

**VISVESVARAYA TECHNOLOGICAL UNIVERSITY**  
**BELAGAVI, KARNATAKA-590 018**



**INTERNSHIP REPORT**  
**ON**

**Machine Learning With Python**

*Submitted in partial fulfilment of the requirements for the **Internship (21INT68)** course of the 6<sup>th</sup> semester.*

**BACHELOR OF ENGINEERING**  
**IN**  
**COMPUTER SCIENCE AND ENGINEERING**

**By**

**Yogesh C [1JS21CS181]**

**Under the guidance of**

**Dr. Supriya B N**

Assistant Professor, CSE Department



**JSS ACADEMY OF TECHNICAL EDUCATION, BENGALURU**  
**Department of Computer Science and Engineering**  
2023 – 2024

JSS MAHAVIDYAPEETHA, MYSURU

# JSS Academy of Technical Education

JSS Campus, Uttarahalli, Kengeri Main Road, Bengaluru – 560060

## Department of Computer Science and Engineering



### CERTIFICATE

This is to certify that the internship entitled “**Data Analytics Using Python**” is a benefited work carried out by **Yogesh C** bearing USN **1JS21CS181** bonafide student of **JSS Academy of Technical Education** in the partial fulfillment for the award of the **Bachelor of Engineering in Computer Science & Engineering** of the **Visvesvaraya Technological University**, Belgaum, during the year 2022-23. It is certified that all corrections / suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The internship report has been approved as it satisfies the academic requirements in respect of internship work prescribed for the said degree.

**Dr. Supriya B N**  
Assistant Professor  
CS&E Department  
JSSATE, Bengaluru

**Dr. P B Mallikarjuna**  
Associate Prof & HOD,  
CS&E Department  
JSSATE, Bengaluru

## ACKNOWLEDGEMENTS

I express our humble pranamams to his holiness Jagadguru **Sri Sri Sri Shivarathri Deshikendra Mahaswamiji** who has showered their blessings on us for framing our career successfully.

I have been lucky enough to have received a lot of help and support from all quarters during the making of this project, so with gratitude, I take this opportunity to acknowledge all those whose guidance and encouragement helped us emerge successful.

I am thankful to the resourceful guidance, timely assistance and graceful gesture of our guide **Dr. Supriya B N**, Assistant Professor, Department of Computer Science and Engineering who has helped me in every aspect of our internship work.

I am also indebted to **Dr. P B Mallikarjuna**, Head of the Department of Computer Science and Engineering for the facilities and support extended towards us.

I express my sincere thanks to our beloved principal, **Dr. Bhimasen Soragoan** for having supported me in my academic endeavors. And last but not least, I would be very pleased to express my heart full thanks to all the teaching and non-teaching staff of CSE department and my friends who have rendered their help, motivation and support.

Yogesh C  
1JS21CS181

## ***COMPANY PROFILE***



**Compsoft Technologies**

Providing a Complete Suite of IT Solutions

Established in 2013, Compsoft Technologies provides a complete suite of IT services focused on digital transformation. The company excels in developing event-driven, real-time applications across various industries, ensuring innovative and efficient production operations.

They are dedicated to optimizing client satisfaction through quality services and the best technological solutions, aligning with industry best practices.

### **Company ambitions :**

Compsoft Technologies aims to drive client success by delivering innovative real-time applications and efficient technology solutions..

### **TALENT**

Our team comprises experts proficient in modern programming languages, AI, machine learning, data analytics, and cloud computing.

### **TECHNOLOGY**

We utilize cutting-edge technologies, including cloud services and advanced AI, to deliver scalable and innovative real-time applications.

### **TRANSFORMATION**

We bring energy and speed to everything we do and everyone we serve. We power our business and our customers towards a new digital world.

## **Portfolio of services**

We offer consulting, custom application development, cloud solutions, data analytics, and AI/ML implementation to optimize business processes. Our services also include IT support, maintenance, and operational performance improvement.

### **Business:**

They operates across multiple business lines, including consulting and strategy to optimize business processes. Our expertise extends to custom application development tailored to diverse industry needs, robust cloud solutions, and advanced data analytics and business intelligence services. Additionally, we specialize in artificial intelligence and machine learning applications, providing comprehensive IT support and maintenance, and enhancing operational performance through strategic initiatives.

### **Care:**

They prioritizes care operations to ensure client satisfaction and seamless service delivery through responsive support and proactive communication.

### **Financial Operations:**

There financial operations focus on optimizing cost-efficiency and financial reporting accuracy, enabling our clients to achieve their strategic business goals effectively.

### **Security Operations:**

The Security operations at Compsoft Technologies are paramount, employing industry best practices and robust protocols to safeguard data integrity and protect against cyber threats. Our key services include monitoring payments, data usage, and new customers applying for connections. We enable peace-of-mind with professionalism and efficiency.

## **ABSTRACT**

In a world driven by data, organizations and individuals are faced with vast amounts of information collected from various sources. Data analytics provides the framework and methodologies to harness this data for better understanding, optimization, and prediction. Through a series of steps, data analytics transforms raw data into actionable insights.

Data analytics using Python constitutes a multifaceted approach to extracting, comprehending, and making sense of intricate datasets. Python's rich ecosystem of libraries, including Pandas, NumPy, and Matplotlib, enables data practitioners to seamlessly manipulate, clean, and transform raw data into structured formats. Further empowered by Scikit-learn and TensorFlow, Python facilitates intricate machine learning models and predictive analysis, unveiling predictive patterns and trends. Coupled with its prowess in data visualization libraries like Seaborn and Plotly, Python provides an avenue to visually communicate complex insights effectively. In amalgamating these capabilities, Python emerges as an all-encompassing platform that empowers analysts to navigate the full spectrum of data analysis, from data preprocessing and statistical exploration to machine learning-driven predictions, culminating in the generation of informed decisions and actionable outcomes.

## TABLE OF CONTENTS

CHAPTER NO	CONTENT	PAGE NO
<b>1.</b>	<b>INTRODUCTION</b>	<b>1</b>
1.1	Understanding Data Analytics	2
1.2	Machine Learning Steps	2
1.3	Types of Machine Learning	3
1.4	Machine Learning Techniques	4
1.5	Machine Learning Tools	5
<b>2.</b>	<b>Training Work Undertaken</b>	
2.1	Setting up a Python environment	7
2.2	Learn the Basic Concepts of Python	7
<b>3.</b>	<b>PROJECTS</b>	
3.1	Sentiment Analysis of Covid-19 Tweets	10
<b>4.</b>	<b>CONCLUSION</b>	<b>21</b>
<b>5.</b>	<b>REFERENCES</b>	<b>22</b>

## CHAPTER 1

### INTRODUCTION

Machine Learning (ML) is a dynamic field at the intersection of computer science and artificial intelligence that empowers machines to learn patterns from data and make decisions or predictions without explicit programming. At its core, ML involves algorithms and statistical models that enable systems to recognize patterns, uncover hidden insights, and improve performance over time through experiences.

ML finds applications in numerous domains, revolutionizing industries by automating tasks, enhancing decision-making processes, and enabling innovations in areas like healthcare, finance, autonomous vehicles, and more. Techniques like neural networks, decision trees, support vector machines, and clustering algorithms form the backbone of ML.

### PROBLEM STATEMENT

The objective of this project is to develop a Python application that performs sentiment analysis on COVID-19-related tweets sourced from Twitter using Machine Learning (ML) techniques. The application prompts the user to input a keyword related to COVID-19, retrieves tweets containing that keyword, and subsequently analyzes the sentiment expressed in those tweets.

The specific tasks of this project include:

**Data Retrieval:** Utilize Twitter's API or an open-source dataset to gather tweets related to the provided keyword or keywords associated with COVID-19.

**Data Preprocessing:** Cleanse and preprocess the collected tweets by removing noise, such as special characters, URLs, and irrelevant information, and perform tokenization, stemming, and lemmatization to standardize the text.

**Feature Extraction:** Transform the preprocessed text data into numerical features suitable for ML algorithms. Techniques such as TF-IDF (Term Frequency-Inverse Document Frequency) or word embeddings (like Word2Vec or GloVe) can be employed for this purpose.

**Sentiment Analysis:** Train a Machine Learning model (e.g., Naive Bayes, Support Vector Machines, or Recurrent Neural Networks) using the preprocessed tweet data and their corresponding sentiment labels (positive, negative, neutral). The model should be capable of predicting the sentiment of new tweets based on the learned patterns.



