VISVESVARAYA TECHNOLOGICAL UNIVERSITY
Jnana Sangama, Belagavi-590018

A

INTERNSHIP REPORT

On

"Sentiment Analysis of Lockdown in USA During Covid-19 A Case Study on Twitter using ML"

Submitted in partial fulfillment of the
INTERNSHIP
In
INFORMATION SCIENCE AND ENGINEERING
VII SEMESTER INTERNSHIP (18CSI85)

By
Akash Tatti (1HK20IS007)
Under the guidance of

Prof. Sharavana K

Associate Professor

Dept. of ISE, HKBKCE Bangalore

Department Of Information Science and Engineering

2023-2024

Accredited by NAAC
HKBK COLLEGE OF ENGINEERING
22/1, Nagawara, Bengaluru – 560045
E-mail:info@hkbk.edu.in, URL:www.hkbk.edu.in

HKBK COLLEGE OF ENGINEERING

Accredited by NAAC
BENGALURU-560045

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

VISVESVARAYATECHNOLOGICAL UNIVERSITY

INTERNSHIP REPORT

On

"Sentiment Analysis Of Lockdown In USA During Covid-19

A Case Study On Twitter using ML"

Submitted in partial fulfillment for

INTERNSHIP (18CSI85)

VII SEMESTER

2023-2024

SUBMITTED BY:

AKASH TATTI (1HK20IS007)

ACKNOWLEDGEMENT

I would like to express my deepest appreciation to all those who provided me the possibility to complete this internship.

First of all, I would take this opportunity to express our gratefulness to **Mr. C.M. Ibrahim**, Chairman, **Mr. C.M. Faiz Mohammed**, Director and for all the facilities provided to complete the internship in time.

I express my sincere gratitude to **Dr Mohammed Riyaz Ahmed**, Principal, HKBKCE for her support which inspired me towards the attainment of knowledge.

I wish to acknowledge the contributions of **Dr. A Syed Mustafa**, HOD, Information Science and Engineering for manifestations in great depths and helped me to complete the assigned task on time.

I sincerely thank my coordinator and guide, **Prof. Sharavana K**, Associate Professor for his constant assistance, support, patience, endurance and constructive suggestions for the betterment of the internship.

Finally, I thank Almighty, all teaching and non-teaching staff of the Department of Information Science and Engineering, my parents, family members and friends for their valuable guidance and cooperation throughout my dissertation.

AKASH TATTI (1HK20IS007)

ABSTRACT

The Covid-19 pandemic, which began in early 2020, led to unprecedented global challenges, including lockdowns implemented to curb the virus's spread. These lockdowns impacted individuals and communities in various ways, sparking a wide range of emotions and opinions. Social media platforms, particularly Twitter, became a prominent space for people to express their thoughts and sentiments during this period. This study presents a comprehensive analysis of public sentiment on Twitter in the USA during the Covid-19 lockdowns, employing machine learning techniques to extract valuable insights. The primary objective of this research is to understand the evolving sentiment trends over different phases of the lockdown and identify key factors influencing sentiment. To achieve this, we collected a vast dataset of tweets related to Covid-19 and the lockdown measures in the USA, spanning from the onset of the pandemic to the easing of restrictions. We employed Natural Language Processing (NLP) and machine learning algorithms to process and analyze this data. Our findings reveal distinct sentiment patterns over time, reflecting the evolving nature of the pandemic and public reactions. Initially, there was a surge in negative sentiment as the lockdowns disrupted daily life and raised concerns about health and economic stability. As time progressed, sentiment shifted towards a more balanced distribution, indicating adaptation and resilience within the community. Furthermore, the study explores the contextual factors contributing to sentiment shifts. We considered various factors, including the severity of Covid-19 cases, government policies, vaccination campaigns, economic relief measures, and public awareness campaigns. Through sentiment analysis, we were able to discern the impact of these factors on public sentiment, providing valuable insights for policymakers and public health officials. This research demonstrates the utility of machine learning and sentiment analysis in understanding the societal impact of major events like the Covid-19 pandemic. It not only sheds light on the emotional responses of the public during times of crisis but also offers a data-driven perspective on the effectiveness of interventions and communication strategies.

