**INTERN PROJECT PHASE – 2**

**Data Analyst Project: Twitter Sentiment Analysis**

**Project - 3:** Twitter Sentiment Analysis

* **Python Code Explanation:**
  + In my Twitter Sentiment Analysis project, these are my project process steps.

**Steps:**

1) Importing Libraries & Packages

2) Importing Dataset

3) Preprocessing Text

4) Analysis of the Text

5) Splitting The Data

6) TF-IDF Vectorizer

7) Transforming Dataset

8) Create & Evaluate Model

9) Saving the Models & Using the model

**Step-1:**

* + I import the necessary libraries and packages for my project.
  + For example, I used Pickle to save the model, WordCloud to display the most highly positive and negative words, WordNetLemmatizer to get the root word, etc.

**Step-2:**

* + Then I import the Twitter sentiment analysis dataset and define the predefined columns for the Twitter sentiment analysis dataset.
  + After that, I extract the ‘Sentiment’ and ‘Text’ columns for my model development.
  + Then I replace the value ‘4’ with ‘1’ in the ‘Sentiment’ column.

**Step-3:**

I used several steps to preprocess the ‘Text’ column data in the Preprocessing of the Text.

**The preprocessing steps taken are:**

1. Lowercasing: Each text is converted to lowercase.

2. Replacing URLs: Links starting with "http" or "https" or "www" are replaced by "URL".

3. Replacing Emojis: Emojis are replaced using a predefined dictionary containing emojis along with their meanings. (e.g., ":)" to "EMOJIsmile")

4. Replacing Usernames: Usernames starting with '@' are replaced with the word "USER". (e.g., "@Kaggle" to "USER")

5. Removing Non-Alphabets: Characters except digits and alphabets are replaced with a space.

6. Removing Consecutive Letters: Three or more consecutive letters are replaced by two letters. (e.g., "Heyyyy" to "Heyy")

7. Removing Short Words: Words with a length less than 2 are removed.

8. Removing Stopwords: Stopwords, which are English words that do not add much meaning to a sentence, are removed. (e.g., "the", "he", "have")

9. Lemmatizing: Lemmatization is the process of converting a word to its base form. (e.g., “Great” to “Good”)

**Step-4:**

After preprocessing, I analyze the data using word clouds to display the most frequent positive and negative tweets from the Twitter dataset.

**Step-5:**

* + Then, we split the train and test data with an 80:20 ratio.
  + After that, I used ‘TfidfVectorizer’ to convert text documents into vectors based on the word's relevance.

**Step-6:**

After splitting the data, I transformed the data into ‘x\_train’ and ‘x\_test’ data.

**Step-7:**

* + Then, I defined a common function for all machine learning models to predict, display the classification report, provide the accuracy score, and display the confusion matrix plot.

**I used four machine-learning models:**

1. Naive Bayes - MultinomialNB, Bernoulli

2. Support Vector Machines - LinearSVC

3. Logistic Regression

* + In evaluating and predicting the sentiment using these four machine learning models, each model gives the following accuracies: MultinomialNB gives 79.54, BernoulliNB gives 79.6, LinearSVC gives 80.06, and Logistic Regression gives 81.26.

**Step-8:**

* + After developing and evaluating the machine learning models, I saved the vectorizer and Logistic Regression model into a pickle file for using the models to predict sentiment analysis.
  + Logistic regression provides the highest accuracy compared to the other three machine learning models.

**Step-9:**

Then, I load the vectorizer and Logistic Regression model for sentiment analysis of new texts. I check whether my model predicts and gives the correct sentiment analysis.

**SCREENSHOT:**

* **Sentiment Column count bar plot:**

**A graph with blue bars

Description automatically generated**

* **A close up of words

  Description automatically generatedWordCloud Plot of Most frequent Negative and Positive words:**

**Positive Word Cloud Plot**

**A close up of words

Description automatically generated**

**Negative Word Cloud Plot**

**A blue squares with white text

Description automatically generated**

* **All Machine Learning models Confusion Matrix:**

**A blue squares with white text

Description automatically generatedA blue squares with white text

Description automatically generated A blue squares with white text

Description automatically generated**