ELEC-E5510 – Speech Recognition
Assignment 2

Nguyen Xuan Binh, 887799

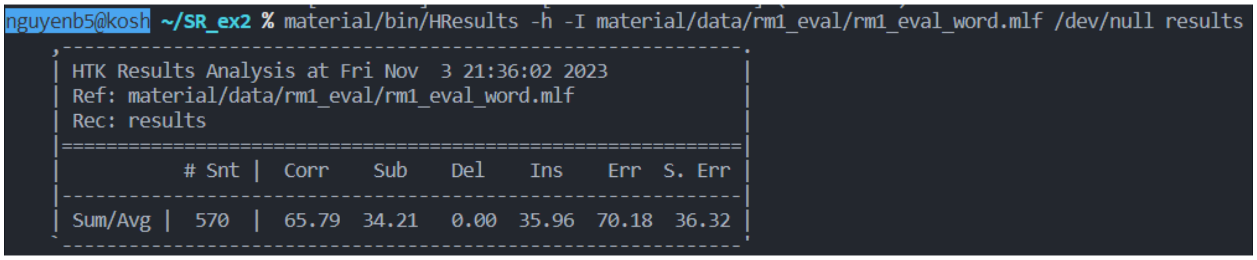
8/11/2023

Question 1

a) The recognition result is very sensitive to the vocabulary of the evaluation data and the lexicon used for recognition. To improve the recognition accuracy, a second lexicon material/ex2/w150.dict is provided which contains only those words that appear in the evaluation set. Build a recognition network from that and run the recognition test over the evaluation set. Compare the results to those of the larger lexicon. Why is the accuracy improving?

In speech recognition, a lexicon is a dictionary that maps words to their phonetic representations. It tells the recognizer which words to expect and how to recognize them based on their phoneme sequences. On the other hand, the recognition network is a graph-like structure that the recognizer uses to match the spoken words against the possible word choices. When we create a network based on a specific lexicon, we essentially instruct the recognizer to ONLY observe the sound of only those words contained in the lexicon.

After following the tutorials from the assignment, this is the output table of the recognition test over the evaluation set. This is the result of lexicon having 600 words



```
nguyenb5@kosh ~/SR_ex2 % material/bin/HResults -h -I material/data/rm1_eval/rm1_eval_word.mlf /dev/null results
    ,-------------------------------------------------------------.
    | HTK Results Analysis at Fri Nov  3 21:36:02 2023            |
    | Ref: material/data/rm1_eval/rm1_eval_word.mlf               |
    | Rec: results                                                |
    |=============================================================|
    |          # Snt |  Corr    Sub    Del    Ins    Err  S. Err |
    |-------------------------------------------------------------|
    | Sum/Avg |  570  |  65.79  34.21   0.00  35.96  70.18  36.32 |
    `-------------------------------------------------------------'
```

Repeating similar pipeline, the commands for Question 1a are:

# Extract the first field from the lexicon, which contains the words
cut -f 1 -d ' ' material/ex2/w150.dict > w150.list

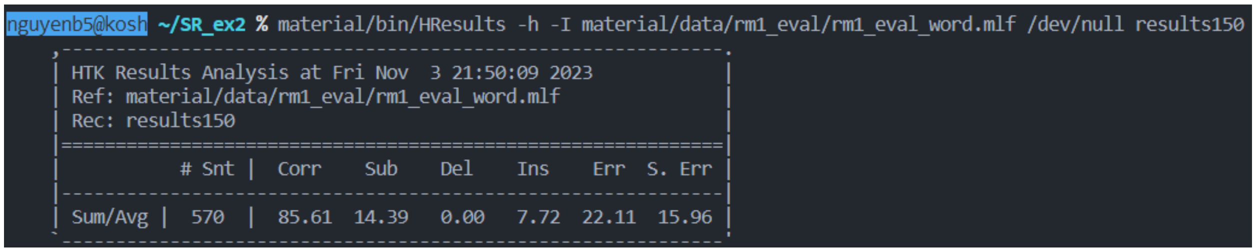
# Build the recognition network using HBuild

material/bin/HBuild -t SILENCE SILENCE -T 1 w150.list w150_loop.htk

# Perform recognition using the new network and lexicon
*material/bin/HVite -T 1 -w w150_loop.htk -H hmm-6/macros -H hmm-6/hmmdefs -C material/ex2/config *
-t 250 -S material/data/rm1_eval/rm1_eval_word.scp -i results150 material/ex2/w150.dict material/ex2/monophones

# Compute the word error rate
material/bin/HResults -h -I material/data/rm1_eval/rm1_eval_word.mlf /dev/null results150

We arrive at the recognition test result over the evaluation set with 150 words lexicon



```
nguyenb5@kosh ~/SR_ex2 % material/bin/HResults -h -I material/data/rm1_eval/rm1_eval_word.mlf /dev/null results150
    ,-------------------------------------------------------------.
    | HTK Results Analysis at Fri Nov  3 21:50:09 2023            |
    | Ref: material/data/rm1_eval/rm1_eval_word.mlf               |
    | Rec: results150                                             |
    |=============================================================|
    |          # Snt |  Corr    Sub    Del    Ins    Err  S. Err |
    |-------------------------------------------------------------|
    | Sum/Avg |  570  |  85.61  14.39   0.00   7.72  22.11  15.96 |
    `-------------------------------------------------------------'
```

How to interpret the table result:

1. Number of Sentences (# Snt): This is the number of sentences that were evaluated.
2. Correct (Corr): This is the percentage of words correctly recognized by the system.
3. Substitutions (Sub): This is the percentage of words that were incorrectly recognized, where the system substituted a different word for the correct one.
4. Deletions (Del): This is the percentage of words that the system should have recognized but missed completely. In both cases, we have 0.00, which means no deletions.
5. Insertions (Ins): This is the number of extra words that the system inserted into the recognized sentence that were not supposed to be there.
6. Error (Err): This is the total error rate, which includes substitutions and insertions.
7. Sentence Error (S. Err): This is the percentage of sentences where there was at least one recognition error.

As we can see, the sentence error rate decreases from 36.32 to 15.96, which means the sentence error rate decreases (or accuracy increases) when we use a smaller dictionary.

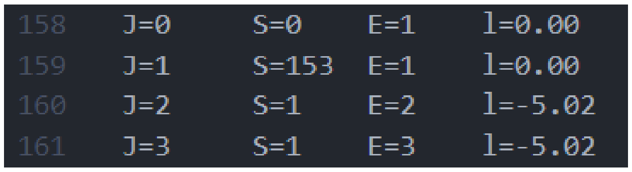
When we run the recognition test with the full lexicon of 600 words, the recognizer has to consider a large number of possible word sequences. This can lead to errors because some words may sound similar to others, or the recognizer may not be very confident in distinguishing between them. By using a smaller lexicon of 150 words that contains only the words from the evaluation set, we reduce the complexity of the task for the recognizer. It has fewer options to choose from, which can lead to better accuracy because the likelihood of confusion between similar-sounding words is reduced.