## 1.1 Understanding Machine Learning:

Machine learning is a powerful field that involves using algorithms and statistical models to analyze data, extracting insights that drive actionable decisions. ML techniques enable the discovery of patterns and trends within data that might not be readily apparent through traditional methods. By harnessing ML, businesses can optimize processes, improve decision-making, and achieve greater efficiency and effectiveness in various domains..

It goes beyond traditional analytics by dynamically adapting algorithms to analyze data and predict outcomes. In gaming, ML determines optimal reward schedules to engage players and enhance their experience. In content companies, ML optimizes content recommendations and organization to maximize user engagement, ensuring continual interaction and satisfaction.

It is essential for businesses as it empowers them to optimize operations and make informed decisions based on predictive insights. By deploying ML models, companies can automate processes, identify patterns in vast datasets, and predict future trends with accuracy. This capability not only enhances operational efficiency but also enables proactive strategies that drive innovation and competitive advantage in various industries.

It also plays a crucial role in personalized customer experiences. By analyzing customer behavior and preferences, ML algorithms can tailor recommendations, promotions, and services to individual users. This level of personalization not only improves customer satisfaction but also increases engagement and loyalty. Moreover, machine learning enables businesses to anticipate customer needs and market trends, allowing them to stay ahead of competitors and adapt quickly to changing market dynamics. Thus, ML not only transforms internal operations but also revolutionizes customer interactions, driving sustainable growth and success in the digital era.

## 1.2 Machine Learning Steps:

The process involved in Machine Learning involves several steps:

1. **Problem Definition:** Clearly define the problem you are trying to solve and determine the objectives and criteria for a successful outcome.
2. **Data Collection:** Gather the data relevant to your problem from various sources. Ensure the data collected is sufficient and relevant.
3. **Data Cleaning:** Handle missing values, duplicates, and inconsistencies in the data. Remove or correct any erroneous data points.

4. **Exploratory Data Analysis (EDA):** Perform initial investigations on the data to discover patterns, spot anomalies, and check assumptions. Use statistical tools and visualization techniques to gain a better understanding of the data.
5. **Feature Selection and Engineering:** Select the most relevant features (variables) that significantly impact the outcome. Create new features from existing ones to improve the model's performance.
6. **Data Splitting:** Split the dataset into training and testing sets to evaluate the model's performance. Sometimes, a validation set is also used to tune model parameters.
7. **Model Selection:** Choose the appropriate machine learning algorithm(s) that best fit the problem and data characteristics. Consider the type of problem (classification, regression, clustering, etc.) and the complexity of the model.
8. **Model Training:** Train the selected model(s) using the training dataset. Optimize the model parameters to improve performance.
9. **Model Evaluation:** Evaluate the trained model using the testing dataset to assess its performance. Use metrics such as accuracy, precision, recall, F1 score, and RMSE, depending on the problem type.
10. **Model Deployment:** Deploy the model into a production environment where it can make real-time predictions. Ensure the deployment process is seamless and that the model can handle the expected load.
11. **Communication and Reporting:** Communicate the findings, model performance, and insights gained from the analysis to stakeholders. Use visualization tools and reports to make the results accessible and understandable.

## 1.3 Types of Machine Learning

Data analytics is broken down into four basic types:

### 1. Supervised Learning:

- **Definition:** The model is trained on labelled data, where the input data is paired with the correct output.
- **Common Algorithms:** Linear Regression, Logistic Regression, Support Vector Machines (SVM), Decision Trees, Random Forests, k-Nearest Neighbours (k-NN), Neural Networks.
- **Applications:** Email spam detection, fraud detection, medical diagnosis, and speech recognition.

### 2. Unsupervised Learning:

- **Definition:** The model is trained on unlabelled data and must find hidden patterns or intrinsic structures in the input data.
- **Common Algorithms:** k-Means Clustering, Hierarchical Clustering, Principal Component Analysis (PCA), Association Rules, Autoencoders.
- **Applications:** Market basket analysis, customer segmentation, anomaly detection, and gene sequence analysis.

### 3. Semi-Supervised Learning:

- **Definition:** Combines a small amount of labelled data with a large amount of unlabelled data during training.
- **Common Algorithms:** Semi-Supervised Support Vector Machines (S3VM), Co-Training, Self-Training, Transductive SVM.
- **Applications:** Image classification, speech recognition, and web content classification.

### 4. Reinforcement Learning:

- **Definition:** The model learns by interacting with an environment and receiving feedback in the form of rewards or penalties.
- **Common Algorithms:** Q-Learning, Deep Q-Networks (DQN), Policy Gradient Methods, Actor-Critic Methods.
- **Applications:** Robotics, game playing, autonomous driving, and recommendation systems.

## 1.4 Machine Learning Techniques

Machine Learning techniques encompass a wide array of methodologies and approaches that aim to extract valuable insights and patterns from data. These techniques are used to analyze data sets of varying sizes and complexity. Here are some common machine learning techniques:

1. **Linear Regression:** A supervised learning technique used for predicting a continuous target variable based on the linear relationship between the input features and the target.
2. **Logistic Regression:** A supervised classification technique that models the probability of a discrete outcome, typically binary, using a logistic function.
3. **Decision Trees:** A supervised learning method that uses a tree-like graph of decisions and their possible consequences to make predictions based on input features.
4. **Random Forests:** An ensemble learning technique that constructs multiple decision trees during training and outputs the mode or mean prediction of individual trees for classification or regression tasks.
5. **Support Vector Machines (SVM):** A supervised learning model used for classification and

- regression by finding the hyperplane that best separates different classes in the feature space.
6. **k-Nearest Neighbors (k-NN):** A simple, instance-based learning algorithm that classifies new instances based on the majority class of their k-nearest neighbors in the training set.
  7. **k-Means Clustering:** An unsupervised learning technique that partitions the data into k distinct clusters based on feature similarity, minimizing the variance within each cluster.
  8. **Hierarchical Clustering:** An unsupervised learning method that builds a hierarchy of clusters by either iteratively merging small clusters or splitting large clusters.
  9. **Principal Component Analysis (PCA):** An unsupervised dimensionality reduction technique that transforms the data into a new coordinate system with the most significant variance captured in fewer dimensions.
  10. **Association Rules:** An unsupervised learning method used for discovering interesting relationships, patterns, or associations among a set of items in large datasets.
  11. **Q-Learning:** A reinforcement learning algorithm that learns the value of actions in states through trial and error, aiming to maximize the total reward over time.
  12. **Deep Q-Networks (DQN):** An extension of Q-Learning using deep neural networks to approximate the Q-values, enabling reinforcement learning in complex environments.
  13. **Policy Gradient Methods:** Reinforcement learning algorithms that optimize the policy directly by maximizing the expected reward, often used in continuous action spaces.
  14. **Autoencoders:** An unsupervised learning neural network used for learning efficient codings by training the network to reconstruct its input, often used for dimensionality reduction and feature learning.
  15. **Neural Networks:** A set of algorithms modeled after the human brain, used in both supervised and unsupervised learning to recognize patterns and make predictions based on complex data inputs.

## 1.5 Machine Learning Tools:

1. **TensorFlow:** An open-source machine learning framework developed by Google. It is widely used for building and deploying deep learning models. TensorFlow supports both CPU and GPU computation.
2. **Keras:** An open-source neural network library written in Python. It is user-friendly, modular, and extensible, making it ideal for rapid prototyping of deep learning models. Keras can run on top of TensorFlow, Theano, or CNTK.
3. **PyTorch:** An open-source machine learning library developed by Facebook's AI Research lab. It is known for its dynamic computational graph and ease of use, making it popular for research and development in deep learning.

4. **Scikit-Learn:** A Python library for machine learning built on NumPy, SciPy, and Matplotlib. It provides simple and efficient tools for data mining and data analysis, supporting various supervised and unsupervised learning algorithms.
5. **Apache Spark MLlib:** A scalable machine learning library built on Apache Spark. It provides a variety of machine learning algorithms and utilities for large-scale data processing and analysis.
6. **H2O.ai:** An open-source platform for machine learning and artificial intelligence. H2O provides a user-friendly interface and supports various machine learning algorithms for building predictive models.
7. **RapidMiner:** An integrated data science platform that provides tools for data preparation, machine learning, deep learning, text mining, and predictive analytics. It offers both visual workflow design and scripting capabilities

## CHAPTER 2

### TRAINING WORK UNDERTAKEN

We were new to machine learning so we use Python as our primary tool, there were several foundational concepts and libraries we learnt as prerequisites. These provided us with a solid understanding of Python programming and the essential tools for data analysis. Here's a roadmap of what we learnt on the Internship 1 as well as the first few days of Internship:

#### 2.1 Setting up a Python environment:

##### 1. Install Visual Studio Code

- Download and install Visual Studio Code from the [official website](#).

