Project Report

AstraZeneca Vaccine Twitter Sentiment Analysis

Manthan Pandey

Table of Contents

INTRODUCTION

In today's digital era, social media has become a crucial platform for organizations to interact with their audience, share updates, and understand public sentiment. AstraZeneca, a global biopharmaceutical company, is no exception. The company's communication strategy on platforms like Twitter reflects its efforts to engage with stakeholders, promote its products, and address public concerns.

Understanding how the public perceives these communications is essential for refining engagement strategies, addressing concerns, and building trust. This sentiment analysis project is designed to classify tweets related to AstraZeneca into three sentiment categories: Positive, Neutral, and Negative. By leveraging machine learning, the project aims to extract actionable insights from public feedback on Twitter.

Project Scope

The project involves:

1. **Text Analysis**: Preprocessing tweets to clean and standardize text for analysis.
2. **Modeling**: Using supervised machine learning to classify sentiments in the tweets.
3. **Insights Extraction**: Understanding the distribution of public sentiment, highlighting strengths and areas for improvement in AstraZeneca's communication strategy.

PROBLEM STATEMENT

The objective of this project is to analyze public sentiment expressed in tweets related to AstraZeneca. By categorizing the sentiments as Positive, Neutral, or Negative, the analysis provides insights into the public's perception of AstraZeneca's communication efforts and brand reputation. This information can support strategic decision-making and communication improvements.

ABSTRACT

This report outlines the process and outcomes of a sentiment analysis project focused on tweets mentioning AstraZeneca. The tweets were processed, vectorized, and analyzed using machine learning techniques. The results demonstrate an overall accuracy of 70.4%, with effective identification of Positive and Neutral sentiments but challenges in detecting Negative sentiments. Key insights and knowledge gained from the analysis are highlighted, including areas for improvement and recommendations for addressing class imbalance.

METHODOLOGY

1. Dataset Exploration:

- The dataset included ***1,552 tweets***, each labeled with a sentiment (***Positive***, ***Neutral***, **Negative**) and accompanied by numerical sentiment scores (***Polarity***, ***Subjectivity***).
- The sentiment distribution revealed class imbalance, with ***739 Positive, 677 Neutral***, and ***136 Negative*** tweets.

2. Data Preprocessing:

- Rows with missing values in any column were removed to ensure a clean dataset.
- URLs, punctuation, special characters, and numbers were removed using regular expressions.
- Tweets were converted to lowercase for uniformity.
- Common stop words (e.g., "the", "and", "is") were removed to reduce noise in the data.
- Text was split into individual words to facilitate further processing.
- Lemmatization or stemming was not applied to keep preprocessing efficient and lightweight.
- A new column, ***Cleaned_Tweet***, was created containing the preprocessed version of each tweet.

3. Modeling:

- Text was vectorized using the ***TF-IDF method***, with a maximum of ***5,000 features***. This approach emphasizes terms that are frequent in a specific tweet but rare across all tweets, capturing important features for classification.
- A Logistic Regression classifier was chosen for its simplicity and performance.
- The dataset was split into ***Training Set (80%)*** which is used to train the sentiment classification model and ***Testing Set (20%)*** which is used to evaluate the model's performance on unseen data.

4. Evaluation:

- Key metrics used: ***Accuracy***, ***Precision***, ***Recall***, and ***F1-Score***.
- Results highlighted a significant gap in performance for the Negative class due to class imbalance.

