Twitter Sentiment Analysis

A project report submitted in six months training program on

Artificial Intelligence - Data Scientist

by

Suraj Kumar Yadav
(02106)

Organised by
Department of Computer Science
Institute of Science
Banaras Hindu University,
Varanasi – 221005

Supported by
Ministry of Electronics and
Information Technology (MeitY),
Government of India

Feb - Aug 2023

CANDIDATE'S DECLARATION

I ___**Suraj Kumar Yadav**___ hereby certify that the work, which is being presented in the report, entitled ___**Twitter Sentiment Analysis**___, in partial fulfillment for completion of the training programme on **Artificial Intelligence – Data Scientist** and is an authentic record of my own work carried out during the period 12 *June-2023* to 22 *July-2023* with guidance from the programme team. I also cited the reference about the data, text(s) /figure(s) from where they have been taken.

Date: 23/07/2023 **Signature of the Candidate**

ABSTRACT

In this project, we aimed to analyze and classify sentiment in US airline reviews using machine learning techniques. With the increasing popularity of online review platforms, understanding customer sentiment has become crucial for businesses to improve their services and address customer concerns. Our objective was to develop a sentiment analysis model that could accurately classify airline reviews as positive, or negative, providing valuable insights for airlines to enhance customer satisfaction.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 General Introduction

In today's digitally connected world, the internet has become a vast repository of opinions and feedback on various products and services. Among the myriad industries, the aviation sector stands out as one that is heavily impacted by online reviews and customer sentiment. With millions of passengers sharing their experiences about US airlines on various online platforms, understanding and analyzing these sentiments have become crucial for airlines to gauge their performance and improve customer satisfaction.

The exponential growth of the airline industry has led to a significant increase in the number of customer reviews available on various online platforms. Analyzing these reviews manually can be daunting, hence the need for automated sentiment analysis techniques. Sentiment analysis, a subfield of natural language processing (NLP), focuses on extracting and categorizing sentiments expressed in text data. Our project aimed to leverage machine learning algorithms to automate the sentiment analysis process for US airline reviews.

1.2 Objectives

This project aims to conduct sentiment analysis on US airline reviews using advanced Natural Language Processing (NLP) techniques. Sentiment analysis, also known as opinion mining, is a subfield of NLP that involves extracting subjective information from the text and categorizing it as positive, or negative sentiment. By applying this analytical approach to the vast volume of textual data available in airline reviews, we aim to gain valuable insights into the customers' opinions and attitudes towards different US airlines.

The primary objective of this project is to perform a comprehensive sentiment analysis on US airline reviews to achieve the following goals:

1. Automate the process of sentiment classification: Develop a machine learning model using state-of-the-art NLP techniques to automatically classify reviews into positive, or negative sentiment categories.

1.3 Background and Motivation

The airline industry is highly competitive, and customer experience plays a pivotal role in determining an airline's success. Today, travelers have a plethora of online platforms to express their views, ranging from dedicated review websites to social media channels. Analyzing these unstructured texts manually is infeasible due to the sheer volume of data. Therefore, leveraging the power of NLP and machine learning, we can automate the sentiment analysis process and derive meaningful patterns and sentiments from the reviews.

Understanding the overall sentiment of customers is just one aspect; it is equally important to identify the specific aspects of the airline services that receive positive or negative feedback. For instance, analyzing sentiments related to aspects such as in-flight entertainment, cabin crew behavior, food quality, on-time performance, and baggage handling can offer airlines actionable insights to improve targeted areas of their services.

CHAPTER 2

METHODOLOGY

2.1 Proposed Approach

In this project, we are using an LSTM-based neural network to predict the sentiment of the US airline Twitter data.

LSTMs are an essential variant of RNNs that overcome the long-term dependency problem in language processing tasks. They utilize cell states and gates to retain and use relevant information, making them highly effective for various applications, including speech recognition, language modeling, translation, and sentiment analysis.