##### 2. Install Python

- Download and install Python from the [official website](#). Ensure you add Python to the system PATH during installation.

##### 3. Install Jupyter Extension in VS Code

- Open VS Code.

Go to the Extensions view by clicking on the Extensions icon in the Activity Bar on the side of the window or by pressing Ctrl+Shift+X.

In the Extensions view, search for "Jupyter" and install the Jupyter extension by Microsoft.

##### 4. Install Required Packages

- With the virtual environment activated, install the necessary packages for your project. For Machine Learning, you typically need packages like NumPy, Pandas, Scikit-learn, and Matplotlib.

#### 2.2 Learn the Basic Concepts of Python:

It is essential that we first understood the fundamental concepts of Python before jumping into any kind of Data Analytics with Python. We started by learning the fundamental concepts of Python programming. This includes variables, data types, loops, conditionals, functions, and basic input/output operations. [3]

- *Python variable names rules*
- *Python reserved words*
- *Functions*
- *Python Datatypes*: Such as int, float, str, bool, etc
- *Strings and it's manipulations*

- *Lists*
- *Sets*
- *Tuples*
- *Dictionaries*
- *Python packages*

Python packages that are specifically designed for data manipulation and analysis. These packages are essential for working with datasets, cleaning and transforming data, and performing various analytical tasks. Here are some of the packages which we learnt:

1. **NumPy:** NumPy is the foundation of numerical computing in Python. It provides support for arrays, matrices, and a wide range of mathematical functions to operate on these structures efficiently. NumPy arrays are n-dimensional, allowing you to work with data in a structured and vectorized manner. It's the go-to package for any computation involving numerical data.

```
In [1]: import numpy as np

In [2]: data = np.array([1, 2, 3, 4, 5])
        mean = np.mean(data)
        print(mean)

        3.0
```

Figure 2.1: Simple code which uses numpy package and its output

2. **Pandas:** Pandas is a powerful data manipulation and analysis library. It introduces the DataFrame and Series data structures, which make it easy to work with structured data. DataFrames are similar to tables in databases, and they provide methods for loading, cleaning, transforming, and analyzing data. Pandas simplifies common tasks like indexing, grouping, and merging datasets.

```
+ Code + Text

✓ [4] # Load Dataset
2s import pandas as pd

df = pd.read_csv("https://github.com/gabrielpreda/covid-19-tweets/raw/8f8ee2b657ee9b78d122e84354d533345c5f0c42/covid19_tweets.csv")

↑ ↓
```

Figure 2.2: Importing the pandas library and reading the dataset

3. **Matplotlib** is a widely used 2D plotting library in Python that facilitates the creation of a wide range of static, animated, and interactive visualizations. It provides a versatile platform for generating various types of graphs, charts, and plots to visually represent data and convey insights.

```
In [3]: import matplotlib.pyplot as plt

# Sample data
x = [1, 2, 3, 4, 5]
y = [2, 4, 6, 8, 10]

# Create a Line plot
plt.plot(x, y, marker='o', linestyle='-', color='blue', label='Data')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.title('Simple Line Plot')
plt.legend()
plt.grid(True)

# Show or save the plot
plt.show()
```

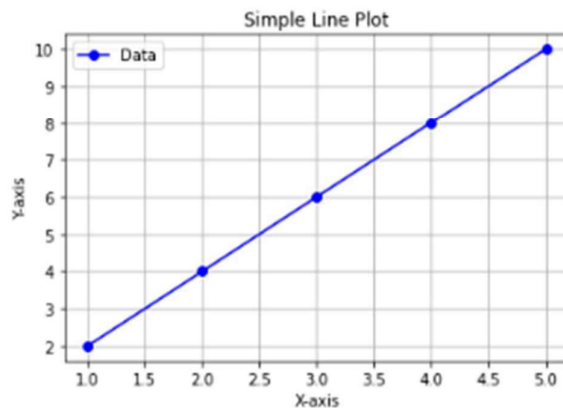


Figure 2.3: Simple program which uses the matplotlib package to plot a graph

4. **Plotly**: Plotly is known for its interactive and web-based visualizations. It offers tools for creating dynamic and interactive graphs, charts, and dashboards. Plotly's integration with Jupyter notebooks is particularly useful.
5. **Seaborn**: Seaborn is built on top of Matplotlib and offers a higher-level interface for creating visually appealing statistical graphics. It simplifies the creation of complex visualizations like heatmaps, distribution plots, and regression plots.



## CHAPTER 3

## PROJECT

### 3.1 Overview of the Project

The primary objectives of the proposed system are as follows:

- **Real-time Sentiment Analysis:** To provide a tool capable of fetching recent tweets related to COVID-19, conducting sentiment analysis, and presenting the sentiment associated with user-input keywords.
- **Improved Accuracy:** To enhance the accuracy of sentiment analysis by utilizing machine learning algorithms trained on a specialized dataset, enabling more precise sentiment classification for COVID-19-related content.
- **User-Friendly Interface:** To develop an intuitive and interactive Python application that allows users to input keywords and easily comprehend the sentiment analysis results in a comprehensible format.

The proposed system aims to fulfill these objectives by leveraging machine learning techniques and providing a practical tool for understanding public sentiment dynamics during the COVID-19 pandemic on social media platform Twitter.

### 3.2 System Requirement

#### 3.2.1 Hardware Requirements:

The hardware specifications for the successful execution of the proposed sentiment analysis application include:

##### 1. Processor:

- Minimum: Dual-core processor (e.g., Intel Core i3 or equivalent)
- Recommended: Quad-core processor (e.g., Intel Core i5 or higher) for faster processing of large datasets
- Memory (RAM):
- Minimum: 4 GB
- Recommended: 8 GB or higher for efficient handling of data processing and machine learning operations

##### 2. Storage:

- Minimum: 50 GB of available disk space
- Recommended: SSD storage for improved data access and faster processing
- Internet Connectivity:
- Stable internet connection for accessing Twitter's API or fetching real-time data from the web.

### 3.2.2 Software Requirements:

The software specifications essential for developing and running the sentiment analysis application are:

1. **Operating System:**

- Compatible with Windows, macOS, or Linux distributions

2. **Python Programming Language:**

- Version 3.x (preferably the latest stable release) as the primary programming language

3. **Development Environment:**

- Jupyter Notebook: For interactive development and data exploration

4. **Python Libraries and Frameworks:**

- **Pandas:** Provides powerful data structures for efficient data manipulation and analysis.
- **NumPy:** Supports large, multi-dimensional arrays and matrices, and a collection of mathematical functions.
- **Seaborn:** Facilitates the creation of attractive and informative statistical graphics.
- **TextBlob:** Simplifies common natural language processing (NLP) tasks.
- **Matplotlib:** Offers extensive options for creating static, animated, and interactive visualizations.
- **Neattext:** Cleans and preprocesses text by removing unwanted characters and anomalies.
- **WordCloud:** Generates word clouds to visualize the prominence of terms in a text corpus.

