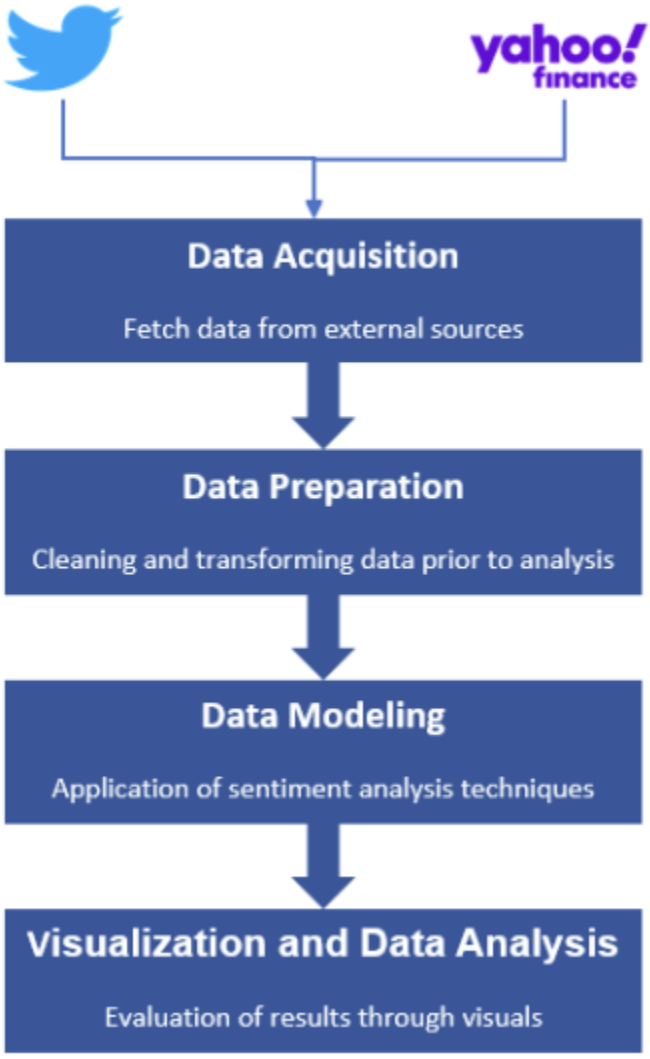
By using these two libraries, we can compare and contrast the sentiment analysis results from both approaches. This will provide us with a more comprehensive understanding of the sentiment towards a particular stock or company on social media.

Furthermore, we can also conduct a comparative analysis of the accuracy of these two libraries in predicting sentiment accurately. This can be done by manually reviewing a subset of the tweets and comparing the human-generated sentiment with the results of both VADER and TextBlob.

Overall, by using VADER and TextBlob, we can ensure that our sentiment analysis results are accurate and reliable. This will provide us with valuable insights into the sentiment of social media users towards a particular stock or company, which can assist investors and traders in making informed decisions.

Overall, this project has the potential to provide valuable insights into the impact of social media sentiment on stock prices. By leveraging sentiment analysis techniques, investors and traders can gain a better understanding of the sentiment of social media users towards a particular stock or company. This information can help them make informed decisions and potentially gain a competitive advantage in the market.

5. Data and Methodology



The above modules are necessary for performing Sentiment Analysis on data acquired from Twitter and Yahoo Finance.

1. Data Acquisition: This module is responsible for retrieving raw data from Twitter and Yahoo Finance and validating it. It is crucial in acquiring relevant data from Twitter and Yahoo Finance. A web scraper can be used to extract tweets from Twitter, and Yahoo Finance can be used to gather information on stock prices listed in the S&P 500. This module also checks whether the data received is what is required for the study.

2. Data Preparation: This module is responsible for cleaning and structuring the data so that it can be used to create an SA model. It prepares the data for the SA model. It involves cleaning and structuring the data to ensure it is ready to be used in the SA model. This

can include removing irrelevant information, formatting the data, and dealing with missing values.

3. Data Modelling: This module sets up an SA model using machine learning (ML) algorithms. The model will describe the data we want to analyse using SA techniques. The ML algorithms can be trained to recognize and classify sentiment in the data, using techniques such as natural language processing (NLP).
4. Visualization and Data Analysis: This module provides visual representations of the SA results to help interpret and assess them. It is responsible for presenting the results of the SA in a way that can be easily interpreted. It provides visual representations of the data, such as charts or graphs, to help understand the sentiment analysis results. The analysis can also involve identifying trends and patterns in the data that may be useful for future decision-making.

1. Data Acquisition

Data Collection sources : Yahoo finance, Kaggle, twitter

The "Tweets Extraction" is a process of extracting tweets related to a particular topic or keyword from the social media platform Twitter. In this context, the tweets related to the stock prices of Tesla and Apple are being extracted for sentiment analysis.

The extraction of the stock price information from Yahoo Finance is the first step in this process. For the sentiment analysis, the data is filtered for the most recent year. Then, using tweepy, a Python package for accessing the Twitter API, the tweets from Twitter linked to the keywords and stock symbols are extracted.

