Simple Poetry Generator

In this assignment, you will construct your own language model and use it to build a simple poetry generator.

The language model is based on small poetry corpus from wellknown Indonesian poets (Khairil Anwar and Sapardi Djoko Damono).

You will need to:

1. Build the unigram model. (score: 15)

2. Build the bigram model. (score: 20)

3. Evaluate the model using perplexity (score: 20). The sentence test set is provided. (Bonus: if you implement a technique to handle the unseen words, so that the probability value would not be zero).

4. Build a poetry generator, that consist of several sentences (35). You may choose how many sentences on your "constructed" poetry, e.g = 5 or 10 sentence.

A whole sentence is generated based on the first word. You can use the random function on choosing the first word. You can set the number of words in a sentence also.

Notes on building the language model:

1. Punctuations such as comma, period, etc were discarded, no need to store it on the model vocabulary.

2. Since all words in a poet have close meaning, on building the bigram model, we also consider the word chain from last word in a sentence and the first word in the next sentence.

Example:

First sentence: Tak ada yang lebih tabah

Second sentence: Dari hujan bulan Juni

You need to build the (tabah, dari) record.

Submit your code on the Github classroom assignment repository. Please use jupyter/ipython notebook if you already master it.

Please add a short conclusion after completing and testing the program, based on the following questions (score: 20):

1. What are the top 10 unigram words having highest probability?

2. What are the top 10 bigrams words having highest probability?

3. What do you think of the perplexity comparison between the unigram and bigram model?

4. Give the example of your generated poetry based on the unigram and the bigram model, and give your comment! (e.g: is it believable as a poetry? :D).

References:

[1] https://github.com/neubig/nlptutorial/blob/master/download/01-unigramlm/nlp-programming-en-01-unigramlm.pdf

[2] https://github.com/neubig/nlptutorial/blob/master/download/02-bigramlm/nlp-programming-en-02-bigramlm.pdf