CS5012: Language and Computation

Practical 1: POS tagging with HMMs

Student No: 180008512

1. Introduction

The aim of this practical is to develop a first-order HMM (Hidden Markov Model) for POS (part-of-speech) tagging in Python, as well as the implementation of three different algorithms for POS tagging.

The features that where implemented are:

1. Algorithm 1 (Viterbi Algorithm)
2. Algorithm 2 (Eager Algorithm and Beam Search)
3. Implementation of code to compute the accuracy of the different algorithms.

2. Training and Testing data

The brown corpus was used. As it was recommended, the training data where the first 10,000 sentences of the corpus, and the following 500 sentences where used as the testing data to reduce the processing time.

It was contemplated to have randomised testing data, of sentences that where not used within the training of the HMM; however, the accuracy of the different algorithms changed as the testing data was different when running the algorithm each time, which made it difficult to compare both algorithms accuracy. Hence, a fixed testing dataset was used instead.

3. Testing and Results

The Viterbi algorithm, using the first 10,000 sentences as training data and the next 500 sentences as testing data, achieves an accuracy of 98%. The Viterbi algorithm achieves such a high accuracy mainly due to the thorough exploration of the different possible tag sequences for a sentence to find the best tag sequence for that sentence. On the contrary, the performance of algorithm 1 with beam search is based on the beam width. In this case, beam width of 1 achieves an accuracy of 94.2%, and grows while the beam width (k) grows, it achieves higher accuracy until it performs a whole Viterbi, which gets the accuracy of the same.

4. How to Run the Program

A README.txt file is provided within the zip submitted which also contains the instructions on how to run it.

To run the program, export the file `hmm.py` to a desired directory, and within a command line, navigate to that directory. Then the file can be executed normally: `py hmm.py`.

Once the program is running, please follow the instructions provided by the "`MAIN MENU`" to run the two algorithms implemented and to exit the program.