**Stock Market Analysis Report**

**Stocks Dataset**

**Overview:**

* This dataset contains 600 rows and 5 columns. 5 columns are Positive, Negative, Neutral, adj\_price, close\_price

**Columns Description:**

1. **Positive** – Proportion of positive (values between 0 and 1)
2. **Negative** – Proportion of negative (values between 0 and 1)
3. **Neutral** – Proportion of neutral (values between 0 and 1)
4. **adj\_price** – Adjusted price of a stock market.
5. **close\_price** – Closing stock price of that day.

**Tweets dataset**

1. **Overview**

* This dataset contains 3,653 rows and 4 columns. 4 columns are Date, closing\_price, adj\_close\_price, Tweets

1. **Columns Description**
   1. **Date** – Date of record.
   2. **closing\_price** – Closing stock price of that day.
   3. **adj\_close\_price** – Adjusted price of a stock market.
   4. **Tweets** – Text data that containing tweets.

**Make Tweets dataset Unstructured to Structured**

**1. Problem Identification**

* The original dataset contained a column that name Tweets.
* Some tweets are written as multi-line text within a single cell, separated by line breaks.
* This made the data inconsistent and difficult to analyse, as each tweet was not cleanly represented in a single row.

**2. Objective**

* Ensure each row in the dataset contains one full tweet in a single row, single line.
* Preserve the associated metadata: Date, closing\_price, and adj\_close\_price.
* Remove unwanted line breaks and whitespace from tweets.

**3. Tools Used**

* Python with the pandas library for data manipulation.
* CSV files used for both input (unstructured) and output (structured).

**4. Steps of making tweets\_dataset Unstructure to Structure**

* Load original tweets\_dataset.
* Prepare for clean data.
* Itearate every row & remove row’s empty space & append break tweet line in a single row

to make dataset structure.

* Create a New DataFrame that containing the clean data.
* Save this DataFrame to a New CSV file.

**5. Result**

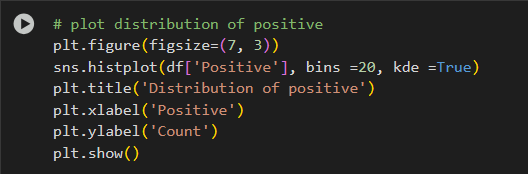
* Every tweet data is now in a single row, single line, free of line breaks.
* Date, closing\_price, and adj\_close\_price data also maintained their line following

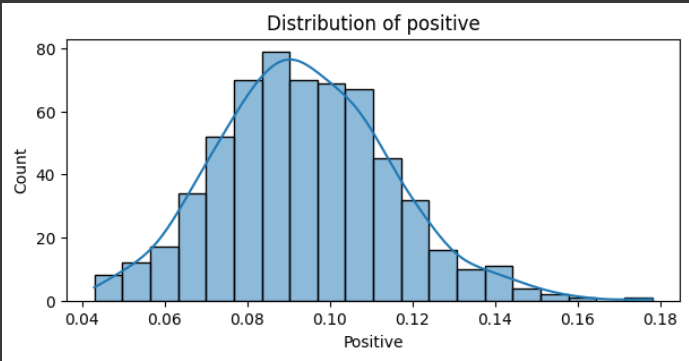
Tweets data.