Additionally, in a large lexicon, it is not feasible to have a unique model for each word because the number of words can be in larger magnitude. Instead, the system models sub-word units, such as phonemes, syllables, or triphones. These sub-word units are combined to form word models. However, this approach has a drawback: sub-word units may not capture the nuances of coarticulation effectively. Word models can capture these effects because the model is for the entire word, but sub-word models may miss them. This could be another reason why smaller lexicon results in smaller error rate. Additionally, ASR may apply some pruning method instead of brute force searching, which can cuts out potentially correct sentences.

b) The recognition network should reflect the recognition task. The task in this exercise contains only single words, but the recognition network contains a loop which allows multiple words to be recognized from a single utterance. Remove the loop by modifying the recognition network (built from w150.dict) with a text editor. Run the recognition test once more and compare to previous results. Explain why the accuracy changed.

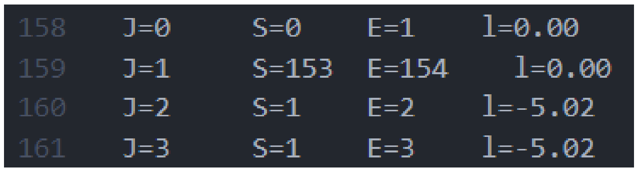
Hint: The recognition network has nodes and transitions. The loop is the only transition pointing backwards to an earlier node. Transitions are lines starting with J=<index>, and have parameters S for the starting node and E for the ending node. As the transitions are indexed, the easiest way is to move the last defined transition instead of the loop transition and update its index. You also need to reduce the count of the transitions in the second line of the file, L=<num>.

The recognition network file is the file w150_loop.htk produced from the previous task. We can see that at the transition J=1, we have the starting node as 153, and ending node as 1, suggesting this is the transition that creates the loop



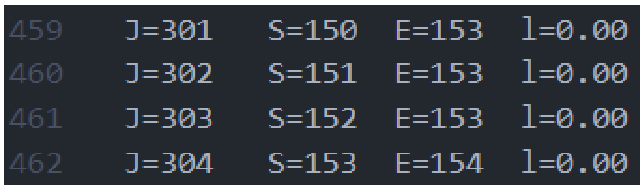
```
J=0     S=0     E=1     l=0.00
J=1     S=153   E=1     l=0.00
J=2     S=1     E=2     l=-5.02
J=3     S=1     E=3     l=-5.02
```

To remove this loop, we update the ending node as the last node in the transition, which is 154



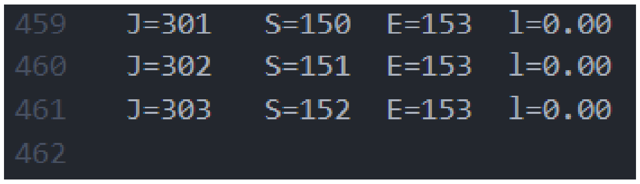
```
J=0     S=0     E=1     l=0.00
J=1     S=153   E=154     l=0.00
J=2     S=1     E=2     l=-5.02
J=3     S=1     E=3     l=-5.02
```

Then we scroll to the end of this file to delete the last line, where S = 153 and E = 154



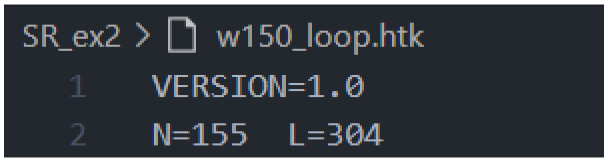
```
J=301   S=150   E=153   l=0.00
J=302   S=151   E=153   l=0.00
J=303   S=152   E=153   l=0.00
J=304   S=153   E=154   l=0.00
```

After deletion, there are now only 304 transitions left.



```
J=301   S=150   E=153   l=0.00
J=302   S=151   E=153   l=0.00
J=303   S=152   E=153   l=0.00
```

We scroll up to the top of this file and update the number of lines as 304

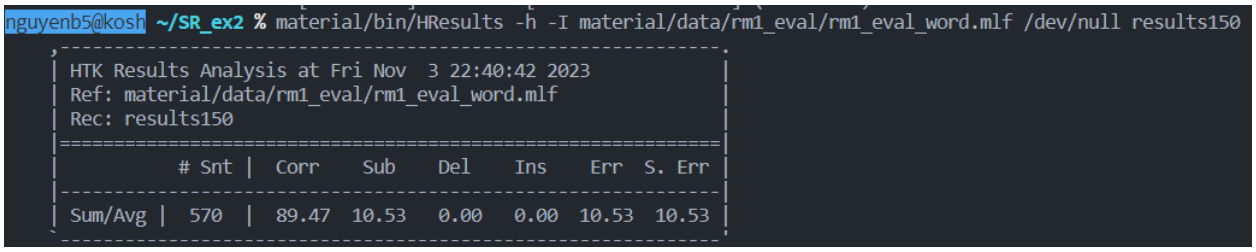


```
SR_ex2 > w150_loop.htk
VERSION=1.0
N=155  L=304
```

the commands for Question 1b are

# Rerun the recognition test using the updated w150_loop.htk
*material/bin/HVite -T 1 -w w150_loop.htk -H hmm-6/macros -H hmm-6/hmmdefs -C material/ex2/config *
-t 250 -S material/data/rm1_eval/rm1_eval_word.scp -i results150 material/ex2/w150.dict material/ex2/monophones

# Compute the word error rate again for the updated result on lexicon 150
material/bin/HResults -h -I material/data/rm1_eval/rm1_eval_word.mlf /dev/null results150



