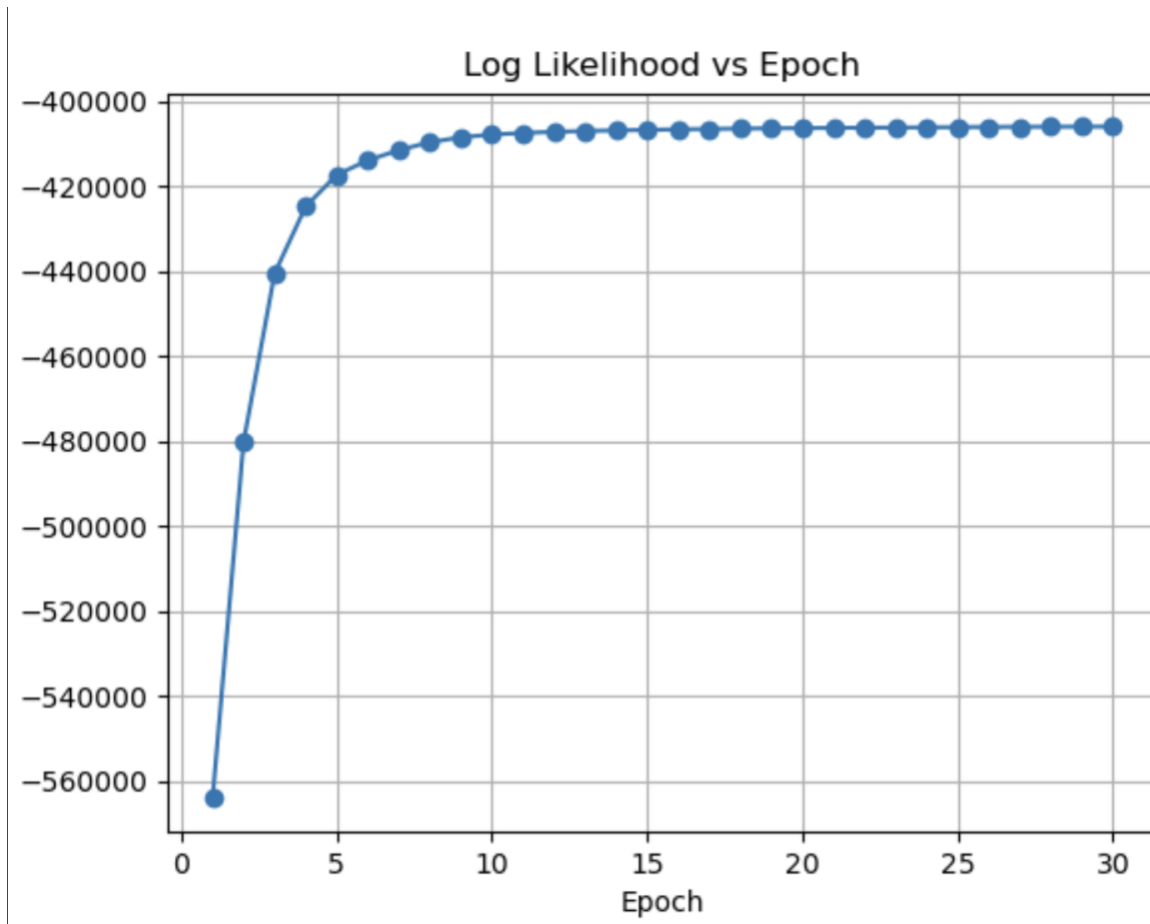


**(520|600).666**  
**Information Extraction from Speech and Text**  
**Project # 2**  
**Lavanya Shankar**

**Question 1:**

Training Log likelihood with 30 epochs:



After running the code for this,

My answer for 1<sup>st</sup> one is,

---- echo: 0 ----

log\_likelihood ==== -564019.6755373772

```
per_frame_log_likelihood === -5.281824933627169
```

the log likelihood kept on increasing,,

My answer for 10<sup>th</sup> one is,

```
---- echo: 10 ----
```

```
log_likelihood ===== -407537.0706880219
```

```
per_frame_log_likelihood === -3.8164261898957896
```

after 10<sup>th</sup> it became stagnant, also the graph if we see its stable

my answer for 30<sup>th</sup> is,

```
---- echo: 29 ----
```

```
log_likelihood ===== -406032.9010899103
```

```
per_frame_log_likelihood === -3.8023402265291035
```

1. Initial Increase in Log-Likelihood - The plot of the training data log-likelihood as a function of the number of iterations likely shows an initial steep increase in the log-likelihood. This indicates rapid improvement in the model's performance as it learns to better fit the training data.

2. After around the 10th iteration, the plot likely shows a plateau or stabilization in the log-likelihood. This suggests that the model has reached a point where further iterations do not significantly increase the log-likelihood. The curve may flatten out, indicating that the model's performance has stabilized.

3. The average per-frame log-likelihood, which is the total log-likelihood divided by the number of acoustic observations in the training data, is an important metric in speech recognition. This metric gives an indication of how well the model is performing on average per observation.

- Initially, as the log-likelihood increases rapidly, the average per-frame log-likelihood would also increase.

- After the log-likelihood reaches a plateau, the average per-frame log-likelihood would stabilize as well. This indicates that, on average, the model is consistently performing at a certain level of log-likelihood per observation.

4. we don't need to run many iterations till 100, the convergence happens at 10 only.

## Question2:

---- echo: 4 ----

log\_likelihood = -417379.38642494177 per\_frame\_log\_likelihood = -3.9085956494352367

Saved to: data/4.mdl.pkl

Predicted Words: ['about', 'about', 'about', 'about', 'after', 'after', 'after', 'after', 'after', 'after', 'after', 'after', 'after', 'after', 'oily', 'oily', 'allow', 'oily', 'about', 'oily', 'along', 'allow', 'along', 'along', 'oily', 'oily', 'also', 'also', 'before', 'also', 'always', 'always', 'always', 'always', 'also', 'always', 'always', 'always', 'always', 'another', 'another', 'another', 'around', 'away', 'away', 'away', 'away', 'always', 'become', 'become', 'become', 'become', 'before', 'before', 'before', 'before', 'before', 'before', 'before', 'brother', 'brother', 'brother', 'brother', 'carry', 'carry', 'carry', 'carry', 'children', 'children', 'children', 'children', 'dinner', 'dinner', 'dinner', 'dinner', 'dinner', 'dinner', 'eating', 'eating', 'many', 'away', 'away', 'enjoy', 'enough', 'enough', 'enough', 'enough', 'enough', 'enough', 'enough', 'about', 'enough', 'enough', 'even', 'even', 'even', 'even', 'even', 'even', 'very', 'carry', 'very', 'very', 'every', 'very', 'extra', 'extra', 'forces', 'forces', 'forces', 'government', 'government', 'government', 'greasy', 'greasy', 'greasy', 'greasy', 'greasy', 'into', 'money', 'into', 'into', 'into', 'into', 'into', 'many', 'many', 'many', 'many', 'many', 'many', 'many', 'money', 'money', 'money', 'money', 'money', 'money', 'money', 'money', 'morning', 'morning', 'morning', 'morning', 'never', 'over', 'never', 'over', 'never', 'often', 'before', 'often', 'often', 'often', 'often', 'oily', 'oily', 'always', 'table', 'oily', 'oily', 'oily', 'oily', 'oily', 'only', 'only', 'only', 'only', 'only', 'only', 'open', 'open', 'open', 'order', 'order', 'order', 'order', 'over', 'other', 'other', 'over', 'over', 'over', 'over', 'away', 'over', 'over', 'people', 'people', 'people', 'people', 'people', 'problem', 'problem', 'problem', 'problem', 'problem', 'several', 'several', 'several', 'several', 'away', 'shellfish', 'shellfish', 'shellfish', 'sometimes', 'sometimes', 'sometimes', 'sometimes', 'study', 'system', 'system', 'sometimes', 'system', 'system', 'table', 'table', 'table', 'table', 'very', 'very', 'very', 'very', 'water', 'water', 'water', 'water', 'without', 'without', 'without', 'without', 'without', 'without', 'without']