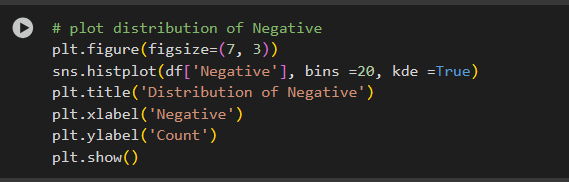
**Report on Exploratory Data Analysis**

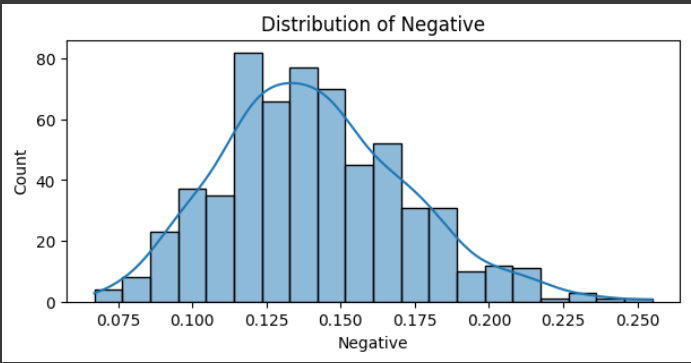
**Stocks Dataset**

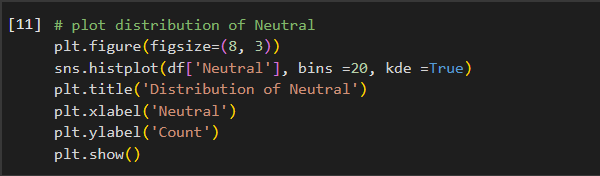
1. **Distribution Analysis**

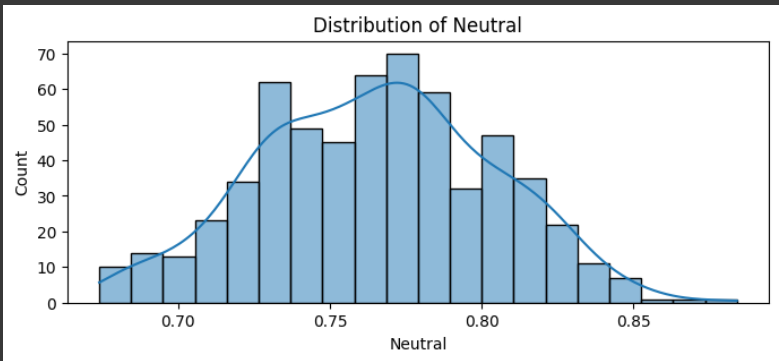
****

****

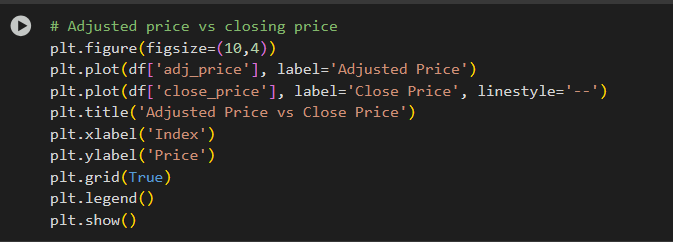


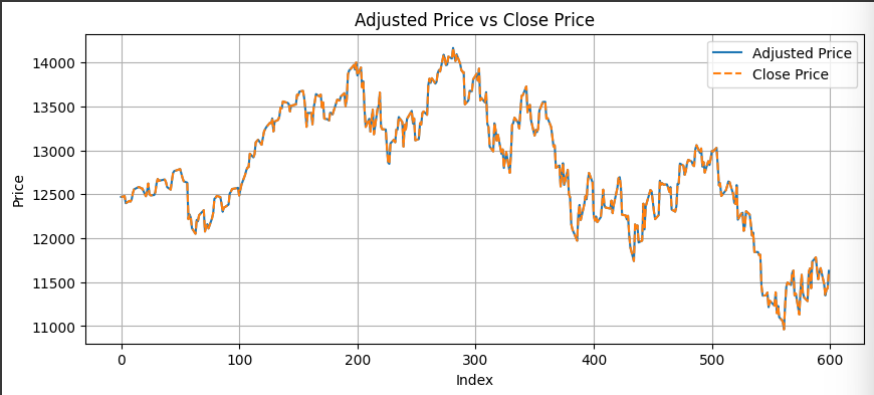




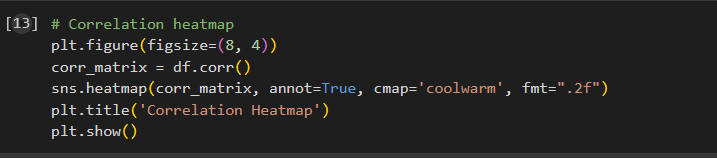


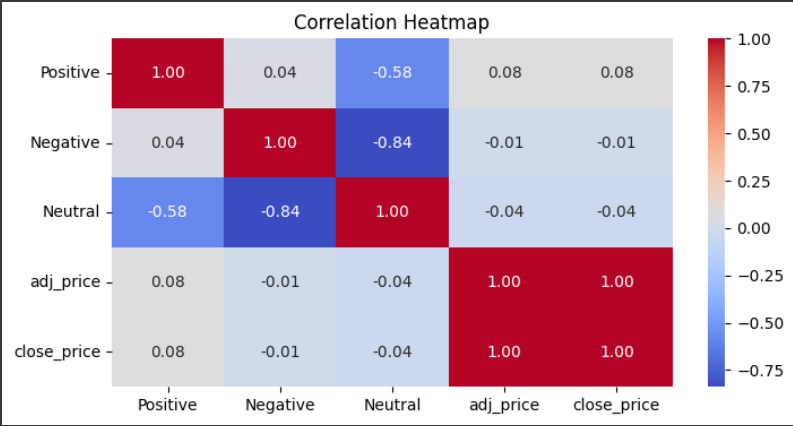
* To visualize the 'Positive', 'Negative', and 'Neutral' variables distributions histograms are plotting. Here KDE (Kernel Density Estimation) is used to provide a smoother representation of the distributions.

1. **** **Adjusted vs Closing price**



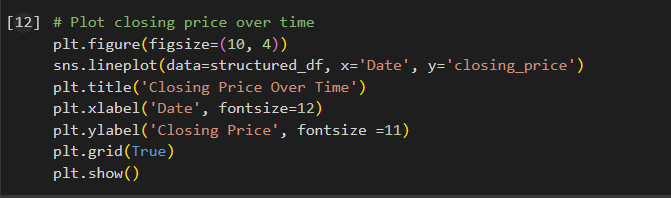
* To visualize the trends of 'adj\_price' (Adjusted Price) and 'close\_price' (Close Price) over time here Line plots are created.
* Here maximum price belongs 200 – 300 rows value of this dataset.

1. **Correlation Analysis**



* Correlation heatmap was generated to visualize the relationships between numerical variables in the dataset. This helps in identifying potential correlations or dependencies between features.

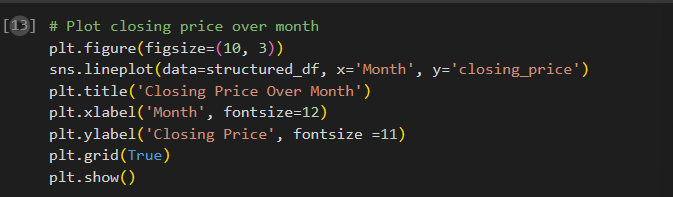
**Tweets Dataset**

1. **Closing Price Over Time**

****

* This plot shows how the closing price has changed over the specified period.
* Closing price are increased Year after year.

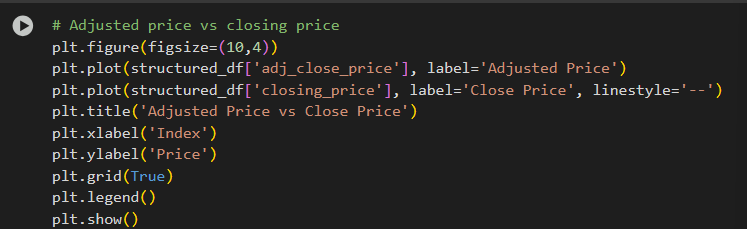
1. **Closing Price Over Month:**

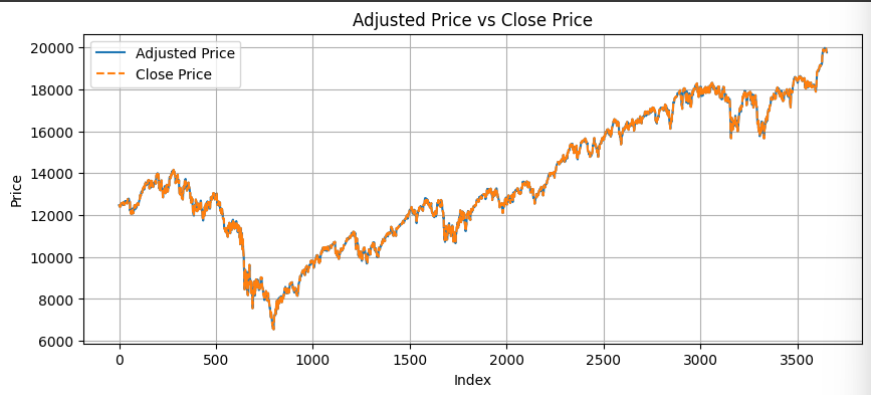
****



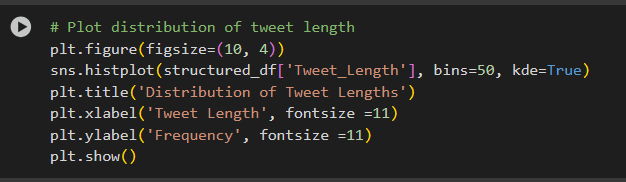
* Monthly Closing price distribution first closing price increased till month 5. In month 6 closing price decreased. Then also increased and decreased.

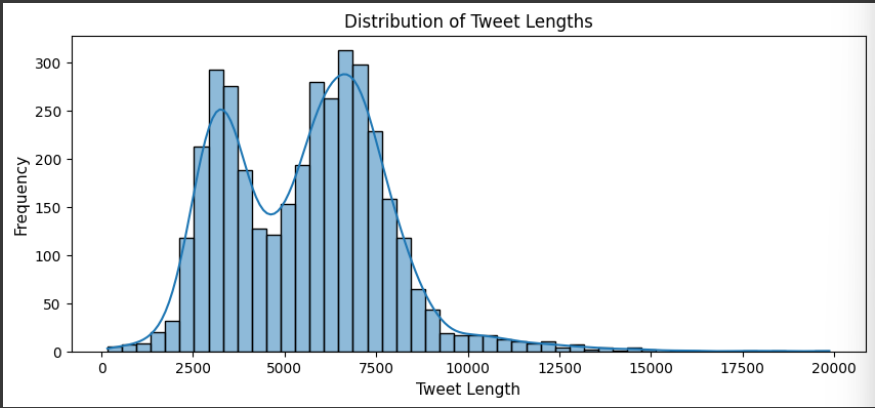
1. **Adjusted vs Closing Price**





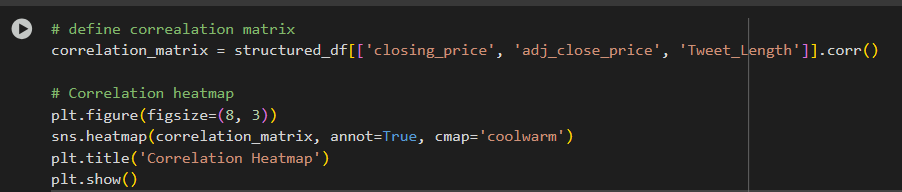
* This visualization helps understand the relationship between the two prices.
* Here Adjusted price and Close price overlaid each other. Both prices similarly decreased and increased.

