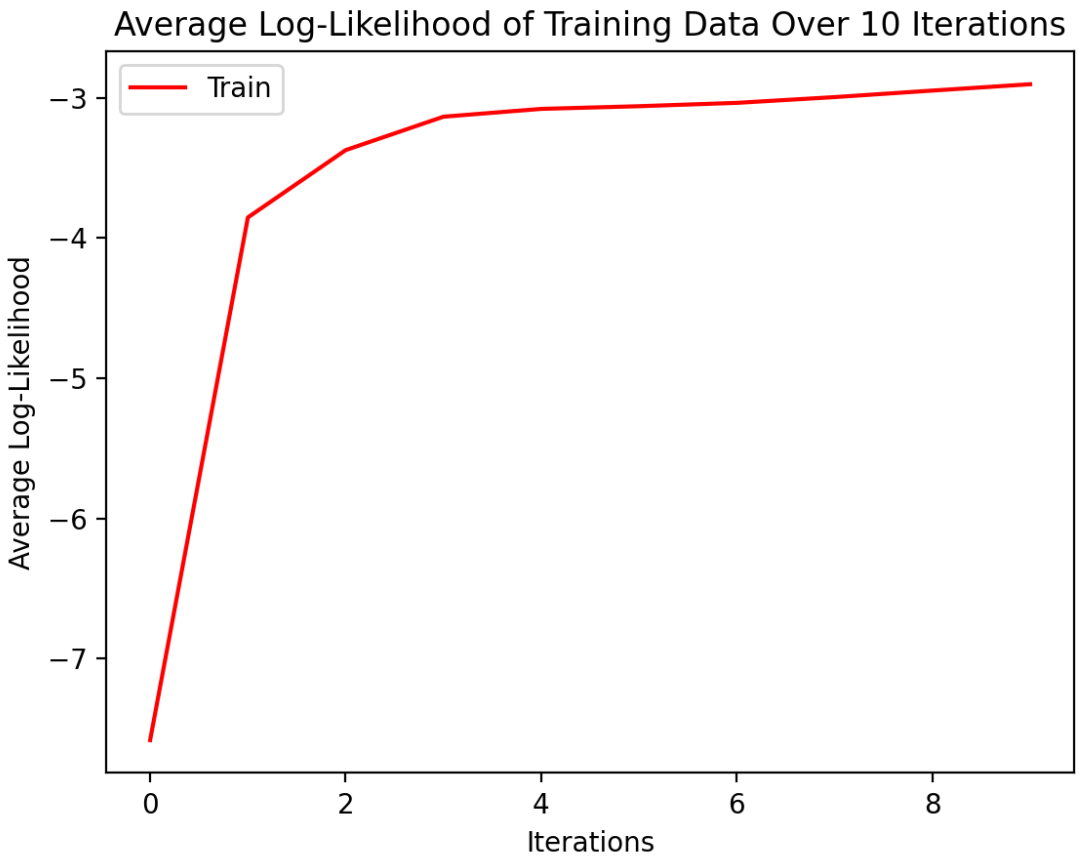
Devanshu Singh
IE Project 1 Report

Primary System

1. Without add-1 smoothing, only 152 labels have non-zero probabilities. This is because I used unigram frequencies from only the parts of the label sequences that did not correspond with silence.
6. It tends to converge after around 5 iterations.

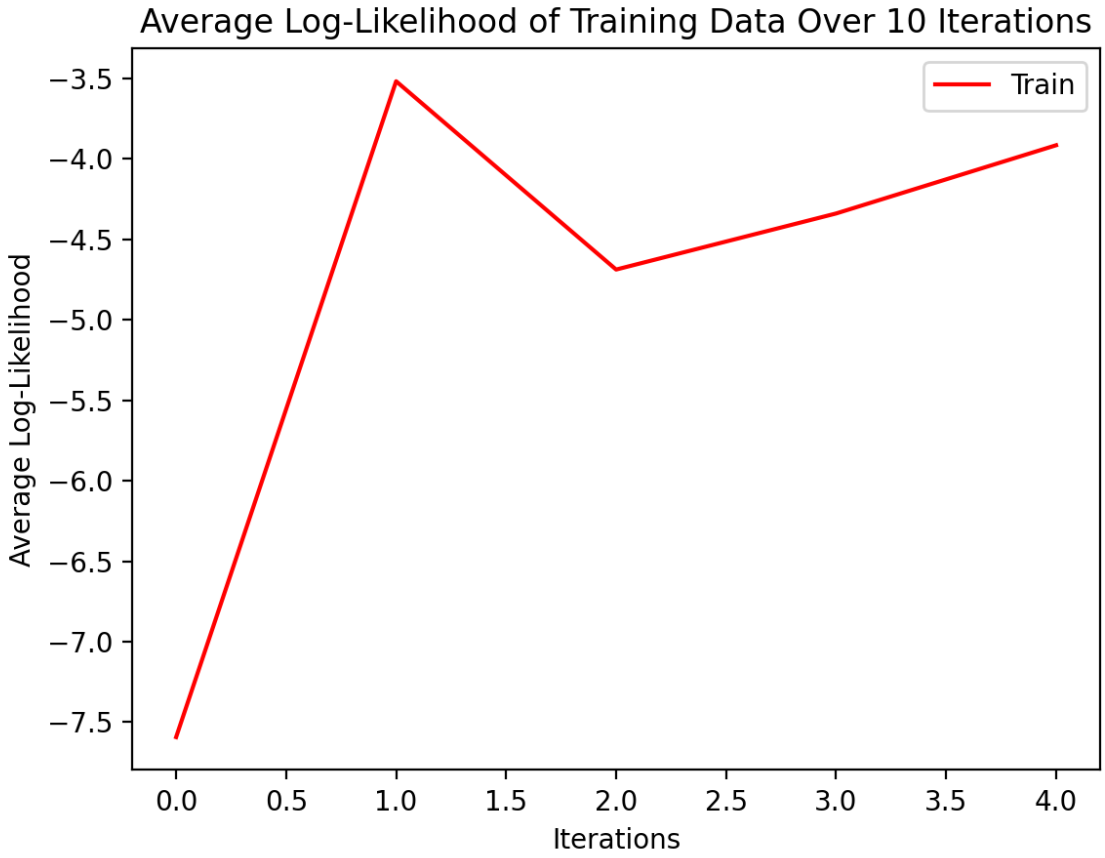
Plot:



Secondary System

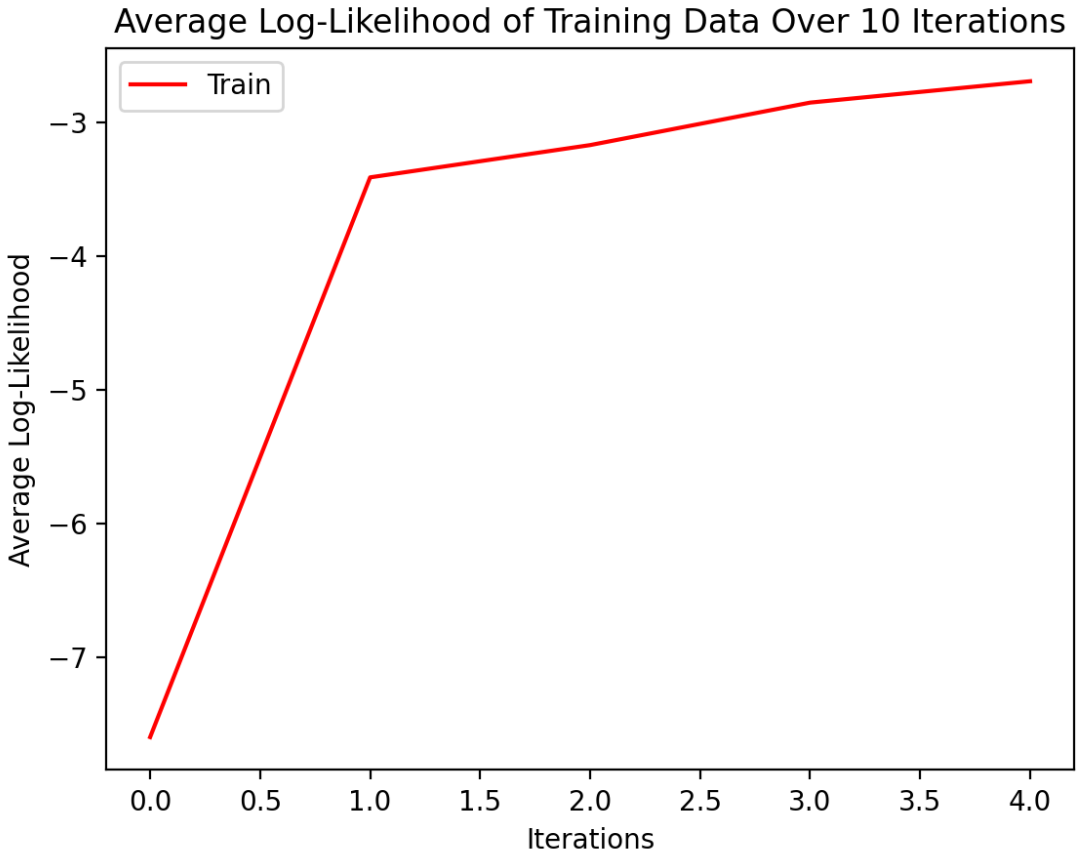
Plot for training the system on kept data:

Devanshu Singh
IE Project 1 Report



Plot for training the system trained using held out method on the entire data:

Devanshu Singh
IE Project 1 Report



Discussion and debugging:
While the plot above shows the total log likelihood increasing monotonically, my system did not work fully. Here I will discuss the various strategies I tried for debugging in the hopes that it gives me some partial credit. I did put in a lot of effort into this project, but ultimately I could not figure out one big problem. I tried various sanity checks which I will also detail

The big problem was the my alphas and betas kept changing for each iteration, even if I removed the random perturbation vectors, and the system predicted “about”, “after” and a few other common words for each utterance. The confidence values were also 0. I tried a creative way to parallelize the forward and backward computation and prevent the multiplication of large sparse matrices. I stored the arc probabilities in one vector instead of a sparse matrix, and did the computations using numpy arrays and my knowledge of which indices correspond to which outgoing and incoming arcs. This made for very complex code but fast computation. However, in trying this creative method, I must have made a mistake early on, because it ended up affecting my accuracy later on.

Sanity checks:

1) My alphas and betas are very close in every iteration. While they are not exactly the same, they are extremely close in each iteration.

Devanshu Singh
IE Project 1 Report

2) The rows of my emission matrix sum to probability 1, 0 in the log domain. The same is true for the transition vector, where all of the probabilities originating from the same left state sum to 1, 0 in log.
3) Multiplying (adding in log) my alphas and betas results in the marginal log probability.
4) My plot increases looks right and the values are in the -10 to 0 range.

Other points:

1) Removing random perturbations made no difference. Originally I thought the random fluctuations in the alphas and betas were due to this, but this was not true.
2) Accuracy on the held out data in the secondary system is very low. However, the early stopping criteria did work and was 4 iterations.