Sentiment Analysis

Submitted by:
Rithin C
Anusha K
Santhoshi S
Krishna G

Submission Date: May 11, 2024

DeVos Graduate School of Management
Northwood University, Midland, MI 48640

Abstract

Social media platforms have become important sources of user-generated content, providing insightful information on sentiment, trends, and public opinion. This thesis investigates sentiment analysis methods used on social media data, particularly on Twitter. Using the Twitter handle of a prominent leader, we have done sentiment analysis of how followers have reacted.

Introduction

Social media platforms have revolutionized communication, providing a vast repository of user-generated content that reflects public opinion, trends, and sentiments. Among the myriad applications of social media data analysis, sentiment analysis stands out as a powerful tool for understanding how people perceive specific topics, events, or brands.

In this context, political parties have increasingly turned to sentiment analysis on platforms like Twitter, Facebook, and Instagram to gain insights into voter sentiment, monitor campaign effectiveness, and track competitor messaging. This paper explores the key aspects of sentiment analysis on social media, the challenges it faces, and its critical role in shaping political strategies.

To extract valuable insights, we explore the models, methodologies, and visualizations involved in sentiment analysis on social media in this study. We also talk about sentiment analysis's consequences for political parties, illuminating how it influences elections and influences policy choices. We hope to further our understanding of the digital environment where politics and public sentiment collide by looking at sentiment analysis's advantages and disadvantages.

Problem Statement

Social media platforms have developed into effective instruments for political communication in recent years. Reputable political figures frequently utilize Twitter to share announcements, policies, and points of view. On the other hand, there is a wide range of reactions to these tweets, from fervent condemnation to exuberant approval. It is important to comprehend the sentiment conveyed in these responses for a number of reasons:

Public Perception and Influence: The tone of followers' responses indicates how the general public interprets the messages of the leader. The credibility, popularity, and overall influence of the leader can be affected by favorable or unfavorable attitude.

Policy Feedback and Adaptation: We can determine how effectively a leader's policies are received by their followers by analyzing sentiment. While negative opinion may point to areas that need improvement or modification, positive sentiment may reflect policy alignment.

Polarization and Acoustic Cavities: Sentiment analysis is useful in identifying polarization among followers. Do different viewpoints coexist or are there echo chambers where only people who share those viewpoints interact?

Methodology

Using Jupyter-Notebook we have run the following code for the dataset that we have acquired from Kaggle.

```
# Step 1: Import Libraries

import pandas as pd

from textblob import TextBlob

import matplotlib.pyplot as plt


# Step 2: Load Dataset with explicit encoding

file_path = "C:\\Users\\Admin\\Downloads\\labeled_dataset.csv"  # Modify the file path here

df = pd.read_csv(file_path, encoding='latin1')  # Try different encodings if needed

# Print column names to identify the correct column for text

print(df.columns)
```

Explanation:

This section imports necessary libraries for data manipulation, sentiment analysis, and visualization: pandas, TextBlob, and matplotlib.pyplot. It specifies the file path of the dataset and loads it into a pandas DataFrame (df). The encoding parameter is used to handle any encoding issues that might arise when reading the CSV file. It prints the column names of the DataFrame to identify the correct column containing the text data.

```
# Step 3: Data Exploration

print(df.head())


# Step 4: TextBlob Sentiment Analysis

def analyze_sentiment(text):

    analysis = TextBlob(text)

    if analysis.sentiment.polarity > 0:
```

```
        return 'positive'

    elif analysis.sentiment.polarity == 0:

        return 'neutral'

    else:

        return 'negative'
```

Assuming 'clean_text' is the correct column name for the text df['Sentiment'] = df['clean_text'].apply(analyze_sentiment)

Print 10 texts with their sentiment

print(df[['clean_text', 'Sentiment']].head(10))

Explanation:

This section explores the dataset by printing the first few rows using df.head().