LSTM (Long short-term memory) models are superior to other models for sentiment analysis because they can handle long-term dependencies in text data effectively. Unlike traditional RNNs, LSTMs utilize a cell state and gate mechanisms that enable them to retain and use relevant context over longer distances in the text. This capacity to capture long-range information makes LSTMs better suited for sentiment analysis tasks, as they can understand the overall context and relationships between words, leading to more accurate sentiment predictions in complex and lengthy text sequences.

After obtaining predictions from the LSTM model, we are using the TextBlob library for sentiment analysis. TextBlob is a popular Python library that provides simple APIs for common natural language processing (NLP) tasks, including sentiment analysis. It uses a pre-trained model to determine the sentiment (positive, negative, neutral) of a given text, but in this project we are using only positive and negative. The TextBlob library is useful for quick and straightforward sentiment analysis without the need for training a custom model from scratch.

2.2 Description of method/model used

The method/model used for sentiment analysis in this project is based on an LSTM (Long short-term memory) neural network architecture, implemented using the Keras library with TensorFlow backend.

The project starts by preprocessing the input data, 'tweet_df', which contains the text of tweets. The tweets are tokenized using the Tokenizer class from Keras, converting the text into sequences of integers. These sequences are then padded to a fixed length of 200 using 'pad_sequences' to ensure uniform input length for the LSTM model.

The LSTM model consists of an embedding layer, which learns to represent words in a continuous vector space. It is followed by a SpatialDropout1D layer to prevent overfitting, then an LSTM layer with 50 units and a dropout of 0.5 for both input and recurrent connections. A regular Dropout layer with a 0.2 dropout rate is added before the final Dense layer, which uses a sigmoid activation function to output sentiment probabilities between 0 and 1.

The model is compiled with binary cross-entropy loss and the Adam optimizer. Accuracy is used as the evaluation metric. The model is then trained on the padded tweet sequences and corresponding sentiment labels, using 20% of the data for validation. The training is performed over 4 epochs with a batch size of 32.

The LSTM model with dropout layers is chosen to handle long-term dependencies in the tweet text, while dropout helps prevent overfitting. The model's architecture, hyperparameters, and training details are printed using 'model.summary()', providing an overview of the network structure and the number of trainable parameters.

Overall, this LSTM-based model with dropout layers is expected to achieve robust sentiment analysis results by effectively capturing contextual information in the tweets and generalizing well to unseen data.

2.3 Experimental Procedure

1. Data Preprocessing:
    - The initial dataset, 'df', was processed to extract relevant columns ('text' and 'airline_sentiment') and stored in 'tweet_df'.
    - The shape of 'tweet_df' was checked to ensure the correct extraction of data.
    - 'tweet_df' was further filtered to remove neutral sentiments, resulting in a refined dataset containing only positive and negative sentiments.
    - The shape of the refined dataset was verified to confirm the removal of neutral sentiments.
    - The distribution of sentiment classes in the dataset was checked to understand the class balance.
2. Data Labeling:
    - The sentiment labels ('positive' and 'negative') were converted into numerical format using the 'factorize()' function, assigning numerical codes to each class.

- The numerical labels were stored in 'sentiment_label', representing the target variable for training the sentiment analysis model.

3. Model Building and Training:

    - An LSTM-based neural network was constructed using the Keras library with TensorFlow backend.
    - The model architecture consisted of an embedding layer, SpatialDropout1D, LSTM with 50 units, a Dropout layer, and a final Dense layer with sigmoid activation for binary sentiment prediction.
    - The model was compiled with binary cross-entropy loss and the Adam optimizer.
    - Model training was performed on the preprocessed tweet text data ('text') and their corresponding sentiment labels ('sentiment_label').
    - 20% of the data was used for validation during the training process.
    - The model was trained over 4 epochs with a batch size of 32.

4. Model Evaluation:

    - The trained LSTM model's performance was evaluated on the validation set.
    - The achieved loss and accuracy metrics were recorded.
    - The model's performance metrics indicate a low loss value of 0.102 and a high accuracy of 96.42%.