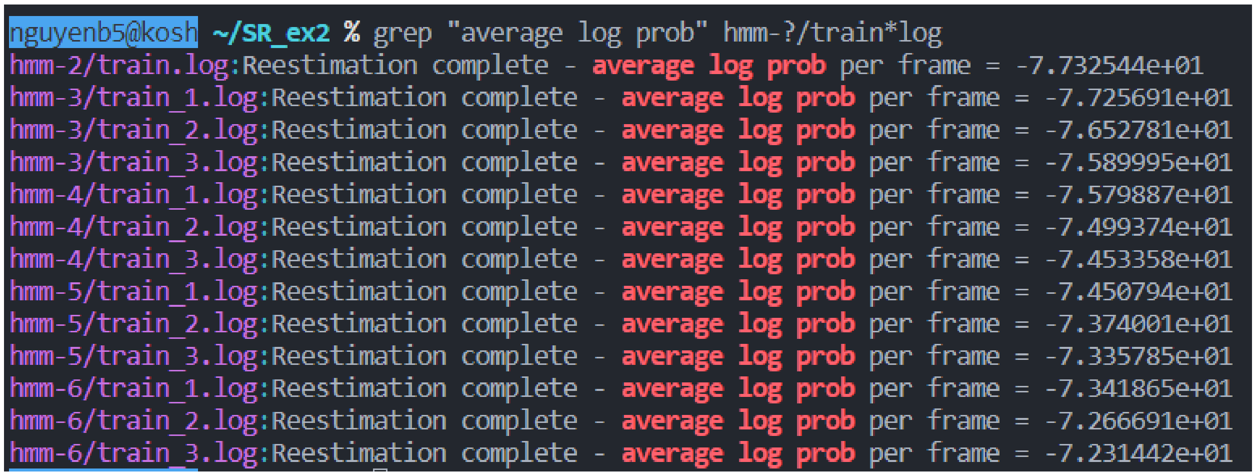
```
nguyenb5@kosh ~/SR_ex2 % material/bin/HResults -h -I material/data/rm1_eval/rm1_eval_word.mlf /dev/null results150
    ,-------------------------------------------------------------.
    | HTK Results Analysis at Fri Nov  3 22:40:42 2023            |
    | Ref: material/data/rm1_eval/rm1_eval_word.mlf               |
    | Rec: results150                                             |
    |=============================================================|
    |           # Snt |  Corr    Sub    Del    Ins    Err  S. Err |
    |-------------------------------------------------------------|
    | Sum/Avg |  570  |  89.47  10.53   0.00   0.00  10.53  10.53 |
    `-------------------------------------------------------------'
```

From the result above, we can see that the sentence error rate has further decreased from 15.96 to 10.53. This decrease (or increase in accuracy) is because without the loop, the network is constrained to recognize only single words per utterance. This means that the recognition process cannot mistakenly insert extra words, which reduces the chance of errors. The decoder has fewer paths to consider, which typically leads to more accurate and faster recognition because the system is not trying to match multiple-word sequences.

Question 2

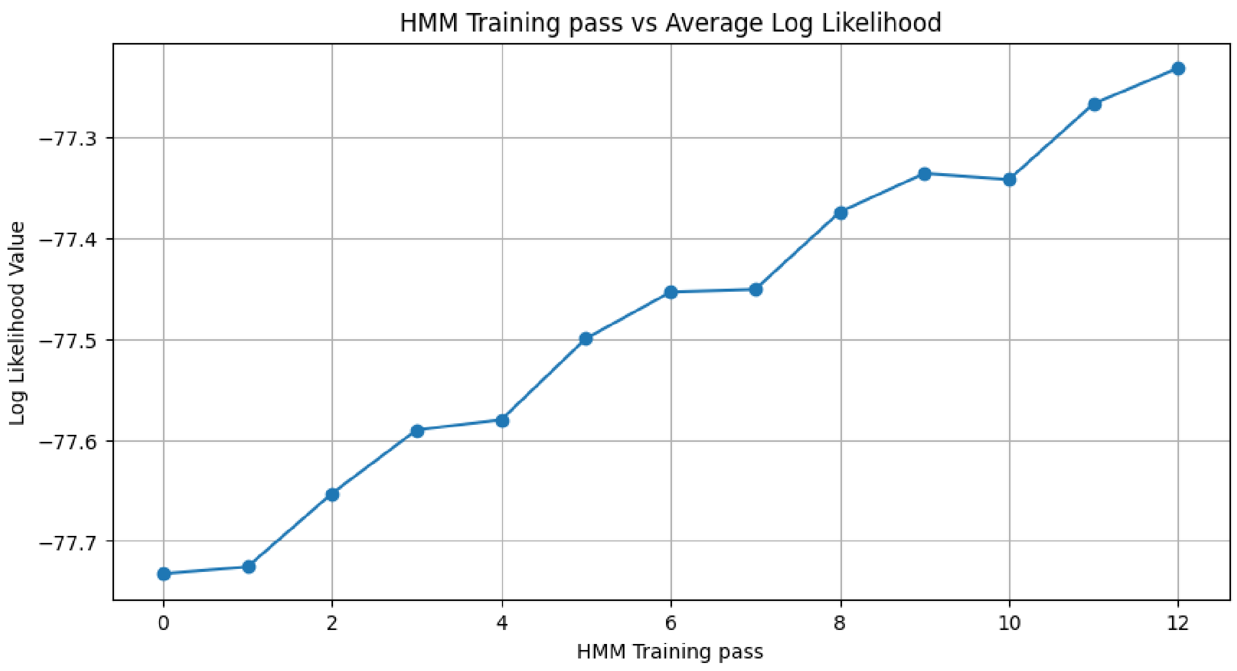
a) An easy way of tracking the HMM training process is to look at the logarithmic likelihood values reported by HERest. You can fetch them with the command grep "average log prob" hmm-?/train*log. Plot the values as one continuous line. (Check that grep gave them in the right order!)

Running the command to extract the logarithmic likelihood values reported by HERest
# Extract the average log probability of the training data from
# the log files from each hmm-{i} folder, i in 1 to 6
grep "average log prob" hmm-?/train*log



```
nguyenb5@kosh ~/SR_ex2 % grep "average log prob" hmm-?/train*log
hmm-2/train.log:Reestimation complete - average log prob per frame = -7.732544e+01
hmm-3/train_1.log:Reestimation complete - average log prob per frame = -7.725691e+01
hmm-3/train_2.log:Reestimation complete - average log prob per frame = -7.652781e+01
hmm-3/train_3.log:Reestimation complete - average log prob per frame = -7.589995e+01
hmm-4/train_1.log:Reestimation complete - average log prob per frame = -7.579887e+01
hmm-4/train_2.log:Reestimation complete - average log prob per frame = -7.499374e+01
hmm-4/train_3.log:Reestimation complete - average log prob per frame = -7.453358e+01
hmm-5/train_1.log:Reestimation complete - average log prob per frame = -7.450794e+01
hmm-5/train_2.log:Reestimation complete - average log prob per frame = -7.374001e+01
hmm-5/train_3.log:Reestimation complete - average log prob per frame = -7.335785e+01
hmm-6/train_1.log:Reestimation complete - average log prob per frame = -7.341865e+01
hmm-6/train_2.log:Reestimation complete - average log prob per frame = -7.266691e+01
hmm-6/train_3.log:Reestimation complete - average log prob per frame = -7.231442e+01
```