## 3.3 Design and Analysis

### 3.3.1 Data

The dataset credited to Gpreda and hosted on Gabriel Preda's GitHub repository consists of COVID-19 tweets sourced from Twitter. This dataset encompasses a diverse range of text-based information, including tweet content, user profiles, timestamp data, and engagement metrics such as retweets and likes. It serves as a comprehensive resource reflecting discussions, opinions, and sentiments related to the COVID-19 pandemic across social media. With its wealth of textual data, the dataset enables analysis to unveil trends, sentiment patterns, and user behavior, offering valuable insights into public discourse, awareness, and reactions concerning the global health crisis on the Twitter platform.

```

Dataset
• Credit (Gpreda)
• https://github.com/gabrielpreda/covid-19-tweets/raw/8f8ee2b657ee9b78d122e84354d533345c5f0c42/covid19\_tweets.csv

### EDA Pkgs
import pandas as pd

# Data Viz Pkg
import matplotlib.pyplot as plt
import seaborn as sns

# Hide warnings
import warnings
warnings.filterwarnings('ignore')

# Load Dataset
df = pd.read_csv("https://github.com/gabrielpreda/covid-19-tweets/raw/8f8ee2b657ee9b78d122e84354d533345c5f0c42/covid19_tweets.csv")

# Preview
df.head()

```

	user_name	user_location	user_description	user_created	user_followers	user_friends	user_favourites	user_verified	date	text
0	🌈🌈🌈	astroworld	wednesday addams as a disney princess keepin l...	2017-05-26 05:46:42	624	950	18775	False	2020-07-25 12:27:21	If I smelled the scent of hand sanitizers toda...
1	Tom Basile us	New York, NY	Husband, Father, Columnist & Commentator. Auth...	2009-04-16 20:06:23	2253	1677	24	True	2020-07-25 12:27:17	Hey @Yankees @YankeesPR and @MLB - wouldn't it...
2	Time4tasticuffs	Pewee Valley, KY	#Christian #Catholic #Conservative #Reagan #Re...	2009-02-28 18:57:41	9275	9525	7254	False	2020-07-25 12:27:14	@diane3443 @wdunlap @realDonaldTrump Trump nev...
3	ethel mertz	Stuck in the Middle	#Browns #Indians #ClevelandProud # [] #Cavs ...	2019-03-07 01:45:06	197	987	1488	False	2020-07-25 12:27:10	@brookbanktv The one gift #COVID19 has give me...
4	DIPR-J&K	Jammu and Kashmir	Official Twitter handle of Department of Inf...	2017-02-12 06:45:15	101009	168	101	False	2020-07-25 12:27:08	25 July : Media Bulletin on Novel #CoronaVirus...

Figure 3.1: Dataset source and preview of data

### 3.3.2 Data Preprocessing

Data preprocessing is a crucial step in machine learning (ML) that involves cleaning and transforming raw data to enhance its quality and usability. Common techniques include handling missing values, removing duplicates, scaling features, and encoding categorical variables. Additionally, normalization and standardization are often applied to ensure consistent data ranges.

```

[9] # Source/ Value Count/Distribution of the Sources
df['source'].value_counts()

Twitter Web App      56891
Twitter for Android  40179
Twitter for iPhone   35472
TweetDeck            8543
Hootsuite Inc.       7321
...
DataBlogger          1
Dear_Assistant       1
OnlyPultCom          1
Washington Square Parkerz  1
Radiology: AI app    1
Name: source, Length: 610, dtype: int64

# Plot the top value_counts
df['source'].value_counts().nlargest(30)

```

Twitter Web App	56891
Twitter for Android	40179
Twitter for iPhone	35472
TweetDeck	8543
Hootsuite Inc.	7321
Twitter for iPad	4336
Buffer	2728
Sprout Social	1833
Instagram	1759

```

# Source/ Value Count/Distribution of the Sources
df['source'].unique()

array(['Twitter for iPhone', 'Twitter for Android', 'Twitter Web App',
      'Buffer', 'TweetDeck', 'Twitter for iPad', 'Africa Newsroom',
      'Blood Donors India', 'TweetCaster for Android',
      'Alexander Higgins', 'IFTTT', 'Hootsuite Inc.', 'Sprout Social',
      'Sprinkl', 'assarofficial', 'IAMBLOG2TWITTER', 'CrowdControlHQ',
      'COVID19-Updates', 'EveryoneSocial', 'Dynamic Signal', 'Instagram',
      'TweetCaster for iOS', 'GlobalPandemic.NET', 'Venrap Radio',
      'HeyOrca', 'Twitter for Advertisers', 'Paper.li',
      'Twitter Media Studio', 'Twitter for Mac', 'dlvr.it',
      'Cheap Bots, Done Quick!', 'Prof. Shanku', 'LaterMedia',
      'SEMrush Social Media Tool', 'Twitterrific for iOS',
      'Sebastian's Twitter Bot', 'Threader_client', 'COVID19FactoidBot',
      'PwC UK SMART', 'tweet pro stiff', 'UK COVID-19 Alerts',
      'Resistbot Open Letters', 'preprint-alert', 'ContentStudio.io',
      'Peeping Moon', 'TweetAutomaticos', 'Orlo', 'AgoraPulse Manager',
      'Meltwater Social', 'Blog2Social APP',
      'Social Genie by Brighter Vision', 'Social Media Publisher App',
      'VoiceToData', 'Hearsay Social', 'Metricool', 'SocialPilot.co',
      'Loomly', 'Owly', 'Facelift-Cloud', 'Khoros', 'Oktopost',
      'coronaData Test', 'SocialOomph', 'SmarterQueue',
      'Salesforce - Social Studio', 'Twittimer', 'Dolar Değiştir',
      'COVID19 Update', 'LinkedIn', 'Socialbakers',
      'Bambu by Sprout Social', 'HubSpot', 'National Herald',
      'Twitter Ads', 'twtotlk', 'WordPress.com',
      'Twitter Media Studio - LiveCut', 'Covid-19 Bot',
      'Tweethot for iOS', 'Zoho Social', 'Mehila Meh (M2)'])

```

Figure 3.2: Identifying source value count distribution in tweets

Data preprocessing is a crucial step in machine learning (ML) that involves cleaning and transforming raw data to enhance its quality and usability. Common techniques include handling missing values, removing duplicates, scaling features, and encoding categorical variables. Additionally, normalization and standardization are often applied to ensure consistent data ranges. Exploring and visualizing data through techniques like histograms or scatter plots can also aid in understanding and preparing the data for ML models.

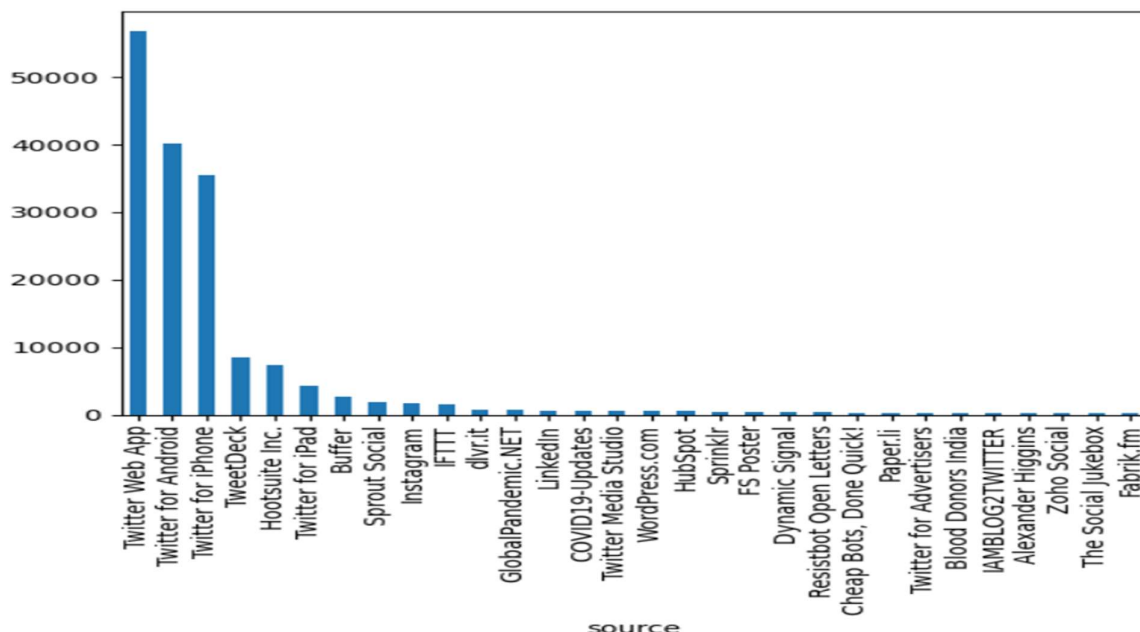


Figure 3.3: plotting source with its count

### 3.3.3 Sentimental Analysis

Sentiment analysis, also known as opinion mining, is a natural language processing (NLP) task

that involves determining the sentiment expressed in a piece of text, such as a review, tweet, or comment. It plays a crucial role in understanding public opinion, customer feedback, and social media trends. In machine learning, sentiment analysis is approached with various techniques and models.

```
[31] from textblob import TextBlob

def get_sentiment(text):
    blob = TextBlob(text)
    sentiment_polarity = blob.sentiment.polarity
    sentiment_subjectivity = blob.sentiment.subjectivity
    if sentiment_polarity > 0:
        sentiment_label = 'Positive'
    elif sentiment_polarity < 0:
        sentiment_label = 'Negative'
    else:
        sentiment_label = 'Neutral'
    result = {'polarity':sentiment_polarity,
              'subjectivity':sentiment_subjectivity,
              'sentiment':sentiment_label}
    return result

[33] # Text
ex1 = df['clean_tweet'].iloc[0]

[34] get_sentiment(ex1)

{'polarity': -0.25, 'subjectivity': 0.25, 'sentiment': 'Negative'}
```