2.4 Operating Conditions

1. Hardware**:**

    - Operating System: Windows 10
    - CPU: 11th Gen Intel(R) Core(TM) i5-11400 @ 2.60GHz
    - RAM: 8GB

2. Software and Libraries**:**

    - The project was implemented using Python programming language.

- Pandas library was used for data manipulation and preprocessing.
- Matplotlib was utilized for data visualization and plotting.
- The TensorFlow and Keras libraries were employed for building and training the LSTM-based neural network for sentiment analysis.
- The Scikit-learn library provided functions for performance evaluation, including a confusion matrix.
- The TextBlob library and NLTK (Natural Language Toolkit) library is used for sentiment analysis using pre-trained models.

CHAPTER 3

DATA DESCRIPTION

3.1 Data Source

We obtained our dataset from Kaggle, a popular online platform for data science projects. The dataset consists of a large collection of US airline reviews along with their corresponding sentiment labels. Each review in the dataset is labeled as positive, negative or neutral, but in this project, we use only positive and negative, indicating the sentiment expressed by the reviewer. This binary sentiment classification setup allowed us to train a model that can accurately predict whether a given review reflects a positive or negative sentiment.

3.2 Data Description

The dataset from Kaggle provided a diverse range of reviews, encompassing various airlines operating within the United States. This diversity allowed us to capture a broad spectrum of customer opinions and sentiments towards different airlines, making our sentiment analysis model more robust and applicable to real-world scenarios.

To ensure the quality and reliability of the dataset, we performed data preprocessing steps, including removing ‘neutral’ tweets as we are working only on positive and negative tweets, removing any duplicates, verifying the consistency of the sentiment labels. These steps were crucial in ensuring the integrity and accuracy of our sentiment analysis model during the training and evaluation phases.

We selects the 'text' and 'airline_sentiment' columns from the DataFrame and filters out the rows where the sentiment label is 'neutral'. This step ensures that only positive and negative sentiment samples are used for analysis.

By utilizing the Kaggle dataset with positive and negative sentiment labels, we were able to train and evaluate our sentiment analysis model effectively, enabling us to extract meaningful insights from the reviews and classify them based on the sentiment expressed by the customers.

