Homework 2:
Building BabyGPT: Language Generation Models

In this homework, you will create your own word-level language model to generate texts. You will then measure the quality of your language model by assessing the complexity of the language used by the model. You will be using the text from 'el principito.txt' for this homework.

1 Recap: Language Models

A language model is a probability function p that assigns probabilities to sequences of symbols — such as words or characters. For example, consider the sequence $S = [' i', ' love', ' Buenos', ' Aires']$. Let's say you turned on the radio at an arbitrary moment and heard someone saying 'I love Buenos'. Your language model will calculate the probability of every word $w \in V$ in your vocabulary (V) and find the word with the highest probability given the preceding sequence 'I love Buenos'. If the probability $P(' Aires')$ is the highest, it will select it as the next word in the sequence.

More formally, we can write this as $p(w_i|w_{i-1}, w_{i-2}, ..., w_1)$, where w_i is the word 'Aires'. For a 3-gram model, this can be written as $p(w_i|w_{i-1}, w_{i-2})$. In this homework, you will first need to create a language model and then use it to generate a sequence of 200 words.

2 Creating a 3-gram Language Model

Given the input text file 'el principito.txt', you should use a Python dictionary to calculate and store $p(w_i|w_{i-1}, w_{i-2})$. In practice, this involves first keeping a count of w_i given w_{i-1} and w_{i-2}, and then using that to calculate the probabilities of each w_i given w_{i-1} and w_{i-2}.

HINT: Use nested dictionaries to keep track of counts and probabilities.

3 Generating the Sequence

Now that you have a Language Model, you can start generating sequences. Your generated text should have as many words as the original text. It should start with the words 'el', 'principito'. You will use a maximum-likelihood estimate

(MLE) to select the next word in the sequence. This simply means that given any w_{i-1} and w_{i-2}, you will pick the word with the highest probability as w_i.

4 Evaluation

Calculate the Flesch-Kincaid Index (FKI) scores of:

- The text generated by this language model
- The text generated by the character-level language model from class (You can find this in your homework 2 folder)
- The original text

Compare the FKI scores of your language models to the original text. How do they differ?

5 Submission

For your final submission, you should submit a Colab notebook (or a .py file) with your work. Additionally, submit a Word document with your observations.