RESULTS

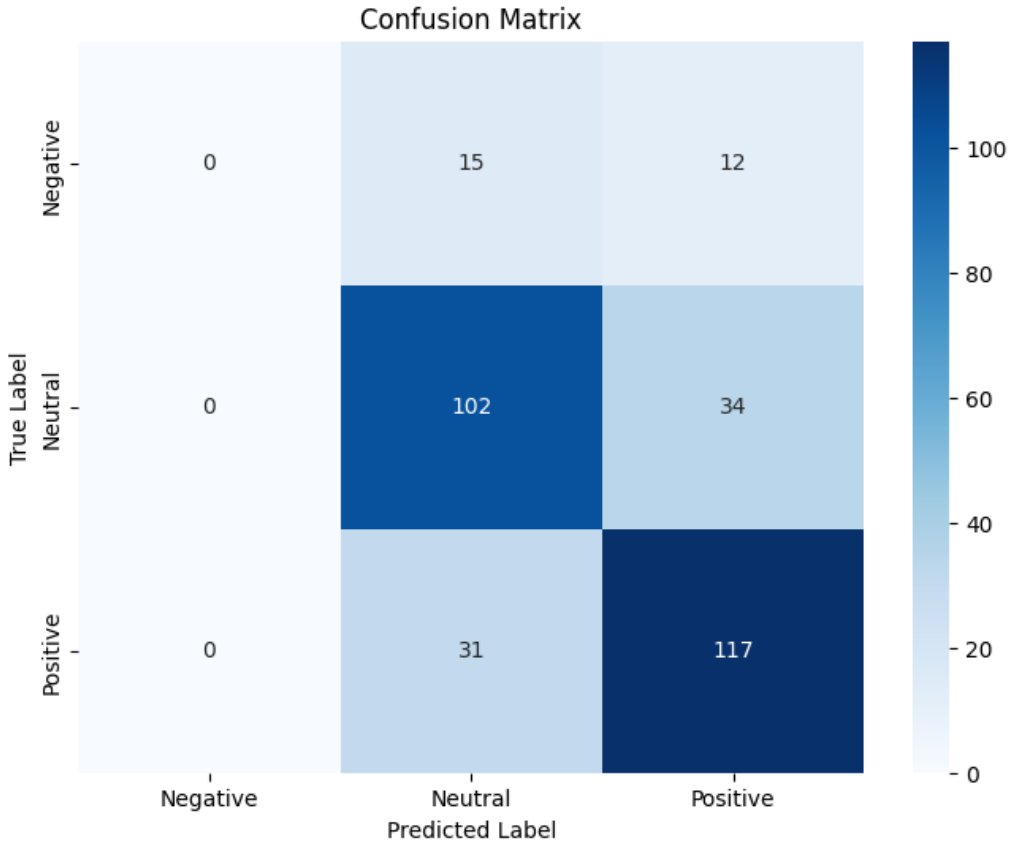
1. Model Performance:

- *Overall Accuracy:* 70.4%
- *Positive Sentiment:* Precision 72%, Recall 79%, F1-Score 75%
- *Neutral Sentiment:* Precision 69%, Recall 75%, F1-Score 72%
- *Negative Sentiment:* Precision, Recall, and F1-Score were 0, as no predictions were made for this class.

