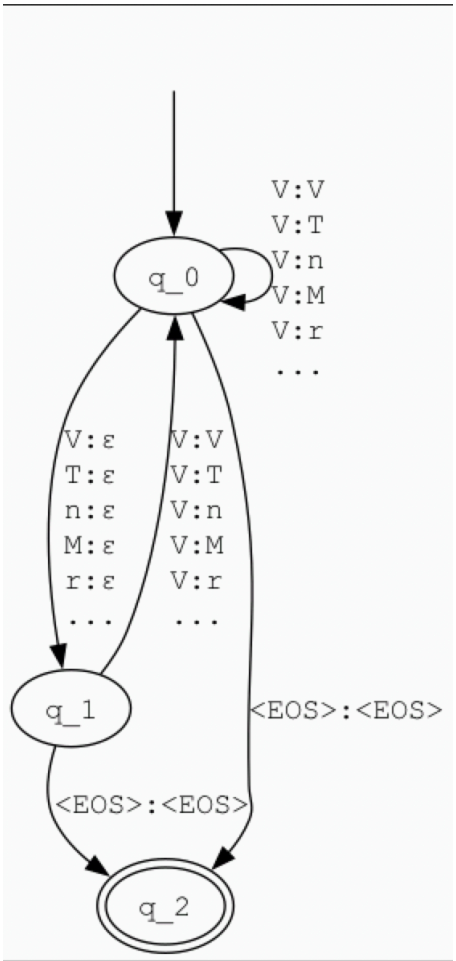
Homework 2

Mehmet Sarioglu
March 28, 2025

Part 1 - Topology and Initializing the model



In our topology, we have three states. We loop back to state 0 for translations and insertions. We move to state 1 when we delete a character. We can move back from state 1 if we do a translation once again. State 2 is used for character deletions. After we have initialized the weights we initialized the model and ran it to see it's performance also printing out the first 10 sequences.

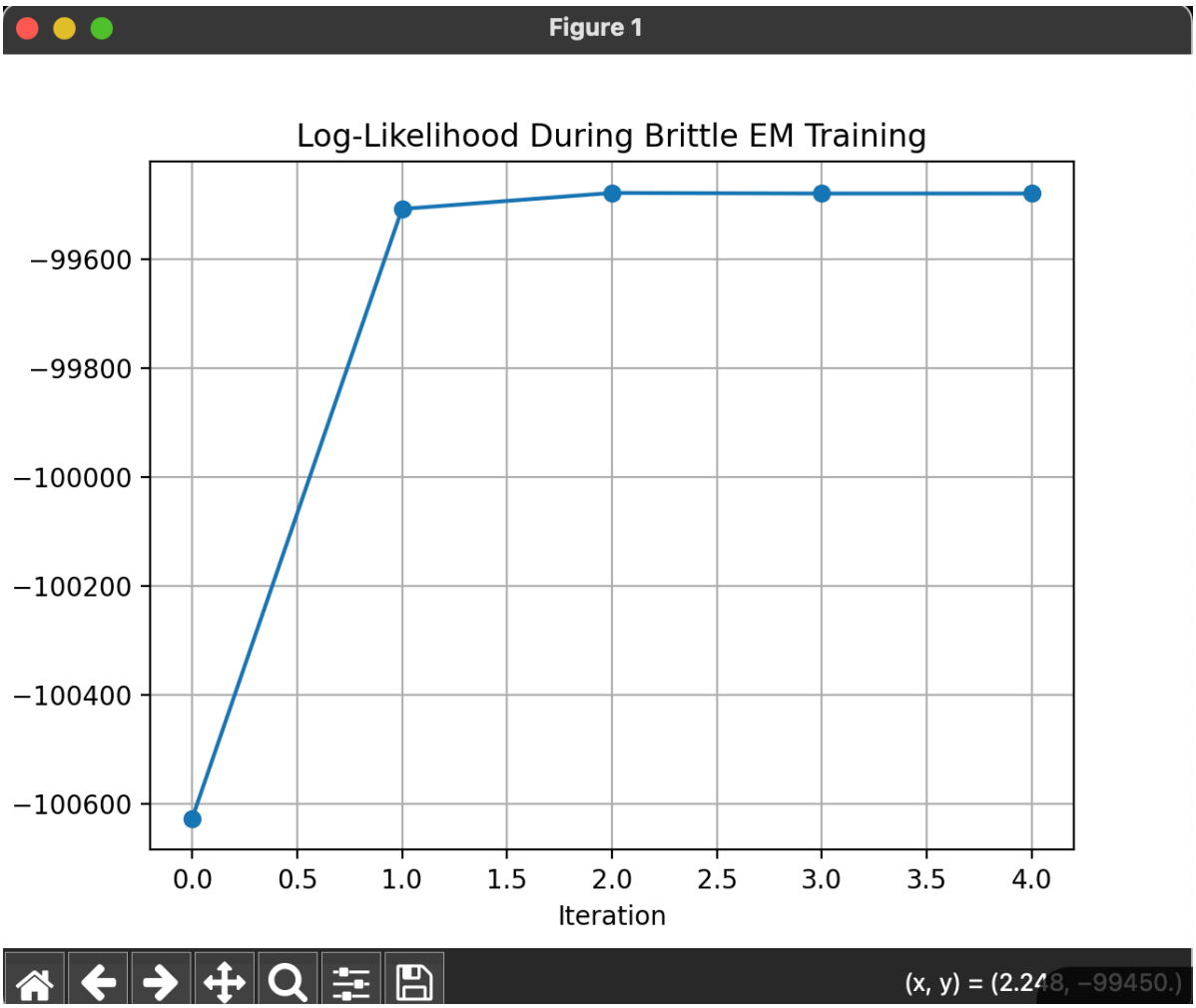
```
(myenv) (base) mehmetborasarioglu@crc-dot1x-nat-10-239-109-89 hw2 % python init.py
First 10 decodings from test set:
 1. [Horatio] Now cracks a noble heart! (logprob=-122.1815)
 2. Good night, sweet prince,   (logprob=-90.5897)
 3. And flights of angels sing thee to thy rest!        (logprob=-145.0738)
 4. Why does the drum come hither?      (logprob=-98.8734)
 5. Enter Fortinbras and English] Ambassador, with Drum,        (logprob=-186.9699)
 6. Colors, and Attendants].    (logprob=-88.1295)
 7. [Fortinbras] Where is this sight?   (logprob=-106.1329)
 8. [Horatio] What is it ye would see?  (logprob=-104.7466)
 9. If aught of woe or wonder, cease your search.       (logprob=-149.3463)
10. [Fortinbras] His quarry cries on havoc. O proud Death,      (logprob=-197.5614)

Character Error Rate (CER): 0.0840
(myenv) (base) mehmetborasarioglu@crc-dot1x-nat-10-239-109-89 hw2 %
```

Part 2 - Implementing Forward and Brittle Train

Part 3 - Implementing Backward and Soft Train

Even though I tried a lot of different weights and different values for smoothing the performance of our modle stayed the same even though the log probabilities



I had the same issue when training the soft em model, even though the confirmation by the TA that my functions were correct we couldn't figure our the reason for the stagnation of the performance. What I saw was that soft EM took lot more time to converge compared to hard em but the improvement was negligable so I decided to terminate its execution early.