Report for HW3

CS 519 Natural Language Processing

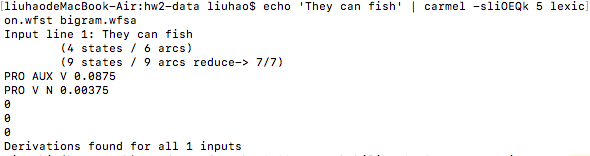
Prof. Liang Huang

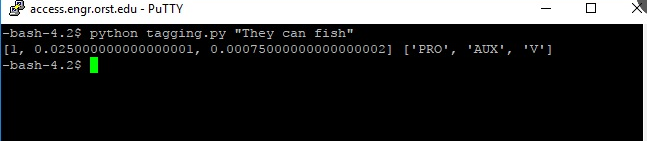
Group member: Gune.Aditya, Hao.Liu, Ke.Huang

# Part1

Here is one result from the bigram-tagging.

We didn’t modify the format but the result is exactly the same comparing to Carmel.





# Part2

Algorithm:

1. First, we store .wfsa and .wfst in 2 dictionary

2. Then we using Viterbi algorithm to find best path from state (k-1) to state (k)

3. Then back track the result as the output

The sub-problems in this question and recurrence relations are like this:

• For k = 1 . . . n,

– For u ∈ K, v ∈ K,

π(k, u, v) = max[w∈K](π(k − 1, w, u) × q(v|w, u) × e(xk|v))

• Return max[u∈K,v∈K] (π(n, u, v) × q(STOP|u, v))

For a given string length n, for all different kinds of English phonemes K, we have K^2 original state, and each state can have K links. So the **time complexity** of this algorithm is O(n). And the **space complexity** is O(n) = O(n)

# Part 3

In part 3 I believe we should use heap to store our value. With the modification of the heap, we can left some results with largest possibility. Limited by time, I regret that I haven’t finish this part.