TABLE OF CONTENTS

CHAPTER -1

COMPANY PROFILE

A Brief History of Company: Varcons technologies, was incorporated with a goal, to provide high quality and optimal Technological Solutions to business requirements of our clients". Every business is different and has a unique business model and so are the technological requirements. They understand this and hence the solutions provided to these requirements are different as well. They focus on clients' requirements and provide them with tailor-made technological solutions. They also understand that Reach of their Product to its targeted market or the automation of the existing process into e-client and simple process are the key features that our clients desire from Technological Solution they are looking for and these are the features that we focus on while designing the solutions for their clients. Varcons is a Technology Organization providing solutions for all web design and development, MYSQL, PYTHON Programming, HTML, CSS, ASP.NET and LINQ. Meeting the ever-increasing automation requirements, Sarva Moola Software Services. specialize in ERP, Connectivity, SEO Services, Conference Management, effective web promotion and tailor made software products, designing solutions best suiting client's requirements. we strive to be the front runner in creativity and innovation in software development through their well-researched expertise and establish it as an out of the box software development company in Bangalore, India. As a software development company, they translate this software development expertise into value for their customers through their professional solutions. They understand that the best desired output can be achieved only by understanding the clients demand better. At our Company we work with them clients and help them to define their exact solution requirement. Sometimes even they wonder that they have completely redefined their solution or new application requirement during the brainstorming session, and here they position themselves as an IT solutions consulting group comprising of high caliber consultants. They believe that Technology when used properly can help any business to scale and achieve new heights of success. It helps Improve its efficiency, profitability, reliability; to put it in one sentence Technology helps you to Delight your Customers and that is what we want to achieve.

CHAPTER-2

STRUCTURE

2.1 Internship Structure

Month 1 [Training]				**[Projects]**
Week 1	**Week 2**	**Week 3**	**Week 4**	
What is AI & ML? Introduction on ML Introduction on Data Science Basic of Python Data Science with Python and its Programming Languages Data Lifecycle Data Manipulation Data Framework	Workflow of ML AND Introduction to Types OF ML	Supervised learning algorithm and its types Unsupervised learning algorithm ad types Reinforcement learning algorithm and types What is Deep Learning? Install Python and Libraries	Loading the datasets Artificial neuron networks Convolution Neural Network Summarizing the datasets Visualizing the datasets Making some predictions	Major Project

2.2 Projects Assigned

Project [Major Project]

Build a python application that asks for a keyword, and you need to identify the sentiment of that. keyword using an open-source dataset.
src: https://colab.research.google.com/drive/1oPzpvhw_EjbrnZhFCWnb0xu_zWWkPdZ8

2.3 Internship Objectives

The Covid-19 pandemic and subsequent lockdowns in the USA have triggered a multitude of emotional responses and opinions among the population. There is a need for an advanced sentiment analysis system that can accurately capture and analyze sentiment trends in Twitter data during different phases of the lockdown, enabling a deeper understanding of the factors influencing public sentiment.

Build a Python application that utilizes natural language processing (NLP) techniques to determine the sentiment of a given keyword. The application will take user input for a keyword, analyze its sentiment using an open-source dataset, and provide feedback on whether the keyword is associated with positive, negative, or neutral sentiments.

CHAPTER-3

INTRODUCTION

3.1 Introduction to ML:

Machine learning is a subfield of artificial intelligence (AI) that focuses on the development of algorithms and statistical models that enable computers to learn from and make predictions or decisions based on data, without being explicitly programmed. It is a rapidly evolving and interdisciplinary field with applications in various domains, including healthcare, finance, marketing, natural language processing, computer vision, and more.

3.2 Overview of Machine Learning:

Fundamental Concepts:

Data: Machine learning relies heavily on data. This data can be structured (e.g., databases) or unstructured (e.g., text or images). The quality and quantity of data play a critical role in the success of a machine learning model. Algorithm: Machine learning algorithms are the core components of the process. These algorithms are responsible for learning patterns and making predictions or decisions based on the data. Model: A model is a representation of the knowledge or patterns learned from the data by a machine learning algorithm. Models can be as simple as linear regression or highly complex, like deep neural networks.

