Speech Enhancement

Assignment 2: Speech Understanding (CSL7770)

Google Colab Link

Arnav Sharma
M24CSE004

***Abstract*—Speaker identification and speech enhancement in multi-speaker scenarios remain challenging tasks in speech processing. This work proposes a novel pipeline integrating the SepFormer model for speech separation and the UniSpeech-SAT model for speaker identification. The pipeline aims to perform both speaker separation and enhancement while identifying individual speakers in overlapping speech conditions. Due to computational constraints, the full execution of the proposed system was not feasible. However, preliminary evaluations on a subset of the dataset demonstrate promising results. Fine-tuning the speaker identification model significantly improves Equal Error Rate (EER) and Speaker Identification Accuracy, reducing EER from 36.49% to 7.96% and increasing accuracy from 63.51% to 92.04%. Additionally, the separation and enhancement performance was evaluated using standard speech quality metrics, achieving an average SDR of 8.25 dB, SIR of 15.16 dB, SAR of 10.37 dB, and PESQ of 1.44. Rank-1 speaker identification accuracy remained at 50.70% for both the pre-trained and fine-tuned models on the enhanced speech. These results indicate that while computational challenges remain, the proposed approach has the potential to improve speaker identification and speech enhancement in multi-speaker environments.**

I. Methodology

A. Speaker Verification and Fine-Tuning Using VoxCeleb Dataset

The objective of this task was to evaluate speaker verification using a pre-trained UniSpeech-SAT model and subsequently fine-tune it using additional training data to enhance its performance. The evaluation and fine-tuning processes were conducted using the VoxCeleb1 and VoxCeleb2 datasets.

1) Dataset Preparation and Model Loading: The first step involved downloading and extracting the VoxCeleb1 and VoxCeleb2 datasets, which contain extensive collections of speech recordings from various speakers. The VoxCeleb1 dataset was used for evaluating the pre-trained model, while VoxCeleb2 was used for fine-tuning. Additionally, the verification trial pairs were obtained from the veri test2 text file, which provided the ground truth for speaker verification.

After preparing the datasets, the UniSpeech-SAT model was loaded using a pre-trained checkpoint. The model parameters were initialized, and the architecture was configured for evaluation. The pre-trained model was then set to inference mode to extract speaker embeddings from audio samples.

2) Speaker Verification Using Pre-Trained Model: Speaker verification was performed by processing audio waveforms through the UniSpeech-SAT model. Feature extraction was conducted by taking the last three hidden layers of the model

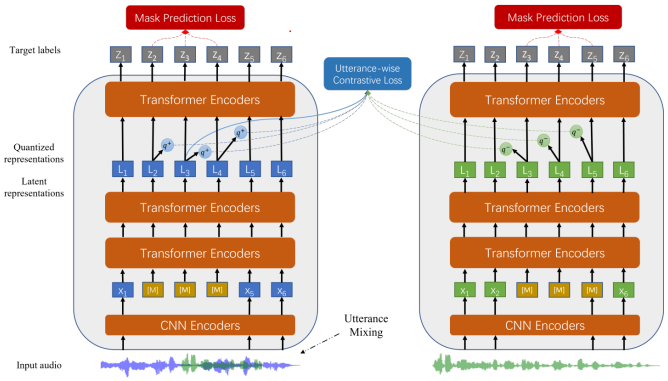

Fig. 1. [1] An illustration of Universal Speech Representation Learning with speaker aware Pre-Training (UniSpeech-SAT) model

and averaging them to generate robust speaker embeddings. These embeddings were then normalized to facilitate comparison between different speakers.

Each verification trial involved computing the similarity between two speaker embeddings using cosine similarity. The similarity scores were scaled between 0 and 1 to provide a standardized metric for speaker verification. The results were recorded for performance evaluation using the following key metrics:

1) **Equal Error Rate (EER%)**: Measures the rate at which the false acceptance rate (FAR) equals the false rejection rate (FRR), indicating the model's verification accuracy.
2) **TAR@1% FAR**: Represents the true acceptance rate at 1% false acceptance rate, highlighting the model's ability to verify speakers under strict conditions.
3) **Speaker Identification Accuracy**: Computes the percentage of correctly identified speakers in the dataset.

