Summary of Approach and Learnings

1. Project Overview:

The **Social Media Sentiment Analyzer** leverages **Streamlit**, **TextBlob**, and **Pandas** to classify tweets into **Positive**, **Neutral**, or **Negative** sentiments. Users upload CSV files containing tweets, and the app performs sentiment analysis, visualizes sentiment distribution in a bar chart, and displays example tweets for each sentiment category.

2. Approach:

- **Data Processing**: We used **Pandas** to handle CSV file uploads and process tweets.
- **Sentiment Analysis**: The sentiment of each tweet is analyzed using **TextBlob**, which computes polarity scores. The thresholds for classifying sentiments were adjusted:
    - **Positive**: Polarity > 0.05
    - **Negative**: Polarity < -0.05
    - **Neutral**: Polarity between -0.05 and 0.05.
- **Visualization**: The app generates a **bar chart** of sentiment distribution using **Matplotlib**, and shows example tweets for each sentiment category.

3. Learnings:

- **Sentiment Classification**: We learned how to fine-tune polarity thresholds for more accurate neutral sentiment detection.
- **Streamlit & Visualization**: We gained experience in building interactive web apps and visualizing sentiment data effectively.
- **Unit Testing**: We wrote unit tests using **unittest** to ensure the sentiment analysis function works correctly across edge cases and common inputs.

4. Challenges and Solutions:

- **Neutral Sentiment**: Initially, neutral tweets were misclassified as positive. We adjusted polarity thresholds to ensure better neutral classification.
- **File Validation**: We ensured proper error handling when the uploaded file didn’t meet the expected format.