3.3 Types of Machine Learning:

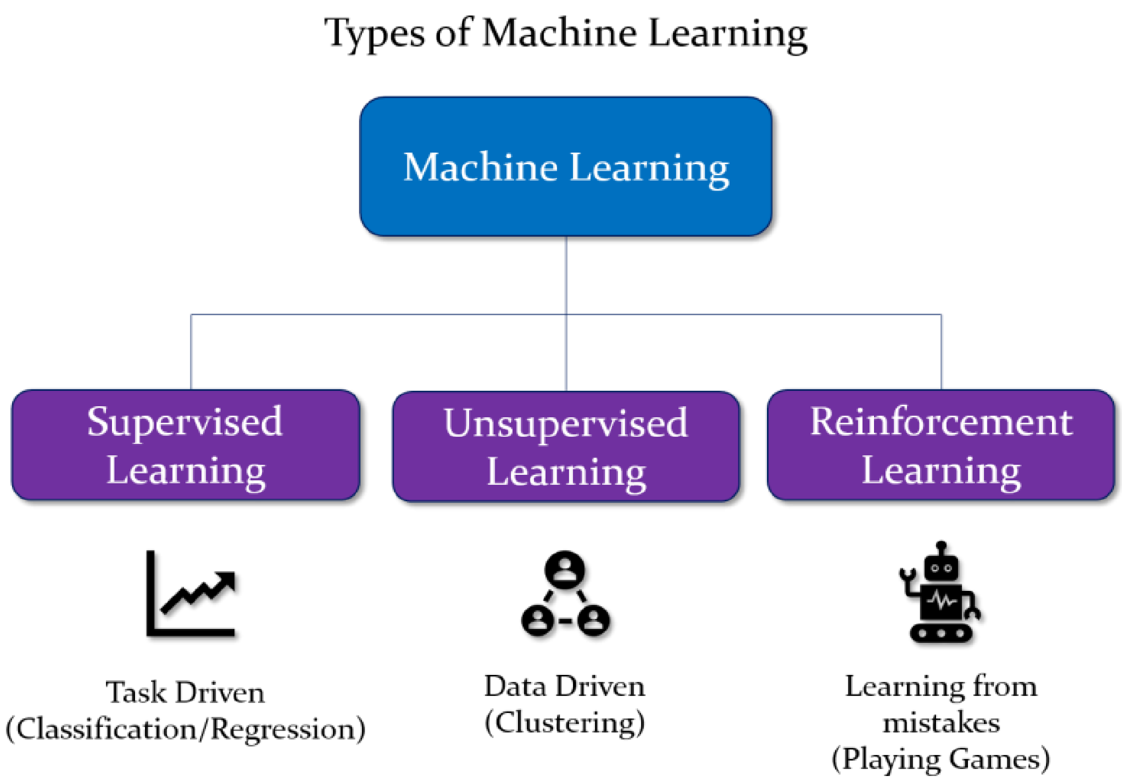


Fig 2.1: Types of machine learning

Supervised Learning: In supervised learning, the algorithm is trained on a labeled dataset, where the input data is paired with the correct output. The goal is to learn mapping from inputs to outputs, making it suitable for tasks like classification and regression.

Unsupervised Learning: Unsupervised learning deals with unlabeled data. Algorithms try to identify patterns or structure within the data, such as clustering similar data points together or reducing data dimensionality.

Semi-Supervised Learning: This is a hybrid of supervised and unsupervised learning, where the model is trained on a combination of labeled and unlabeled data. It can be useful when acquiring labeled data is expensive or time-consuming.

Reinforcement Learning: In reinforcement learning, agents learn to make sequences of decisions to maximize a cumulative reward in an environment. This approach is often used in autonomous systems and gaming.

3.4 Steps in Machine Learning:

1. Data Collection: Gathering relevant data is the first step. This may involve data cleaning, preprocessing, and feature engineering to prepare the data for training.
2. Model Selection: Choosing the appropriate algorithm or model architecture for the task at hand is crucial. This selection depends on the nature of the data and the problem to be solved.
3. Training: During this phase, the model learns from the training data. It optimizes its internal parameters to make accurate predictions or decisions.
4. Evaluation: After training, the model is assessed using a separate dataset (validation or test data) to measure its performance. Common evaluation metrics include accuracy, precision, recall, F1-score, and more, depending on the task.
5. Hyperparameter Tuning: Fine-tuning hyperparameters (e.g., learning rate, batch size, depth of a neural network) can significantly impact a model's performance

CHAPTER-4

MAJOR PROJECT

4.1 Software Requirements

The following software are required to carry out this project:

- Python (v3.10.0 or higher)
- Any editor/platform you want such as: VSCode, Anaconda, Jupyter, Python Idle, Spyder etc.

