Hidden Markov Models

In this project, I implemented Hidden Markov Models (HMM) and applied HMM to Part-of-Speech Tagging problem.

Firstly, I implemented various steps involved in learning HMM by breaking it down into two procedures:

1. **The Evaluation Problem**: Given HMM Model and a sequence of observations, finding the probability that the observations are generated by the model?

   Two algorithms are usually used for the evaluation problem: forward algorithm or the backward algorithm. Based on the result of forward algorithm and backward algorithm, we calculated probability of sequence and posterior probability of state.

2. **The Decoding Problem**: Given a model and a sequence of observations, finding the most likely state sequence in the model which produced the observation sequence.

   For decoding problem, we used Viterbi algorithm.

Then I calculated HMM parameters from the data and used it to solve Part-of-Speech Tagging problem.

HMM Class

Here, I have abstracted Hidden Markov Model as a class. Each Hidden Markov Model initialized with Pi, A, B, obs_dict and state_dict. HMM class has 6 inner functions:

- forward function
- backward function
- sequence_prob function
- posterior_prob function
- likelihood_prob function (likelihood of transition from state s at time t to state s' at time $t+1$)
- Viterbi function (dynamic programming algorithm for finding the most likely sequence of hidden states)

Application to Speech Tagging

Part-of-Speech (POS) is a category of words which have similar grammatical properties. (Example: noun, verb, adjective, adverb, pronoun, preposition, conjunction, interjection, and some numerical, article, or determiner.)

Part-of-Speech Tagging (POST) is the process of marking up a word in a text (corpus) as corresponding to a particular part of speech, based on both its definition and its context.

Here I used HMM to do POST by calculating the parameters (pi, A, B, obs_dict, state_dict) of HMM first and then apply Viterbi algorithm to do speech-tagging. (I have assumed that the emission probability from each state to a new unseen word is 10^{-6}(a very low probability)).

After successful implementation of HMM class, I was able to get an accuracy of 78% for Part-of-Speech Tagging.