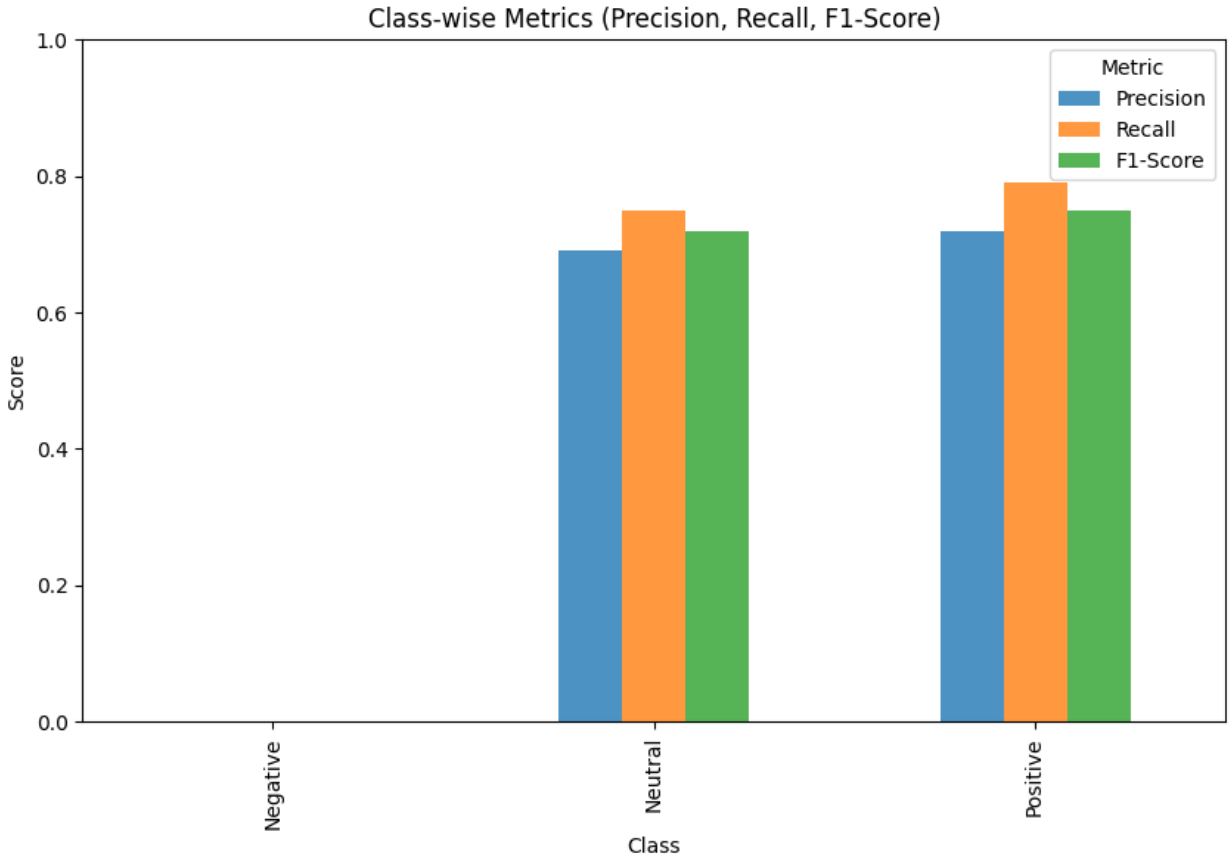
2. Insights from Confusion Matrix:

- Strong performance in predicting Neutral and Positive sentiments.
- Complete lack of predictions for Negative sentiments, reflecting the class imbalance.

3. Visual Insights:



(Fig 3.1:- A heatmap of the confusion matrix showed where the model performed well and where it failed.)



(Fig 3.2:- A bar chart of class-wise metrics revealed clear disparities in performance across sentiment categories.)

FINDINGS AND INSIGHTS

1. Public Sentiment Distribution:

- The majority of tweets about AstraZeneca are Positive or Neutral, suggesting favorable public perception.
- A smaller but significant Negative sentiment group highlights areas for concern that require attention.

2. Impact of Class Imbalance:

- The dataset's imbalance (only ~9% Negative tweets) negatively impacted the model's ability to predict this class.
- Balancing the dataset or adjusting model parameters will likely improve performance.

3. Modeling Effectiveness:

- Logistic Regression proved effective for Neutral and Positive predictions but failed for imbalanced classes.
- More advanced models, such as Random Forest or neural networks, might better handle such disparities.

4. Preprocessing Techniques:

- Simplified text preprocessing (e.g., removing stop words, lemmatization) was effective in preparing data for vectorization.
- Further domain-specific preprocessing could improve results.

5. Communication Strategy Insights:

- Positive public perception offers a strong foundation for AstraZeneca's messaging.
- Addressing concerns raised in Negative tweets could improve brand reputation and trust.

RECOMENDATIONS

1. Dataset Improvement:

- Address class imbalance using techniques like oversampling (e.g., SMOTE) or adjusting class weights.
- Regularly update the dataset with recent tweets to reflect current sentiments.