4.2 Hardware Requirements

- SYSTEM: Intel Core i3-540 Processor or higher
- HARD DISK: 40 GB or above
- RAM: 256 MB or above

4.3 Steps Followed

- You are required to load/import the libraries first.
- Load the Dataset
- Summarize the Dataset
- Visualize the dataset
- Use 5-6 Algorithms which you feel might fit perfect for such classifications such as Logistic Regression, KNN, Decision Trees, Gaussian Naive Bayes, Support Vector Machine.
- Split your dataset into train and test spit.
- Determine which algorithm gives you the most accurate results/highest accuracy scores and use that for further computations and building the final model.
- Evaluation of the model and further predictions.

4.4 Source Code with Snapshots

```
X_train = ["This was really awesome an awesome movie",
           "Great movie! Ilikes it a lot",
           "Happy Ending! Awesome Acting by hero",
           "loved it!",
           "Bad not upto the mark",
           "Could have been better",
           "really Dissapointed by the movie"]
#X_test = ["it was really awesome and really disspntd"]

y_train = ["positive","positive","positive","positive","negative","negative","negative"] # 1- Positive class, 0- negative class
```

```
[4] from nltk.tokenize import RegexpTokenizer
    # NLTK -> Tokenize -> RegexpTokenizer

[ ] # Stemming
    # "Playing" -> "Play"
    # "Working" -> "Work"

    from nltk.stem.porter import PorterStemmer
    # NLTK -> Stem -> Porter -> PorterStemmer

    from nltk.corpus import stopwords
    # NLTK -> Corpus -> stopwords
```

```
[6] # Downloading the stopwords
    import nltk
    nltk.download('stopwords')

    [nltk_data] Downloading package stopwords to /root/nltk_data...
    [nltk_data]   Unzipping corpora/stopwords.zip.
    True

[7] tokenizer = RegexpTokenizer(r"\w+")
    en_stopwords = set(stopwords.words('english'))
    ps = PorterStemmer()
```

```
def getCleanedText(text):
  text = text.lower()

  # tokenizing
  tokens = tokenizer.tokenize(text)
  new_tokens = [token for token in tokens if token not in en_stopwords]
  stemmed_tokens = [ps.stem(tokens) for tokens in new_tokens]
  clean_text = " ".join(stemmed_tokens)
  return clean_text
```

```
[9]  X_test = ["It was good"]

     X_clean = [getCleanedText(i) for i in X_train]
     xt_clean = [getCleanedText(i) for i in X_test]

[11] X_clean

     ['realli awesom awesom movi',
      'great movi ilik lot',
      'happi end awesom act hero',
      'love',
      'bad upto mark',
      'could better',
      'realli dissapoint movi']

[12] xt_clean

     ['good']
```

```
[14] from sklearn.feature_extraction.text import CountVectorizer

[15] cv = CountVectorizer(ngram_range = (1,2))
     # "I am PyDev" -> "i am", "am Pydev"

[16] X_vec = cv.fit_transform(X_clean).toarray()
```

```
     print(cv.get_feature_names_out())

     ['act' 'act hero' 'awesom' 'awesom act' 'awesom awesom' 'awesom movi'
      'bad' 'bad upto' 'better' 'could' 'could better' 'dissapoint'
      'dissapoint movi' 'end' 'end awesom' 'great' 'great movi' 'happi'
      'happi end' 'hero' 'ilik' 'ilik lot' 'lot' 'love' 'mark' 'movi'
      'movi ilik' 'realli' 'realli awesom' 'realli dissapoint' 'upto'
      'upto mark']

[19] Xt_vect = cv.transform(xt_clean).toarray()

[20] Xt_vect

     array([[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
             0, 0, 0, 0, 0, 0, 0, 0, 0, 0]])
```

```
X_vec

array([[0, 0, 2, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
        0, 0, 0, 1, 0, 1, 1, 0, 0, 0],
       [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 1,
        1, 0, 0, 1, 1, 0, 0, 0, 0, 0],
       [1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 1, 0, 0,
        0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
       [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
        0, 1, 0, 0, 0, 0, 0, 0, 0, 0],
       [0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
        0, 0, 1, 0, 0, 0, 0, 0, 1, 1],
       [0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
        0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
       [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0,
        0, 0, 0, 1, 0, 1, 0, 1, 0, 0]])
```

```
[21] from sklearn.naive_bayes import MultinomialNB

[22] mn = MultinomialNB()

mn.fit(X_vec, y_train)

▾ MultinomialNB
MultinomialNB()
```

```
y_pred = mn.predict(X_vec)

[26] y_pred

array(['positive', 'positive', 'positive', 'positive', 'negative',
       'negative', 'negative'], dtype='<U8')

[27] y_pred = mn.predict(Xt_vect)

[28] y_pred

array(['positive'], dtype='<U8')

[29] import matplotlib.pyplot as plt
```