It defines a function analyze_sentiment() that takes a text input, analyzes its sentiment using TextBlob, and returns a sentiment label ('positive', 'neutral', or 'negative') based on the polarity of the sentiment score. The function is then applied to a column named 'clean_text' in the DataFrame using the apply() function, and the results are stored in a new column named 'Sentiment'. It prints the first 10 texts along with their corresponding sentiment labels.

Step 5: Data Analysis

sentiment_counts = df['Sentiment'].value_counts()

Plotting the sentiment distribution as a pie chart

plt.figure(figsize=(8, 6))

plt.pie(sentiment_counts, labels=sentiment_counts.index, autopct='%1.1f%%', colors=['green', 'blue', 'red'], startangle=140)

plt.title('Sentiment Analysis')

plt.axis('equal') # Equal aspect ratio ensures that pie is drawn as a circle.

Display the plot

plt.show()

Explanation:

This section calculates the count of each sentiment category ('positive', 'neutral', 'negative') using the value_counts() function. It plots a pie chart to visualize the sentiment distribution using matplotlib.pyplot. The pie chart is customized with appropriate labels, colors, and title. The plot is displayed using plt.show().

print(sentiment_counts)

Explanation:

This section prints the count of each sentiment category to provide a numerical summary of the sentiment analysis results. It gives an overview of the sentiment distribution in the dataset.

Below is the screenshot of the code.

```python
# Step 1: Import Libraries
import pandas as pd
from textblob import TextBlob
import matplotlib.pyplot as plt

# Step 2: Load Dataset with explicit encoding
file_path = "C:\\Users\\Admin\\Downloads\\labeled_dataset.csv"   # Modify the file path here
df = pd.read_csv(file_path, encoding='latin1')   # Try different encodings if needed

# Print column names to identify the correct column for text
print(df.columns)

# Step 3: Data Exploration
print(df.head())

# Step 4: TextBlob Sentiment Analysis
def analyze_sentiment(text):
    analysis = TextBlob(text)
    if analysis.sentiment.polarity > 0:
        return 'positive'
    elif analysis.sentiment.polarity == 0:
        return 'neutral'
    else:
        return 'negative'

# Assuming 'clean_text' is the correct column name for the text
df['Sentiment'] = df['clean_text'].apply(analyze_sentiment)

# Print 10 texts with their sentiment
print(df[['clean_text', 'Sentiment']].head(10))

# Step 5: Data Analysis
sentiment_counts = df['Sentiment'].value_counts()

# Plotting the sentiment distribution as a pie chart
plt.figure(figsize=(8, 6))
plt.pie(sentiment_counts, labels=sentiment_counts.index, autopct='%1.1f%%', colors=['green', 'blue', 'red'], startangle=140)
plt.title('Sentiment Analysis')
plt.axis('equal')   # Equal aspect ratio ensures that pie is drawn as a circle.


print(sentiment_counts)
```
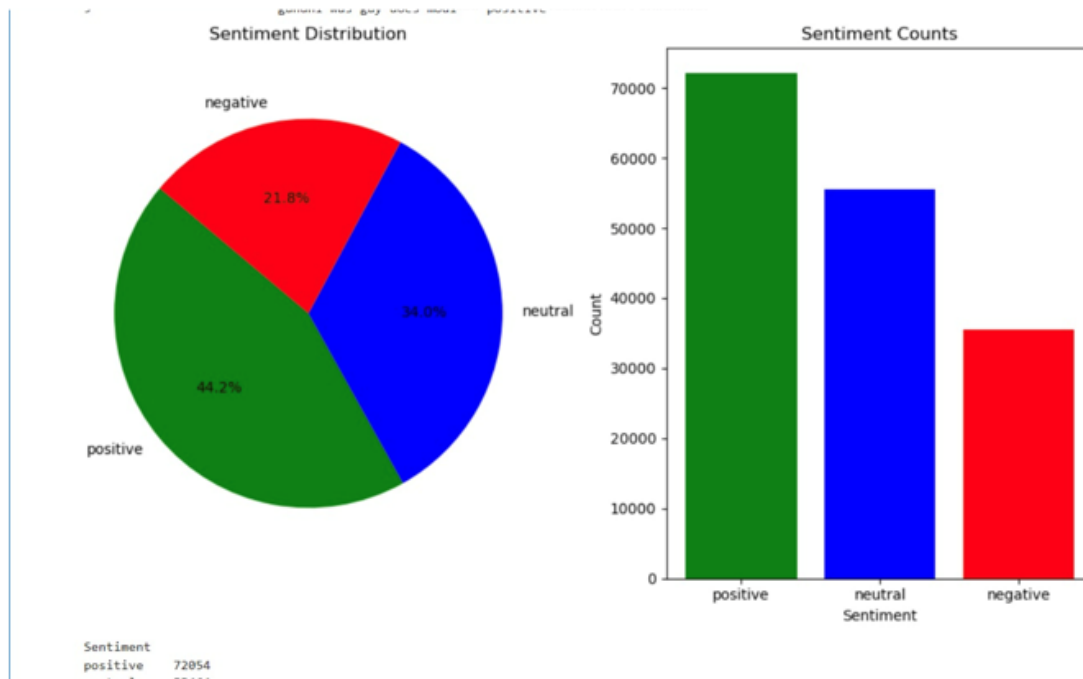
Index(['clean_text', 'sentiment'], dtype='object')
                      clean_text sentiment