Following the extraction of the tweets, sentiment analysis is carried out to determine how people feel generally about the stock prices of Apple and Tesla. Natural Language Processing (NLP) techniques are used in the sentiment analysis to recognize and extract the sentiment from the text data. Based on the analysis, the sentiment might be categorized as positive, negative, or neutral.

Overall, the Tweets Extraction process is a powerful tool for analyzing the sentiments of people towards a particular topic or issue. It can be used in various industries such as marketing, politics, and finance, to gain insights into the public opinion on a particular matter.

2. Data Preparation

2.1 Merging the DataFrames to form one single DataFrame

```
[ ]  # Part1: Merge raw_company_tweet_db, raw_company_db
     raw_company_tweet_db = pd.merge(raw_company_tweet_db, raw_company_db,on='ticker_symbol')

     # Part2: Merge raw_company_tweet_db, raw_company_db
     raw_tweet_db = pd.merge(raw_tweet_db, raw_company_tweet_db, on="tweet_id")
```

Initially, multiple DataFrames were merged together into a single DataFrame based on shared columns.

In Part 1, two DataFrames, raw_company_tweet_db and raw_company_db, were merged together using the pd.merge function. This was done by combining the DataFrames using a common column called ticker_symbol. The resulting DataFrame contained all the columns from both DataFrames, with rows that had matching ticker_symbol values merged together.

In Part 2, the resulting merged DataFrame from Part 1, raw_company_tweet_db, was merged with another DataFrame called raw_tweet_db. This was also done using the pd.merge function, but this time based on a different common column called tweet_id. The resulting DataFrame contained all the columns from both DataFrames, with rows that had matching tweet_id values merged together.

By merging the DataFrames in this way, the resulting DataFrame contained all the information from the original three DataFrames merged together into a single DataFrame. This made it easier to analyze the data and draw conclusions from it.

```
# Let us check the number of rows and columns in the dataframes
[print(f"For Dataframe {df.name}\nNumber of Rows are {df.shape[0]}\nNumber of Columns are {df.shape[1]}\n\n") for df in list_df]
```

```
For Dataframe raw_tweet_db
Number of Rows are 4336445
Number of Columns are 9


For Dataframe raw_companyvalue_db
Number of Rows are 3522
Number of Columns are 7
```

A list comprehension was used to print out information about each DataFrame's shape, i.e., the number of rows and columns it contains.
The output shows that the raw_tweet_db DataFrame has 4,336,445 rows and 9 columns, and the raw_companyvalue_db DataFrame has 3,522 rows and 7 columns.

2.2 Missing Values Check

List comprehension was used to loop over two DataFrames, raw_tweet_db and raw_companyvalue_db, stored in a list called list_df. The code then printed information about the missing values in each DataFrame as below.

```
[print(f"For DataFrame {df.name}, we have missing values check as\n{df.isna().sum()}\n\n") for df in list_df]
```

```
For DataFrame raw_tweet_db, we have missing values check as
tweet_id            0
writer          55919
post_date           0
body                0
comment_num         0
retweet_num         0
like_num            0
ticker_symbol       0
company_name        0
dtype: int64


For DataFrame raw_companyvalue_db, we have missing values check as
day_date        0
open_value      0
high_value      0
low_value       0
close_value     0
volume          0
ticker_symbol   0
dtype: int64


[None, None]
```

In the print statement inside the list comprehension, an f-string was used to display the name of the DataFrame (df.name) and the number of missing values in each column of the DataFrame (df.isna().sum()).

To calculate the number of missing values in each column of the DataFrame, the df.isna().sum() method was used. This method returned a DataFrame with the same shape as the original, but with True values in places where there were missing values, and False elsewhere. The sum() method was then used to add up the number of True values in each column, which gave the number of missing values in each column.

The output displayed the number of missing values in each column of each DataFrame, which could help identify columns with a lot of missing values and provide insight into the overall data quality.

```
raw_tweet_db.writer = raw_tweet_db.writer.fillna('anonymous')
```

fillna() method filled any missing values in the writer column of the raw_tweet_db DataFrame with the string 'anonymous'.

By filling in missing values with a default string value, this code helped to handle missing data in the writer column of the raw_tweet_db DataFrame. This is important to ensure that subsequent analysis or modeling tasks on the data are not affected by missing values.

2.3 Converting data type of columns

```
[print(f"for DataFrame {df.name}\n{df.info()}\n") for df in list_df]
```

info() method provided summary of the DataFrame, including the total number of rows and columns, data type and the amount of memory used by the DataFrame.

By examining the data type of the columns in the dataframes, we were able to determine that certain columns needed to be converted to different data types in order to properly conduct sentiment analysis. Therefore, checking the data type of the columns was an important step in preparing the data for accurate and effective sentiment analysis.

```
[ ] raw_tweet_db.post_date = pd.to_datetime(raw_tweet_db.post_date, unit="s")
    raw_companyvalue_db.day_date = pd.to_datetime(raw_companyvalue_db.day_date)
```

The columns 'post_date' in the raw_tweet_db dataframe and 'day_date' in the raw_companyvalue_db dataframe were converted to datetime format using the pd.to_datetime() function. This was done in order to make it easier to work with dates and times in the dataframes.