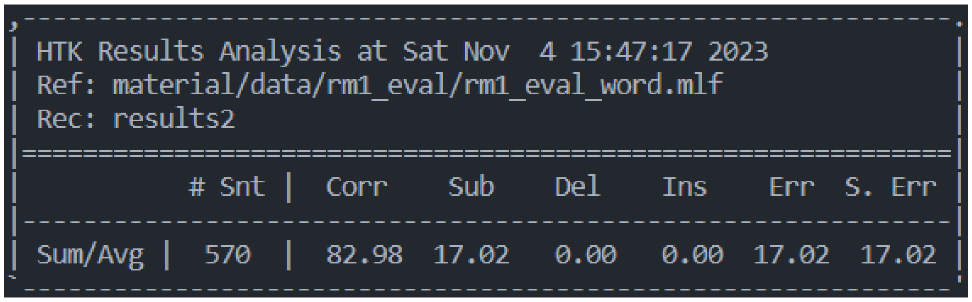
These values can be plotted in order. This is a graph plotted in Python



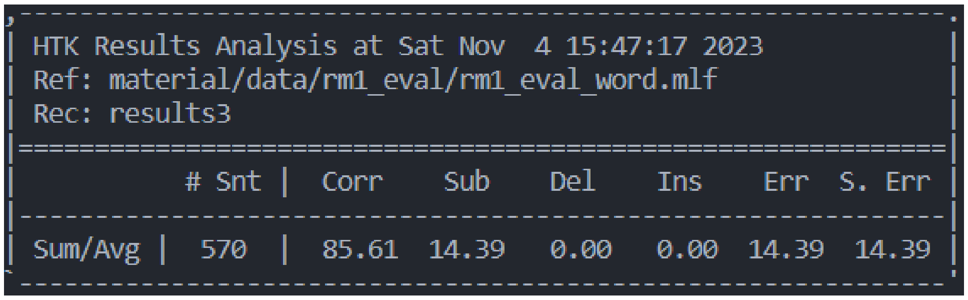
b) Using the modified w150 network, recognize the evaluation set using all the models through hmm-2 to hmm-6. Graphically plot the word error rate with respect to the model number. Comment on the relationship of the two curves. Do you think further iterations would improve the recognition result?

To recognize the evaluation set using all models from hmm-2 to hmm-6, we can use a for-loop that iterates over folders hmm-2 to hmm-6. The commands for this part are

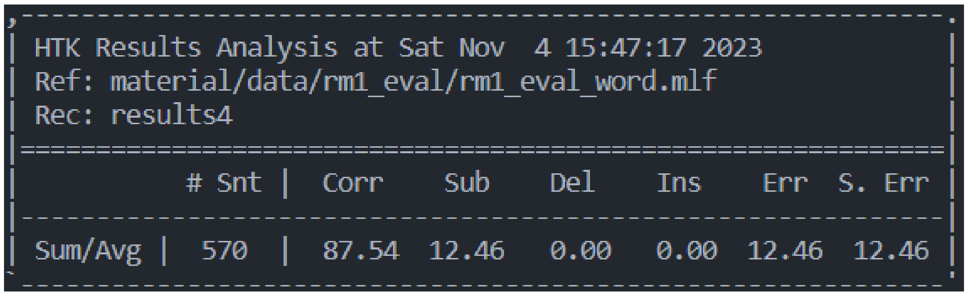
```
# Recognize evaluation set using all models from hmm-2 to hmm-6
for i in $(seq 2 6); do \
material/bin/HVite -T 1 -w w150_loop.htk -H hmm-${i}/macros -H hmm-${i}/hmmdefs -C material/ex2/config \
-t 250 -S material/data/rm1_eval/rm1_eval_word.scp -i results${i} material/ex2/w150.dict material/ex2/monophones; done

# Computing the error rate of each result from hmm-2 to hmm-6
for i in $(seq 2 6); do \
material/bin/HResults -h -I material/data/rm1_eval/rm1_eval_word.mlf /dev/null results${i}; done
```



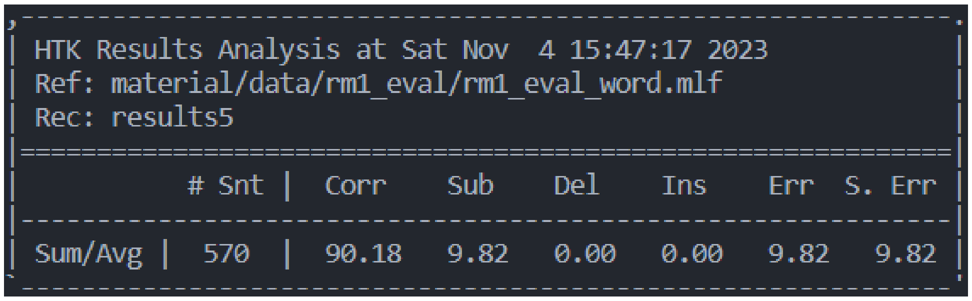
```
,-------------------------------------------------------------.
| HTK Results Analysis at Sat Nov  4 15:47:17 2023            |
| Ref: material/data/rm1_eval/rm1_eval_word.mlf               |
| Rec: results2                                               |
|=============================================================|
|          # Snt |  Corr    Sub    Del    Ins    Err  S. Err |
|-------------------------------------------------------------|
| Sum/Avg |  570  |  82.98  17.02   0.00   0.00  17.02  17.02 |
`-------------------------------------------------------------'
```



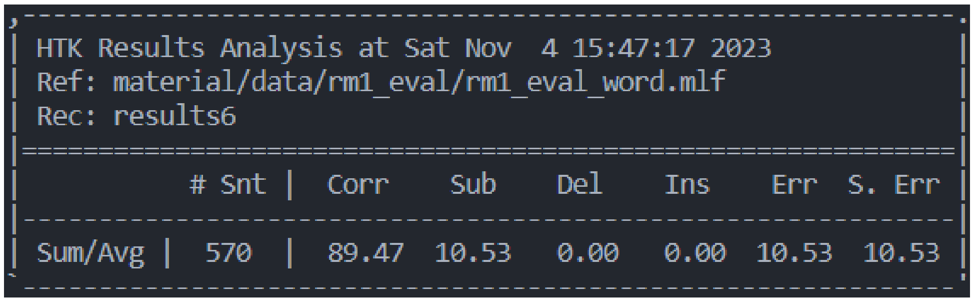
```
,-------------------------------------------------------------.
| HTK Results Analysis at Sat Nov  4 15:47:17 2023            |
| Ref: material/data/rm1_eval/rm1_eval_word.mlf               |
| Rec: results3                                               |
|=============================================================|
|          # Snt |  Corr    Sub    Del    Ins    Err  S. Err |
|-------------------------------------------------------------|
| Sum/Avg |  570  |  85.61  14.39   0.00   0.00  14.39  14.39 |
`-------------------------------------------------------------'
```



```
,-------------------------------------------------------------.
| HTK Results Analysis at Sat Nov  4 15:47:17 2023            |
| Ref: material/data/rm1_eval/rm1_eval_word.mlf               |
| Rec: results4                                               |
|=============================================================|
|          # Snt |  Corr    Sub    Del    Ins    Err  S. Err |
|-------------------------------------------------------------|
| Sum/Avg |  570  |  87.54  12.46   0.00   0.00  12.46  12.46 |
`-------------------------------------------------------------'
```