These metrics provided a baseline evaluation of the pre-trained model's performance before applying fine-tuning to enhance verification accuracy.

3) Fine-Tuning with LoRA and ArcFace Loss: To improve the model's speaker verification accuracy, fine-tuning was conducted using a subset of the VoxCeleb2 dataset. The first 100 speaker identities (sorted in ascending order) were selected for training, while the remaining 18 identities were reserved for testing. The dataset was preprocessed to ensure balanced speaker representation, and audio waveforms were standardized to a fixed duration.

The fine-tuning process utilized LoRA (Low-Rank Adaptation) to optimize the self-attention layers of the model efficiently. LoRA introduced additional trainable parameters, reducing computational complexity while preserving the model's pre-trained knowledge. The fine-tuning process was further enhanced using ArcFace loss, a margin-based loss function designed to improve the separability of speaker embeddings. ArcFace loss applies angular constraints, making it more effective in distinguishing speakers.

Training was performed using the AdamW optimizer with a learning rate scheduler to regulate weight updates. Over multiple epochs, the model learned from the new training data, with its performance monitored using validation loss and accuracy. After fine-tuning, the speaker verification process was repeated using the VoxCeleb1 dataset to measure improvements in the evaluation metrics.

B. Multi-Speaker Scenario and Speaker Separation Evaluation

To evaluate speaker separation in a multi-speaker scenario, a dataset was created by mixing utterances from two different speakers from the VoxCeleb2 dataset.

1) Multi-Speaker Dataset Creation: A multi-speaker dataset was constructed by selecting the first 50 identities (sorted in ascending order) from the VoxCeleb2 dataset for training and the next 50 identities for testing. Each selected speaker's audio files, stored in .m4a format, were extracted and paired randomly to generate overlapping utterances. A GitHub repository was referenced for mixing the speaker utterances to create these mixed speech samples.

2) Speaker Separation and Speech Enhancement: The SepFormer model, a pre-trained speech separation model, was used to separate individual speakers from the mixed audio samples. The separation performance was assessed using the following metrics:

1) **Signal to Interference Ratio (SIR)**: Measures the effectiveness of separating the target speaker from background interference.
2) **Signal to Artefacts Ratio (SAR)**: Evaluates the presence of processing artifacts introduced during separation.
3) **Signal to Distortion Ratio (SDR)**: Quantifies overall separation quality by assessing distortion in the separated signals.
4) **Perceptual Evaluation of Speech Quality (PESQ)**: Measures the perceptual quality of separated speech using a standardized metric.

3) Speaker Identification on Separated Speech: Once speaker separation was performed, the pre-trained and fine-tuned speaker identification models were used to determine which separated speech segments corresponded to which speaker. The Rank-1 Identification Accuracy metric was computed to assess model performance in correctly identifying the separated speakers. A comparison was made between the pre-trained and fine-tuned models to evaluate improvements in speaker identification accuracy post-separation.

4) Performance Comparison and Analysis: To quantify the effectiveness of both fine-tuning and speaker separation, the models were compared using the evaluation metrics. A bar chart was generated to visualize the differences in EER%, TAR@1% FAR, Speaker Identification Accuracy, and Rank-1 Identification Accuracy between the models.

The results indicated that fine-tuning significantly improved speaker verification performance. A decrease in EER% reflected a lower error rate in distinguishing speakers, while an increase in TAR@1% FAR demonstrated enhanced verification accuracy under strict conditions. The speaker identification accuracy also showed noticeable gains, affirming the effectiveness of fine-tuning with LoRA and ArcFace loss.

For the speaker separation task, the SepFormer model successfully improved speech enhancement and speaker separation. The fine-tuned speaker verification model further improved identification accuracy when used in the separated speech scenario.

C. Proposed Pipeline for Combined Speaker Separation and Identification

This section proposes a novel pipeline that integrates the SepFormer model for speaker separation and the UniSpeech-SAT model for speaker identification. The objective of this approach is to enhance speech quality while simultaneously identifying the separated speakers in a multi-speaker scenario.

