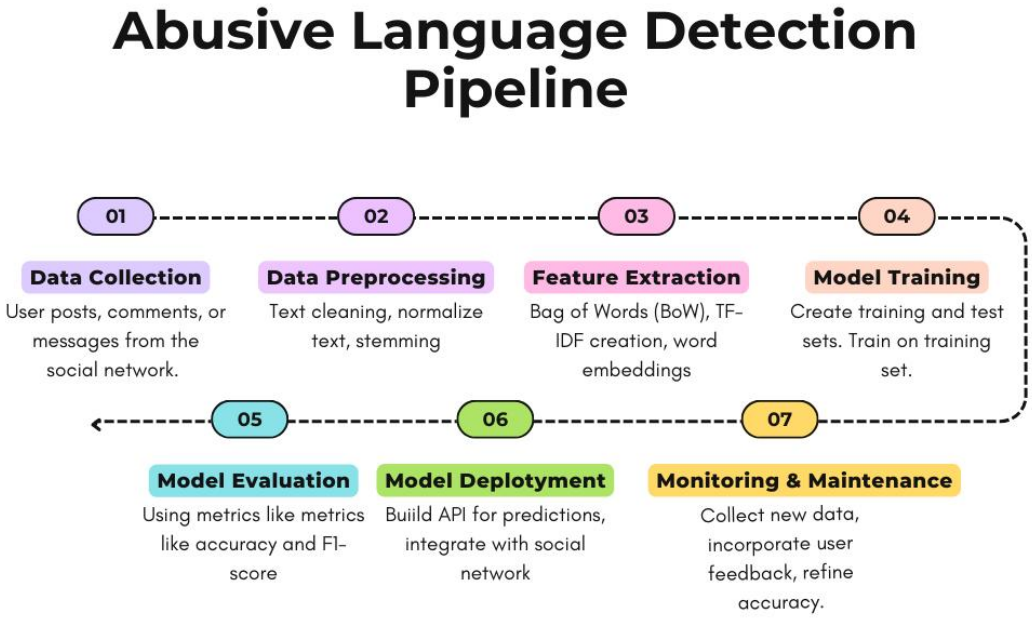
Question 1: Describe a pipeline for detecting abusive language on a social network. You may use diagrams. (Your response must not exceed one page.)

Answer:

- The first step **is Data Collection,** a dataset of posts with both abusive and non-abusive examples is collected.
- The dataset is then **preprocessed,** the text is cleaned, and some of the preprocessing steps include (removing special characters, tokenizing, lowercase, and removing stop words), and then stemming is applied.
- After data preprocessing, the next step is **Feature Extraction**, a matrix of token counts using Bag of Words and TF-IDF is created, and word embeddings are used for better representations.
- The extracted features are then used for **Model Training,** the features are into test, training, and evaluation subsets, and then an ML model is trained on the training set.
- The trained model is tested in **Model Evaluation**, the model performance is evaluated with metrics like accuracy and F1-score, and errors using a confusion matrix.
- The model is then deployed in Model **Deployment,** an API for predictions is built and integrated with the social network for real-time monitoring of abusive content.
- The final step is **Monitoring and Maintenance:** the model is maintained with new data and incorporates user feedback to refine its accuracy.



Question 2: Explain the process of using a hybrid CNN-LSTM algorithm to detect aggressive content in a post.

Answer:

A process of combining both the Convolutional Neural Networks (CNNs) and Long Short-Term Memory networks (LSTMs) for detecting aggressive content in posts takes advantage of the strength of both architecture. The CNN is used for feature extraction as it is able to capture several patterns in data like spatial patterns, while the LSTMs would be used for sequence modelling.

The high-level overview of the pipeline includes:

1. **Text Preprocessing:** the text is cleaned by removing special characters, URLs, and whitespace; then the text is Tokenized and converted to lowercase. stop words are removed and stemming is applied.
2. **Embedding Layer:** the text is converted into word embeddings using pre-trained embeddings like Word2Vec.3.CNN Layer: the convolutional layers are then used to extract local features and patterns from the word embeddings, and pooling layers are used for dimensionality reduction and to capture the most important features.
3. **LSTM Layer:** the output of the CNN layers is passed into LSTM layers to capture long-term dependencies and contextual information in the sequence of words.
4. **Fully Connected Layer:** the output of the LSTM layers is then passed through fully connected layers to make the final classification.
5. **Output Layer:** In this layer, the sigmoid activation function is implemented to output the probability of the post being aggressive or non-aggressive as this is a binary classification task.

Question 3: Using Python 3, implement a Facebook and/or Instagram connector to collect posts (images, text, and comments related to the images) based on a defined topic, for example, "the death of President Jacques Chirac."
It is recommended to store the text and images in a MongoDB database.

Answer:
I have answered this question and explained the steps I followed to arrive at the solution I got on a Jupyter Notebook which is available on my Github at this link:

https://github.com/Aliamichael/Facebook_connector