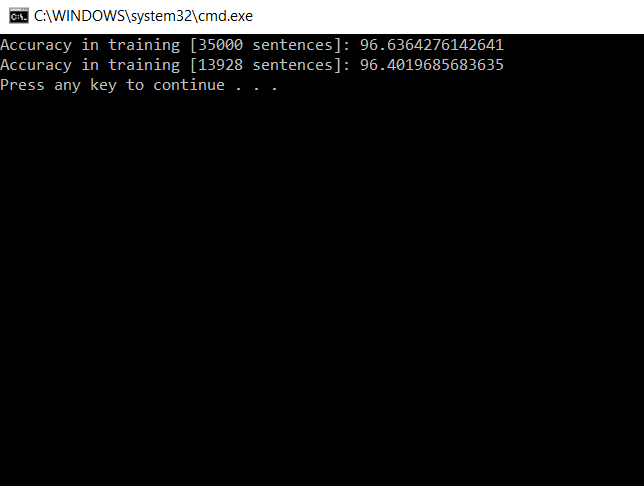
NATURAL LANGUAGE PROCESSING – CSCE 5290

ASSIGNMENT 3

ZAKIR SHAIK – 11145367

2. I have created a model where I have defined a map of map known as individual\_count where word – tag increments are done. In prediction, for each sentence, for each token I have compared the individual word-tag count with that of maximum value. If the value is more then that tag is assigned to the current token. If unknown words are found, “NN” is assigned. I got 96.6% accuracy for training and 96.4% for testing. It is taking 6-8 seconds for execution.

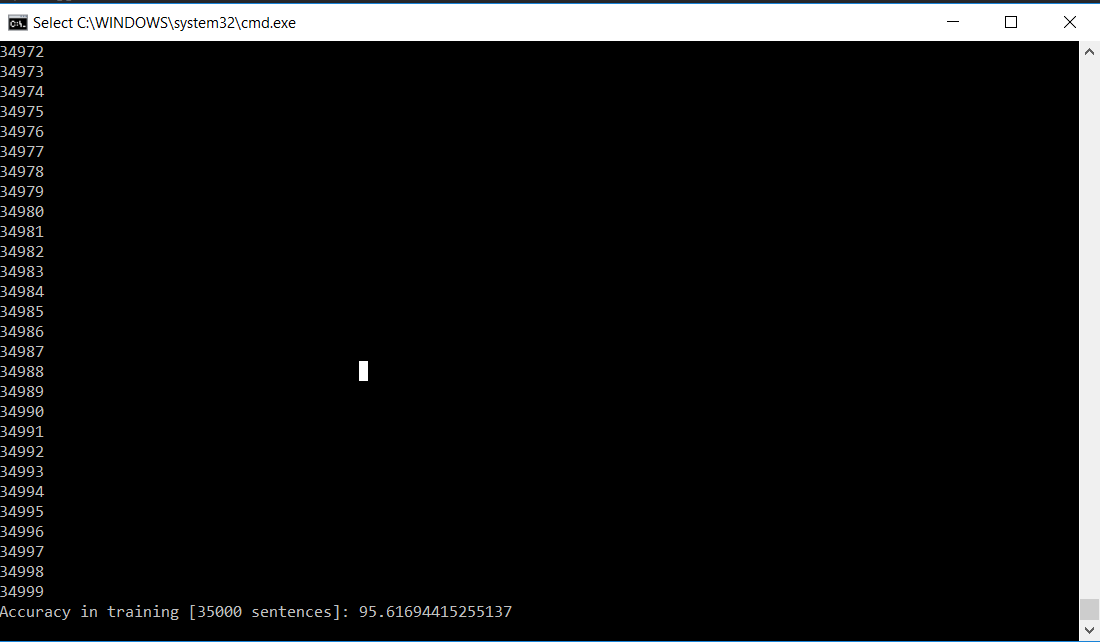
To run the code just open the solution file in visual studio and click ctrl+F5 as I have already given the arguments in properties. The same is with question 3.



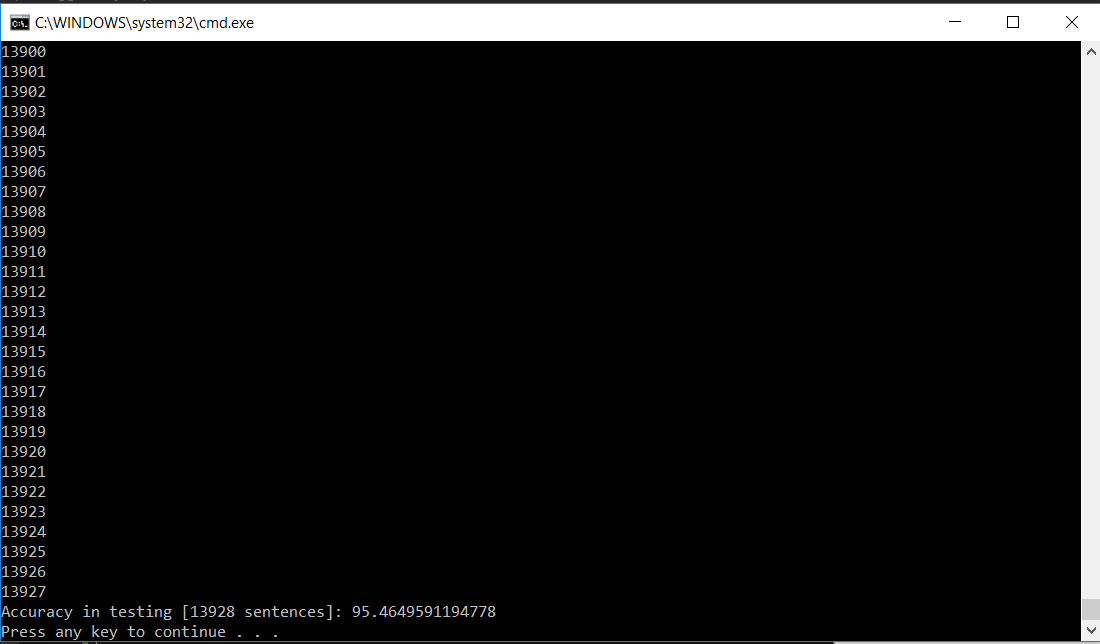
3. For bigram HMM Viterbi POS tagging, In modeling, I have defined three maps transition, individual\_count and likelihood which holds the prior, individual count of tags and the likelihood values. Then for each sentence, I have assigned the starting tag as ‘<s>’ and then I have ended the sentence with </s> tag. I have calculated probability values for both prior and likelihood by dividing total count by individual count.

In predict\_tags function, I have written the edge case first where the first token in sentence is dealt with. I have also defined statements which will assign tags for unseen words. After the tags are assigned, probability of word|tag is found and I have updated the likelihood values.

Then I have calculated the Viterbi matrix and updated the cells for each of the possible tags. Finally the code is taking 6 minutes to run for training set with an accuracy of 95.6%. Here the numbers are the sentence numbers ie. Sentence count starting from 0.



For testing I have got an accuracy of 95.4% with an average execution time of 2 seconds.



While executing code I have got many errors like Divide by zero where I have smoothed. Also I have done laplace smoothing for the transition probabilities.