2.4 Finalize dataset date range

```
# for checking the chronologically first tweet in the dataframe, we sort the dataframe by date column
raw_tweet_db.sort_values(by="post_date", inplace=True)
raw_companyvalue_db.sort_values(by="day_date", inplace=True)
```

Checking the chronologically first and last tweet in the dataframe was helpful in finalizing the dataset date range. By identifying the earliest and latest dates, we were able to ensure that our dataset only included tweets within the desired time frame and avoid any potential data leakage or bias.

To determine the date range for our analysis, we looked at the chronologically first and last tweet in the **raw_tweet_db** and **raw_companyvalue_db** dataframe. By looking at the head and tail end of the dataframe, we identified the time period during which we should consider the tweets for our sentiment analysis.

```
[ ] raw_companyvalue_db = raw_companyvalue_db[raw_companyvalue_db.day_date < "2020-01-01"]
```

```
[ ] raw_companyvalue_db = raw_companyvalue_db[raw_companyvalue_db.day_date > "2019-01-01"]
```

```
raw_companyvalue_db.tail()
```

	day_date	open_value	high_value	low_value	close_value	volume	ticker_symbol
3521	2019-12-30	158.990005	159.020004	156.729996	157.589996	16348400	MSFT
1256	2019-12-30	72.364998	73.172501	71.305000	72.879997	144114400	AAPL
2766	2019-12-30	85.758003	85.800003	81.851997	82.940002	62932000	TSLA
1257	2019-12-31	72.482498	73.419998	72.379997	73.412498	100805600	AAPL
2767	2019-12-31	81.000000	84.258003	80.416000	83.666000	51428500	TSLA

We also made sure that this time period aligns with the available stock values, so that our analysis can provide meaningful insights. Since we wanted to align the tweets with the stock values, we decided to drop all the tweets posted in the year 2020 from our analysis. Based on this, we decided to consider all tweets posted from 1st January 2019 to 31st December 2019.

2.5 Adding Columns

```
# Let us add column to the stock price dataframe which shows the max stock price fluctuation
raw_companyvalue_db['fluctuation'] = raw_companyvalue_db.high_value - raw_companyvalue_db.low_value

# Let us add column to the stock price dataframe which shows the net rise in stock price
raw_companyvalue_db['price_gain'] = raw_companyvalue_db.close_value - raw_companyvalue_db.open_value

# Let us add column to the stock price dataframe which shows the total valuation at the end of the day
raw_companyvalue_db['total_valuation_EOD'] = raw_companyvalue_db.volume * raw_companyvalue_db.close_value
```

The three columns added to the stock price dataframe (raw_companyvalue_db) - 'fluctuation', 'price_gain', and 'total_valuation_EOD' - to calculate and analyze the stock value performance.

Fluctuation: This column shows the maximum fluctuation in stock prices during the day. High fluctuations could indicate a volatile market, which could affect the sentiment of the investors. For example, if the fluctuation is high, investors may feel uncertain about the future of the company, leading to a negative sentiment.

Price_gain: This column shows the net rise or fall in stock prices during the day. The sentiment of investors could be affected by the trend of the stock price. For example, if the stock price is rising, investors may feel positive about the future prospects of the company, leading to a positive sentiment.

Total_valuation_EOD: This column shows the total valuation of the company at the end of the day. The sentiment of investors could be influenced by the overall performance of the company. For example, if the company's valuation is high, investors may feel positive about the future prospects of the company, leading to a positive sentiment.

3. Data Modeling

3.1 Why Vader?

NLTK's Sentiment Intensity Analyzer utilizes a sentiment lexicon known as VADER, which is essentially a list of words with associated sentiment scores. Each word in the lexicon is assigned a polarity (positive or negative) and a magnitude (intensity). By analyzing the presence and frequency of positive and negative words in a sentence, the Sentiment Intensity Analyzer determines the overall sentiment of the sentence.

The approach is based on the idea that if a sentence contains more positive words, it is likely to be perceived as more positive, while a higher occurrence of negative words suggests a more negative sentiment. Additionally, VADER takes into account other

linguistic features such as capitalization, which can influence the sentiment analysis outcome.
For more detailed information on the Sentiment Intensity Analyzer, you can refer to the classifier's documentation or the original research paper it is based on.

VADER is specifically built to handle social media text, which includes slang, emojis, abbreviations, and other language variations commonly found in tweets.

VADER's lexicon and rule-based approach are fine-tuned for social media text, including Twitter. It takes into account domain-specific features like capitalization, exclamation marks, and repeated letters, which are common in tweets.

VADER is a pre-trained model and doesn't require additional training on a specific dataset. This makes it convenient to use out-of-the-box for sentiment analysis tasks, especially in scenarios where annotated training data might be limited or not readily available.