2. Advanced Modeling:

- Explore models like Random Forest, Support Vector Machines (SVM), or neural networks.
- Consider using domain-specific embeddings (e.g., Word2Vec, GloVe, or BERT) for better text understanding.

3. Actionable Communication Strategies:

- Monitor Negative tweets for recurring themes or issues and address them proactively.
- Leverage insights from Positive and Neutral tweets to reinforce successful communication strategies.

4. Future Studies:

- Incorporate sentiment trends over time to assess the impact of campaigns or events.
- Expand analysis to include other social media platforms for a more comprehensive view.

CONCLUSION

The sentiment analysis project successfully categorized tweets related to AstraZeneca into Positive, Neutral, and Negative sentiment categories, providing valuable insights into public perception. The analysis revealed that the majority of tweets were Positive (47.6%) or Neutral (43.6%), indicating an overall favorable or balanced perception of AstraZeneca. However, 8.8% of the tweets reflected Negative sentiments, which highlights areas where public concerns or dissatisfaction may exist and require attention.

The machine learning model, a Logistic Regression classifier, achieved an overall accuracy of **70.4%**, with strong performance in identifying Positive and Neutral sentiments. The model's precision, recall, and F1-scores for these categories were satisfactory, demonstrating its ability to classify these sentiments effectively. However, the model struggled to classify Negative sentiments, failing to make any predictions for this category. This shortfall was primarily due to the class imbalance in the dataset, with the Negative sentiment being significantly underrepresented.

The sentiment analysis also highlighted the importance of regularly updating datasets to include newer tweets, ensuring that the model captures evolving trends in public sentiment. Additionally, domain-specific language and context should be considered for better text understanding.

In conclusion, this analysis provides AstraZeneca with a strong foundation for understanding public sentiment and offers actionable insights to enhance its communication strategies. By addressing the highlighted challenges and incorporating the recommended improvements, AstraZeneca can further strengthen its engagement with the public, mitigate negative perceptions, and build greater trust with its audience.

APPENDICES

Appendix A:- Data Pre Processing Code

The following Python code was used to preprocess the text data:

```
import re


# Function to clean and preprocess text
def simplified_preprocess_text(text):
    # Remove URLs
    text = re.sub(r"http\S+|www\S+|https\S+", '', text,
flags=re.MULTILINE)
    # Remove punctuation, numbers, and special characters
    text = re.sub(r"[^a-zA-Z\s]", '', text)
    # Convert to lowercase
    text = text.lower()
    # Remove common stop words
    common_stop_words = ['the', 'and', 'is', 'to', 'in', 'for', 'on',
'at', 'a', 'an', 'of', 'with', 'by', 'this', 'it', 'as']
    words = text.split()
    words = [word for word in words if word not in common_stop_words]
    # Join back into a single string
    return ' '.join(words)


# Apply preprocessing
cleaned_data['Cleaned_Tweet'] =
cleaned_data['Tweet'].apply(simplified_preprocess_text)
```

Appendix B:- Model Training and Evaluation Code

The following Python code was used to train and evaluate the Logistic Regression model:

```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix

# Vectorize text data
tfidf_vectorizer = TfidfVectorizer(max_features=5000)
X = tfidf_vectorizer.fit_transform(cleaned_data['Cleaned_Tweet'])
y = cleaned_data['Target']

# Train-test split
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42, stratify=y)

# Train Logistic Regression
log_reg_model = LogisticRegression(max_iter=1000, random_state=42)
log_reg_model.fit(X_train, y_train)

# Evaluate the model
y_pred = log_reg_model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
classification_rep = classification_report(y_test, y_pred)
conf_matrix = confusion_matrix(y_test, y_pred)
```