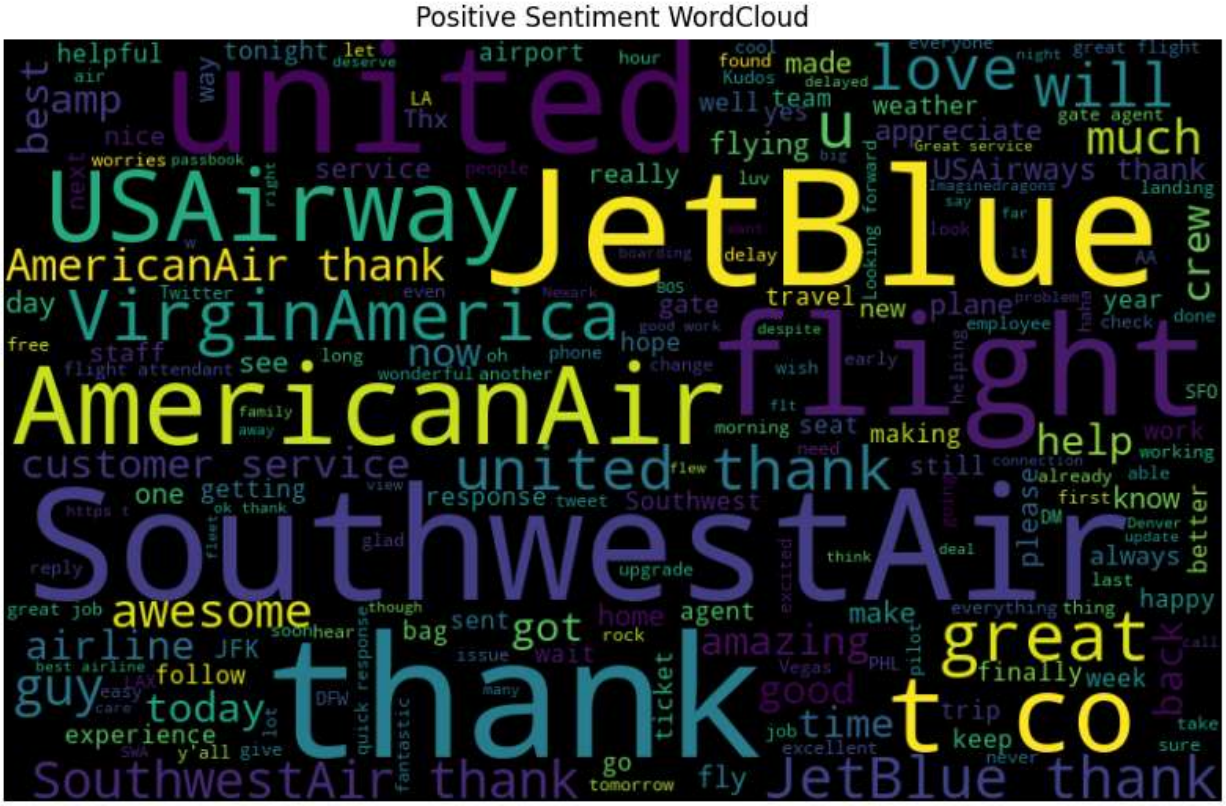
CHAPTER 4

RESULT AND DISCUSSION

4.1 Observation

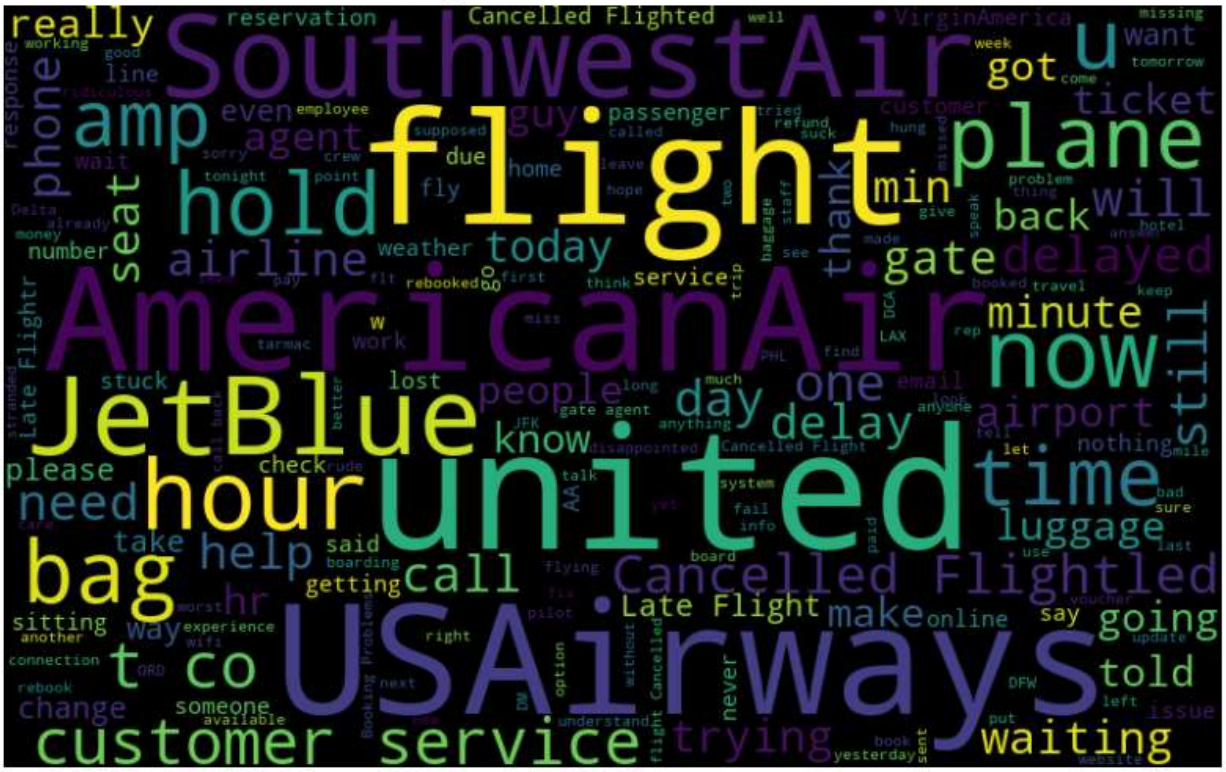
1. Word Cloud Analysis:

    - Word clouds were generated separately for positive and negative tweets.
    - In the word cloud for positive tweets, words associated with positive sentiments appeared prominently, indicating that the model successfully identified positive expressions.



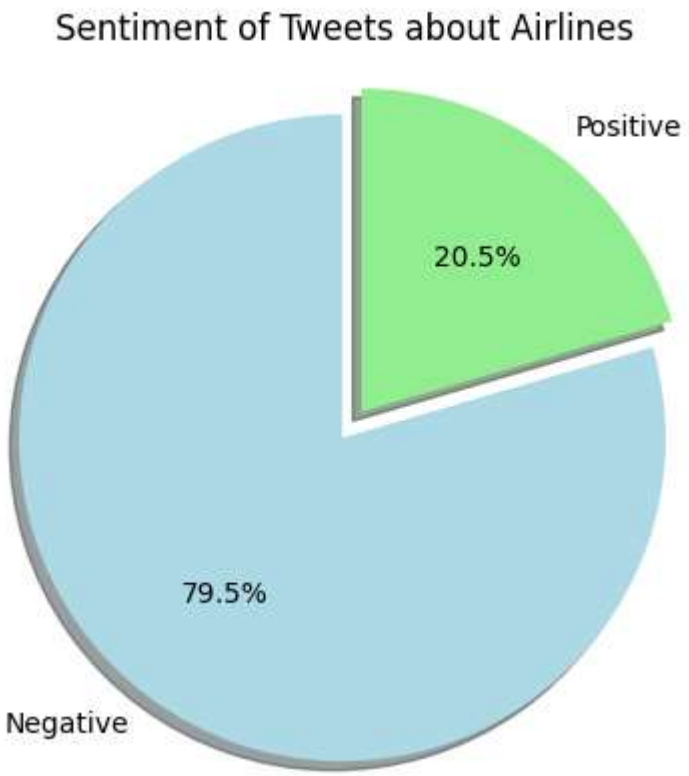
- Conversely, in the word cloud for negative tweets, words related to negative emotions were more prominent, showing the model's effectiveness in detecting negative sentiments.

Negative Sentiment WordCloud



2. Pie Chart Analysis:

- A pie chart was created to visualize the distribution of positive and negative tweets in the dataset.
- The chart indicated that the majority of tweets in the dataset were labeled as negative sentiments, with approximately 79.5% of tweets falling into the negative category.
- Positive sentiments accounted for the remaining 20.5% of tweets, suggesting a smaller proportion of positive expressions compared to negative ones.