Figure 3.4: Marking polarity, subjectivity, sentiment

### 3.3.4 Noise Removal

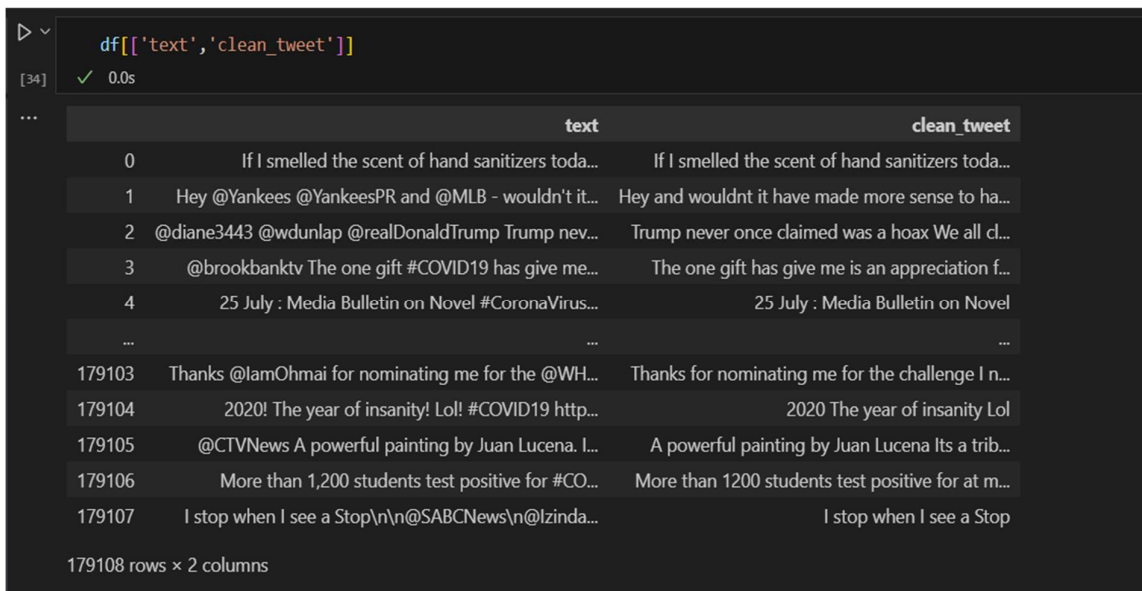
**Mentions/User Handles:** User mentions (e.g., @username) are typically removed to prevent bias towards specific users and to maintain privacy.

- **Hashtags:** Hashtags (e.g., #hashtag) are often removed as they can introduce noise, although in some cases, they can be extracted and analyzed separately.
- **URLs:** Links to websites (e.g., http://example.com) are removed to avoid irrelevant content and reduce the noise in the text.
- **Emojis:** Emojis can be removed or translated into text to simplify the analysis and ensure consistency in the data.
- **Special Characters:** Characters such as punctuation marks and symbols (e.g., !, @, #, \$) are removed to clean the text and improve the clarity of the analysis.

```
df['extracted_hashtags'] = df['text'].apply(nfx.extract_hashtags)
df[['extracted_hashtags', 'hashtags']]
df['clean_tweet'] = df['text'].apply(nfx.remove_hashtags)
df[['text', 'clean_tweet']]
df['clean_tweet'] = df['clean_tweet'].apply(lambda x: nfx.remove_userhandles(x))
df[['text', 'clean_tweet']]

[38] ✓ 1.6s
```

Figure 3.5: Noise Removal



```
df[['text', 'clean_tweet']]
```

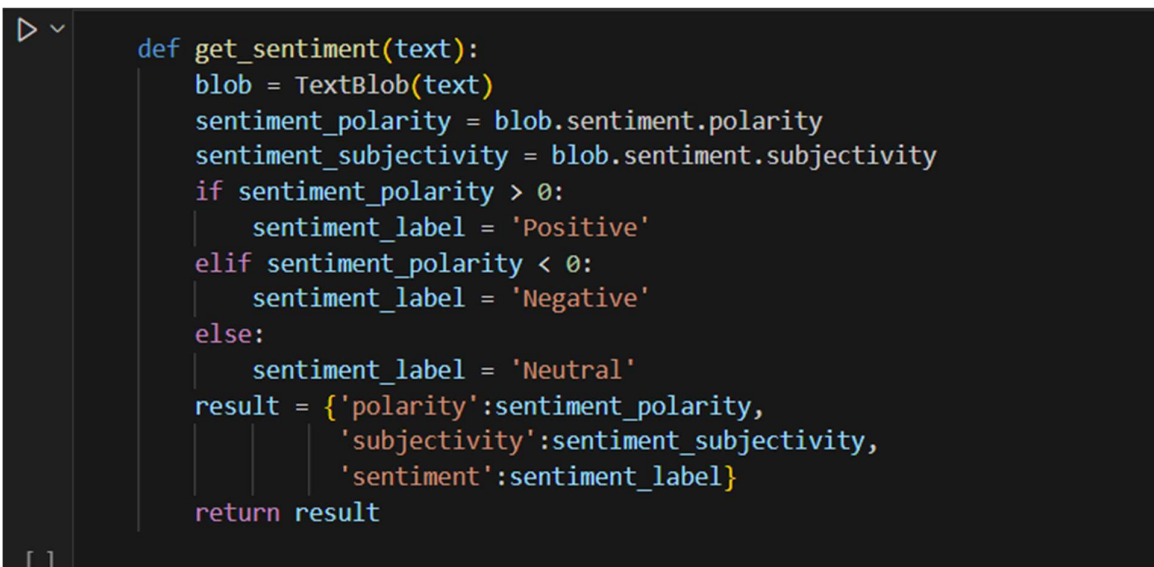
	text	clean_tweet
0	If I smelled the scent of hand sanitizers toda...	If I smelled the scent of hand sanitizers toda...
1	Hey @Yankees @YankeesPR and @MLB - wouldn't it...	Hey and wouldnt it have made more sense to ha...
2	@diane3443 @wdunlap @realDonaldTrump Trump nev...	Trump never once claimed was a hoax We all cl...
3	@brookbanktv The one gift #COVID19 has give me...	The one gift has give me is an appreciation f...
4	25 July : Media Bulletin on Novel #CoronaVirus...	25 July : Media Bulletin on Novel
...	...	...
179103	Thanks @lamOhmai for nominating me for the @WH...	Thanks for nominating me for the challenge I n...
179104	2020! The year of insanity! Lol! #COVID19 http...	2020 The year of insanity Lol
179105	@CTVNews A powerful painting by Juan Lucena. I...	A powerful painting by Juan Lucena Its a trib...
179106	More than 1,200 students test positive for #CO...	More than 1200 students test positive for at m...
179107	I stop when I see a Stop\n\n@SABCNews\n@lzinda...	I stop when I see a Stop

179108 rows x 2 columns

Figure 3.6: comparison between tweet and cleaned tweet

### 3.3.4 Sentiment Analysis

Sentiment analysis, also known as opinion mining, is a natural language processing (NLP) task that involves determining the sentiment expressed in a piece of text, such as a review, tweet, or comment. It plays a crucial role in understanding public opinion, customer feedback, and social media trends. In machine learning, sentiment analysis is approached with various techniques and models.



```
def get_sentiment(text):
    blob = TextBlob(text)
    sentiment_polarity = blob.sentiment.polarity
    sentiment_subjectivity = blob.sentiment.subjectivity
    if sentiment_polarity > 0:
        sentiment_label = 'Positive'
    elif sentiment_polarity < 0:
        sentiment_label = 'Negative'
    else:
        sentiment_label = 'Neutral'
    result = {'polarity':sentiment_polarity,
              'subjectivity':sentiment_subjectivity,
              'sentiment':sentiment_label}
    return result
```

Figure 3.7: function for labelling the sentiment

The primary goal of sentiment analysis is to classify the sentiment of a given text into categories like positive, negative, or neutral. This classification enables businesses to gain insights into customer opinions, monitor brand reputation, and make data-driven decisions based on public



sentiment.