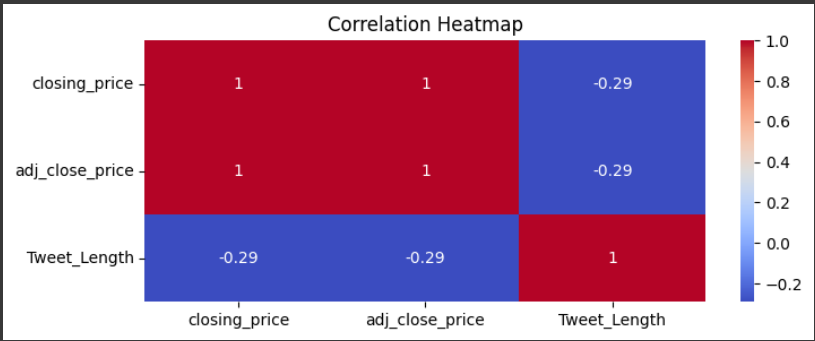
1. **Tweet Length Distribution**

****

* We create a new column 'Tweet\_Length' to store the length of each tweet.
* Here in this plot Tweet lengths distribution sometime increased, sometime decreased.

1. **Correlation Analysis**

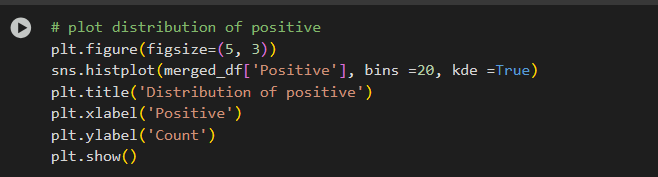
****

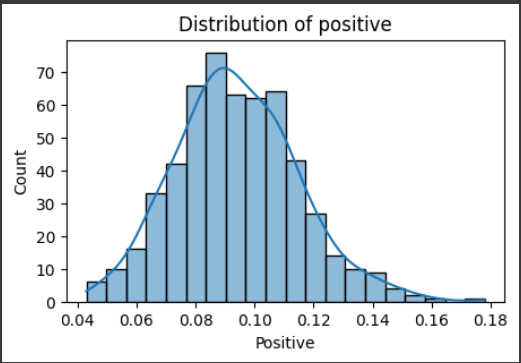
****

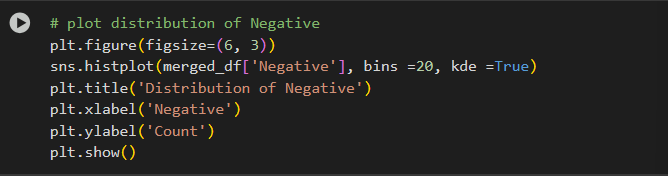
* Here in this correlation heatmap Closing price, Adjusted price, Tweet Length these variables correlation is visualized.
* A Correlation heatmap was generated to visualize the correlations between variables. Correlation helps understand potential relationships between variables.

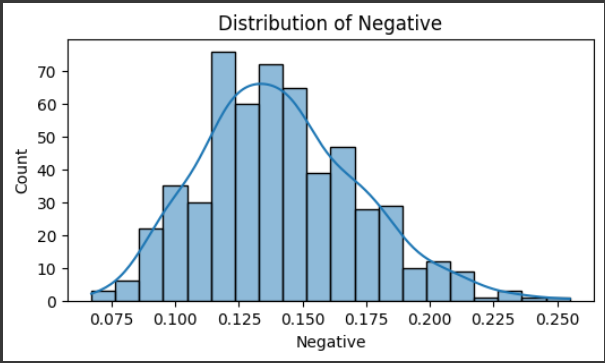
**Merged Dataset**

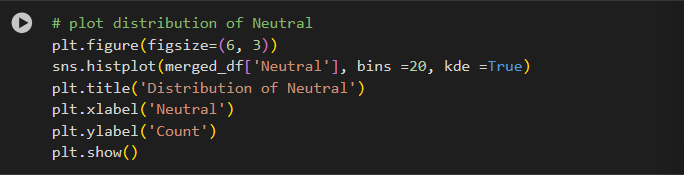
**1.** **Distribution of Sentiments:**

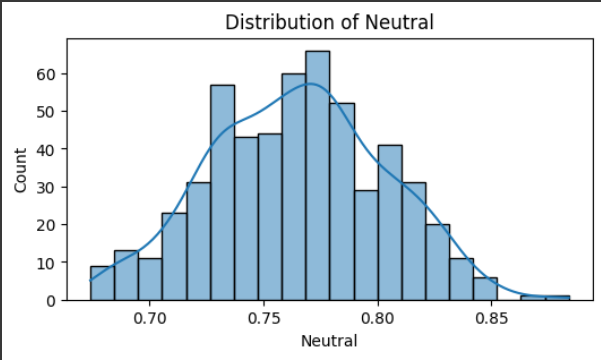
****

****

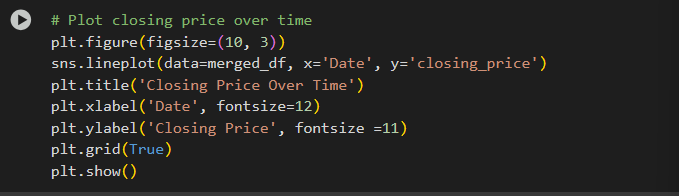
****

****

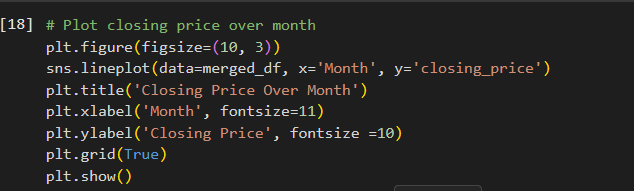
****

****

* Distribution of Positive, Negative, and Neutral Sentiments
* To visualize the 'Positive', 'Negative', and 'Neutral' variables distributions histograms are plotting. These histograms illustrate the frequency of tweets as positive, negative, and neutral.
  1. **Closing Price Over Time:**

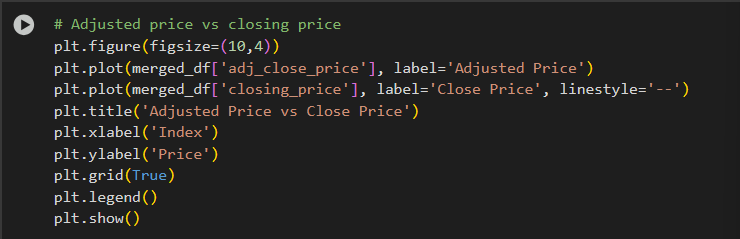
****

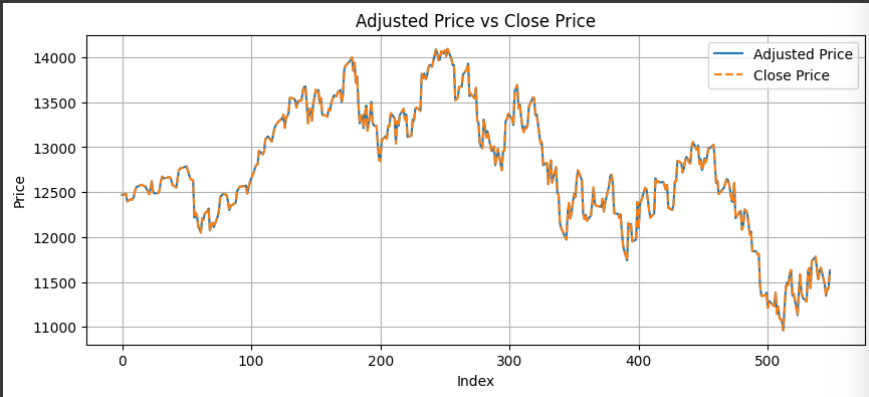
****

* This plot shows how closing price change over specified period.
* The distribution of closing price over time sometime increased, sometime decreased.
  1. **Closing Price by Month:**