Actual Words: ['about', 'about', 'about', 'about', 'after', 'after', 'after', 'after', 'after', 'after', 'after', 'after', 'after', 'after', 'allow', 'allow', 'allow', 'allow', 'allow', 'along', 'along', 'along', 'along', 'along', 'also', 'also', 'also', 'also', 'always', 'always', 'always', 'always', 'always', 'always', 'always', 'always', 'always', 'another', 'another', 'another', 'around', 'away', 'away', 'away', 'away', 'become', 'become', 'become', 'become', 'before', 'before', 'before', 'before', 'before', 'before', 'brother', 'brother', 'brother', 'brother', 'carry', 'carry', 'carry', 'children', 'children', 'children', 'children', 'dinner', 'dinner', 'dinner', 'dinner', 'dinner', 'dinner', 'eating', 'eating', 'eating', 'enjoy', 'enjoy', 'enjoy', 'enough', 'enough', 'enough', 'enough', 'enough', 'enough', 'enough', 'enough', 'even', 'even', 'even', 'even', 'even', 'every', 'every', 'every', 'every', 'every', 'every', 'extra', 'extra', 'forces', 'forces', 'forces', 'government', 'government', 'government', 'greasy', 'greasy', 'greasy', 'greasy', 'greasy', 'into', 'into', 'into', 'into', 'into', 'into', 'into', 'into', 'many', 'many', 'many', 'many', 'many', 'many', 'money', 'money', 'money', 'money', 'money', 'money', 'money', 'money', 'morning', 'morning', 'morning', 'morning', 'never', 'never', 'never', 'never', 'never', 'often', 'often', 'often', 'often', 'often', 'often', 'oily', 'oily', 'oily', 'oily', 'oily', 'oily', 'oily', 'oily', 'only',

'only', 'only', 'only', 'only', 'only', 'only', 'only', 'only', 'only', 'open', 'open', 'open', 'order', 'order',  
'order', 'order', 'other', 'other', 'other', 'over', 'over', 'over', 'over', 'over', 'over', 'over', 'people',  
'people', 'people', 'people', 'people', 'problem', 'problem', 'problem', 'problem', 'problem',  
'several', 'several', 'several', 'several', 'shellfish', 'shellfish', 'shellfish', 'shellfish', 'sometimes',  
'sometimes', 'sometimes', 'sometimes', 'study', 'system', 'system', 'system', 'system',  
'system', 'table', 'table', 'table', 'table', 'table', 'very', 'very', 'very', 'very', 'water', 'water', 'water',  
'water', 'without', 'without', 'without', 'without', 'without', 'without', 'without']