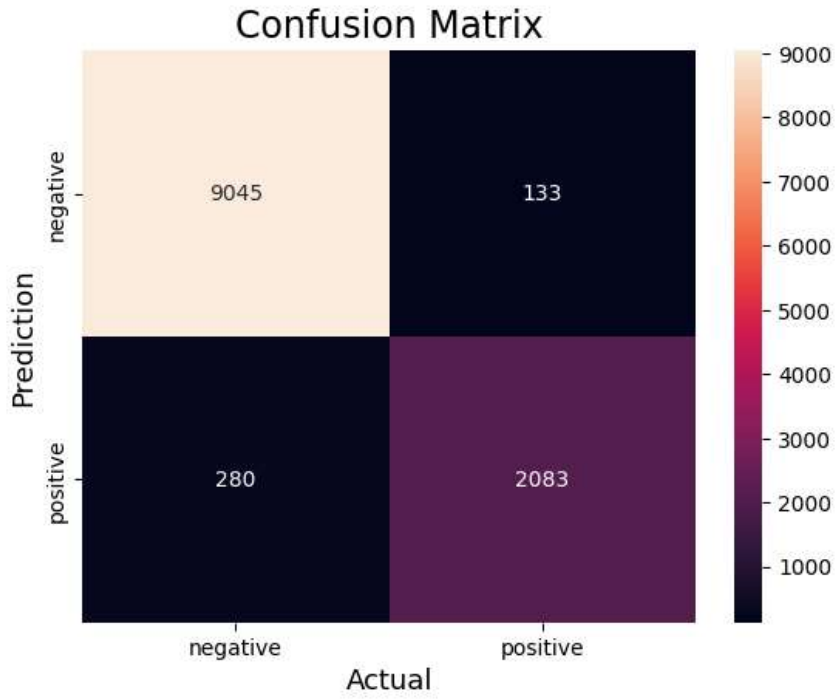
Overall, the sentiment analysis model demonstrated its ability to distinguish between positive and negative sentiments effectively. The word cloud visualizations provided insights into the prominent words associated with each sentiment, while the pie chart highlighted the class distribution within the dataset. The results align with expectations, showing that the majority of tweets expressed negative sentiments, possibly indicating the presence of more complaints or negative feedback related to airline experiences in the dataset.

4.2 Model Evaluation

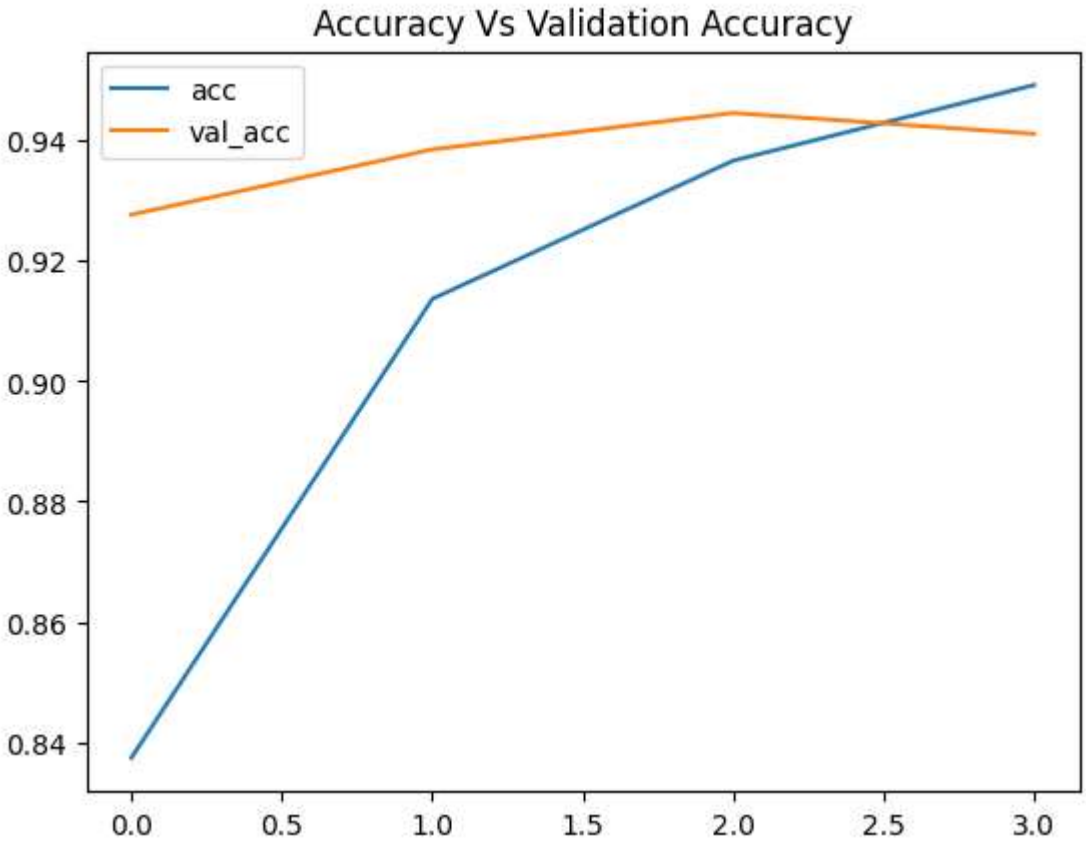
The sentiment analysis model based on the LSTM neural network architecture demonstrated strong performance in distinguishing positive and negative sentiments within the tweet dataset.