3.2 Testing sentiment analysis using vader vs textblob

We were trying to decide whether to use vader or textblob. So we have taken a sample data set and have done sentiment analysis manually and used that data as a sample set to test the accuracy given by vader and textblob.

```
!pip install twython
import nltk
nltk.download('vader_lexicon')
from nltk.sentiment.vader import SentimentIntensityAnalyzer

sid = SentimentIntensityAnalyzer()


def sentiment_analysis_vader(tweet):
    return sid.polarity_scores(tweet)["compound"]
```

To perform sentiment analysis on a tweet using VADER, we can call the sentiment_analysis_vader() function and pass the tweet text as an argument. The function will return a sentiment score between -1 and 1, where negative values indicate negative sentiment, positive values indicate positive sentiment, and values close to zero indicate neutral sentiment.

```
!pip install textblob

from textblob import TextBlob
import tweepy


def sentiment_analysis_textblob(tweet):
    analysis = TextBlob(tweet)
    return analysis.sentiment.polarity
```

To perform sentiment analysis on a tweet using TextBlob, we can call the sentiment_analysis_textblob() function and pass the tweet text as an argument. The function will return a polarity score between -1 and 1, where negative values indicate negative sentiment, positive values indicate positive sentiment, and values close to zero indicate neutral sentiment.

▸ Applying sentiment analysis to the tweets using vadern and textblob to check which would be a better fit:

```
[101] sample_raw_tweet_db = raw_tweet_db.sample(n=100, random_state=42)

[102] sample_raw_tweet_db['sentiment_vader'] = sample_raw_tweet_db['body'].apply(lambda x : sentiment_analysis_vader(x))

[103] sample_raw_tweet_db['sentiment_textblob'] = sample_raw_tweet_db['body'].apply(lambda x : sentiment_analysis_textblob(x))

[104] print(sample_raw_tweet_db[['body','sentiment_vader','sentiment_textblob']].head())
```

In this piece of code, we have taken a sample data frame named sample_raw_tweet_db on which we have applied the functions written above to give a polarity score between -1 and 1.

```
sample_raw_tweet_db.to_csv('sample_raw_tweet_db.csv', index=False)
```

```
sample_train_raw_tweet_db = pd.read_csv('sample_train_raw_tweet_db.csv')
```

```
def categorize_sentiment(score):
    if score > 0:
        return 'Positive'
    elif score < 0:
        return 'Negative'
    else:
        return 'Neutral'
```

```
sample_train_raw_tweet_db['sentiment_vader'] = sample_train_raw_tweet_db['sentiment_vader'].apply(categorize_sentiment)
sample_train_raw_tweet_db['sentiment_textblob'] = sample_train_raw_tweet_db['sentiment_textblob'].apply(categorize_sentiment)
```

```
sample_train_raw_tweet_db.head()
```

We have taken the sample data into a csv and have used that data to classify our analysis based on the tweets. We have added it as a new column Training_sentiment, which we will be using to analyze how accurate both the sentiment analysis are.

Once this is done, we have manually placed the file at the location of the directory where the code is being run.

A function has been written to classify the sentiment from vader and textblob to Positive, Negative or Neutral.

The columns have been replaced to tell the sentiment.

```
# Define a function to calculate accuracy for vader
def calculate_accuracy_vader(row):
    if row['sentiment_vader'] == row['Training_sentiment']:
        return 1
    else:
        return 0

# Define a function to calculate accuracy for textblob
def calculate_accuracy_textblob(row):
    if row['sentiment_textblob'] == row['Training_sentiment']:
        return 1
    else:
        return 0
```

This function is written to calculate the accuracy of both the sentiment analysis from the sample data.

```
# Apply the accuracy function to sentiment_vader column
sample_train_raw_tweet_db['accuracy_vader'] = sample_train_raw_tweet_db.apply(lambda row: calculate_accuracy_vader(row), axis=1)

# Apply the accuracy function to sentiment_textblob column
sample_train_raw_tweet_db['accuracy_textblob'] = sample_train_raw_tweet_db.apply(lambda row: calculate_accuracy_textblob(row), axis=1)

# Calculate average accuracy for sentiment_vader and sentiment_textblob
accuracy_vader = sample_train_raw_tweet_db['accuracy_vader'].mean() * 100
accuracy_textblob = sample_train_raw_tweet_db['accuracy_textblob'].mean() * 100

# Print the accuracy results
print("Accuracy for sentiment_vader: {:.2f}%".format(accuracy_vader))
print("Accuracy for sentiment_textblob: {:.2f}%".format(accuracy_textblob))
```

