TWITTER SENTIMENT ANALYSIS

BY: Khushali Sheth
Rutva Gandhi
Anisha Sharma
Akshitha Maram
Varun Dubey

TABLE OF CONTENTS

KHUSHALI SHETH

INTRODUCTION

- In today's digital era, social media platforms like Twitter have become a wide area for getting the public sentiment on a wide range of topics.
- In this project the aim is to showcase a Twitter sentiment analysis using NLP, neural network model, Leveraging the Sentiment dataset.
- This analysis helps understand public opinion on various topics as expressed on Twitter our approach involves meticulous preprocessing of textual data, including the removal of URLs, mentions, and non-alphanumeric characters, followed by tokenization and sequence padding.This project aims to accurately classify tweets into positive or negative sentiments.

DATA OVERVIEW

Dataset Description

- 1.6 million tweets sourced from Twitter.
- Each tweet labeled as positive or negative based on emoticons.
- Tweets limits characters from 140-280, covering diverse topics.
- Used widely for training and evaluating sentiment analysis models.
- Valuable resource for studying social media sentiment trends.

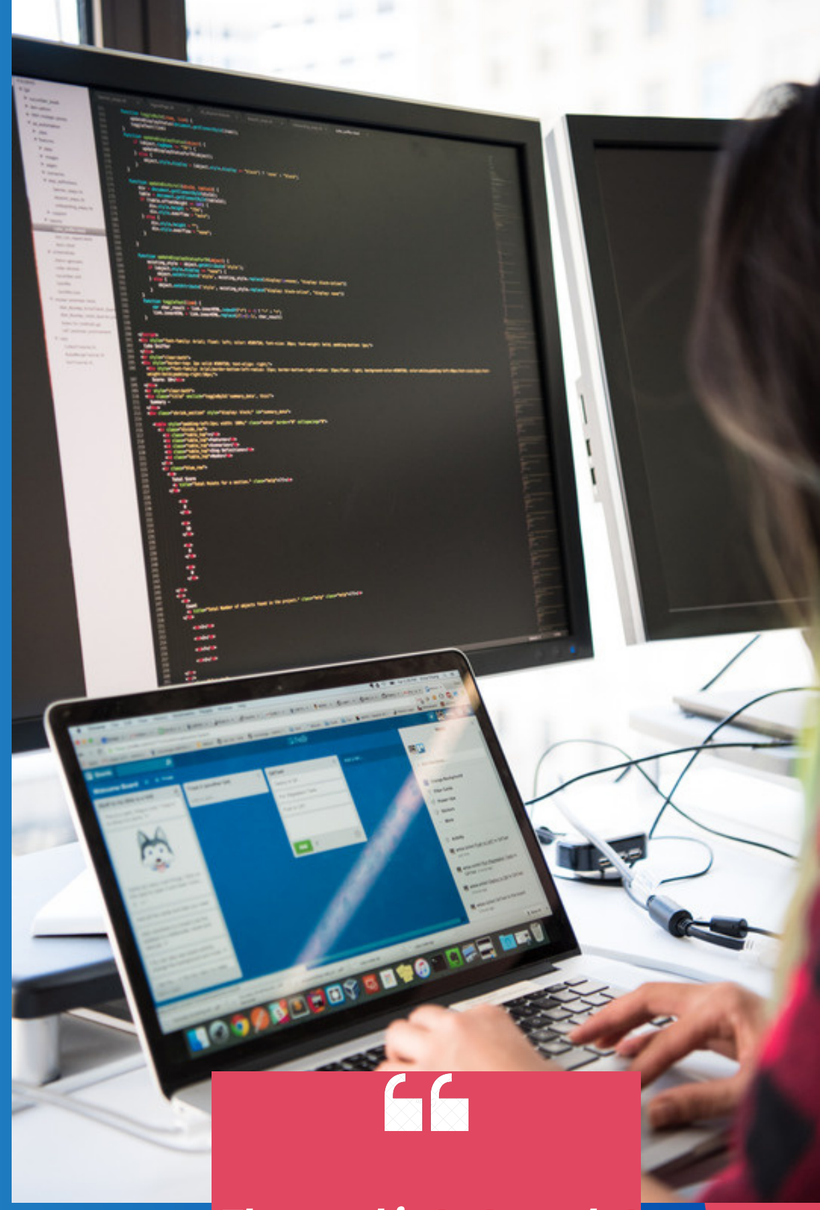
SOURCE: KAGGLE

KHUSHALI SHETH

METHODOLOGY - THEORETICAL FRAMEWORK

LSTM (Long Short-Term Memory)