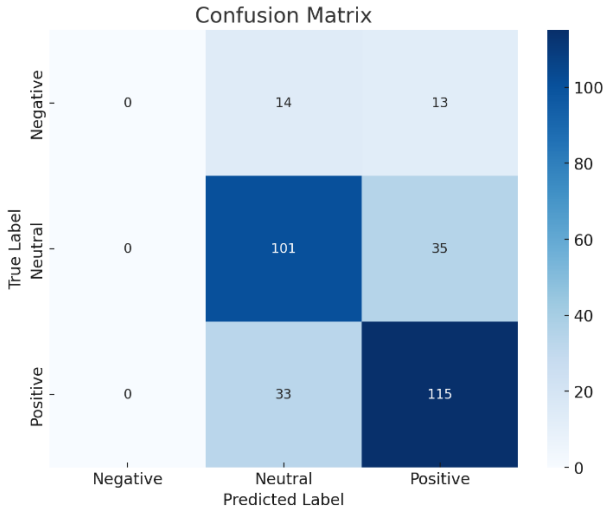
Appendix C:- Data Dictionary

Column Name	Description
id	Unique identifier for each tweet.
Tweet	The raw text of the tweet.
Subjectivity	A numerical score (0 to 1) indicating how subjective the tweet's content is.
Polarity	A numerical score (-1 to 1) indicating the sentiment polarity of the tweet.
Target	Sentiment label (Positive, Neutral, Negative) for the tweet.
Cleaned Target	Preprocessed version of the Tweet, cleaned for analysis.
Calculated Polarity	Sentiment polarity calculated using TextBlob during analysis.
Calculated Sentiment	Sentiment label derived from Calculated_Polarity during analysis.

Appendix C:- Visualizations

1. Confusion Matrix:-

- Highlights the model's performance by showing the true and predicted sentiment classifications.
- Demonstrates strong classification for Neutral and Positive sentiments, but no predictions for Negative sentiments due to class imbalance.



(Fig 9.1:- Visualization of actual vs. predicted sentiment labels to highlight the model's performance.)

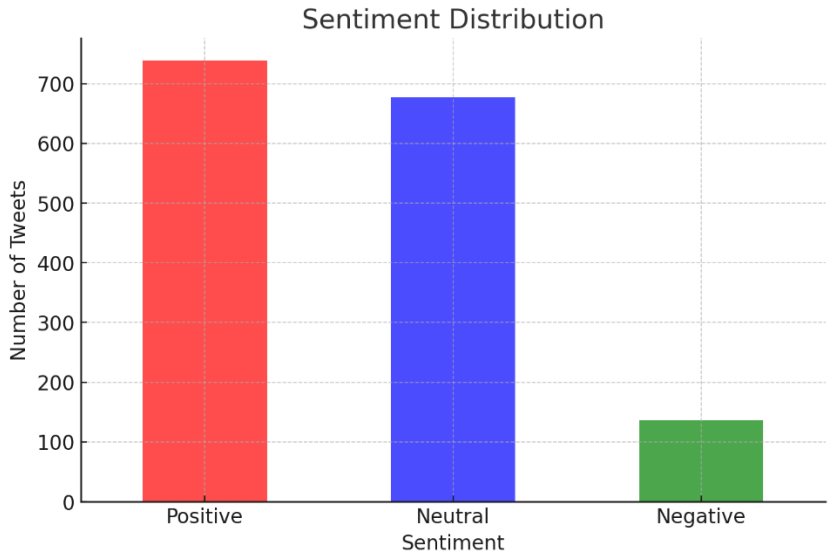
2. Class-wise Metrics Bar Chart:-

- Displays the Precision, Recall, and F1-Score for each sentiment class (Positive, Neutral, Negative).
- Clearly illustrates disparities in the performance across classes, with Negative sentiment having no measurable metrics.

(Fig 9.2:- Precision, Recall, and F1-Score metrics were visualized in a bar chart.)

3. Sentiment Distribution Bar Chart:

- Provides an overview of the distribution of sentiment labels in the dataset.
- Shows a dominance of Positive and Neutral tweets, with relatively few Negative tweets.



(Fig 9.3:- Showed strong performance for Positive and Neutral sentiments but no predictions for Negative.)