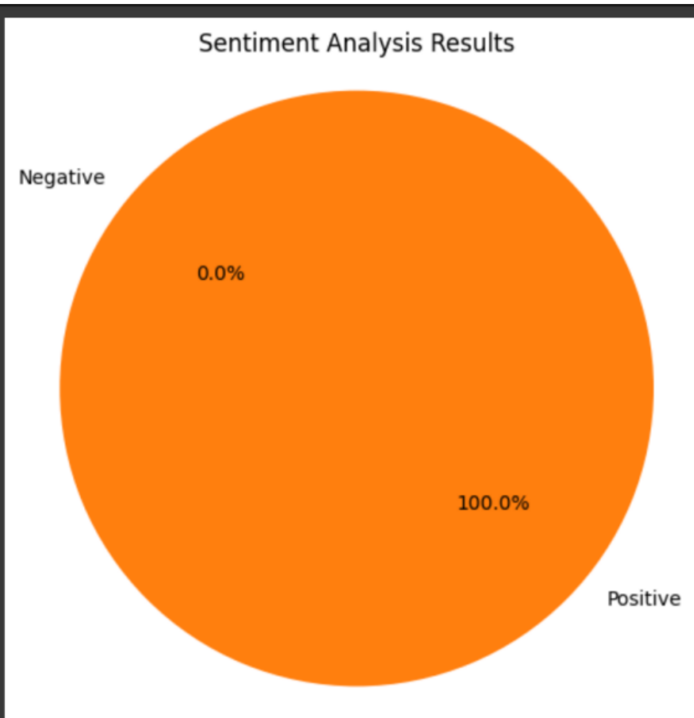
```
[30] mn.fit(X_vec, y_train)
```

```
▾ MultinomialNB
MultinomialNB()
```

```
[31] y_pred = mn.predict(Xt_vect)
```

```
[32] labels = ['Negative', 'Positive']
     count_negative = (y_pred == 'negative').sum()
     count_positive = (y_pred == 'positive').sum()
     sentiment_counts = [count_negative, count_positive]
```

```
plt.figure(figsize=(6, 6))
plt.pie(sentiment_counts, labels=labels, autopct='%1.1f%%', startangle=140)
plt.title('Sentiment Analysis Results')
plt.axis('equal')  # Equal aspect ratio ensures that pie is drawn as a circle.
plt.show()
```

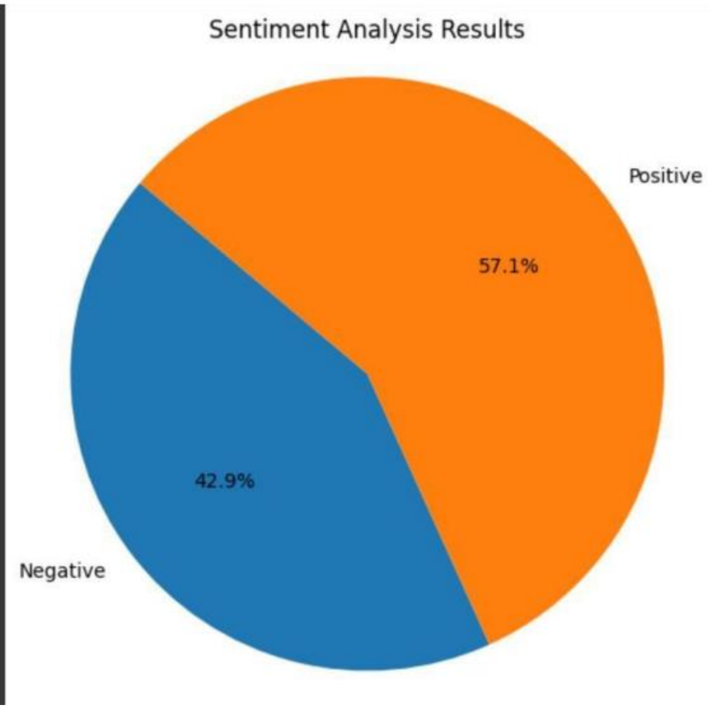


```
[34] y_pred = mn.predict(X_vec)
```

```
[35] labels = ['Negative', 'Positive']
     count_negative = (y_pred == 'negative').sum()
     count_positive = (y_pred == 'positive').sum()
     sentiment_counts = [count_negative, count_positive]
```

```
[36] plt.figure(figsize=(6, 6))
     plt.pie(sentiment_counts, labels=labels, autopct='%1.1f%%', startangle=140)
     plt.title('Sentiment Analysis Results')
     plt.axis('equal')  # Equal aspect ratio ensures that pie is drawn as a circle.
     plt.show()
```