- **Handling Long Sequences:** LSTMs mitigate gradient issues, handling lengthy sequences effectively.
- **Memory Cells:** Equipped with memory cells, LSTMs retain key information for sentiment analysis, excluding noise.
- **Gating Mechanisms:** LSTMs use gates to preserve sentiment-related data while discarding less relevant information.
- **Capturing Contextual Information:** LSTMs excel at understanding contextual nuances critical for sentiment analysis.
- **Learning Temporal Patterns:** LSTMs adeptly recognize complex temporal patterns, crucial for interpreting emotions in text with long-range dependencies.

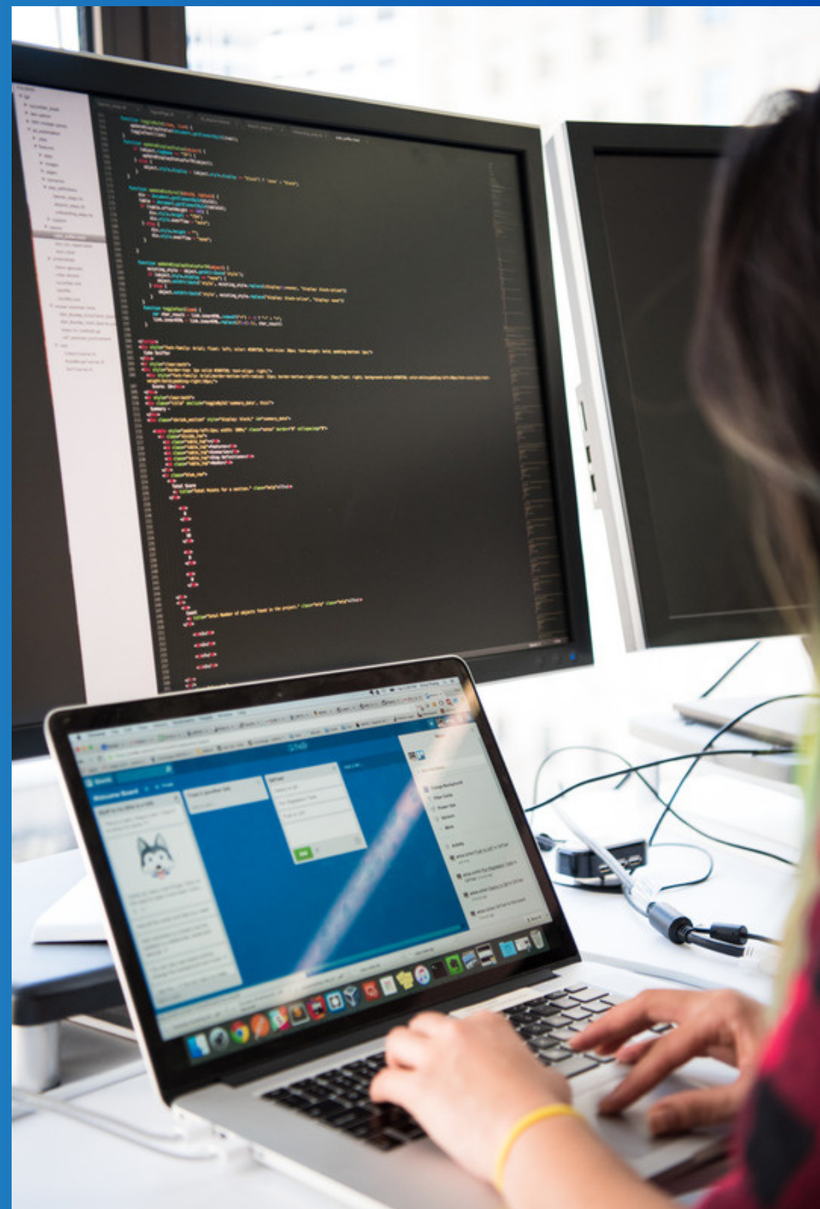


> ***The goal is to turn data into information, and information into insight.***

RUTVA GANDHI

Embedding Layers

- **Conversion to Numeric Vectors:** Embedding layers transform text into numerical vectors for machine learning.
- **Semantic Relationship Encoding:** They create dense vectors positioning similar words close together, capturing word meanings.
- **Dimension Reduction:** Mapping high-dimensional words into fixed-size vectors reduces computational complexity.
- **Contextual Comprehension:** Embeddings encode not just word meanings but also their contextual usage.
- **Transfer Learning Utility:** Pre-trained embeddings allow models to leverage general language semantics for specific tasks.
- **Neural Network Input:** Dense vectors from embeddings serve as inputs, preserving vital semantic information for NLP tasks.