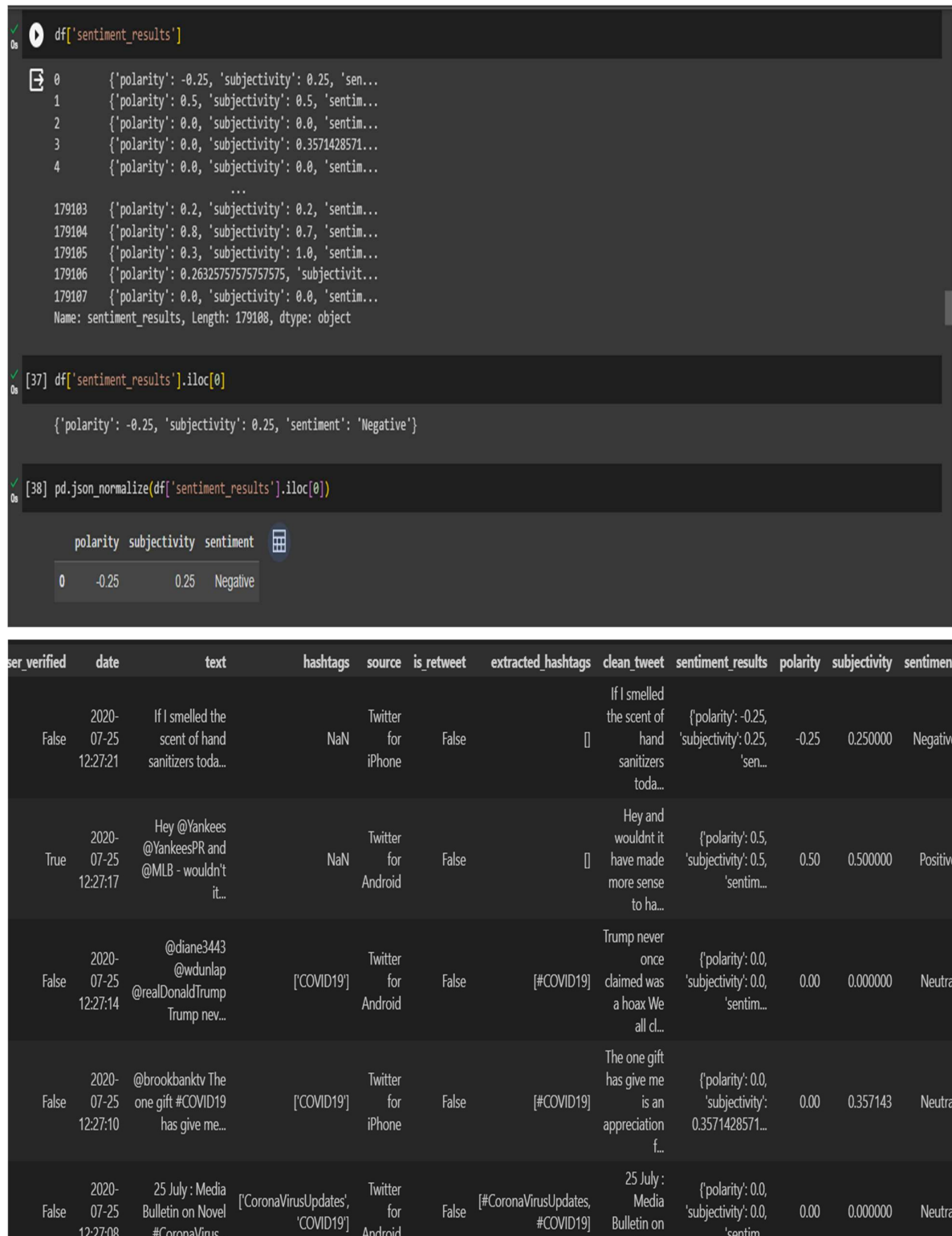


Figure 3.8: Result of sentiment analysis

### 3.3.5 Entity Extraction

Entity extraction, also known as named entity recognition (NER), is a process in natural language processing (NLP) that identifies and categorizes key information (entities) in text. These entities

can include names of people, organizations, locations, dates, and more. By extracting entities, businesses can gain structured insights from unstructured text data, enabling them to analyze customer feedback, monitor brand mentions, and track important events. Entity extraction helps in transforming raw text into meaningful information, which can be used for various applications like sentiment analysis, trend analysis, and content summarization.

```
def plot_wordcloud(docx):  
    plt.figure(figsize=(20,10))  
    mywordcloud = WordCloud().generate(docx)  
    plt.imshow(mywordcloud, interpolation='bilinear')  
    plt.axis('off')  
    plt.show()
```

