Assignment 2: Speaker Verification and Source Separation

Alli Khadga Jyoth - M23CSA003

Github link : [KhadgaA/Speaker-verification-and-speration (github.com)]

Question 1: Speaker Verification

Task 1 & 2: Replicating the Results and Comparisons

The task is to evaluate the pretrained models on the VoxCeleb1-H dataset and compare the results with the results reported in the paper.

Models	Reported %EER	Ours %EER	Delta
Hubert Large	1.678	2.023	0.345
WavLM Base+	1.758	2.018	0.26
WavLM Large	1.318	1.761	0.443

The difference observed in the reported results and the results observed by me is because of not using Adaptive-S Norm for the embeddings. In the literature it is observed that using AS-Norm increases the performance of model by 5 - 10 %, which is inline with what we observe. The reason for not using AS-Norm in our testing is, it requires a complicated process of finding Cohorts, imposters and using them to find the adaptive normalization coefficients.

Task 3: Evaluating models on Kathbadh Dataset

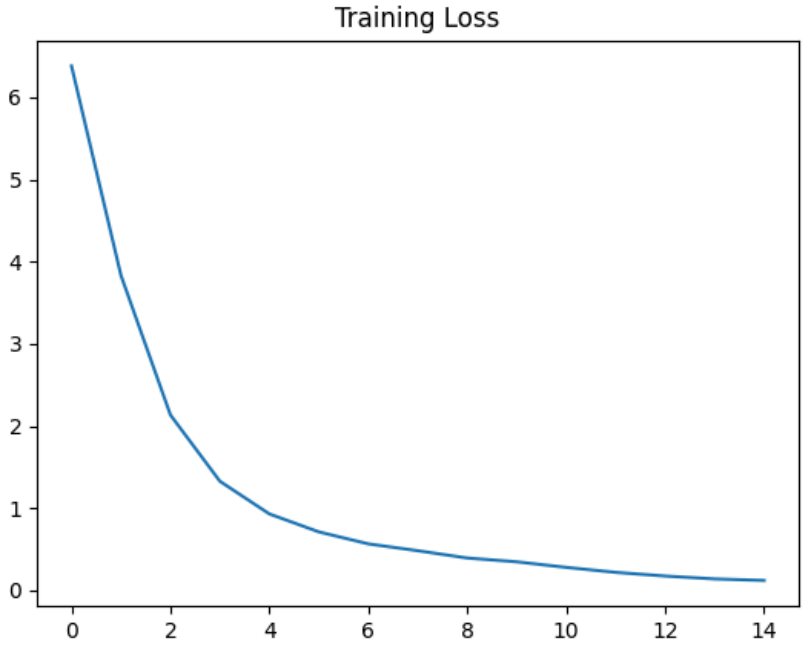
In this task we are required to select one language from the Kathbadh dataset and evaluate our models on the test set. So our language of choice is **Telugu.**

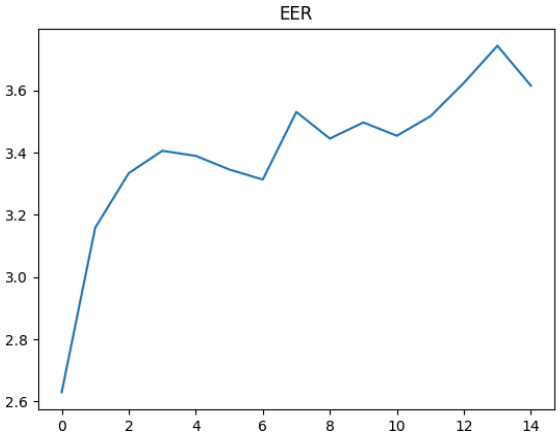
Models	Ours %EER
Hubert Large	3.873
WavLM Base+	3.708
WavLM Large	**3.402**

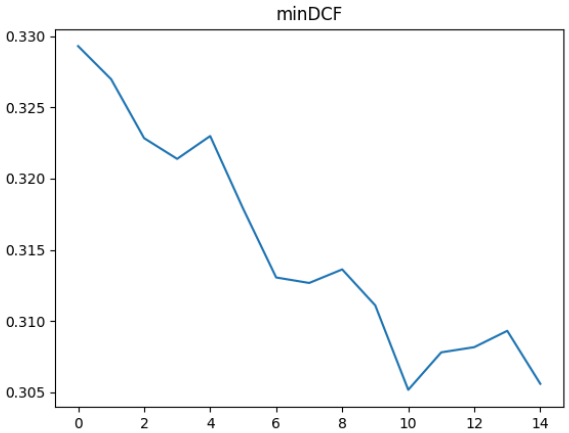
Task 3: Fine Tuning on Kathbadh Dataset

In this task we are required to select the best performing model on the Kathbadh dataset and then fine tune the model on the dev set. And report the results.

For the loss function we are using additive angular margin Softmax (AAM) loss, with a margin of 0.3. The learning rate is set to 5e-6. The batch size is set to 32. With Adam optimizer. The model is finetuned for 15 epochs.







Results:

Model	%EER
WavLM Large	**3.615**

Task 5: Analysis of results

From the figure of train loss, it is evident that the training loss is decreasing continuously, but the EER is increasing on the test data, this might be due to

1. Overfitting: The discrepancy between training loss and test EER suggests overfitting. Regularization techniques can help solve this.
2. Model Complexity: Fine-tuning may require careful hyperparameter tuning and architectural adjustments.