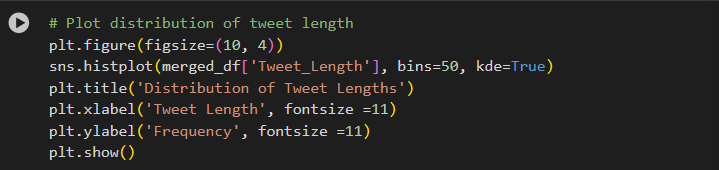
****

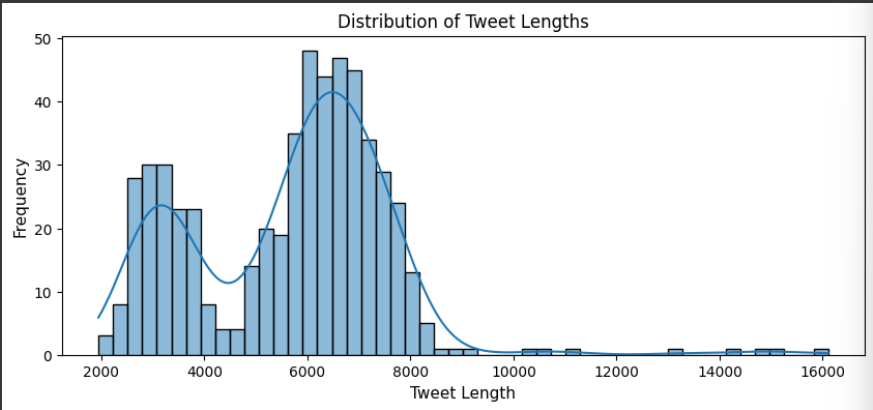
* Closing price distribution over month sometime decreased, sometime increased. Closing price was highest in the 10th month.
* It helps to identify any seasonal patterns, indicating if certain months tend to have higher or lower closing prices on average.
  1. **Adjusted Price vs. Closing Price:**

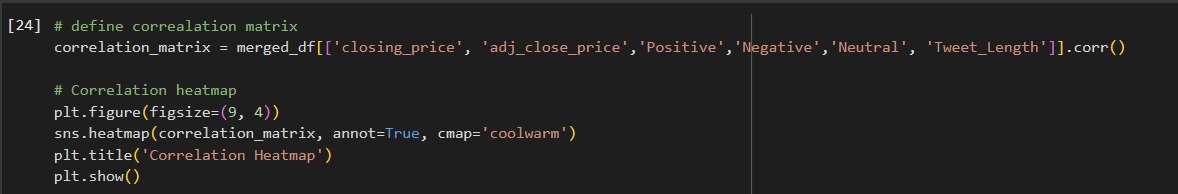
****

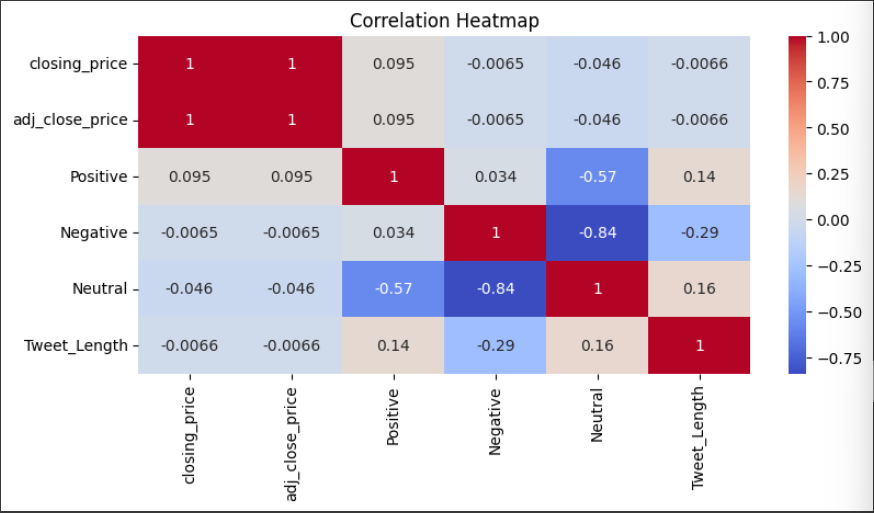
****

* This plot displays the adjusted and closing prices together, allowing for a direct comparison.
* Adjusted price and Close price both prices similarly increased and decreased.
  1. **Distribution of Tweet Lengths:**





* This histogram displays the distribution of tweet lengths, showing how many tweets fall into various length categories.
* It helps to understand the typical length of tweets in the dataset.
  1. **Correlation Heatmap:**

****

* This heatmap visualizes the correlation between closing price, Adjusted price, Positive, Negative, Neutral, Tweet Length these variables.
* Correlation helps identify potential relationships and dependencies between variables.

**Conclusion:**

Exploratory Data Analysis provides valuable insights from the dataset, particularly through the visualizations, plotting and correlation heatmap.

**Report on Natural Language Processing**

**Twitter and Merged Dataset**

**1. Text Cleaning and Preprocessing:**

The initial step in the NLP is the cleaning and preprocessing of the raw tweet text. This is important to remove noise and prepare the text for analysis. The following preprocessing steps are:

* **Removal of URLs**: URLs are often irrelevant to the sentiment of a tweet and were removed using regular expressions.
* **Removal of Mentions and Hashtags**: Mentions (@username) and hashtags (#stock) were removed as they may not contribute to the sentiment analysis in this context.
* **Removal of Punctuation**: Punctuation marks were removed to focus on the words themselves.
* **Lowercasing**: All text was converted to lowercase to ensure that words with different capitalization are treated as the same.
* **Removal of Stop Words**: Common English stop words ('the', 'a', 'is') were removed as they typically do not carry significant sentiment.

**2. Tokenization:**

* The cleaned tweets are tokenized, splitting each tweet into individual words. This is a fundamental step in NLP that breaks down text.
* The resulting tokens are stored in a new column called 'tokens'.

**3. Stemming:**

Using the Porter Stemmer, words are reduced by removing suffixes or prefixes ("running" becomes "run"). This is a more aggressive approach and aims to find root word removing suffix or prefix.

* Stemmed tokens are stored in 'stemmed\_tokens'.

**4. Lemmatization:**

Using the WordNet Lemmatizer, words are reduced to their dictionary or base form ("running" becomes "run", "better" becomes "good"). This is a less aggressive approach and aims for grammatically correct base forms.

* Lemmatized tokens are stored in 'lemmatized\_tokens'.

**5. Corpus Creation:**

* A corpus is created from the 'Clean\_Tweet' column. A corpus is a collection of text documents, which is essential for many NLP tasks, including text vectorization and topic modeling.
* The tolist() method is used to convert the pandas Series of cleaned tweets into a list, forming the corpus.

**6. TF-IDF Vectorization**

The TF-IDF (Term Frequency-Inverse Document Frequency) technique is used to convert the cleaned text data (the corpus) into a numerical representation.This method assigns weights to words based on their frequency in a document and across the entire corpus.

