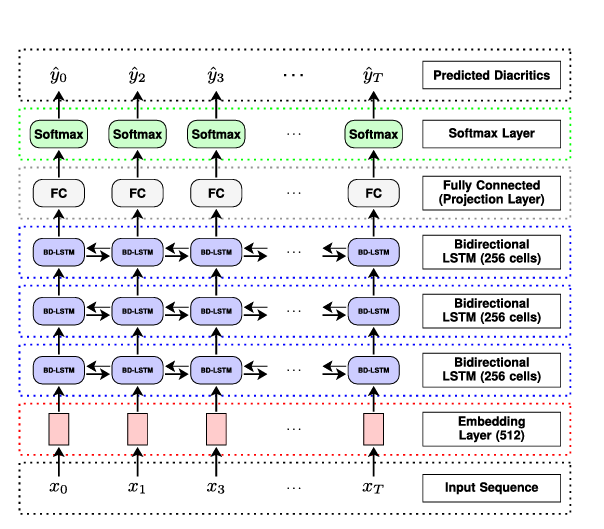
**NLP Project**

**Arabic-text-diacritization**

|  |  |
| --- | --- |
| **Team Members** | |
| **Name** | **Email** |
| **Abdelaziz Salah Mohammed Abdo** | **Abdelaziz**[**132001@gmail.com**](mailto:132001@gmail.com) |
| **Abdelrahman fathi** |  |
| **Khaled Hesham Elsayed Taha** |  |
| **Kirollos Samy** |  |

1. **Project Pipeline:**
2. **Detailed Description of each phase:**

**2.1- Data preprocessing:**

* The data preprocessing can be divided into two stages:

1. Data Cleaning: The corpus undergoes a meticulous cleaning process, where all characters are retained except for Arabic letters, diacritics, and punctuation. This step is crucial as it selectively preserves characters contributing to diacritization, essential for the models to learn. During the inference stage, our designed algorithms proficiently restore the removed characters, placing them back in their correct positions.
2. Data Splitting: This phase involves segmenting the corpus into sentences. The absence of stringent punctuation rules in Arabic poses a challenge in identifying sentence boundaries, especially when compared to languages like English[40]. Arabic sentences can be linked with coordinators such as (w)wa and (f) fa, merging with the first word in the subsequent sentence. Further complicating matters is the fact that many Arabic words can incorporate these coordinators as their initial letter, posing a challenge to text segmentation.
3. Adding to the complexity, the primary indicator of sentence endings in Arabic is a period, akin to many other languages. However, relying solely on a period would result in excessively long sentences, as Arabic writings, particularly classical ones, often link sentences with coordinators, using a period only at the end of each paragraph. To address this, the corpus is split using multiple delimiters to generate a comprehensive set of sentences. For computational efficiency, we opt for sentences with lengths at most 600 characters, a choice informed by experiments with various models indicating that a larger length necessitates a smaller batch size (< 16).
4. To execute the corpus split, we employ a period, generating numerous sentences, some exceeding 600 characters. Sentences with lengths under 601 are directly employed in training, while others undergo additional processing in the subsequent step. An iterative process ensues for sentences larger than 600 characters, involving a split on punctuation, commencing with ",", followed by ";", and then ":". Any sentences exceeding 600 characters after these steps are excluded from the training data.

**2.2- Feature extraction:**

**2.3- Model training:**

1. **Evaluation:**
2. **Used Model in the test set submission on Kaggle:**