Choice of Data Sources

- **Relevance to the Task:** The selected sources contain a mix of Greeklish and English text, making them suitable for training a classifier to distinguish between the two languages.
- **Availability & Accessibility:** The chosen websites allow easy access to text data, either through open forums, blogs, or discussions, ensuring a steady flow of content.
- **Diversity of Sources:** Gathering data from different platforms helps create a well-rounded dataset and prevents bias.

Data Scraping Methods and Preprocessing Steps

Data Scraping Methods

1. **Web Scraping with Python:**
    - Used libraries like requests, BeautifulSoup, and Scrapy to extract text data from selected websites.
    - Focused on retrieving relevant content such as forum posts, article text, and user comments.
2. **Reddit API for User Feeds:**
    - Collected Greeklish text from personal Reddit feeds using the praw library.
    - Extracted post titles, comments, and descriptions for better text diversity.

Preprocessing Steps

1. **Text Cleaning:**
    - Removed URLs, special characters, and unnecessary spaces.
    - Standardized text to lowercase for uniformity.
2. **Tokenization:**
    - Converted sentences into sequences of words using Tokenizer from Keras for deep learning models.
    - Applied TfidfVectorizer for traditional ML models to convert text into numerical features.
3. **Padding**

- Ensured all text samples had a uniform length using pad_sequences in deep learning models.

4. **Augmentation for Balance:**
    - Used word shuffling and sentence reordering to create more training samples.
    - Ensured both Greeklish and English classes had equal representation.

These steps helped in cleaning raw text data and transforming it into a format suitable for model training.

Rationale for Model Selection, Training Process, and Evaluation

1. Model Selection

We used two different approaches for classification:

1. **LSTM (Long Short-Term Memory) with Word Embeddings**
    - LSTM is well-suited for handling sequential text data, making it effective for distinguishing Greeklish from English.
    - The **Embedding layer** was used to automatically learn word representations, improving classification performance.
    - **Batch Normalization & Dropout** were added to stabilize training and prevent overfitting.
2. **Logistic Regression with TF-IDF**
    - TF-IDF (Term Frequency-Inverse Document Frequency) was used to convert text into numerical vectors based on word importance.
    - Logistic Regression was chosen as a lightweight, efficient baseline model to compare performance against LSTM.
    - This model is computationally efficient and interpretable, making it a strong choice for text classification.

2. Training Process

- **LSTM Model:**

- **Data Preparation:** Tokenization and padding were applied to ensure consistent input size.
- **Hyperparameters:**
  - **Embedding Dimension:** 128
  - **LSTM Layers:** Two stacked layers (128 and 64 units)
  - **Dropout Rate:** 30%
  - **Batch Size:** 32
  - **Epochs:** 25
  - **Optimizer:** Adam
  - **Loss Function:** Binary Crossentropy
- The model was trained using 80% of the data, with 20% reserved for validation.

- **Logistic Regression Model:**
  - TF-IDF was applied to convert text data into numerical form.
  - The model was trained on 80% of the data, with the remaining 20% used for testing.

3. Model Evaluation

- **LSTM Model:**
  - **Accuracy Score:** Tracked during training and validation.
  - **Loss Curve Analysis:** Checked for signs of overfitting.
  - **Classification Report:** Evaluated precision, recall, and F1-score.(given in notebook)
- **Logistic Regression Model:**
  - **Accuracy Score:** Compared against LSTM to measure effectiveness.
  - **Precision & Recall:** Assessed the model's ability to distinguish between Greeklish and English.
  - **F1-Score:** Ensured a balanced evaluation of performance. .(given in notebook)

-

Using both models allowed us to assess whether a deep learning approach (LSTM) provided significant improvements over a traditional machine learning approach (Logistic Regression).

Challenges Faced and Solutions Provided

1. Challenge: Data Scraping and Collection

- **Issue:** Finding high-quality Greeklish and English text data was difficult since most sources primarily contain standard Greek or English.
- **Solution:** I manually identified diverse sources such as forums, blogs, and news sites where Greeklish is commonly used. Additionally, I ensured that the scraped data had a balance between Greeklish and English to avoid bias.