Confidence: [0.9879372 1. 0.99999988 1. 0.99999999 0.99999404  
0.98295968 0.99999988 0.9362512 0.98420952 1. 0.9974939  
1. 0.99945001 0.71071408 0.95532767 0.50022921 0.99259651  
0.76194161 0.61164302 0.99999999 0.99838834 0.99507339 0.99999978  
0.99068121 0.98455326 1. 1. 0.99994352 1.  
0.99813167 0.89228221 1. 1. 0.94562764 1.  
1. 0.99999955 1. 1. 0.90418037 1.  
1. 1. 1. 1. 0.99999748 1.  
1. 1. 1. 1. 1. 1.  
1. 1. 1. 1. 1. 1.  
1. 1. 0.99999913 1. 1. 1.  
1. 1. 1. 1. 1. 0.99999981  
1. 0.99999933 1. 1. 1. 1.  
1. 0.8968584 1. 1. 1. 0.9999988  
1. 0.99999486 0.99999933 0.99999999 1. 0.99999996  
0.94428263 1. 0.99915985 1. 0.56282432 1.  
1. 0.99989818 0.70406221 0.57996252 0.75306811 0.81112204  
0.91619866 0.99584037 0.72908483 1. 1. 1.  
1. 1. 1. 1. 1. 1.  
1. 1. 1. 1. 0.99999965 0.66503179  
0.99939117 0.88178877 1. 1. 0.99999839 1.  
0.99999968 0.98206446 0.99999998 0.99996175 0.99499425 0.99987816  
0.99973161 0.99999875 0.99999999 0.86662704 0.99951524 0.99912025  
0.99999968 0.63803191 1. 1. 0.99707281 0.99999953  
0.99999995 0.97084597 1. 0.97616632 1. 1.  
0.9992146 0.99999999 1. 1. 0.99999997 0.99999901  
0.99999486 0.96615652 0.99980577 1. 1. 0.98603579  
0.99999999 1. 0.99948361 0.99999263 0.88980693 0.94088032  
0.99823758 0.87514497 0.90737791 0.99941129 0.99996853 0.98931105  
0.99999956 0.99999996 0.99999999 0.96755243 1. 0.99983434  
1. 0.9924024 0.99999806 0.99999699 1. 0.99997909  
0.71293599 1. 0.97569676 0.78303235 1. 1.  
1. 1. 0.99960591 1. 1. 0.99995155  
1. 1. 1. 1. 1. 1.  
1. 1. 1. 1. 1. 1.  
1. 1. 1. 1. 1. 1.]

```
0.99999999 1.    0.99999996 1.    1.    1.
1.    1.    0.97605956 0.94078464 0.99216766 0.9994188
1.    1.    1.    1.    1.    1.
1.    0.99999983 1.    1.    1.    ]
```

The confidence is calculated for each word:

If predicted word is about and actual word is about then the confidence is 0.99 or 1 → most likely

If predicted word is about and actual word is table then the confidence is 0.70 → word is not likely

### Running the code:

I have shared all the output in the text file.

Output1.txt is for the first question without heldout data

Output2.txt is for the second question with the heldout data

To run the code without heldout use train

```
def main():
    n_epochs = 8
    wr = Word_Recognizer()
    wr.train(num_epochs=n_epochs)
    wr.plot_log_likelihood()
```

To run the code with heldout use train\_2

```
def main():  
    n_epochs = 8  
    wr = Word_Recognizer()  
    wr.train_2(num_epochs=n_epochs)  
    wr.plot_log_likelihood()
```