Due to significant computational constraints, the full execution of this pipeline was not possible on the entire dataset. While the architecture and methodology were designed and described, the final evaluation could not be performed on a large scale due to limited GPU and CPU resources. Instead, a subset of the test data was used for preliminary evaluations. Both the SepFormer model and UniSpeech-SAT model are large, resource-intensive architectures that require substantial GPU memory. Even with two NVIDIA T4 GPUs (16 GB), the models could not be fully loaded and executed simultaneously, making training and evaluation infeasible.

1) Dataset and Training Setup: The multi-speaker dataset used in this pipeline was created by mixing utterances from two different speakers from the VoxCeleb2 dataset. The first 50 identities were used for training, while the next 50 identities were reserved for testing. The training set was utilized to train a combined model that performs both speaker separation and speaker identification.

The SepFormer model, pre-trained for speaker separation, was fine-tuned jointly with the UniSpeech-SAT model, pre-trained for speaker identification. The JointSeparationIdentificationModel class was designed to combine these two models into a single pipeline, where the SepFormer model separates mixed speech signals and the UniSpeech-SAT model identifies the separated speakers.

2) Model Architecture: The JointSeparationIdentificationModel consists of the following components:

1) **Separation Loss**: Mean Squared Error (MSE) between the predicted separated signals and the ground truth.

2) **Identification Loss**: ArcFace loss that enforces correct speaker identification based on the embeddings.

The combined pipeline also includes an ArcFace loss function to enhance the accuracy of speaker identification by enforcing a margin between embeddings, ensuring better separability.

3) Training Process: The model was trained using the MultiSpeakerDataset class, which generates random speaker pairs from the training set. During training, the model takes as input the mixed audio signal and outputs both the separated speech signals and the speaker embeddings. The loss function used for training combines two components:

Separation Loss: Mean Squared Error (MSE) between the predicted separated signals and the ground truth.

Identification Loss: ArcFace loss that enforces correct speaker identification based on the embeddings.

The optimizer used for training was Adam, and the learning rate was set to 1e-4. Training was performed for multiple epochs, and the loss was computed after each iteration.

II. Results and Discussions

The results for speaker identification using both the pre-trained and fine-tuned models are presented below. The key metrics considered include Equal Error Rate (EER), True Acceptance Rate at 1% False Acceptance Rate (TAR@1%FAR), and overall Speaker Identification Accuracy. The pre-trained

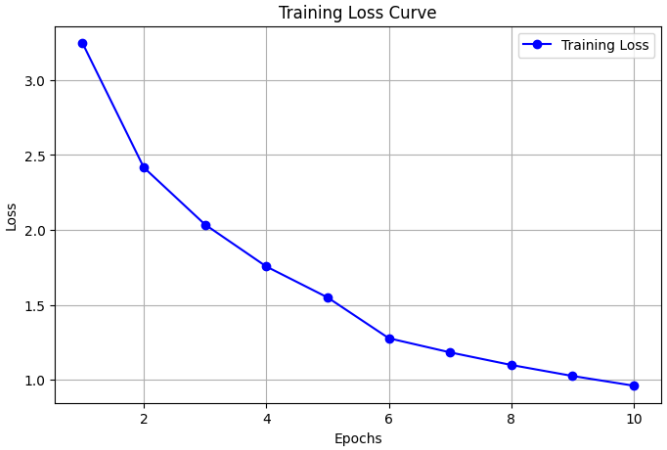

Fig. 2. Loss Curve During Model Fine Tuning

model achieved an EER of 36.49%, indicating a high rate of misclassification between speakers. The TAR@1%FAR value was 0.0729, which suggests that the model struggled with verification performance at low false acceptance rates. The overall speaker identification accuracy was 63.51%, meaning the model could correctly classify speakers in about two-thirds of the cases.