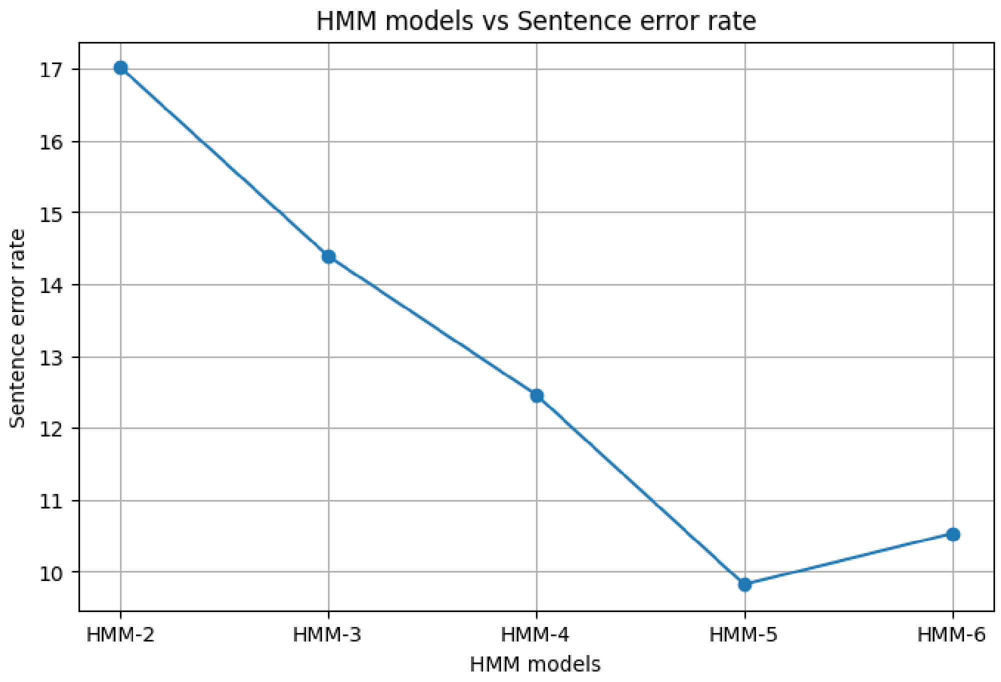


```
,-------------------------------------------------------------.
| HTK Results Analysis at Sat Nov  4 15:47:17 2023            |
| Ref: material/data/rm1_eval/rm1_eval_word.mlf               |
| Rec: results5                                               |
|=============================================================|
|         # Snt |  Corr    Sub    Del    Ins    Err  S. Err  |
|-------------------------------------------------------------|
| Sum/Avg |  570  |  90.18   9.82   0.00   0.00   9.82   9.82 |
`-------------------------------------------------------------'
```



```
,-------------------------------------------------------------.
| HTK Results Analysis at Sat Nov  4 15:47:17 2023            |
| Ref: material/data/rm1_eval/rm1_eval_word.mlf               |
| Rec: results6                                               |
|=============================================================|
|         # Snt |  Corr    Sub    Del    Ins    Err  S. Err  |
|-------------------------------------------------------------|
| Sum/Avg |  570  |  89.47  10.53   0.00   0.00  10.53  10.53 |
`-------------------------------------------------------------'
```

We can plot the S. Error rate against the HMM models in Python



The relationship of the two curves from plot (2a) and plot (2b) have an opposite trend. As the average log likelihood of the data given the model increases, this suggests the model is better at explaining the data, which often corresponds to a lower error rate. The error rate starts to increase again after a certain model complexity (HMM-5), which suggests that the model has begun to overfit the training data and is becoming less generalizable. This is because the complexity of the model increases with more states (HMM-2 to HMM-6 indicate more complex models). The conclusion is that further iteration would not improve the recognition results.

Question 3

a) Recognize the evaluation set with the gender dependent models, using correct evaluation subsets _f for females and _m for males. Use the modified w150 network and hmm-6 models. Analyze the possible benefits and drawbacks of gender dependent models and their use as a combined system. Assume our evaluation data sets are representative of the real case. Back your analysis with further sufficient experiments; can you prove your statements with results, actual numbers?

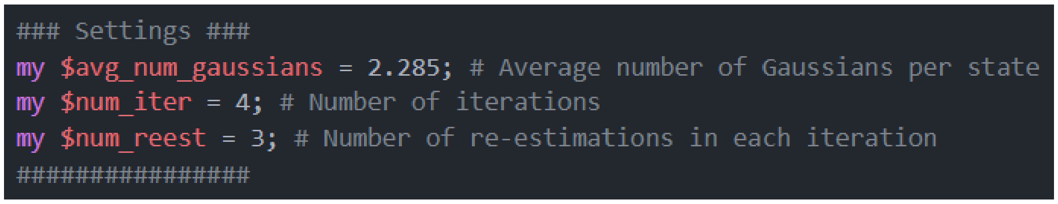
NOTE: Given the gender specific result files, you can get the combined results with HResult simply by providing both input files at the same time.

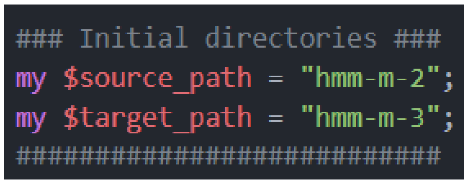
First, I create two copies of the folder hmm-2 and rename them to add the _m and _f tag

```
> hmm-f-2
```

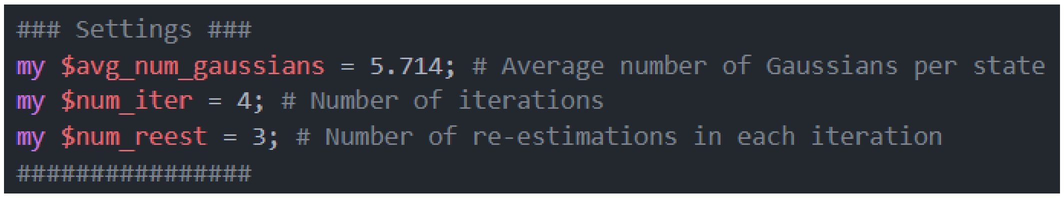
```
> hmm-m-2
```

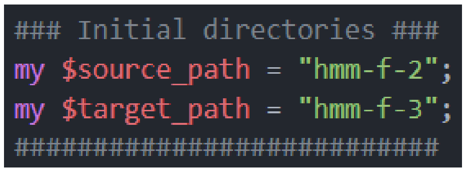
For the female model, since the proportion is 2 parts out of 7 (2 for females and 5 for males), we multiply the average number of Gaussians (8) by the proportion for females (2/7) = 2.285
For the male model, we multiply 8 by the proportion for males (5/7) = 5.714
Create a copy of the file hmm_train.pl and rename it to hmm_train_f.pl and I change the average number of Gaussians and also rename the source and target path to not override the old results (general model)


```
### Settings ###
my $avg_num_gaussians = 2.285; # Average number of Gaussians per state
my $num_iter = 4; # Number of iterations
my $num_reest = 3; # Number of re-estimations in each iteration
################
```


```
### Initial directories ###
my $source_path = "hmm-m-2";
my $target_path = "hmm-m-3";
############################
```