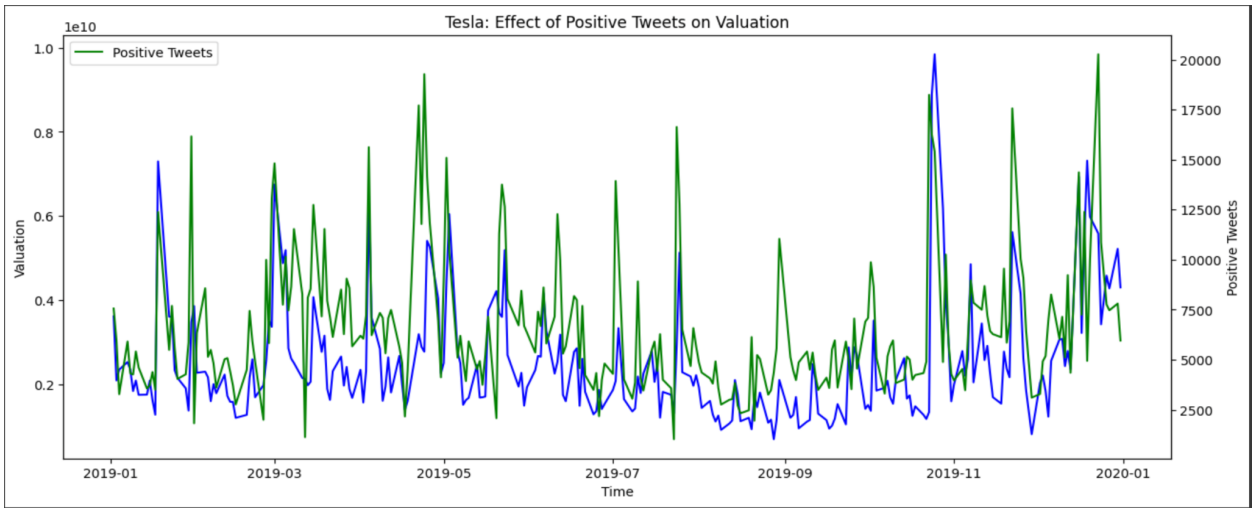
```
Accuracy for sentiment_vader: 81.00%
Accuracy for sentiment_textblob: 69.00%
```

In this piece of code, we have added 2 columns to check the accuracy of both the analysis for each tweet in the sample data for which we have classified the sentiment.

We then check how accurate they are.

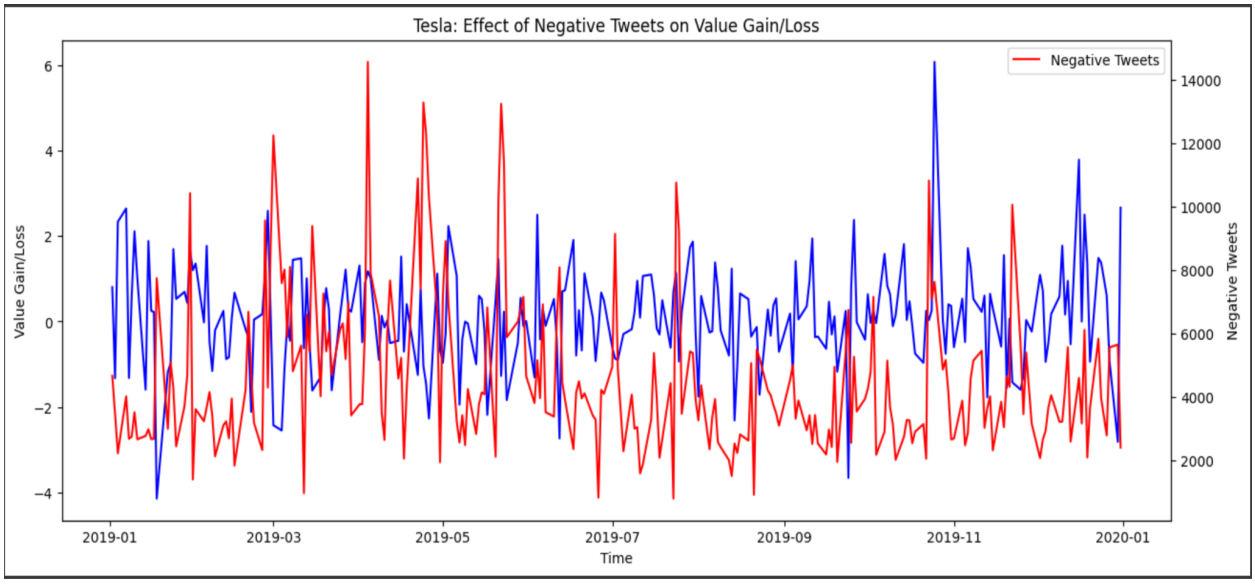
Here we found that vader was more accurate to the analysis which we have manually done for the sample data.

4. Visualization and Data Analysis



The observation made is that when there is a spike in the number of positive tweets, there is also a spike in the valuation of the stocks. In other words, an increase in positive sentiment expressed in tweets is accompanied by an increase in the overall value of the stocks.