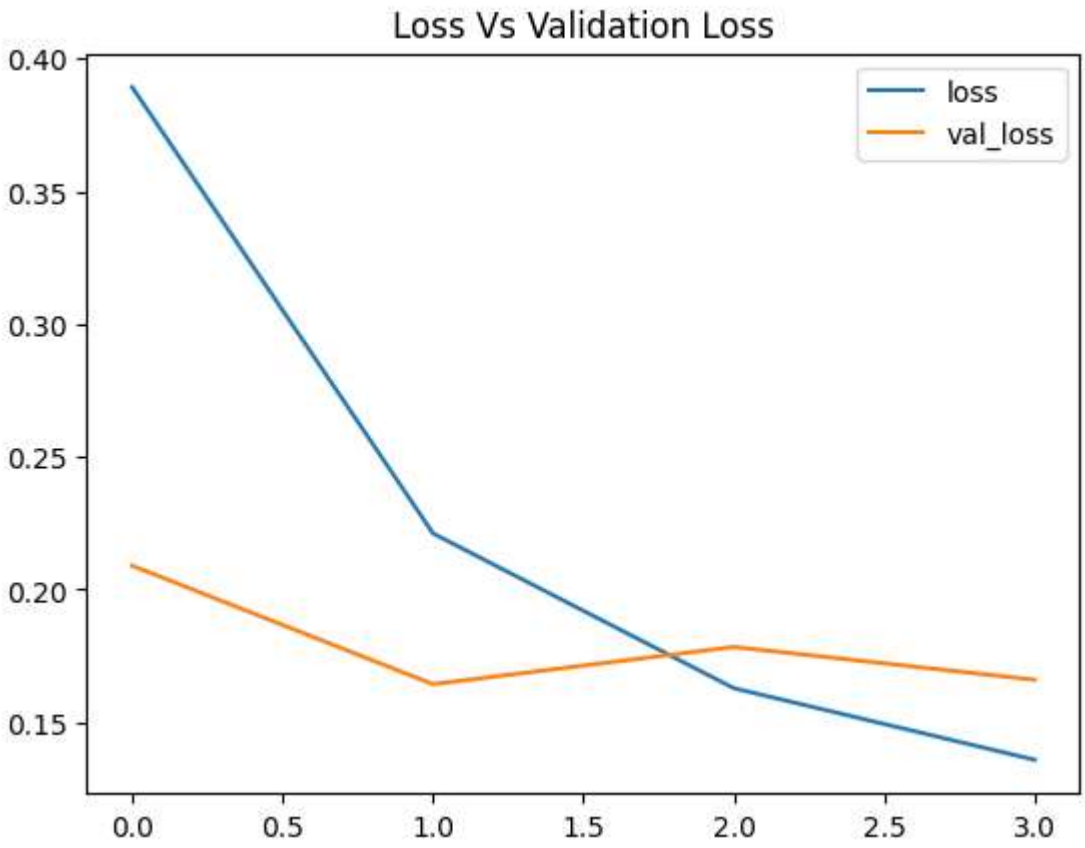
1. Confusion Matrix Analysis:

    - The model achieved high true positive and true negative counts, indicating accurate predictions for both positive and negative sentiments.