I create the file hmm_train_m.pl and do the same


```
### Settings ###
my $avg_num_gaussians = 5.714; # Average number of Gaussians per state
my $num_iter = 4; # Number of iterations
my $num_reest = 3; # Number of re-estimations in each iteration
################
```


```
### Initial directories ###
my $source_path = "hmm-f-2";
my $target_path = "hmm-f-3";
############################
```

The running commands for Question 3a are:

```
mkdir ex2_f
mkdir ex2_m
mkdir ex2_both
```

# Female dependent model

```
./hmm_train_f.pl material/bin material/data/rm1_train/rm1_train.mlf
material/data/rm1_train/rm1_train_f.scp \
material/ex2/monophones material/ex2/config

# Perform recognition using the new network and lexicon
material/bin/HVite -T 1 -w w150_loop.htk -H hmm-f-6/macros -H hmm-f-6/hmmdefs -C
material/ex2/config \
-t 250 -S material/data/rm1_eval/rm1_eval_word_f.scp -i ex2_f/results material/ex2/w150.dict
material/ex2/monophones

material/bin/HResults -h -I material/data/rm1_eval/rm1_eval_word.mlf /dev/null ex2_f/results
```

# Male dependent model

```
./hmm_train_m.pl material/bin material/data/rm1_train/rm1_train.mlf
material/data/rm1_train/rm1_train.scp \
material/ex2/monophones material/ex2/config

# Perform recognition using the new network and lexicon
material/bin/HVite -T 1 -w w150_loop.htk -H hmm-m-6/macros -H hmm-m-6/hmmdefs -C
material/ex2/config \
-t 250 -S material/data/rm1_eval/rm1_eval_word_m.scp -i ex2_m/results material/ex2/w150.dict
material/ex2/monophones

material/bin/HResults -h -I material/data/rm1_eval/rm1_eval_word.mlf /dev/null ex2_m/results
```

# Combined model

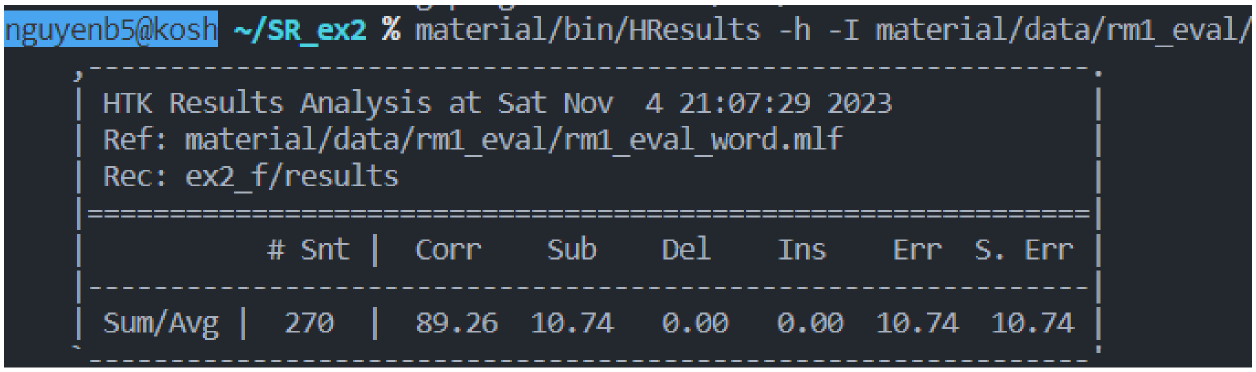
```
./hmm_train.pl material/bin material/data/rm1_train/rm1_train.mlf
material/data/rm1_train/rm1_train.scp \
material/ex2/monophones material/ex2/config

material/bin/HVite -T 1 -w w150_loop.htk -H hmm-6/macros -H hmm-6/hmmdefs -C
material/ex2/config \
```

-t 250 -S material/data/rm1_eval/rm1_eval_word.scp -i ex2_both/results material/ex2/w150.dict material/ex2/monophones

material/bin/HResults -h -I material/data/rm1_eval/rm1_eval_word.mlf /dev/null ex2_both/results ex2_both/results

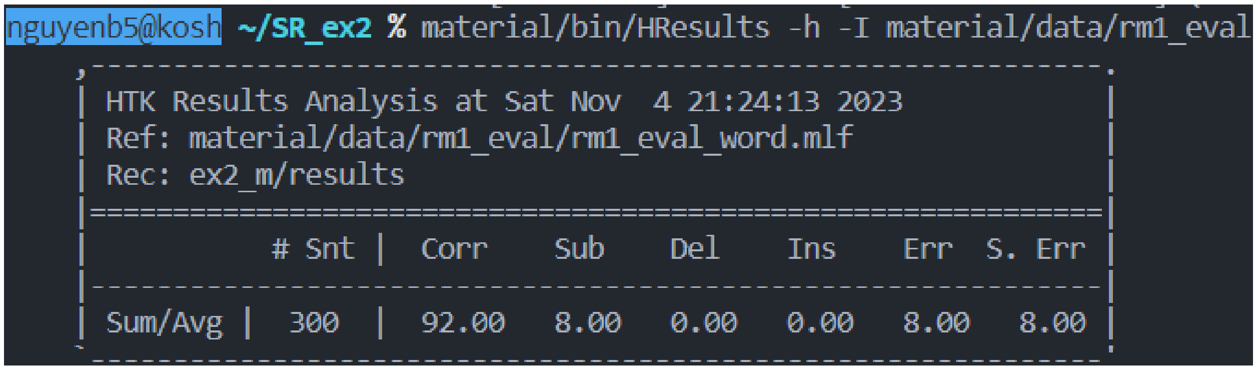
We arrive at these results

Recognition results for Female only:



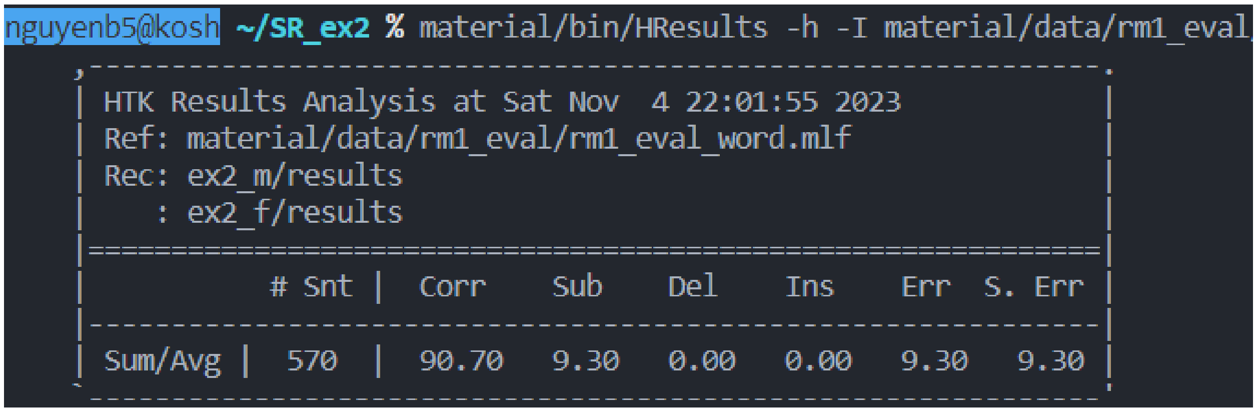
```
nguyenb5@kosh ~/SR_ex2 % material/bin/HResults -h -I material/data/rm1_eval/
    ,-------------------------------------------------------------.
    | HTK Results Analysis at Sat Nov  4 21:07:29 2023             |
    | Ref: material/data/rm1_eval/rm1_eval_word.mlf                |
    | Rec: ex2_f/results                                           |
    |==============================================================|
    |          # Snt |  Corr    Sub    Del    Ins    Err  S. Err  |
    |--------------------------------------------------------------|
    | Sum/Avg |  270  |  89.26  10.74   0.00   0.00  10.74  10.74  |
    `--------------------------------------------------------------'
```

Recognition results for Males only:



```
nguyenb5@kosh ~/SR_ex2 % material/bin/HResults -h -I material/data/rm1_eval
    ,-------------------------------------------------------------.
    | HTK Results Analysis at Sat Nov  4 21:24:13 2023             |
    | Ref: material/data/rm1_eval/rm1_eval_word.mlf                |
    | Rec: ex2_m/results                                           |
    |==============================================================|
    |          # Snt |  Corr    Sub    Del    Ins    Err  S. Err  |
    |--------------------------------------------------------------|
    | Sum/Avg |  300  |  92.00   8.00   0.00   0.00   8.00   8.00  |
    `--------------------------------------------------------------'
```

Recognition results for combined model:



```
nguyenb5@kosh ~/SR_ex2 % material/bin/HResults -h -I material/data/rm1_eval
    ,-------------------------------------------------------------.
    | HTK Results Analysis at Sat Nov  4 22:01:55 2023             |
    | Ref: material/data/rm1_eval/rm1_eval_word.mlf                |
    | Rec: ex2_m/results                                           |
    |    : ex2_f/results                                           |
    |==============================================================|
    |          # Snt |  Corr    Sub    Del    Ins    Err  S. Err  |
    |--------------------------------------------------------------|
    | Sum/Avg |  570  |  90.70   9.30   0.00   0.00   9.30   9.30  |
    `--------------------------------------------------------------'
```

From the results, the female model has the highest error rate (10.74%) and male model has the lowest error rate (8%). Meanwhile the combined model has the intermediate error rate (9.3%)

Benefits of gender dependent model:

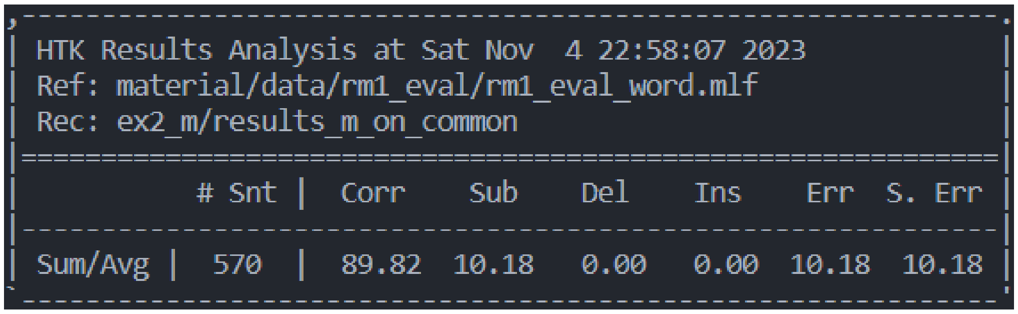
- There are inherent differences in the vocal tract length and shape between males and females. Gender-dependent models can be optimized to better capture these characteristics. These models can better account for the variability within a gender group, which can be less than the variability across genders.
- If the dataset contains an equal number of male and female voice samples, gender-dependent models are likely to perform very well

Drawbacks of gender dependent model:

- To optimize the performance of each gender-dependent model, as well as their overall efficiency, it is necessary that the training datasets for males and females are balanced. Observing the performance above, it appears that the female model does not achieve the same level of accuracy as its male counterpart, which can be attributed to the difference in the amount of training data—the female data being only 2/5 the size of the male dataset. Consequently, when the training data is skewed towards one gender, models specifically designed for each gender may struggle to generalize effectively.

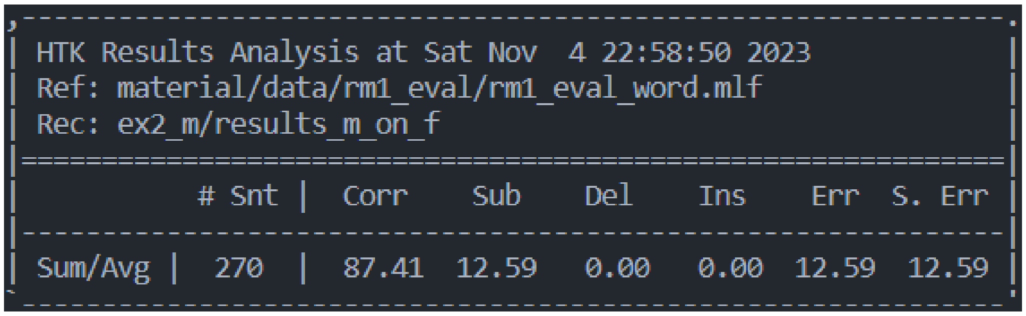
To test this theory, we can test the male and female model on the combined dataset and also on the opposite gender dataset