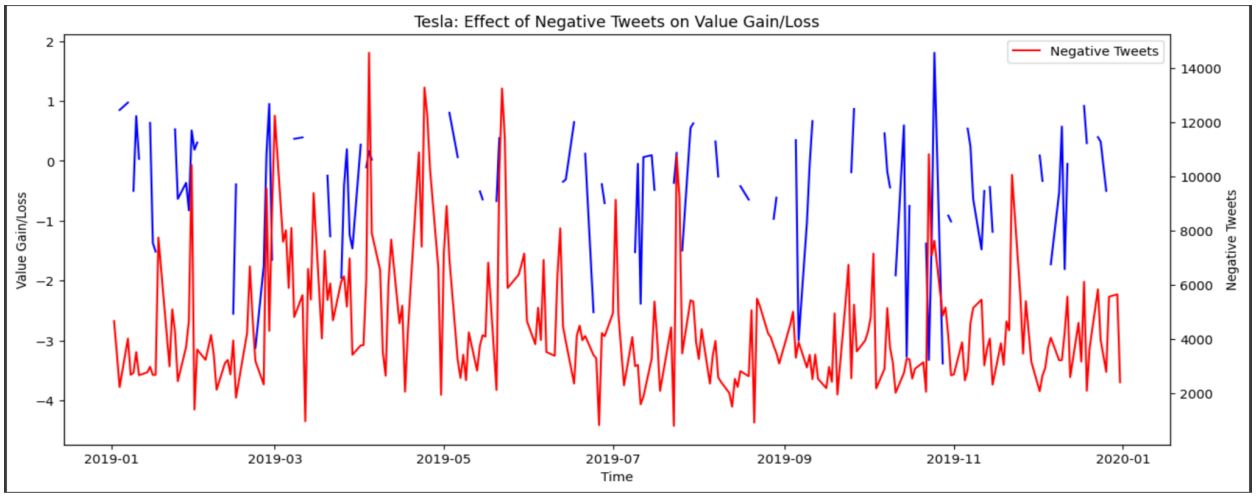
This suggests a positive correlation between the sentiment expressed in tweets and the valuation of the stocks. When more people are expressing positive opinions, optimism, or favorable views about the stocks on social media platforms, it tends to create a positive sentiment in the market. This positive sentiment can influence investors and traders, leading to an increase in demand for the stocks and subsequently driving up their valuation.



The above graph suggests that the effect of negative tweets on stock prices is not as straightforward or evident compared to the effect of positive tweets. To gain a better understanding of the relationship between negative tweets and stock prices, the suggestion is to plot the data using the logarithmic values of the "price_gain" variable.

By taking the logarithmic values of "price_gain," we can potentially uncover any underlying patterns or relationships that may not be apparent when using the original values. Logarithmic transformations are commonly used in financial analysis to normalize data, reduce skewness, and amplify small changes in values.

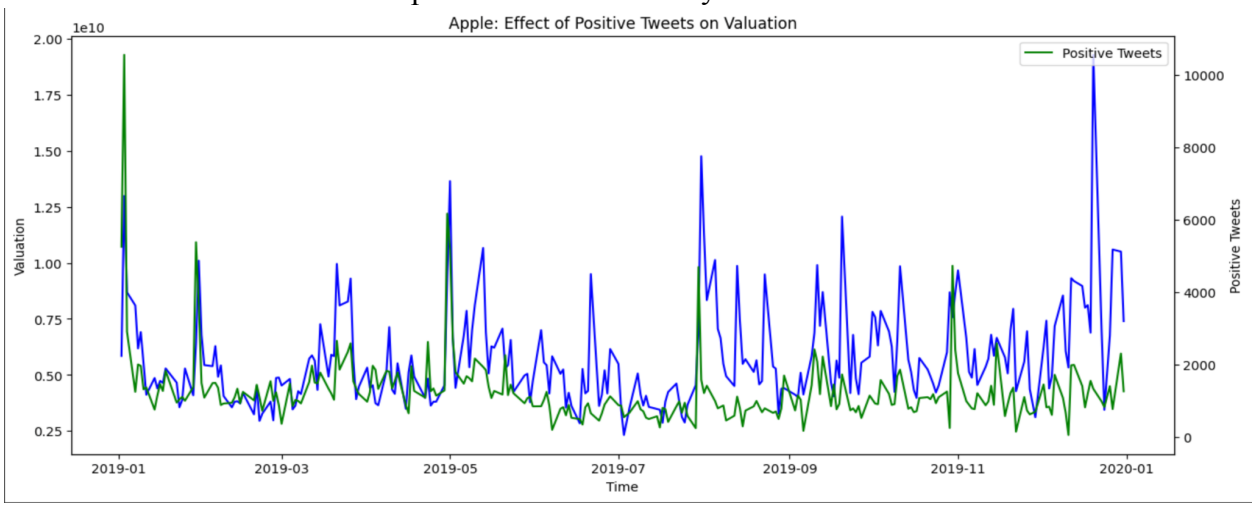
Plotting the data with the logarithmic values of "price_gain" will allow us to examine the relationship between negative tweets and stock prices from a different perspective. This approach provides an alternative way to assess the impact of negative sentiment expressed in tweets on the stock prices and potentially uncover any hidden patterns in the data.