	sentiment	id	date	query	user_id	text
0	0	1467810369	Mon Apr 06 22:19:45 PDT 2009	NO_QUERY	_TheSpecialOne_	@switchfoot http://twitpic.com/2y1zl - Awww, t...
1	0	1467810672	Mon Apr 06 22:19:49 PDT 2009	NO_QUERY	scotthamilton	is upset that he can't update his Facebook by ...
2	0	1467810917	Mon Apr 06 22:19:53 PDT 2009	NO_QUERY	mattycus	@Kenichan I dived many times for the ball. Man...
3	0	1467811184	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	ElleCTF	my whole body feels itchy and like its on fire
4	0	1467811193	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	Karoli	@nationwideclass no, it's not behaving at all....

DATA PREPARATION:

01 Text Preprocessing:

- **URL Removal:** The code uses a regular expression (regex) to remove URLs from the text. This is done using re.sub(r"http\S+|www\S+|https\S+", "", text, flags=re.MULTILINE), which targets strings starting with "http", "www", or "https" followed by any non-whitespace characters.
- **Mentions and Hashtags Removal**: The code also removes Twitter mentions (e.g., @username) and hashtags using another regex: re.sub(r"\@\w+|\#", "", text).
- **Non-Alphanumeric Character Removal:** The code employs re.sub(r"([A-Za-z0-9]+)", r"\1", text) to retain alphanumeric characters, effectively removing non-alphanumeric characters from the text.

DATA PREPARATION:

02 Tokenization and Padding:

- **Tokenization:** The code uses the Tokenizer class from Keras, specifying a maximum vocabulary size of 5000 words (max_words = 5000). It tokenizes the text, converting each unique word into a specific integer. The Tokenizer is fitted on the training data.
- **Conversion to Sequences**: The texts (tweets) are then converted into sequences of integers using texts_to_sequences().
- **Padding:** The sequences are padded to a fixed length (max_length = 30) using pad_sequences(). This ensures that all input sequences to the neural network have the same length. Padding is done post the actual sequence with zeros if the sequence is shorter than the specified max_length.

ANISHA SHARMA

ANALYSIS AND MODEL TRAINING

Natural Language Processing (NLP): NLP involves the ability of a computer to understand, interpret, and generate human-like text. In sentiment analysis, NLP techniques are used to analyze the content of tweets and extract meaningful information.

Sentiment Classification: Tweets are typically classified into three main categories: positive, negative, or neutral. Some more advanced sentiment analysis models may use a scale, allowing for more nuanced sentiment classification.

Text Classification: It is a process involved in Sentiment Analysis. It is classification of peoples opinion or expressions into different sentiments. Sentiments include Positive, Neutral, and Negative, Review Ratings and Happy, Sad. Sentiment Analysis can be done on different consumer centered industries to analyse people's opinion on a particular product or subject.

ANALYSIS AND MODEL TRAINING

Machine Learning: Machine learning algorithms are trained on labeled datasets to recognize patterns and associations between words, phrases, and sentiments. These algorithms can then be applied to analyze new, unseen tweets and predict their sentiment.

Recurent Neural Networks can handle a seqence of data and learn a pattern of input sequence to give either sequence or scalar value as output. In our case, the Neural Network outputs a scalar value prediction.

LSTM - Long Short Term Memory, its a variant of RNN which has memory state cell to learn the context of words which are at further along the text to carry contextual meaning rather than just neighbouring words as in case of RNN.

ANALYSIS AND MODEL TRAINING

Performance Metrics:

- **Accuracy and Loss**: The model uses accuracy and loss as performance metrics. Accuracy measures the proportion of correctly predicted observations to the total observations, which is straightforward and useful for classification problems. Loss, particularly binary cross-entropy loss, quantifies how far off predictions are from the actual results, providing a more nuanced view of model performance.