Testing male model on the common evaluation dataset



```
,-------------------------------------------------------------.
| HTK Results Analysis at Sat Nov  4 22:58:07 2023            |
| Ref: material/data/rm1_eval/rm1_eval_word.mlf               |
| Rec: ex2_m/results_m_on_common                              |
|=============================================================|
|          # Snt |  Corr    Sub    Del    Ins    Err  S. Err |
|-------------------------------------------------------------|
| Sum/Avg |  570  |  89.82  10.18   0.00   0.00  10.18  10.18 |
`-------------------------------------------------------------'
```

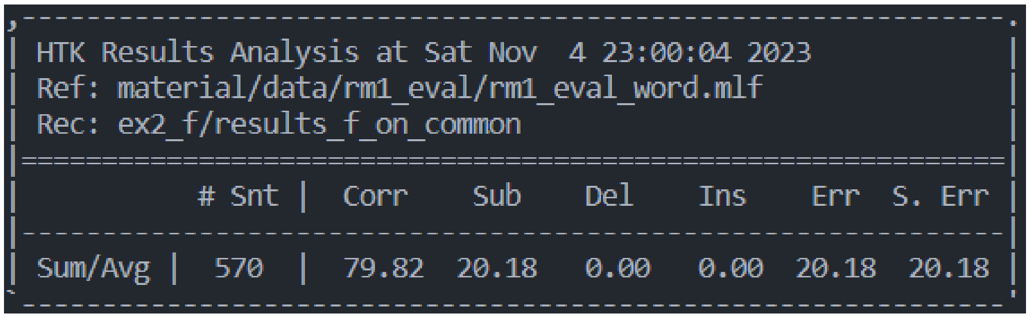
Testing male model on the female dataset



```
,-------------------------------------------------------------.
| HTK Results Analysis at Sat Nov  4 22:58:50 2023            |
| Ref: material/data/rm1_eval/rm1_eval_word.mlf               |
| Rec: ex2_m/results_m_on_f                                   |
|=============================================================|
|          # Snt |  Corr    Sub    Del    Ins    Err  S. Err |
|-------------------------------------------------------------|
| Sum/Avg |  270  |  87.41  12.59   0.00   0.00  12.59  12.59 |
`-------------------------------------------------------------'
```

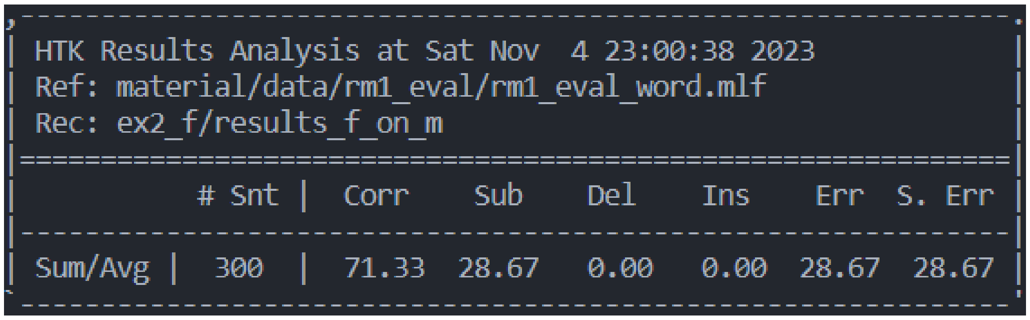
In both cases, the error rates are higher than the male model on male dataset (8% error)

Testing female model on the common evaluation dataset



```
,-------------------------------------------------------------.
| HTK Results Analysis at Sat Nov  4 23:00:04 2023            |
| Ref: material/data/rm1_eval/rm1_eval_word.mlf               |
| Rec: ex2_f/results_f_on_common                              |
|=============================================================|
|          # Snt |  Corr    Sub    Del    Ins    Err  S. Err |
|-------------------------------------------------------------|
| Sum/Avg |  570  |  79.82  20.18   0.00   0.00  20.18  20.18 |
`-------------------------------------------------------------'
```

Testing female model on the male dataset



```
,-------------------------------------------------------------.
| HTK Results Analysis at Sat Nov  4 23:00:38 2023            |
| Ref: material/data/rm1_eval/rm1_eval_word.mlf               |
| Rec: ex2_f/results_f_on_m                                   |
|=============================================================|
|          # Snt |  Corr    Sub    Del    Ins    Err  S. Err |
|-------------------------------------------------------------|
| Sum/Avg |  300  |  71.33  28.67   0.00   0.00  28.67  28.67 |
`-------------------------------------------------------------'
```

In both cases, the error rates are much higher than the female model on the female dataset (10.74% error). This is because the females have much fewer training data than male dataset

In conclusion, the gender dependent model only works well on data that contains mostly of that given gender, and the combined model works with good generalization when the male and female datasets are balanced.

b) Assume that you know the gender of the speaker beforehand. Now that you have the male and female gender dependent models, and the gender independent model, what would be the best configuration to use for recognition?

With the measured error rates of the three models above, we can conclude that if the speaker is male, it is advisable to use the male-dependent model because of its best performance compared to the other models, as it has the lowest error rate of 8.0. However, if we know the speaker is female, we should choose the combined model with an error rate of 9.30.

c) Report the number of Gaussians in the final models. For a model under hmm-6, you can fetch it with command "grep MEAN hmm-6/hmmdefs | wc -l"

The commands for printing the number of Gaussian models are:
grep MEAN hmm-2/hmmdefs | *wc -l*
grep MEAN hmm-3/hmmdefs | *wc -l*
grep MEAN hmm-4/hmmdefs | *wc -l*
grep MEAN hmm-5/hmmdefs | *wc -l*
grep MEAN hmm-6/hmmdefs | *wc -l*

```
grep MEAN hmm-f-2/hmmdefs | wc -l
grep MEAN hmm-f-3/hmmdefs | wc -l
grep MEAN hmm-f-4/hmmdefs | wc -l
grep MEAN hmm-f-5/hmmdefs | wc -l
grep MEAN hmm-f-6/hmmdefs | wc -l

grep MEAN hmm-m-2/hmmdefs | wc -l
grep MEAN hmm-m-3/hmmdefs | wc -l
grep MEAN hmm-m-4/hmmdefs | wc -l
grep MEAN hmm-m-5/hmmdefs | wc -l
grep MEAN hmm-m-6/hmmdefs | wc -l
```

The number of Gaussian models are reported below

- For combined model:

hmm-2: 133
hmm-3: 266
hmm-4: 425
hmm-5: 658
hmm-6: 1058

- For female model:

hmm-2: 133
hmm-3: 224
hmm-4: 266
hmm-5: 287
hmm-6: 304

- For male model:

hmm-2: 133
hmm-3: 261
hmm-4: 383
hmm-5: 535
hmm-6: 760