* **TF**: Frequency of a word in a document
* **IDF**: Rarity of the word across all documents

**7. Sentiment Analysis**

* Sentiment analysis was performed to determine the emotional tone of each tweet.
* Sentiment Intensity Analyzer is used to calculate a compound sentiment score for each cleaned tweet. The compound score provides a single metric representing the overall sentiment (positive, negative, or neutral).

VADER (Valence Aware Dictionary and Sentiment Reasoner) sentiment analysis tool is employed to determine the emotional tone of the tweets. VADER is specifically designed for social media text and is sensitive to emojis, slang, and punctuation.

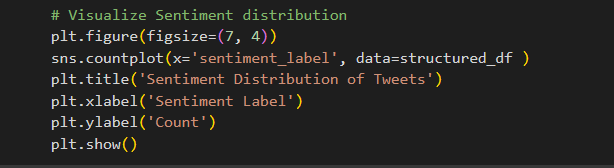
**8. Sentiment Categorization:**

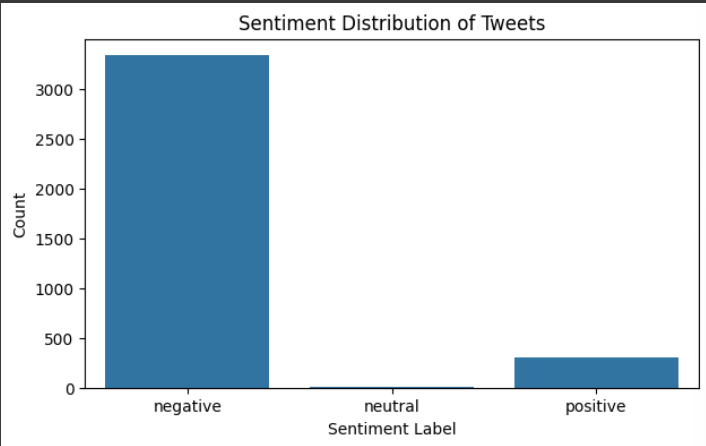
The numerical 'sentiment\_score' was categorized into three discrete sentiment labels: 'negative', 'neutral', and 'positive'. These labels are stored in the sentiment\_label column.

* Scores from -1 to -0.05 were labeled as 'negative'.
* Scores from -0.05 to 0.05 were labeled as 'neutral'.
* Scores from 0.05 to 1 were labeled as 'positive'.

This categorization provides a more intuitive representation of sentiment and allows for easier visualization and analysis of the sentiment distribution.

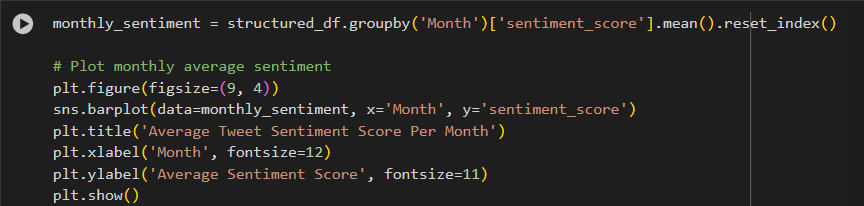
**Twitter Dataset**

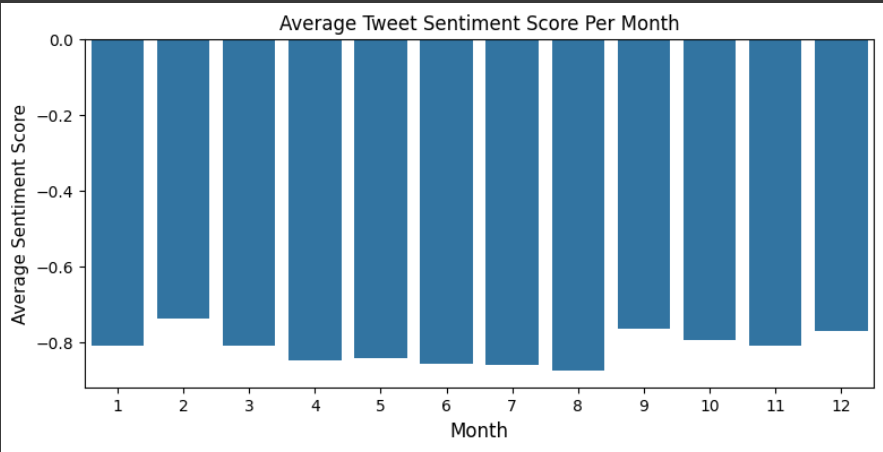
**9. Sentiment Distribution over Time:**

****

* A countplot was generated to visualize the distribution of the categorized sentiment labels 'negative', 'neutral', 'positive'.
* This plot provides a clear overview of the proportion of tweets falling into each sentiment category.

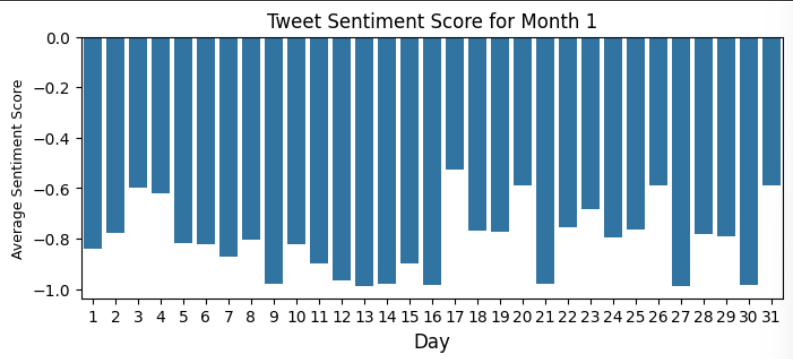
**10.** **Monthly Sentiment Distribution:**

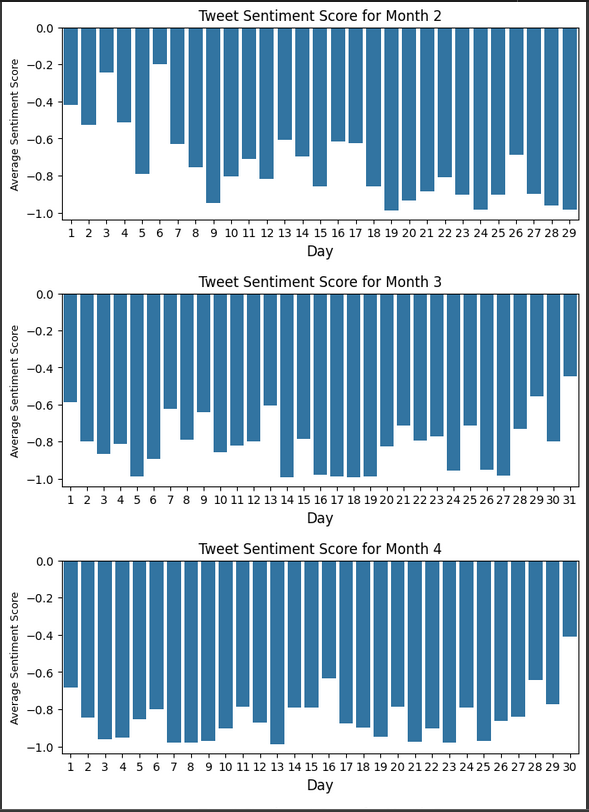
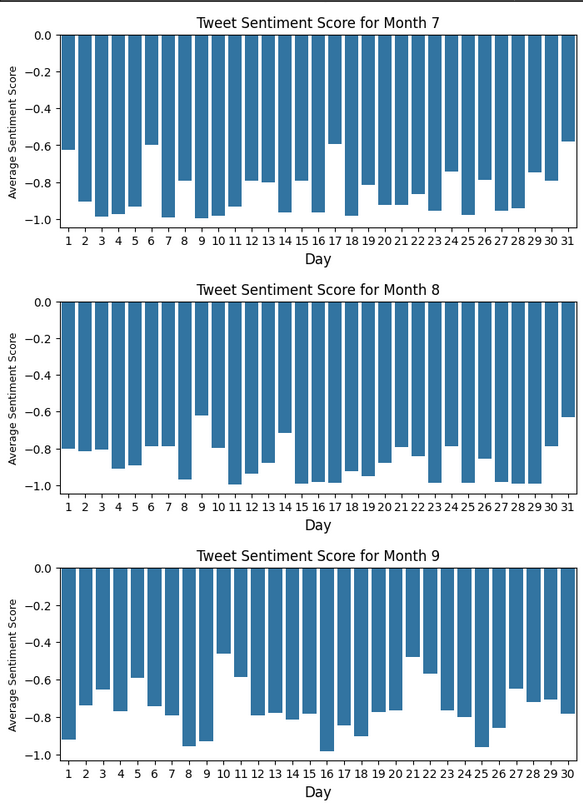