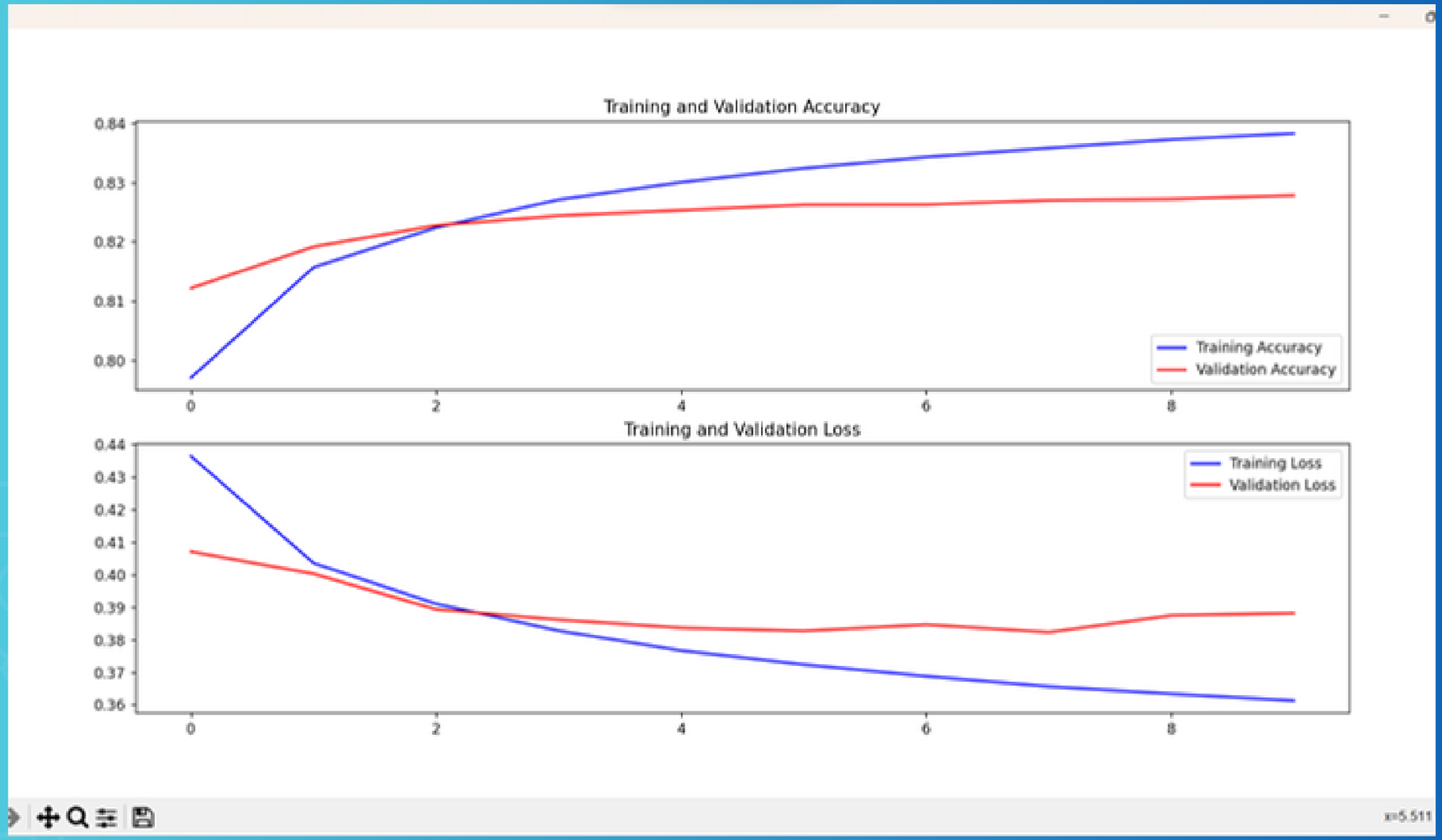
ANALYSIS AND MODEL TRAINING

Results Interpretation:

- The results, as observed in the learning curves, would show how the model's accuracy and loss evolved over training cycles for both training and validation sets. An ideal outcome would be increasing accuracy and decreasing loss over time.
- Training and validation accuracy trends can indicate if the model is learning effectively, while the trends in loss can point out overfitting (if the validation loss starts increasing after a point).
- The final evaluation of the model on the test dataset gives an overall idea of how well the model can generalize to new, unseen data.