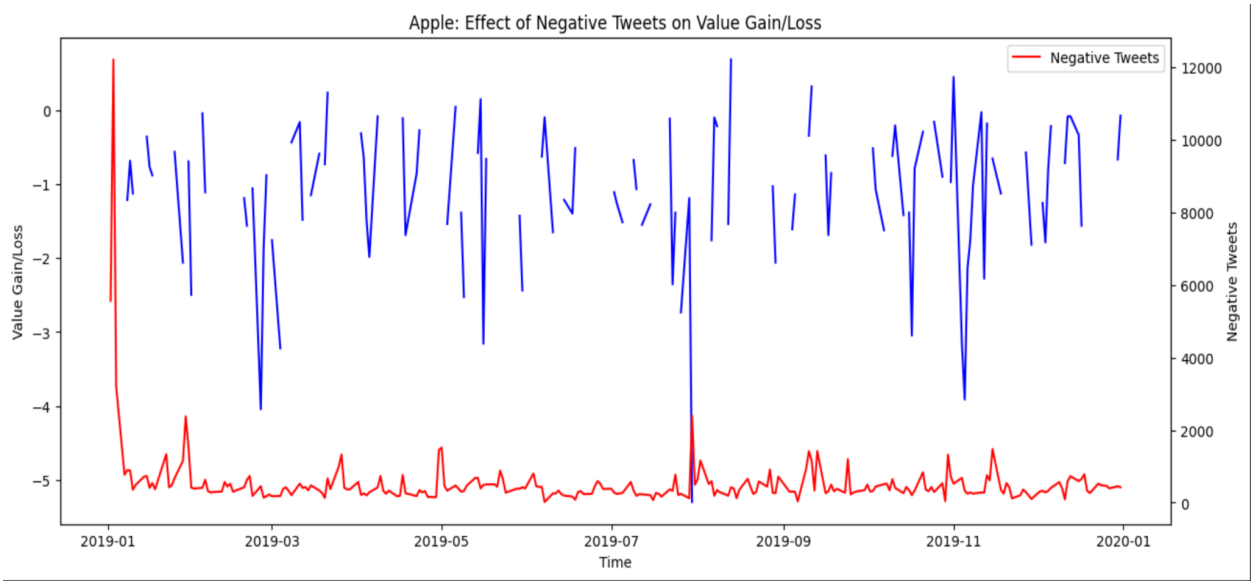
Taking logarithmic values of the 'price_gain' column provides a clearer and more informative graph. By applying the logarithmic transformation, the data is scaled in a way that makes small changes more visible and allows for better visualization of the relationship between negative tweets and price gain.

From the resulting graph, it can be observed that whenever there is a spike in the number of negative tweets, the corresponding price gain is negative. This indicates a drop in valuation or a decrease in stock prices. The negative sentiment expressed in tweets appears to have an impact on Tesla's valuation, suggesting that public opinion and sentiment play a role in influencing the stock market.

Based on this observation, it can be concluded that Tesla's valuation is indeed affected by the sentiments expressed in tweets. The graph provides visual evidence of the relationship between negative tweets and price gain, reinforcing the notion that social media sentiment can have an impact on stock market dynamics.



We can see a similar pattern is observed for Apple as well. Just like Tesla, when there is a spike in the number of positive tweets related to Apple, there is also a spike in the valuation of Apple's stock. This observation indicates a positive correlation between positive tweets and the valuation of Apple.



As seen in the graph above, there is no strong correlation between negative tweets and a drop in the share value of Apple.

Based on the graph or analysis mentioned, it is observed that spikes in negative tweets about Apple do not consistently align with a significant decrease in the stock's value. This implies that negative sentiment expressed in tweets about Apple does not consistently lead to a noticeable negative impact on the stock's valuation.

The lack of a strong correlation between negative tweets and a drop in Apple's share value could be attributed to several factors. It's important to consider that the stock market is influenced by various complex and interconnected factors, such as overall market conditions, company performance, economic indicators, investor sentiment, and global events. Negative tweets alone may not be sufficient to drive a significant decline in the share value of Apple if other positive factors or market conditions outweigh the impact of those negative sentiments.

6. Results

The observations drawn from the analysis of the given data is as follows:

1. Positive Tweet Spikes (high trending): It has been observed that when there is a significant increase in the number of positive tweets related to the stocks, there is a coinciding rise in the stock value. This suggests a positive correlation between positive sentiment expressed in tweets and the increase in stock value.
2. Negative Tweet Spikes (high trending): Similarly, when there is a significant increase in the number of negative tweets related to the stocks, there is an overall coinciding drop in the stock value. However, the correlation between negative sentiment in tweets and the decrease in stock value is not as obvious or pronounced as in the case of positive tweets.
3. Other factors contributing to stock value: It is important to note that while there is a correlation between tweet sentiment and stock value, there are also other factors at play. The rise or fall in stock values cannot be solely attributed to tweet sentiment. This is evident from the lack of a clear relationship between low to medium trending tweets and the rise or fall of stock values.

In summary, the analysis indicates that there is a relationship between tweet sentiment and stock value, particularly with positive tweets. However, it is important to consider other factors and not solely rely on tweet sentiment as the determining factor for stock value fluctuations.

7. Conclusion

Throughout this study, we have conducted a thorough analysis of the correlation between tweet sentiments and stock prices. To achieve this, we utilized two popular sentiment analysis tools, NLTK and TextBlob, and compared their effectiveness with the VADER Sentiment Analyzer.

