Natural Language Processing

P-2 Part A Report

Roll No - 20161072

Implement a HMM using the Brown Corpus

- ### Calculating Transition and Emission Probabilities

  Transistion prob. (State_from - > State_to) can be computed by making bigrams and trigrams models of tags given along the sentence in Brown Corpus.
  And, for Emission prob. We have to make count of word given a tag. That we can easily compute from given Brown Corpus (may use any approach for this).
  And, State Prob. can be computed from unigram model of tags.

- ### Probability of the Observation Sequence

  Now, Transistion, State, Emission Prob. are available
  When a sequence is given for Evaluation (i.e Computing Probability of sequence), then we can either use Forward or backward procedure to calculate value of alpha and beta resp. Which further used in computing probability of given sequence.

- ### Viterbi Algorithm

  Given a sequence of words it will return the most likely state sequence. Work similar to dynamic programming by max. the Posteriori Probability.
  Also additive smoothing can be done for avoiding Zero Probabilities, this can be done by updating the State, Emission and Transition Prob. with best values using alpha and beta calculated above using Forward and Backward Procedure.

Observations

- ### Difficulties Faced

  As the length of observation sequence increases and the number of observations increases, the Prob. of State, Emission and Transition tends to zero.
  So, I tried handling them using scaling and Smoothing(not implemented).
  Log probabilities in this case will not work and I tried it out.

- ### Output Analysis

  After running certain number of iterations for training the HMM on the entire brown corpus, I got the Emission Prob. Many of the top 10 emitted words for each tag are same. The top 10 words for each tag get varied, as we run the algorithm again & again or you can say i.e a 10 fold validation. Such pattern could also be explained on the basis of transition probabilities between different tags as in the case of POS tagging also. Because the occurrence of one tag is dependent on the other tags too which is incorporated in Transition Prob.