python3 Project2.py – running the code

### **Contrastive system:**

#### **Held out data:**

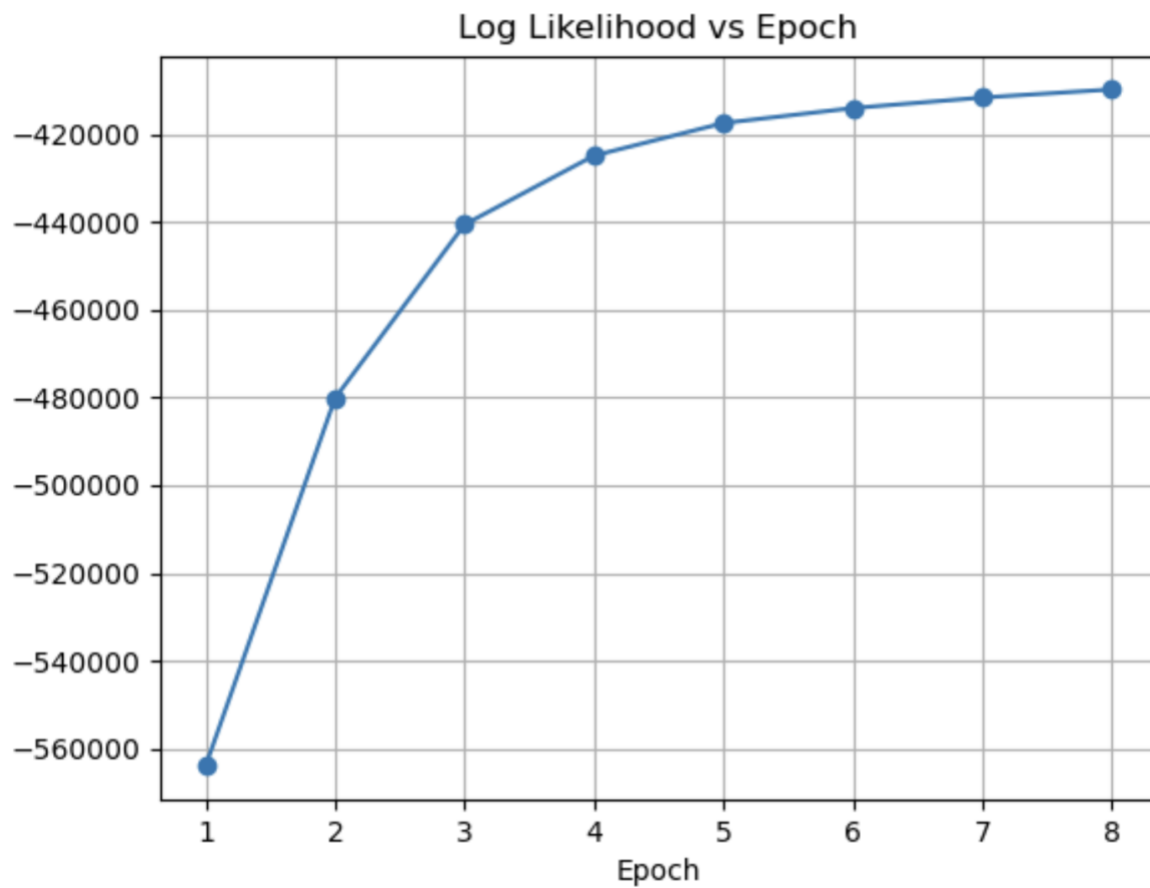
I got an accuracy of 86% at epoch 6

Accuracy: 0.8661087866108786

The accuracy kept increasing till 6<sup>th</sup> iteration and after that it reduces.

After a certain number of iterations (in this case, after the 6th iteration), the model starts to overfit to the training data. Overfitting occurs when the model learns to memorize the training data rather than capturing its underlying patterns. As a result, the model's performance on the test data begins to deteriorate because it cannot generalize well to unseen data. the decrease in accuracy on the test data after the 6th iteration indicates that the model's ability to generalize to new data is diminishing. It is making more errors on unseen examples, which results in a decrease in accuracy

