**Sentiment Analysis Web Application Documentation**

**Overview**

The Sentiment Analysis Web Application is a Python-based tool that applies natural language processing (NLP) to determine the sentiment of tweets. The application uses a machine learning model trained on tweet data to classify sentiment as positive, negative, or neutral. The project features a Streamlit-powered web interface for easy interaction.

**Project Structure**

The project is organized into several directories and files, each serving a specific purpose in the application's ecosystem:

- `app/`: Contains the Streamlit web application.

- `\_\_init\_\_.py`: Identifies the directory as a Python module.

- `streamlit\_app.py`: The main script for the Streamlit web application.

- `data/`: Stores the datasets used for training and testing the model.

- `train.csv`: The training dataset.

- `test.csv`: The testing dataset.

- `models/`: Houses the trained machine learning model and vectorizer.

- `sentiment\_model.pkl`: The serialized machine learning model.

- `tfidf\_vectorizer.pkl`: The serialized TF-IDF vectorizer.

- `notebooks/`: Contains Jupyter notebooks used for exploratory analysis and model training.

- `model\_development.ipynb`: The notebook for model development and evaluation.

- `src/`: Includes the source code for data preprocessing, feature extraction, and model training.

- `\_\_init\_\_.py`: Marks the directory as a Python module.

- `data\_preprocessing.py`: Contains functions for data cleaning and preprocessing.

- `feature\_extraction.py`: Implements the feature extraction process using TF-IDF.

- `train\_model.py`: Script for training the sentiment analysis model.

- `tests/`: Contains unit tests for the application.

- `test\_preprocessing.py`: Provides test cases for data preprocessing.

- `.gitignore`: Lists files and directories that should be ignored by Git.

- `LICENSE`: The license file for the project.

- `README.md`: Offers a detailed project overview, setup, and usage instructions.

- `requirements.txt`: Lists the Python dependencies required for the project.

**Setup and Installation**

To set up the project, follow these steps:

1. Clone the repository to your local machine.

🡪 git clone <https://github.com/saimaharana4/sentiment-analysis-webapp.git>

1. Navigate to the project directory.

🡪 cd sentiment-analysis-webapp

1. Create and activate a virtual environment.

🡪 venv\Scripts\activate

1. Install the required dependencies from `requirements.txt`.

🡪 pip install -r requirements.txt

1. Train the machine learning model using `train\_model.py`.

🡪 python src/ train\_model.py

1. Run the Streamlit web application with the `streamlit\_app.py` script.

🡪 SET PYTHONPATH=%PYTHONPATH%;D:\sentiment-analysis-webapp

🡪 streamlit run app/streamlit\_app.py

**Usage Instructions**

To use the web application:

1. Start the Streamlit server by running `streamlit run app/streamlit\_app.py`.

2. Open the provided local URL in your web browser.

3. Enter the text of a tweet into the input field on the web page.

4. Click the "Predict Sentiment" button to classify the sentiment of the tweet.

**Maintenance and Testing**

To maintain and test the application:

- Update the `requirements.txt` file as dependencies evolve.

- Re-train the model periodically with new data to improve accuracy.