Results

Below is the pie chart of sentiment distribution.

Sentiment Distribution (Pie Chart): The sentiment distribution in the examined tweets is displayed in the pie chart on the left.
Positive Sentiment (Green): A positive sentiment is present in about 44.2% of the tweets.
Neutral Sentiment (Blue): About 34.7% of tweets are classified as neutral.
Negative mood (Red): A negative mood is expressed in about 21.1% of the tweets.

Sentiment Counts (Bar Graph): A count for each sentiment category is shown in the bar graph on the right.
Positive: There are more than 60,000 positive tweets.
Neutral: There are over fifty thousand neutral tweets.
Negative: There are over 30,000 tweets that express negativity.

Future Scope & Conclusion

With its ability to analyze vast amounts of text data, such as news articles, social media posts, opinions, and recommendations, MLOps can be quite useful in forecasting election results. Voter preferences can be ascertained by using sentiment analysis to determine how the public feels about political parties, candidates, and issues. During elections, MLOps enables constant

social media platform monitoring.

Political campaigns and government organizations can better grasp public opinion, spot patterns, and modify their tactics by utilizing real-time sentiment analysis.

MLOps models are able to identify contentious issues and false information that is propagating on social media. Governments may combat misinformation and uphold openness by proactively addressing stuff that polarizes people. Sentiment analysis is useful for evaluating how the general public will respond to new laws or proposed policies. This data can be used by governments to enhance citizen communication, address issues, and improve policies.

MLOps can assess sentiment in times of crisis (natural disasters, pandemics, etc.) to ascertain public opinion and customize responses. Sentiment information can help governments better allocate resources and meet the needs of their constituents. Candidates can better evaluate the impact of their campaigns by using MLOps to measure sentiment surrounding them. Additionally, it can see possible problems early on, enabling campaigns to modify their messaging.

References

1. Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. Foundations and Trends® in Information Retrieval, 2(1–2), 1–135.
2. Agarwal, A., Xie, B., Vovsha, I., Rambow, O., & Passonneau, R. (2011). Sentiment analysis of Twitter data. Proceedings of the Workshop on Languages in Social Media, 30–38.
3. Barberá, P. (2015). Birds of the same feather tweet together: Bayesian ideal point estimation using Twitter data. Political Analysis, 23(1), 76–91.
4. Gayo-Avello, D. (2012). I wanted to predict elections with Twitter and all I got was this lousy paper: A balanced survey on election prediction using Twitter data. arXiv preprint arXiv:1204.6441.
5. Jungherr, A. (2016). Twitter use in election campaigns: A systematic literature review. Journal of Information Technology & Politics, 13(1), 72–91.

Video link

Recording-20240512_142805.webm

Github link

[Upload files • CR461/Final-project (github.com)](github.com)