DATA OVERVIEW AND VIZUALIZATION

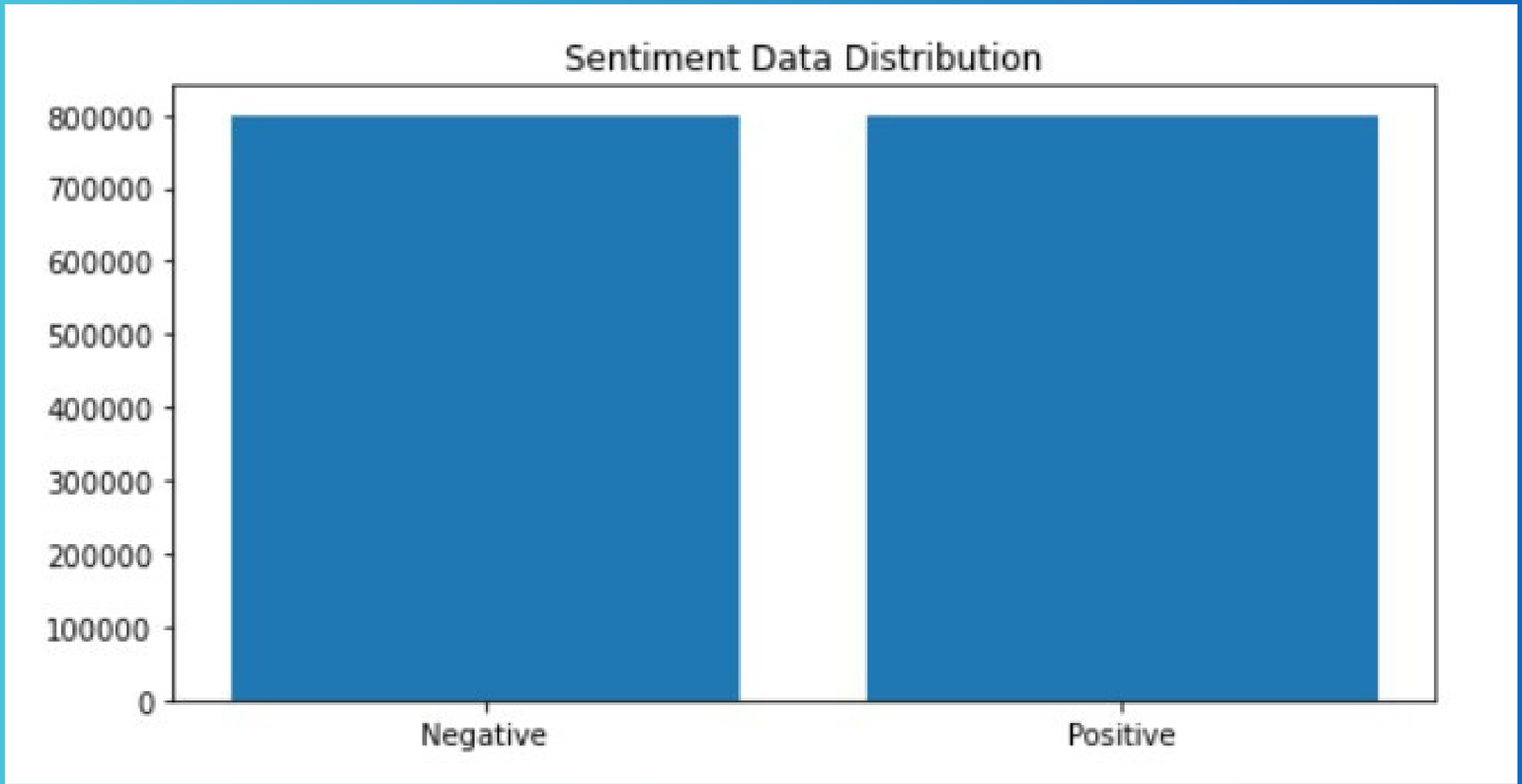
AKSHITHA MARAM



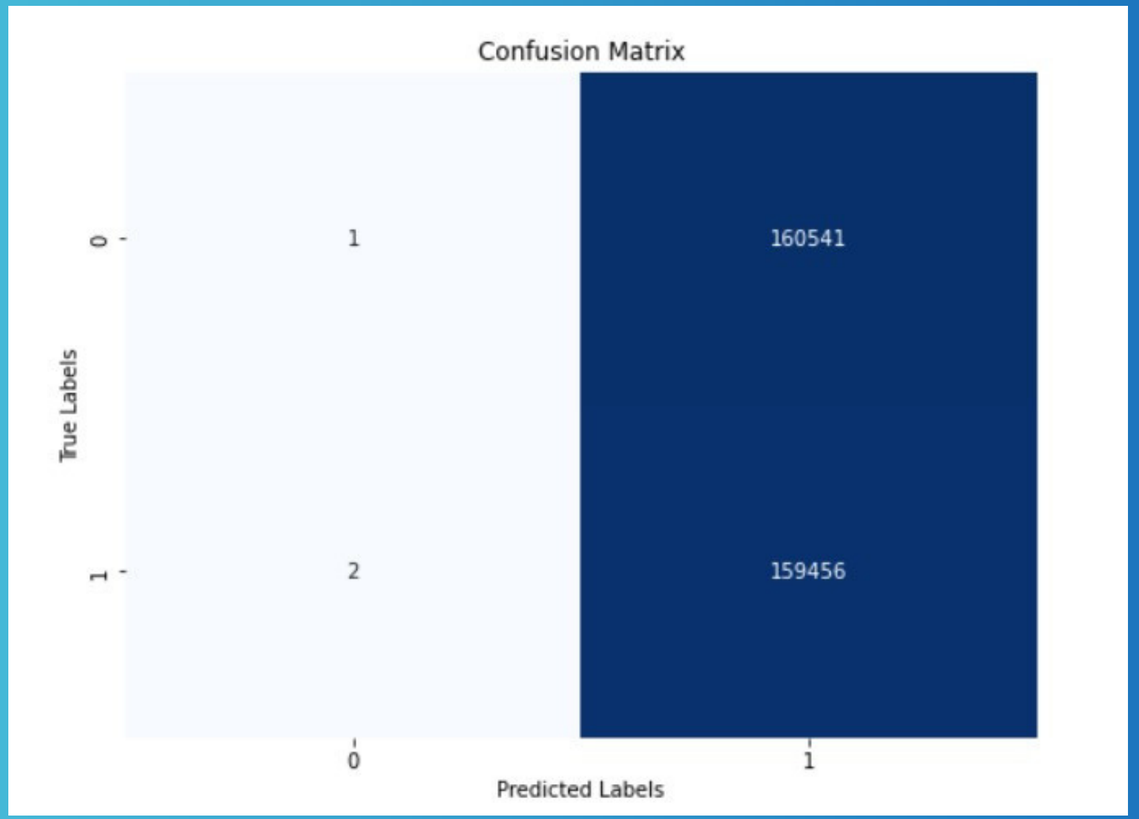
DATA OVERVIEW AND VIZUALIZATION

```
y: 0.5019
Epoch 1/5 - Train Accuracy: 0.5007, Train Loss: 27.0743, Validation Accuracy: 0.4983, Validation Loss: 0.7549
Epoch 2/5 - Train Accuracy: 0.4993, Train Loss: 0.8157, Validation Accuracy: 0.5017, Validation Loss: 0.7189
Epoch 3/5 - Train Accuracy: 0.5011, Train Loss: 0.8341, Validation Accuracy: 0.4983, Validation Loss: 0.7089
Epoch 4/5 - Train Accuracy: 0.5003, Train Loss: 0.7611, Validation Accuracy: 0.4983, Validation Loss: 0.7136
Epoch 5/5 - Train Accuracy: 0.4999, Train Loss: 0.7298, Validation Accuracy: 0.5017, Validation Loss: 0.7078
10000/10000 [==============================] - 10s 1ms/step
Test Set Metrics:
Accuracy: 50.1697 %
Confusion Matrix:
 [[160539      3]
 [159454      4]]
Classification Report:
              precision    recall  f1-score   support

           0       0.50      1.00      0.67    160542
           1       0.57      0.00      0.00    159458

    accuracy                           0.50    320000
   macro avg       0.54      0.50      0.33    320000
weighted avg       0.54      0.50      0.34    320000
```

DATA OVERVIEW AND VIZUALIZATION



DATA OVERVIEW AND VIZUALIZATION



VARUN DUBEY

CONCLUSION

In conclusion, the culmination of our Twitter Sentiment Analysis project highlights the pivotal role of sentiment analysis in comprehending and interpreting public sentiments and opinions expressed on the Twitter platform. Through the utilization of advanced Natural Language Processing (NLP) techniques and leveraging the Sentiment140 dataset comprising 1.6 million labeled tweets, our project has successfully contributed significant insights into the understanding of so sentiments.

THANK YOU