Figure 3.9: Function for plotting world\_cloud

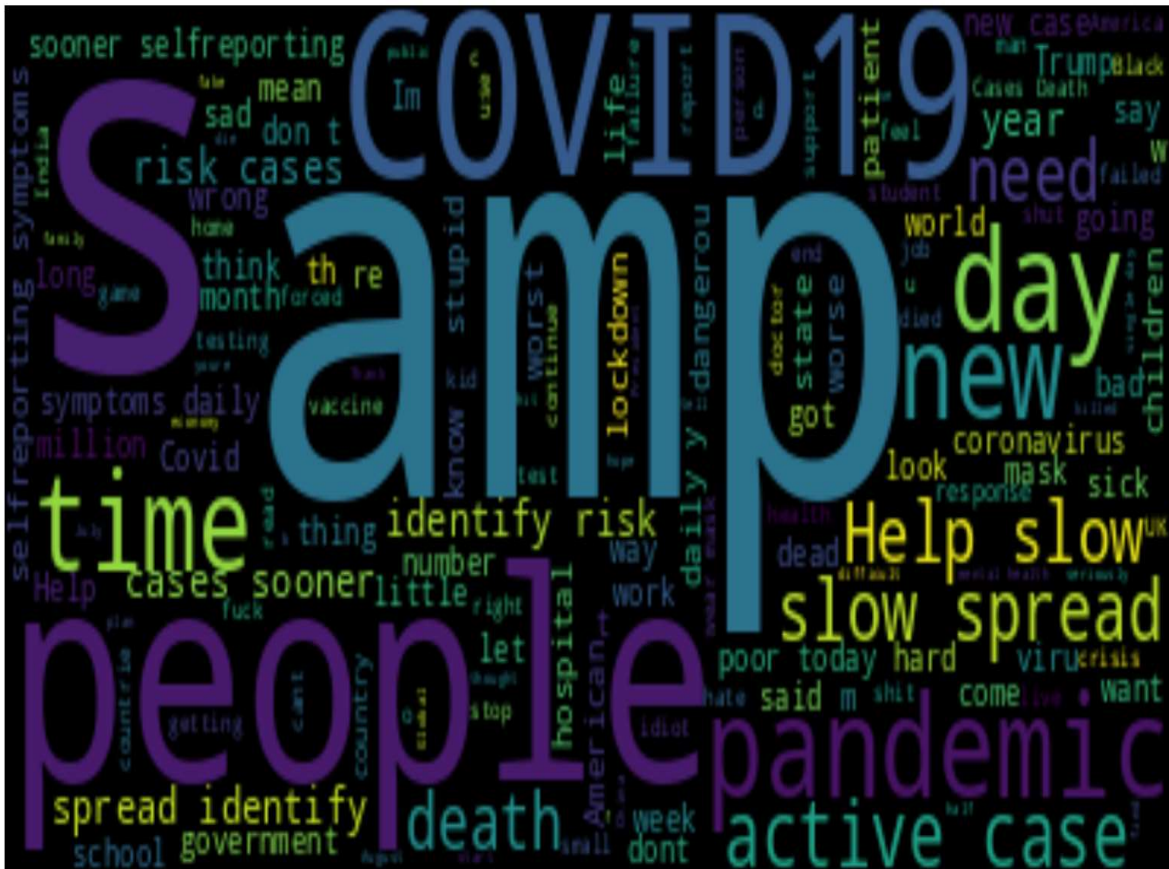


Figure 3.10: world cloud of negative docx



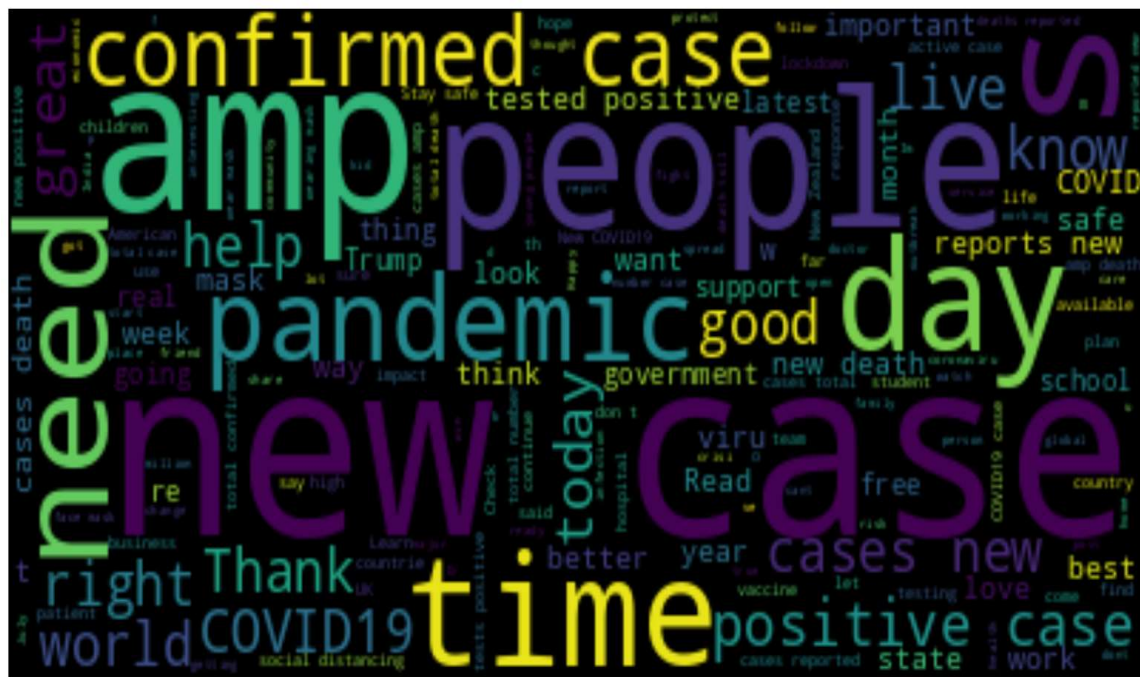


Figure 3.11: world\_cloud of neutral docx

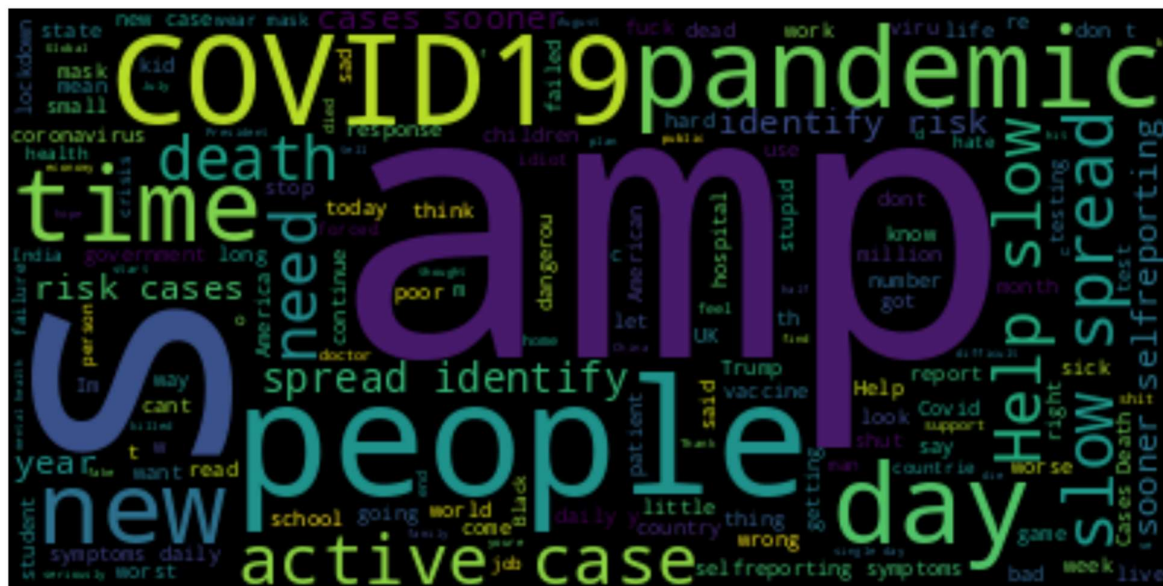


Figure 3.12: world\_cloud of positive docx

## RESULTS

### 3.4 Technical Skills Developed:

During my internship analyzing COVID-19 tweets sourced from Twitter, I honed a range of technical skills crucial to data analysis and natural language processing. I adeptly cleaned and preprocessed the dataset, addressing issues such as missing data and duplicates, and applied advanced text preprocessing techniques like tokenization, stop-word removal, and stemming/lemmatization to enhance data quality. Utilizing tools such as TextBlob and Neattext, I implemented sentiment analysis to classify tweets into positive, negative, and neutral categories, gaining proficiency in understanding and manipulating textual data. Additionally, I conducted time-series analysis to track trends in tweet content over different periods, correlating these trends with major COVID-19 events to extract meaningful insights. Skills in Python programming, particularly leveraging Pandas for data manipulation and NumPy for numerical operations, were fundamental in executing these tasks effectively, further bolstering my technical toolkit in data science.