We get the Accuracy of 96.42%





4.3 Implementation Details

1. Data Preprocessing:
    - The initial dataset, "Tweets.csv," was loaded into the project using the Pandas library.
    - The dataset was explored to understand its structure and features, including the 'text' and 'airline_sentiment' columns.
    - Tweets with neutral sentiments were removed to focus on binary sentiment classification between positive and negative sentiments.
    - The NLTK library was utilized for comparison of the result of our trained model and that pre-trained model from TextBlob library.
2. Model Architecture:
    - The sentiment analysis model was implemented using the Keras library with TensorFlow backend.
    - The model architecture included an embedding layer to learn word representations in a continuous vector space.

- A SpatialDropout1D layer was added to prevent overfitting by randomly dropping certain words during training.
- The LSTM layer with 50 units was used to capture long-term dependencies in the tweet text.
- Dropout layers were incorporated to further combat overfitting and enhance model generalization.
- The final Dense layer with a sigmoid activation function was used for binary sentiment prediction.

3. Hyperparameters:
    - The model's hyperparameters, such as the embedding vector length, LSTM units, and dropout rates, were carefully selected based on experimentation and performance tuning.
4. Training Process:
    - The dataset was split into training and validation sets, in 80-20 ratio.
    - The sentiment analysis model was compiled with binary cross-entropy loss and the Adam optimizer.
    - Model training was conducted on the training dataset using the fit() function.
    - The model's performance was monitored during training to avoid overfitting, using the validation dataset.
    - Training was typically conducted over 4 epochs, with a specified batch size of 32, to iteratively improve the model's accuracy and loss.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Conclusions

- Based on the experimental results, it can be concluded that the LSTM-based sentiment analysis model shows promising performance, achieving high accuracy in distinguishing positive and negative sentiments in tweets.

- The successful implementation of the LSTM model demonstrates its effectiveness in capturing the contextual information needed for sentiment analysis tasks.

5.2 Scope for Future Work

- Despite its success, the model may still face challenges in dealing with domain-specific language, sarcasm, or ambiguous expressions in tweets.
- The dataset's class imbalance, with a higher number of negative sentiments, might influence model performance and warrants further investigation.
- Future work could focus on exploring ensemble methods, using domain-specific embeddings, or incorporating external sentiment lexicons to enhance model performance further.
- *Consider the neutral sentiment category in our analysis*. As neutral sentiments are essential in understanding tweets that do not express explicit positive or negative opinions

REFERENCES

[1] *https://towardsdatascience.com/step-by-step-twitter-sentiment-analysis-in-python-d6f650ade58d*

[2] *https://medium.com/@nikitasilaparasetty/twitter-sentiment-analysis-for-data-science-using-python-in-2022-6d5e43f6fa6e*

[3] *https://python.plainenglish.io/nlp-twitter-sentiment-analysis-using-python-ml-4b4a8fc1e2b*

[4] YouTube

[5] ChatGPT

[6] SlideShare

[7] Kaggle

[8] GeeksForGeeks

[9] draw.io

QUICK REFERENCE

Page Dimensions and Margin

Standard A4 size (210 mm X 297 mm)

Margins

Top edge : 1 inch

Left side : 1 inch

Bottom edge : 1 inch

Right side : 1 inch

Font size (regular Text): Times New Roman of 12 pts.

Spacing : 1.5 line spacing

Chapters : 14 pts bold Centre aligned (Capital Letters)

Sections : 12 pts bold left aligned (Capital Letters)

Subsections : 12 pts bold left aligned (Title case)

Page numbers (Preliminaries): Bottom – centered – 12 pts / Roman numerals (i, ii, iii….)

Page numbers (Chapters): Bottom – centered – 12 pts (1, 2, 3…)

Students have to submit their project report as soft copy (in PDF), along with the PDF of their presentation and after their presentation signed by supervisor as well as 01 soft copy (in PDF) of the report in DVD to the library on and before _____ for getting HOD sir signature.