[36]



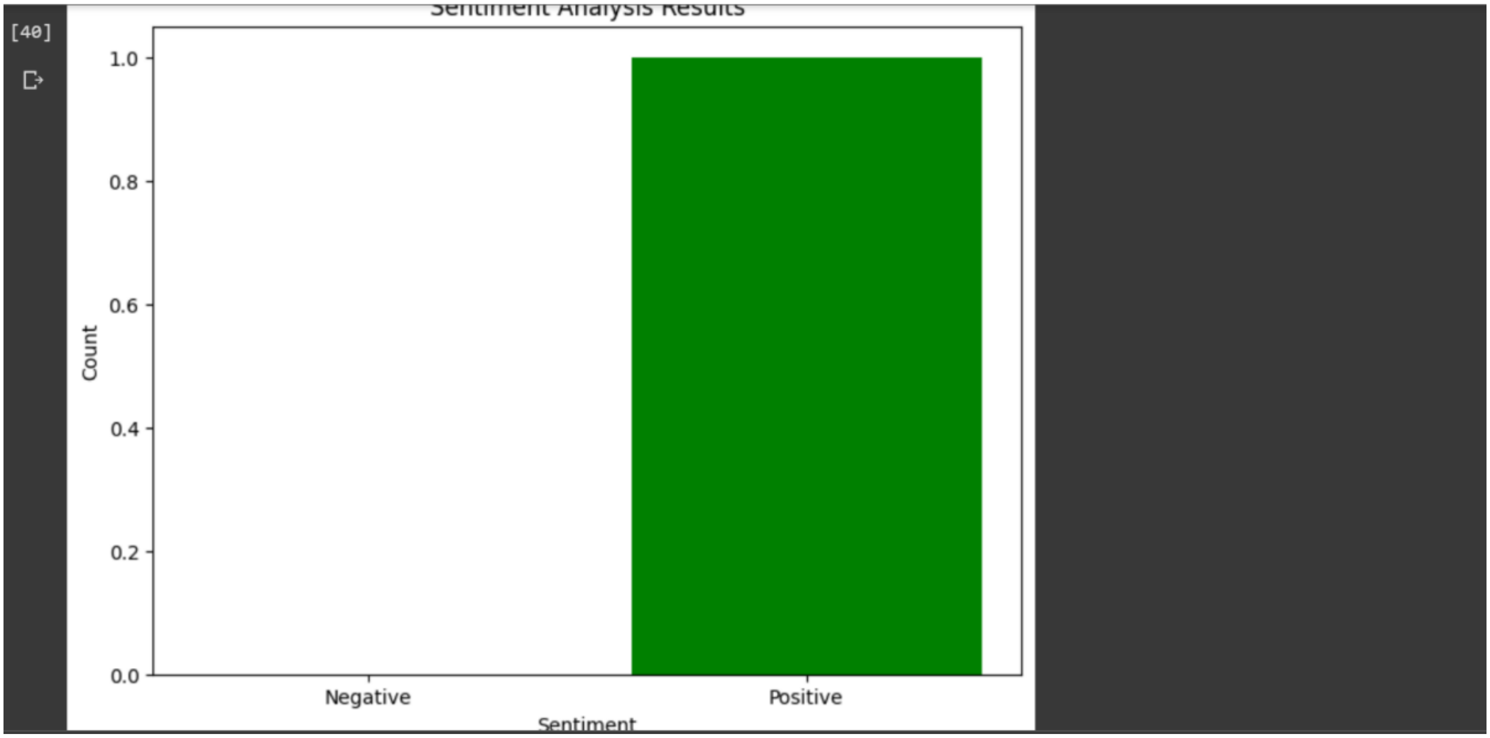

```
[37] mn.fit(X_vec, y_train)
```

```
▾ MultinomialNB
MultinomialNB()
```

```
[38] y_pred = mn.predict(Xt_vect)
```

```
[39] labels = ['Negative', 'Positive']
     count_negative = (y_pred == 'negative').sum()
     count_positive = (y_pred == 'positive').sum()
     sentiment_counts = [count_negative, count_positive]
```