### 3.5 Snapshots:

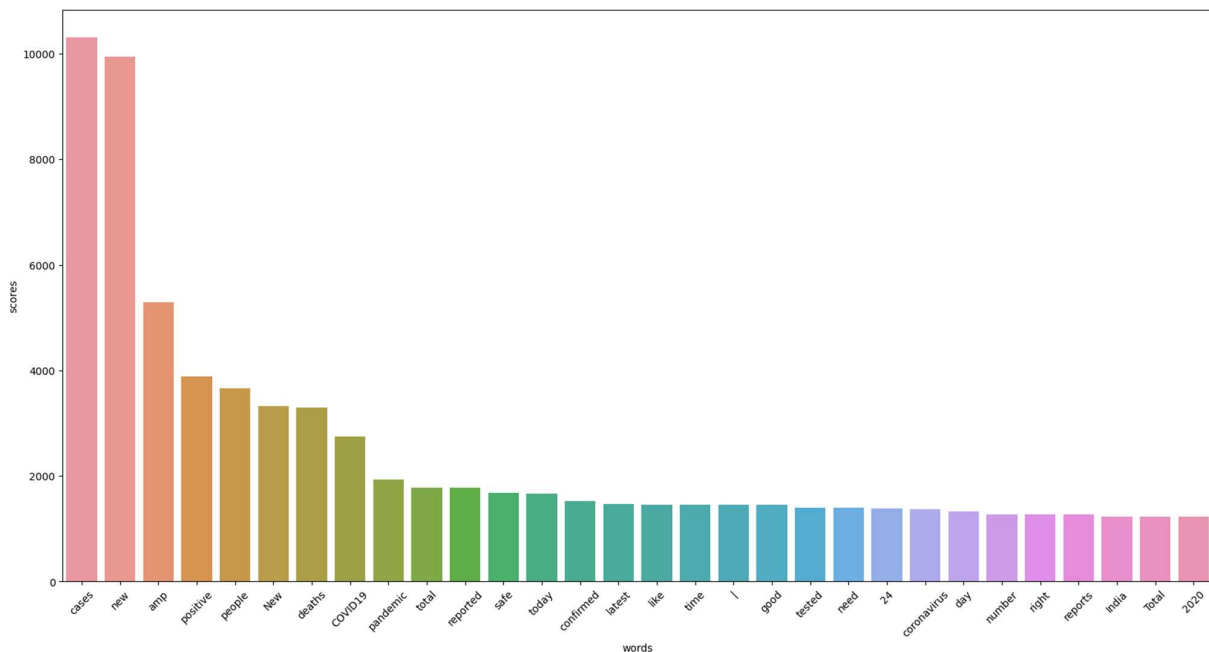


Figure 3.13: Bar graph of positive docx

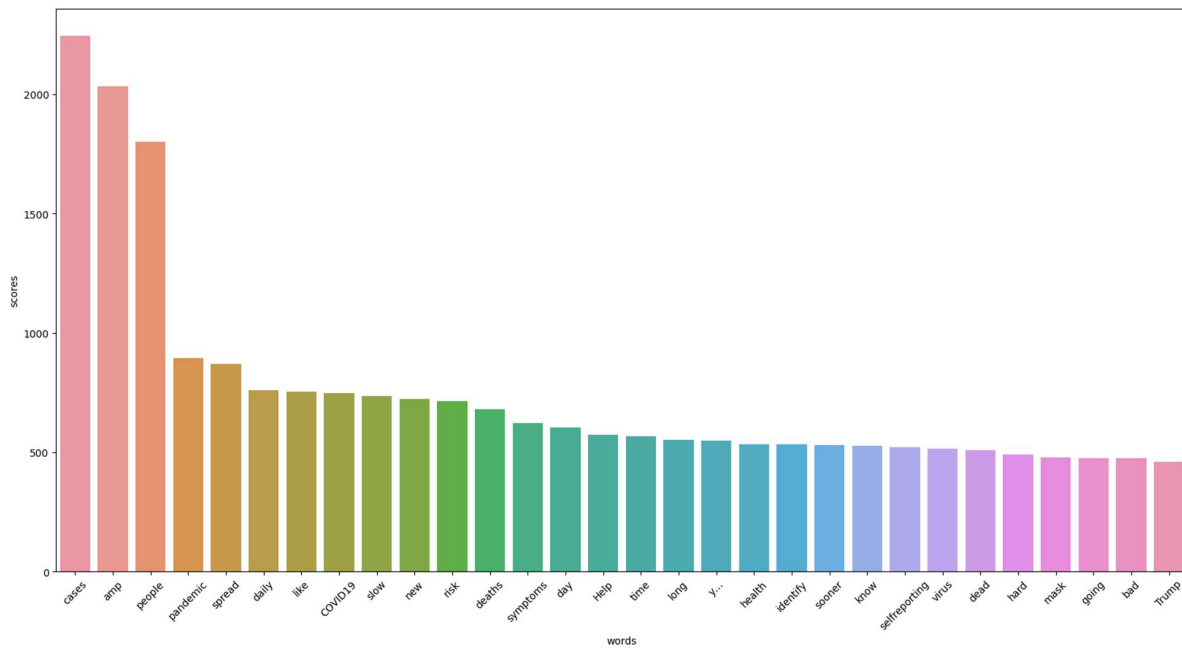


Figure 3.14: Bar graph of negative docx

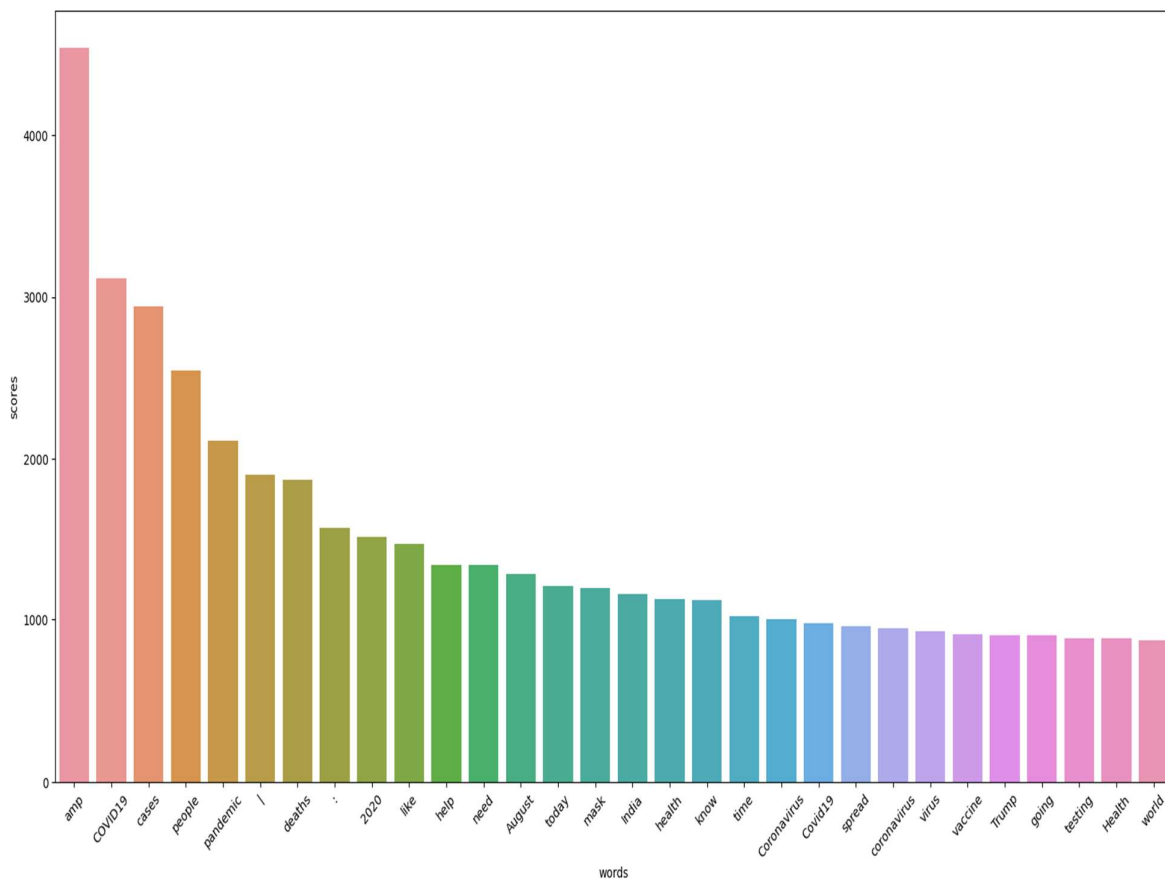


Figure 3.15: Bar graph of neutral docx

## CHAPTER 4

### CONCLUSION

The COVID-19 pandemic has had an unprecedented impact on global society, affecting health, economies, and daily life. Understanding public sentiment during such a critical period can provide valuable insights into public opinion, emotional responses, and potential areas of concern. This project aimed to develop a machine learning-based sentiment analysis system to evaluate public sentiment regarding COVID-19 on Twitter.

Preprocessing the collected tweets was a crucial step. Tweets often contain noise in the form of emojis, URLs, mentions, and hashtags. We implemented various preprocessing techniques such as tokenization, stop-word removal, and stemming to clean the textual data and convert it into a suitable format for analysis. This ensured that our machine learning model received high-quality input, leading to more accurate sentiment predictions.

Our sentiment analysis system successfully identified prevalent sentiments within the collected tweets. The analysis revealed that public sentiment fluctuated over time, often corresponding to significant events related to the pandemic, such as lockdown announcements, vaccine developments, and changes in government policies. Positive sentiments were often associated with news of recovery rates and vaccine efficacy, while negative sentiments spiked during periods of rising cases and fatalities.

The system also highlighted the presence of misinformation and panic-inducing content, which had a substantial impact on public sentiment. By identifying these trends, our sentiment analysis system can aid in understanding the spread of misinformation and its effects on public opinion. This insight can help policymakers and public health officials in crafting effective communication strategies to address public concerns and mitigate the spread of false information.

Future work could focus on expanding the dataset to include multilingual tweets, exploring more sophisticated models, and integrating additional data sources such as news articles and social media platforms to provide a more comprehensive analysis of public sentiment. By continually refining and improving these models, we can better understand and respond to public sentiment in times of crisis, ultimately contributing to more informed and effective decision-making.

## CHAPTER 5

### REFERENCES

- [1] Python Basics - Code with Harry - <https://www.youtube.com/watch?v=7IWOYhfAcVg>
- [2] Pandas - W3Schools - <https://www.w3schools.com/python/pandas/default.asp>
- [3] Numpy - W3Schools - <https://www.w3schools.com/python/numpy/default.asp>
- [4] Mathplotlib - W3Schools - <https://www.w3schools.com/python/mathplotlib/default.asp>
- [5] Dataset - github - <https://github.com/gabrielpreda/covid/tweets/>