Our findings indicate that VADER was the most effective tool for sentiment analysis classification using Twitter data. VADER's ability to quickly and accurately classify large amounts of data makes it a valuable tool for analyzing sentiment on social media platforms. We also identified some limitations in our study, such as the use of a small volume of data and a general lexicon for categorizing specific data. However, we plan to address these limitations in future work by using larger volumes of data, a specific lexicon, and a corpus for training the data to obtain better results.

Our findings suggest that tweet sentiments can have a significant impact on the stock prices of Tesla and Apple. We observed that fluctuations in stock prices and market cap were often influenced by tweet sentiments, indicating the importance of monitoring social media platforms and analyzing sentiment to gain insights into customers' opinions.

Overall, our study highlights the importance of sentiment analysis in understanding the impact of social media on stock prices. By analyzing tweet sentiments, companies can gain valuable insights into their customers' opinions and make informed decisions. We hope that our findings will encourage further research in this area and lead to the development of more effective sentiment analysis tools.

8. Future Scope

While this study has achieved its objectives, there are several areas that could be addressed in future research. One potential avenue for future work is to improve sentiment indicators as momentum indicators for the stock market. This could involve exploring new methods for sentiment analysis and incorporating additional data sources beyond Twitter.

Another area for future research is to expand the analysis beyond English sentences. While Twitter has many international users, this study solely focused on English sentences. Therefore, it would be interesting to explore how sentiment analysis can be applied to categorize sentiment in other languages as well.

Additionally, neutral tweets should not be disregarded, and adequate consideration should be given to neutral sentiment to obtain more accurate results. Furthermore, to improve the accuracy of sentiment analysis, it would be beneficial to exclude fake Twitter accounts that mislead the calculation of sentiment.

Finally, future research could focus on training and testing the Twitter data rather than relying on pre-existing libraries such as TextBlob and VADER. This would help to improve the accuracy of the sentiment analysis model and avoid misinterpretation of positive comments as negative ones. Overall, these areas of future research have the potential to enhance the effectiveness of sentiment analysis in predicting stock market trends and provide valuable insights for companies and investors.

References

1. V. D. Nguyen, B. Varghese, A. Barker, The royal birth of 2013: Analysing and visualising public sentiment in the uk using twitter, in: 2013 IEEE International Conference on Big Data, IEEE, 2013, pp. 46–54.
2. A. Java, X. Song, T. Finin, B. Tseng, Why we twitter: understanding microblogging usage and communities, in: Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis, 2007, pp. 56–65.
3. G. Ranco, D. Aleksovski, G. Caldarelli, M. Grčar, I. Mozetič, The effects of twitter sentiment on stock price returns, PloS one 10 (2015) e0138441.
4. B. J. Jansen, M. Zhang, K. Sobel, and A. Chowdury, "Twitter power: Tweets as electronic word of mouth," J. Am. Soc. Inf. Sci. Technol., vol. 60, no. 11, pp. 2169–2188, 2009.
5. V. Kharde and P. Sonawane, "Sentiment analysis of twitter data: a survey of techniques," arXiv Prepr. arXiv1601.06971, 2016.
6. S. Bird, E. Klein, and E. Loper, Natural language processing with Python: analyzing text with the natural language toolkit. " O'Reilly Media, Inc.," 2009.
7. Gilbert, C. H. E., & Hutto, E. (2014, June). Vader: A parsimonious rule-based model for sentiment analysis of social media text. In Eighth International Conference on Weblogs and Social Media (ICWSM-14). Available at (20/04/16) http://comp. social. gatech. edu/papers/icwsm14.vader. hutto.pdf (Vol. 81, p. 82).
8. Batra, R., & Daudpota, S. M. (2018, March). Integrating StockTwits with sentiment analysis for better prediction of stock price movement. In 2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET) (pp. 1-5). IEEE
9. S. Elbagir and J. Yang. "Sentiment Analysis on Twitter with Python's Natural Language Toolkit and VADER Sentiment Analyzer." In: Lecture Notes in Engineering and Computer Science. Vol. 2239. 2020, pp. 63–80. isbn: 9789881404855. doi: 10.1142/9789811215094_0005.
10. Pagolu, V. S., Reddy, K. N., Panda, G., & Majhi, B. (2016, October). Sentiment analysis of Twitter data for predicting stock market movements. In 2016 international conference on signal processing, communication, power and embedded system (SCOPES) (pp. 1345-1350). IEEE.
11. J. M. G. F. F. Sacramento, Sentiment Analysis in the Stock Market Based on Twitter Data, Ph.D. thesis, ISCTE-Instituto Universitario de Lisboa (Portugal), 2021.
12. Mitali Desai, Mayuri A. Mehta, "Techniques for Sentiment Analysisof Twitter Data: A Comprehensive Survey", International Conferenceon Computing, Communication, and Automation (ICCCA2016).
13. Sanjeev Kumar Sharma, "Sentiment Analysis: An analysis on its past,present and future scope", International Journal of advanced researchin Science and Engineering, Vol.No.6, Issue No.07, July 2017