**Design Document: Twitter Sentiment Classification Project**

**Overview**

The Twitter Sentiment Classification Project is a machine learning-based application designed to classify tweets into sentiment categories: Negative, Neutral, Positive, and Irrelevant. Using a pre trained Logistic Regression model with a TF-IDF feature extractor, the project achieves an accuracy of 80% on the validation dataset.

The project involves the following components:

1. Data preprocessing to clean and standardize the tweets.
2. Feature extraction to convert text data into numerical representations using TF-IDF.
3. A Logistic Regression model for sentiment classification.
4. An interactive interface to classify new user-provided sentences.

**Purpose**

The purpose of this project is to:

* Develop a pipeline for sentiment analysis of text data.
* Provide a practical example of machine learning applications in natural language processing (NLP).
* Build an interactive tool that allows users to input sentences and obtain sentiment predictions in real-time.

**System Architecture**

**Input**

The input consists of a CSV file containing tweet data with the following columns:

* TweetID: A unique identifier for each tweet.
* Entity: The entity associated with the tweet (optional).
* Sentiment: The sentiment label for each tweet (e.g., Negative, Neutral, Positive, Irrelevant).
* Message: The text content of the tweet.

**Output**

The output is a predicted sentiment for user-provided text or sentences from the dataset.

**Key Components**

1. **Data Preprocessing**:
   * Noise removal (e.g., URLs, mentions, hashtags, and special characters).
   * Conversion of text to lowercase for consistency.
   * Removal of stopwords (common words like "the" and "is" that don’t contribute to sentiment analysis).
   * Lemmatization to reduce words to their base forms.
2. **Feature Extraction**:
   * Text data is converted into numerical vectors using TF-IDF (Term Frequency-Inverse Document Frequency).
   * This process captures the importance of terms relative to their frequency in the dataset.
3. **Model Training**:
   * A Logistic Regression model is trained using the TF-IDF features and sentiment labels.
   * The model learns to associate specific words or patterns with sentiment categories.
4. **Interactive Interface**:
   * Users can input sentences through an interactive interface.
   * The system preprocesses the input, converts it to a numerical vector, and predicts its sentiment using the trained model.

**Key Functions**

**Preprocessing**

The preprocessing pipeline standardizes tweets by removing irrelevant content, normalizing case, and reducing word variations. This step ensures the data is clean and ready for feature extraction.

**TF-IDF Feature Extraction**

TF-IDF transforms cleaned tweets into numerical representations that capture the importance of terms relative to the dataset. This step ensures the model focuses on meaningful patterns in the text.

**Logistic Regression Model**

The Logistic Regression model is trained on the TF-IDF features to classify tweets into one of the four sentiment categories. It uses learned weights for each term to predict the probability of each sentiment.

**Interactive Interface**

The interface allows users to input sentences and receive sentiment predictions in real time. It preprocesses the input text, applies TF-IDF transformation, and uses the Logistic Regression model to generate predictions.