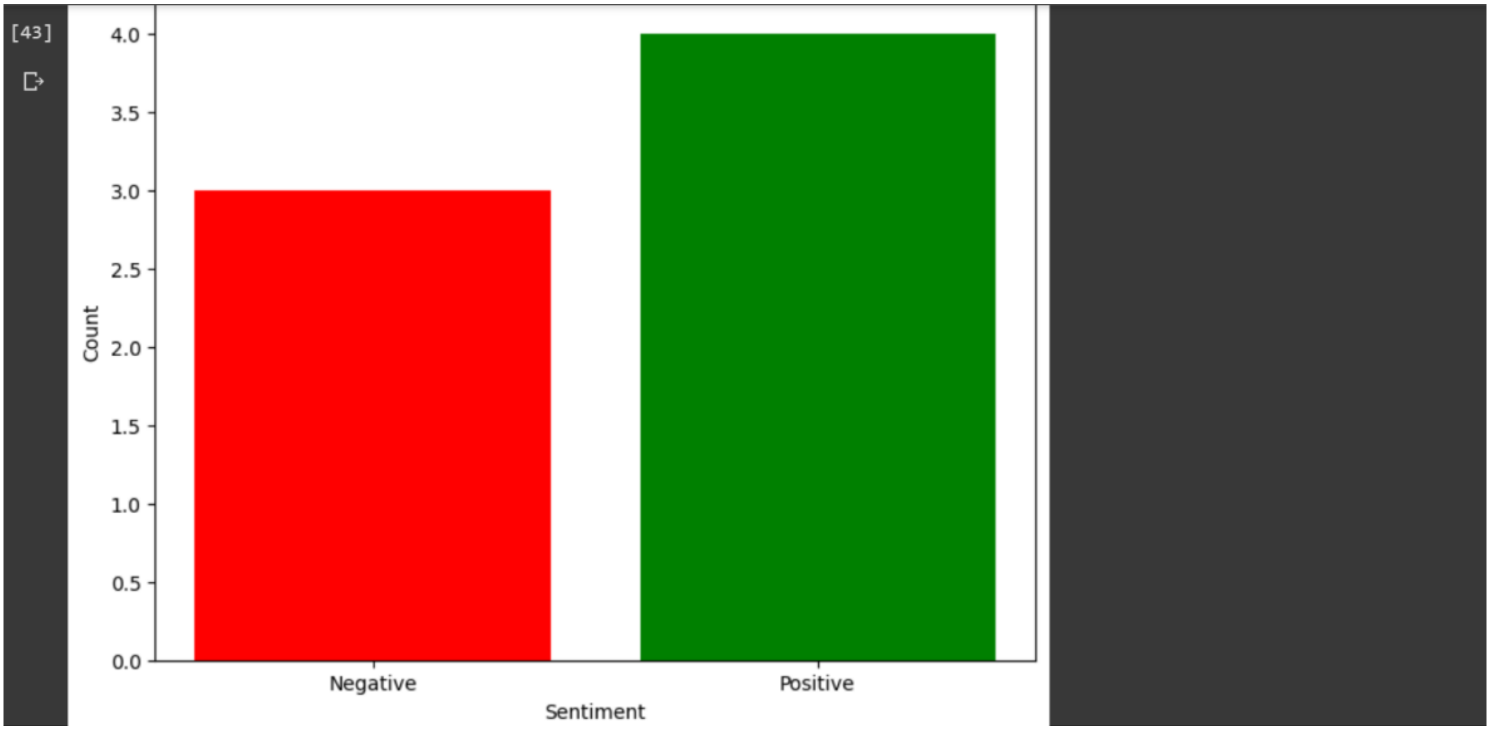
```
[40] plt.figure(figsize=(8, 6))
     plt.bar(labels, sentiment_counts, color=['red', 'green'])
     plt.xlabel('Sentiment')
     plt.ylabel('Count')
     plt.title('Sentiment Analysis Results')
     plt.show()
```




```
[41] y_pred = mn.predict(X_vec)
```

```
[42] labels = ['Negative', 'Positive']
     count_negative = (y_pred == 'negative').sum()
     count_positive = (y_pred == 'positive').sum()
     sentiment_counts = [count_negative, count_positive]
```

```
[43] plt.figure(figsize=(8, 6))
     plt.bar(labels, sentiment_counts, color=['red', 'green'])
     plt.xlabel('Sentiment')
     plt.ylabel('Count')
     plt.title('Sentiment Analysis Results')
     plt.show()
```


