HMM and POSTagging Assignment: My Own POSTagger

Due date: Saturday, March 3rd 2018, 23:59

In this assignment, you are asked to build your own POSTagger using several methods that we learned in class. This time, you will build Indonesian POSTagger, with universal dependency (UD) POSTag label. For the dataset, please download annotated Bahasa Indonesia treebank from this link: https://github.com/UniversalDependencies/UD_Indonesian-GSD/blob/master/id-ud-train.conllu

You are required to use first 100 sentences, split it randomly 90 sentences for training and 10 sentences for testing. However, at the end, you could improve your experiment by using the whole sentences in the dataset, which will give you bonus score. Please do this assignment using ipython/jupyter notebook, so we could trace the program easily.

You may use any method that we already discuss in class to build your POSTagger. You can choose the simplest one first and then move to more complex method. I will divide the steps you need to do into several parts:

No	Step	Score
1	Create the vocabulary, frequency, and tag information table.	10
2	Create the probability of a word labeled with a tag (emission probability) table.	20
3	Create a POSTagger based on highest emission probability table.	20
4	Create a tag transition probability table (bigram model, only record one previous tag)	20
5	**Bonus**: Using the whole sentences in the dataset	10
6	Create a bigram POSTagger using Viterbi method	30
7	**Bonus**: applying smoothing technique	10
8	**Bonus**: applying OOV handling technique such as by defining rules based on affix.	10
9	**Bonus**: evaluating the POSTagger using precision, recall, and F-measure metrics *). Calculate the precision, recall, and F-measure for each tag. You can see a simple explanation about precision, recall, and F-measure from:	10

*)brief information about precision, recall, and F-measure:

https://hpi.de/fileadmin/user_upload/fachgebiete/plattner/teaching/NaturalLanguageProcessing/NLP2015/NLP_Exercise2.pdf