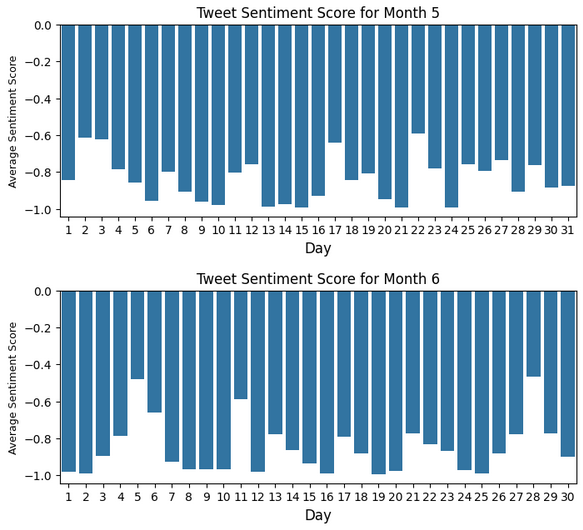


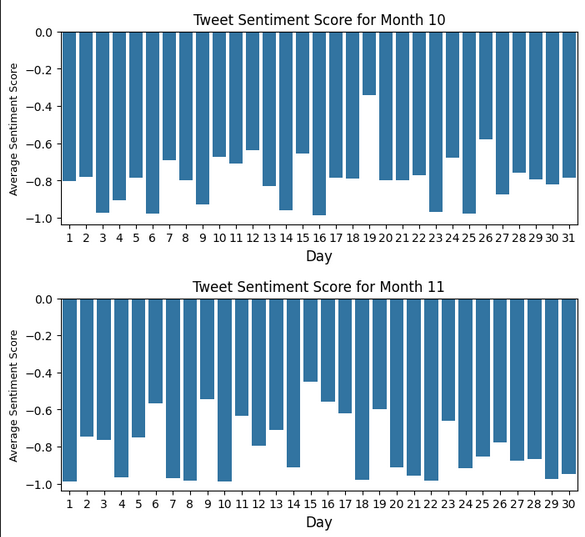
* The average sentiment score was calculated for each month to observe trends in sentiment over time.
* A bar plot was created to visualize the average sentiment score per month, allowing for identification of months with positive, negative, or neutral sentiment.

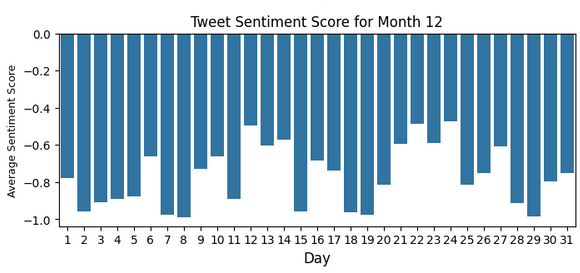
**11.** **Daily Average Sentiment per Month Distribution:**

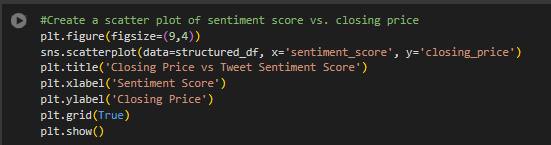


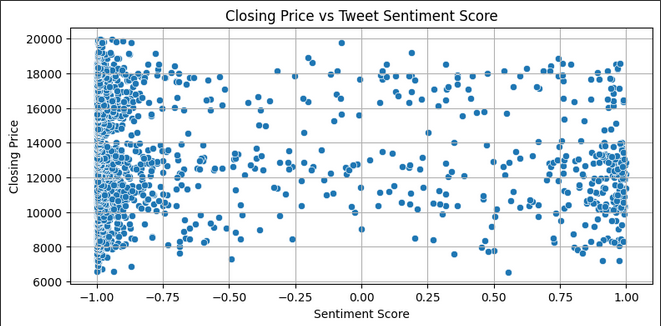




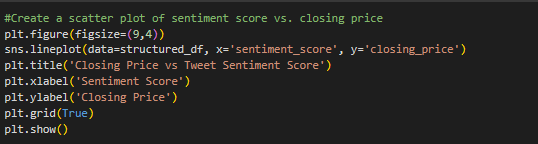


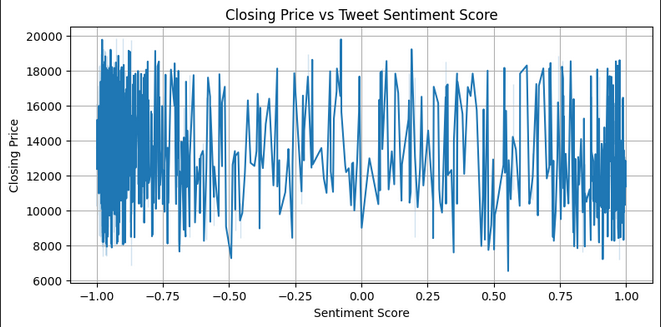


* To visualize the average sentiment score on a daily basis for each month bar plots were generated.
  1. **Closing Price vs Tweets Sentiment Score (Scatter Plot):**

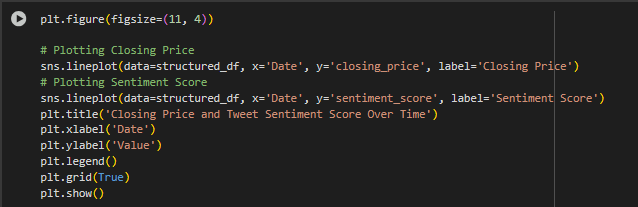


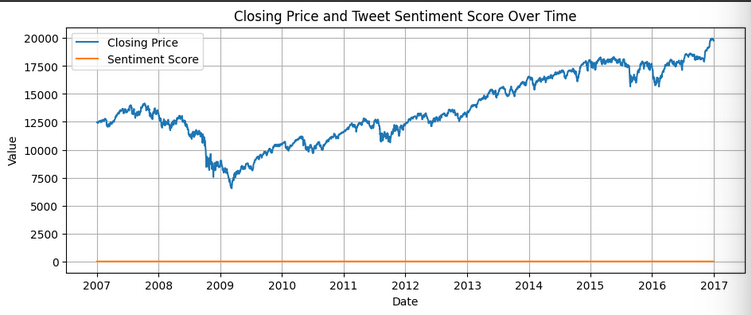
* A scatter plot was created with sentiment score on the x-axis and closing price on the y-axis.
* Scatter plot was created to visualize the relationship between the tweet sentiment score and the closing price.
  1. **Closing Price vs Tweet Sentiment Score (Line Plot):**