Log likelihood for this system with epoch 8



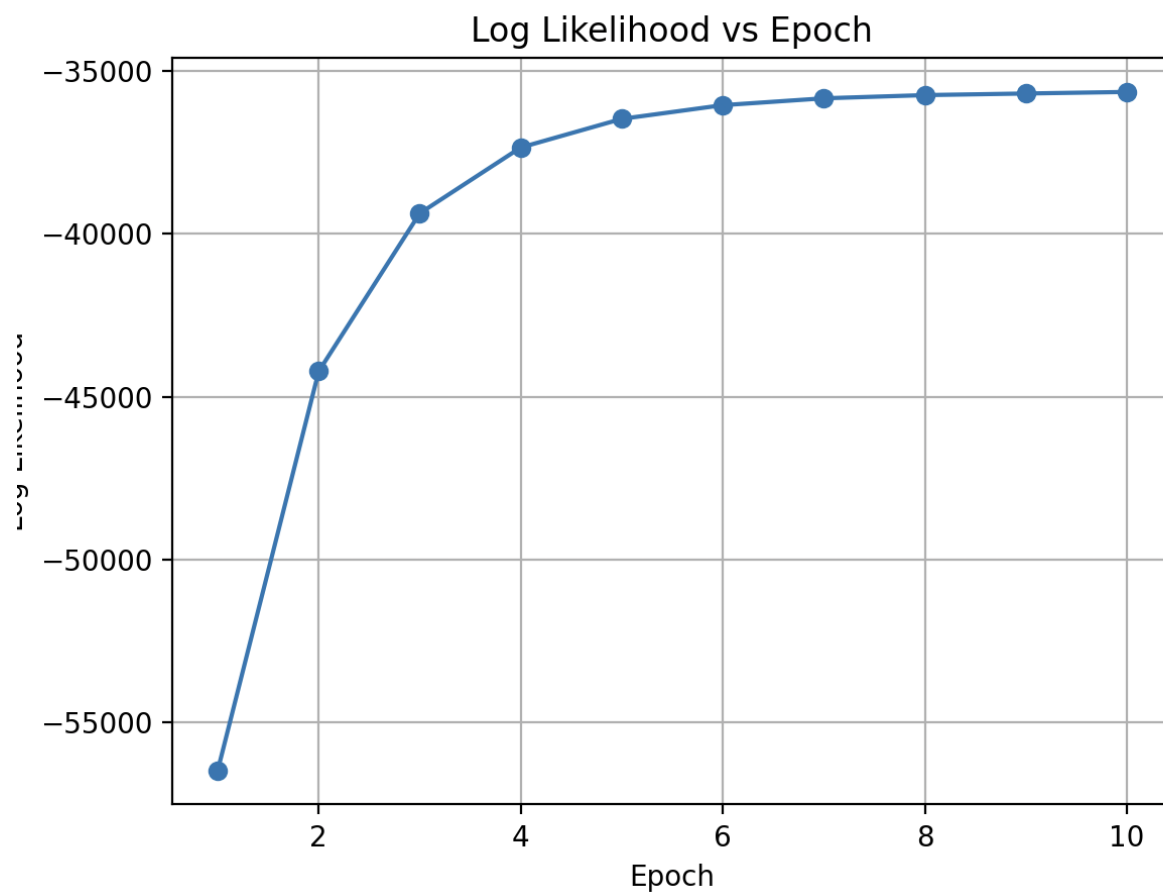
**Contrastive system:**

**Phoneme model:**

Instead of alphabets I used phoneme.  
Here if we see the convergence is much faster.  
Its happening at 7-8  
It converged 2-3 iterations before only

This is a good extension,

This is the graph



This is the output:

```
(base) lavanya@lavanyas-mbp Project2 % python Project2_phoneme.py
```

```
reading file: data/clsp.lblnames
```

```
256 lines, done
```

```
reading file: data/clsp.trnlbls
```

```
798 lines, done
```

```
reading file: data/clsp.endpts
```

```
798 lines, done
```

```
reading file: data/clsp.trnscr
```

```
798 lines, done
```

```
reading file: data/clsp.devlbls
```

```
393 lines, done
```



```
---- epoch: 0 ----
log_likelihood = -56479.24421020095 per_frame_log_likelihood = -5.3191979855152525
---- epoch: 1 ----
log_likelihood = -44223.88938683607 per_frame_log_likelihood = -4.164992407876819
---- epoch: 2 ----
log_likelihood = -39373.661883794346 per_frame_log_likelihood = -3.708199461649496
---- epoch: 3 ----
log_likelihood = -37351.17038803217 per_frame_log_likelihood = -3.517721829726142
---- epoch: 4 ----
log_likelihood = -36472.38762028216 per_frame_log_likelihood = -3.4349583368131626
---- epoch: 5 ----
log_likelihood = -36051.61865968275 per_frame_log_likelihood = -3.395330444498281
---- epoch: 6 ----
log_likelihood = -35841.50162222928 per_frame_log_likelihood = -3.3755416860264913
---- epoch: 7 ----
log_likelihood = -35743.14572584222 per_frame_log_likelihood = -3.3662785577172936
---- epoch: 8 ----
log_likelihood = -35693.68923037486 per_frame_log_likelihood = -3.361620760065442
---- epoch: 9 ----
log_likelihood = -35645.673508281565 per_frame_log_likelihood = -3.357098654010319
[-56479.24421020095, -44223.88938683607, -39373.661883794346, -37351.17038803217, -
36472.38762028216, -36051.61865968275, -35841.50162222928, -35743.14572584222, -
35693.68923037486, -35645.673508281565]
```