[43]
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
Count
Negative
Positive
Sentiment

CHAPTER-5

SPECIFIC OUTCOMES

5.1 Technical Outcomes

- Know and get a basic overview of how a basic machine learning model is trained, tested, and used.
- Get familiar with the basic ML terms like classes, test data, train data, accuracy, training model.
- Know how to use teachable machines for basic ML projects and make simple models that can be exported/downloaded/copied and used in your apps, websites, and more.
- Have a general template of classification problem that you can use on for various datasets.
- To code and make your own classification model.
- To visualize your dataset and perform train test split.
- Understanding the Evaluation metrics and accuracy score.
- Understanding the hands-on implementation of AI and ML concepts theoretically learned.

5.2 Non-Technical Outcomes

- Communication skills
- Gaining work experience
- Attaining confidence
- Problem Solving skills.
- Exploration of a desired career path
- Effective email writing
- Telephone etiquettes

CHAPTER-6

FUTURE ENHANCEMENT

6.1 Scope of Future Improvement

With respect to the model training, enhancements can be made if models are trained,for the sentiment analysis application involve refining its capabilities to provide more accurate and nuanced results. Integrating state-of-the-art sentiment analysis models, such as BERT or GPT-based models, could significantly improve the application's understanding of context and subtle nuances in language, enhancing its overall performance. Additionally, expanding the dataset used for training the model to include domain-specific or user-generated content would make the analysis more tailored to specific contexts. The application's user interface can be enhanced to provide a more intuitive and interactive experience, allowing users to input multiple keywords, visualize sentiment trends over time, and receive detailed insights. Real-time data integration, possibly through web scraping or API integration, would keep the application updated with the latest trends and sentiments. Incorporating advanced natural language processing techniques for better text preprocessing, including stemming, lemmatization, and handling of emojis and slang, would contribute to more accurate sentiment analysis. Furthermore, supporting multiple languages would broaden the application's scope, making it more accessible and relevant in a global context. These enhancements collectively aim to elevate the application's performance, usability, and adaptability, positioning it as a robust tool for comprehensive sentiment analysis in various domains and scenarios.

CONCLUSION

Twitter sentiment analysis through machine learning techniques serves as a potent tool for understanding and harnessing public sentiment in the digital age. With the explosive growth of social media, Twitter has become a microcosm of real-time public opinion, making it a rich source of data for businesses, researchers, and policymakers. Through data collection, preprocessing, and the application of machine learning algorithms, sentiment analysis empowers us to gauge the prevailing emotions within the Twitter verse. Whether it's tracking
reactions to a product launch, monitoring public sentiment during a crisis, or assessing political discourse, sentiment analysis unveils trends and insights that can inform strategies and decision-making. However, it is crucial to recognize the challenges, including the need for representative datasets, addressing class imbalances, and accounting for the nuances of human language such as sarcasm and irony. Additionally, staying updated with the evolving linguistic landscape on Twitter is essential. Sentiment analysis is not limited to positive, negative, or neutral classifications. Subtler sentiments like anxiety, excitement, or frustration can also be detected, providing a deeper understanding of public sentiment. In an era where public perception can shape reputations, influence stock markets, and sway elections, Twitter sentiment analysis using machine learning is an indispensable tool for businesses, governments, and researchers alike. It enables us to navigate the vast sea of social media data, extracting actionable insights and helping us make more informed decisions in an ever-connected world.

REFERENCES

- Geeks For Geeks https://www.geeksforgeeks.org/machine-learning/
- Wikipedia https://en.m.wikipedia.org/wiki/Machine_learning
- Open Ai : https://chat.openai.com/c/d0496bac-1bf3-4840-a79a-db1d601a2eb4.
- JAVA POINT: https://www.javatpoint.com/machine-learnings