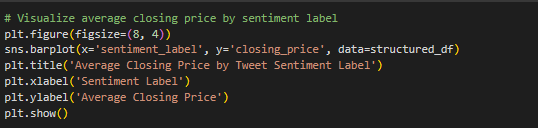


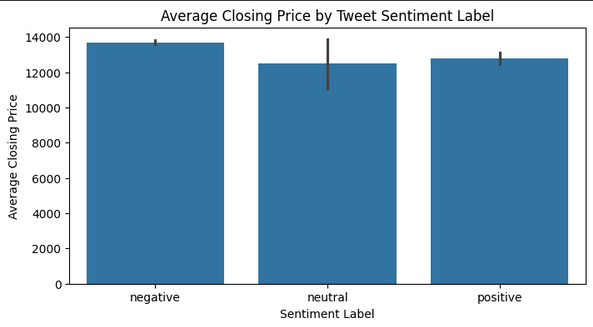


* This plot provides a visual representation of how the closing price changes with varying sentiment scores.
  1. **Closing Price and Sentiment Score Over Time:**



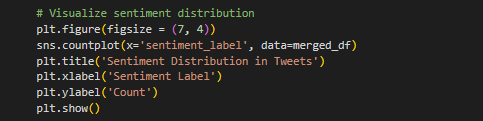


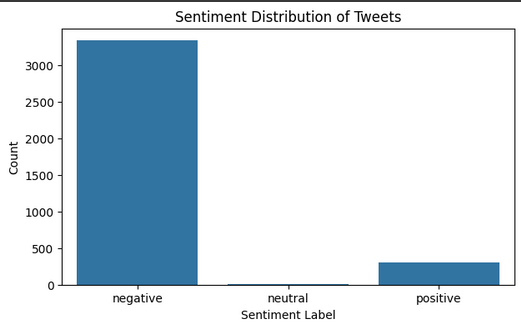
* This dual-axis line plot displaying both the closing price and the average sentiment score over time.
* This plot allows for a direct visual comparison of the trends of closing price and sentiment over time.
  1. **Average Closing Price by Sentiment Label:**



* This bar plot comparing the average closing price for each sentiment label ('negative', 'neutral', 'positive').
* This visualization helps to see a noticeable difference between average closing price and different sentiment categories.

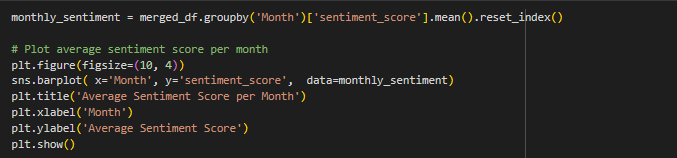
**Merged Dataset**

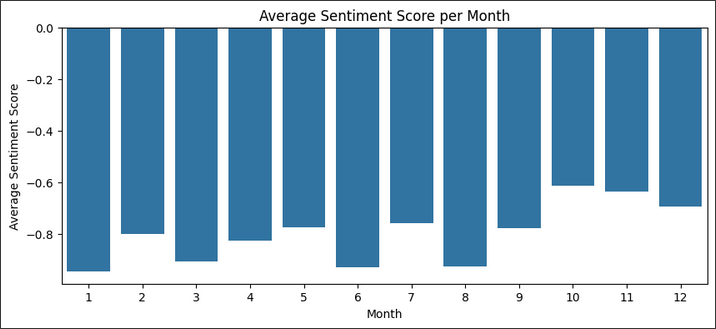
**9. Sentiment Distribution over Time:**

****

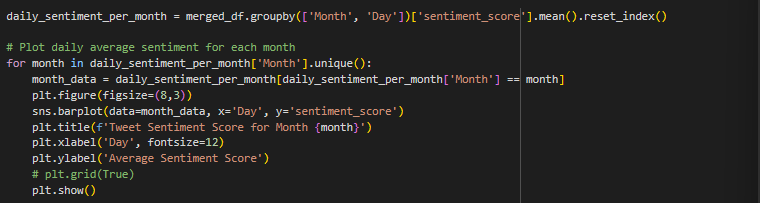
* A count plot was generated to visualize the distribution of the categorized sentiment labels 'negative', 'neutral', 'positive'.
* This plot provides a clear overview of the proportion of tweets falling into each sentiment category.

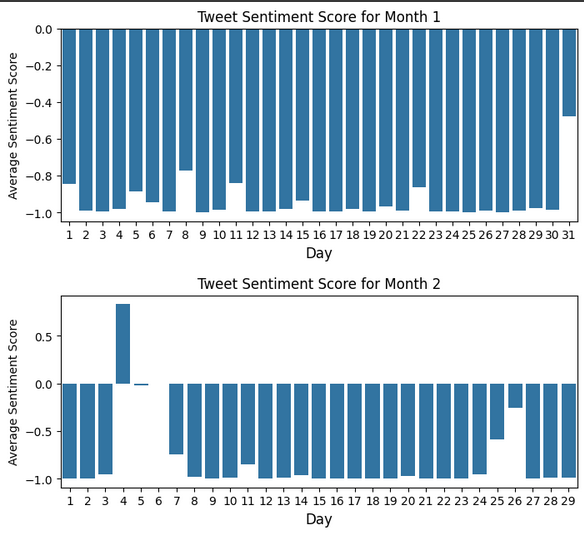
**10.** **Monthly Sentiment Distribution:**

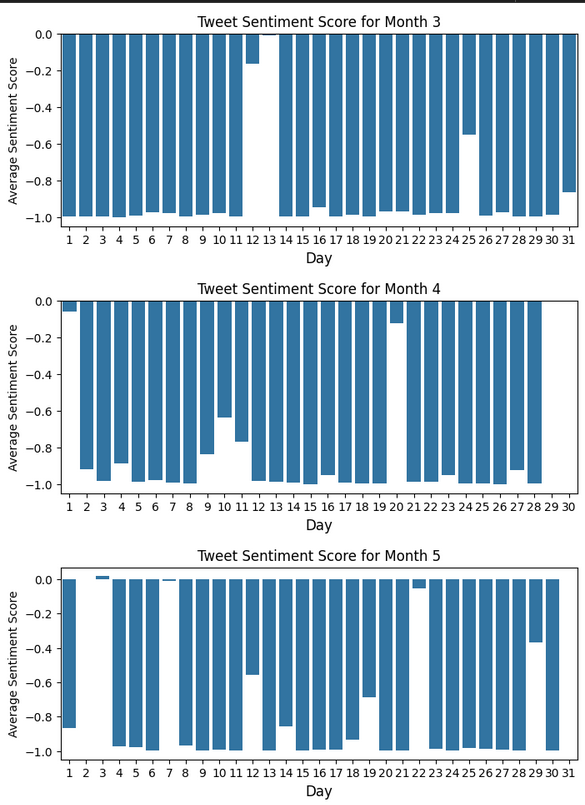
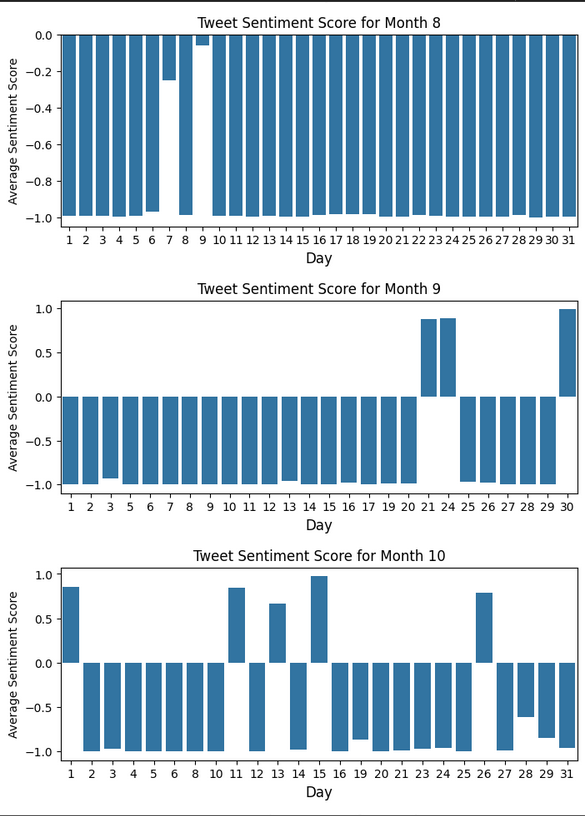


