Homework #5

1. (20 pt) Starting from the LM code discussed in class, build a 4-gram, 5-gram, 6-gram language model and test them.
    a. How does performance change as you increase *n*? Can you make *qualitative* and *quantitative* judgments?
    b. How does training time / model size change as you increase *n*?

    You can use a package and/or do part of the exercise coding from scratch.

2. (40 pts) Consider again Markov language models, but now instead of building n-grams with *words,* we build n-grams with *chars*: for example, with n=2, we have that “butter and” becomes: “START b, b u, u t, t e, e r, r SPACE, SPACE a, a n, d STOP”.

    Use nltk or refactor the code in the notebook to build a char-based language model in a dataset of choice. Build a LM with n=2, 3 and 4, show your work and reason about the following questions:

    - When you use the LM as a generative tool, is the text produced by these char models coherent?
    - What is a sensible pre-processing for this type of LM? (hint: what do we do with numbers? How do we treat consecutive spaces?)
    - Is the estimation problem easier or harder compared to the word version? (hint: is data more sparse with words or char?) What are the advantages and disadvantages of char-based models?