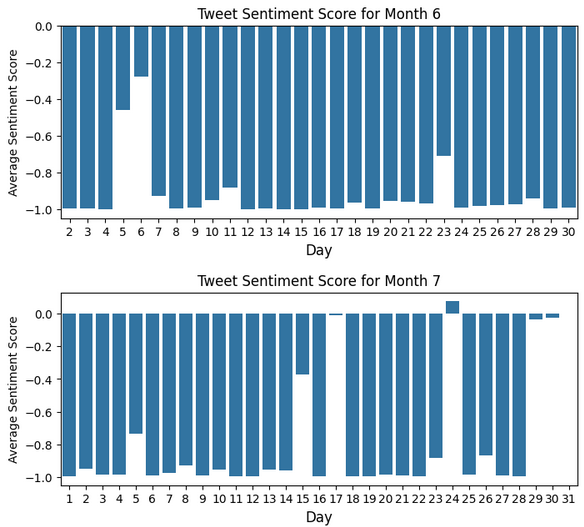


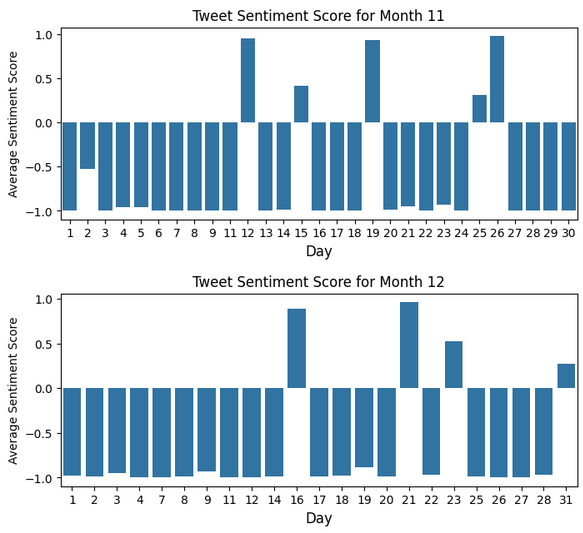
* The average sentiment score was calculated for each month to observe trends in sentiment over time.
* A bar plot was created to visualize the average sentiment score per month, allowing for identification of months with positive, negative, or neutral sentiment.

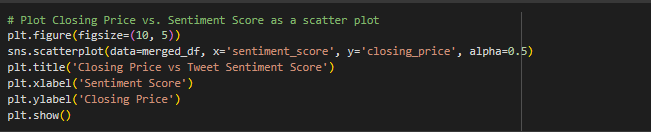
**11.** **Daily Average Sentiment per Month Distribution:**

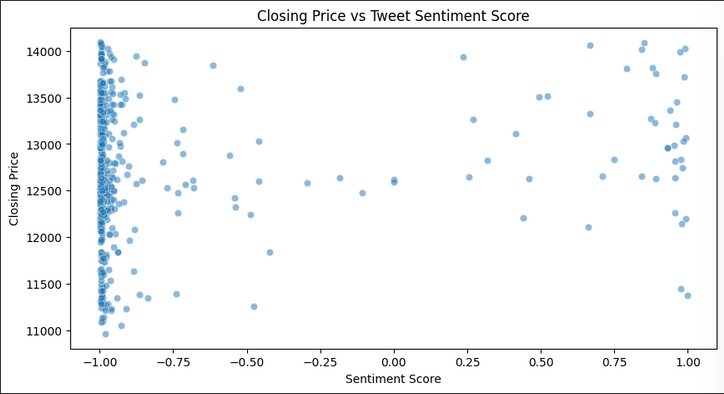


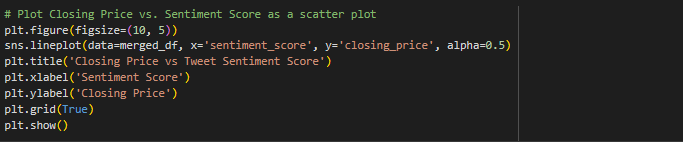




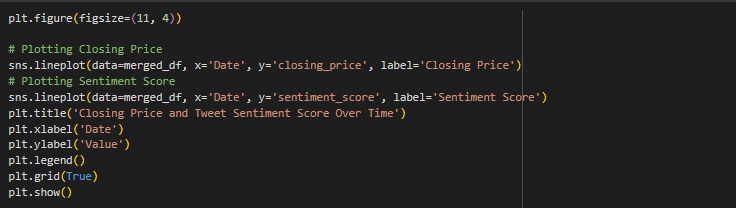


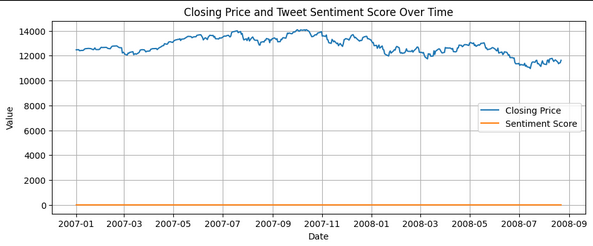
* To visualize the average sentiment score on a daily basis for each month bar plots were generated.
  1. **Closing Price vs Tweets Sentiment Score (Scatter Plot):**



* A scatter plot was created with sentiment score on the x-axis and closing price on the y-axis.
* Scatter plot was created to visualize the relationship between the tweet sentiment score and the closing price.
  1. **Closing Price vs Tweet Sentiment Score (Line Plot):**



* This plot provides a visual representation of how the closing price changes with varying sentiment scores.
  1. **Closing Price and Sentiment Score Over Time:**



* This dual-axis line plot displaying both the closing price and the average sentiment score over time.
* This plot allows for a direct visual comparison of the trends of closing price and sentiment over time.

**Conclusion: -**

This report presents a comprehensive analysis of stock market trends and sentiment impact using structured and unstructured data. Through detailed data preprocessing, exploratory data analysis (EDA), and natural language processing (NLP), we derived meaningful insights from both numerical and textual sources.

The **NLP and sentiment analysis** portion of the report transformed raw tweets into actionable sentiment scores. Using techniques like tokenization, stemming, lemmatization, and TF-IDF vectorization, we effectively quantified tweet content. VADER-based sentiment scoring provided a reliable measure of tweet tone, which was then correlated with market trends.