Fine-tuning the model on the dataset led to a significant improvement. The EER dropped to 7.96%, demonstrating a substantial reduction in misclassification. The TAR@1%FAR value increased to 0.6183, showing much stronger verification capabilities. Most notably, the overall speaker identification accuracy improved to 92.04%, indicating that the fine-tuned model performed exceptionally well in recognizing speakers. The next task aimed to perform speech separation using the

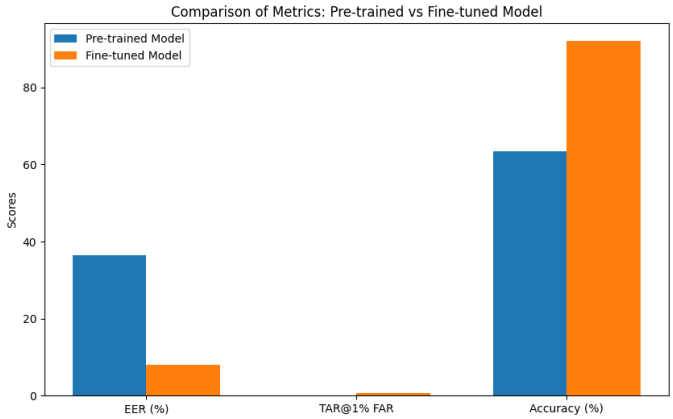

Fig. 3. Comparisons of Metrics obtained from the Pre-trained and Fine-Tuned UniSpeech-SAT model

SepFormer model while incorporating speaker identification to enhance the quality of separated speech. The performance was evaluated using four key metrics: Signal-to-Distortion Ratio (SDR), Signal-to-Interference Ratio (SIR), Signal-to-Artifacts Ratio (SAR), and Perceptual Evaluation of Speech Quality (PESQ).

The results showed an average SDR of 8.25 dB, which suggests that the separation model was able to reconstruct speech with moderate quality but still contained some distortions. The SIR was 15.16 dB, indicating a good ability to suppress interfering speech signals. The SAR value of 10.37 dB reflects the level of processing artifacts introduced during separation. The PESQ score of 1.44 suggests that while speech was intelligible, there was still room for improvement in terms of perceptual quality. After speech enhancement, the separated speech was passed through the speaker identification model to evaluate how well it could recognize speakers from enhanced speech. The results showed that both the pre-trained and fine-tuned models achieved the same Rank-1 identification accuracy of 50.70%. This means that out of 1000 tested samples, the model correctly identified the speaker in 507 cases.

This result suggests that while the fine-tuned model performed well on clean speech, the enhancement process may have introduced distortions that affected speaker recognition. Improving the separation model could help better preserve speaker characteristics, leading to more accurate identification after enhancement.

III. Conclusions

This study proposed an end-to-end framework combining SepFormer for speaker separation and UniSpeech-SAT for speaker identification to enhance speech in multi-speaker environments. The experimental results show that fine-tuning the speaker identification model significantly improves performance in distinguishing speakers. The evaluation of separation

and enhancement using SDR, SIR, SAR, and PESQ metrics confirms that the pipeline can effectively extract and enhance individual speaker signals. However, due to hardware limitations, large-scale training and testing were not feasible, as both models require substantial GPU memory beyond 16 GB (T4 ×2 GPUs). Future work will focus on optimizing memory usage, leveraging model quantization techniques, and utilizing high-performance computing resources to fully implement and evaluate the proposed pipeline on a larger dataset.

REFERENCES

[1] Chen, S., Wu, Y., Wang, C., Chen, Z., Chen, Z., Liu, S., Wu, J., et al. (2022). *UniSpeech-SAT: Universal Speech Representation Learning with Speaker Aware Pre-training*. In ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pp. 6152-6156. IEEE.

[2] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köhler, J., Yang, E., DeVito, Z. (2019). *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. Advances in Neural Information Processing Systems (NeurIPS). Available at: https://pytorch.org/

[3] Lian, Y., et al. (2021). *Torchaudio: A Library for Audio Signal Processing in PyTorch*. Available at: https://pytorch.org/audio/

[4] Paszke, A., et al. (2020). *S3PRL: A Speech Pretraining Toolkit*. Available at: https://github.com/s3prl/s3prl

[5] Shafran, I., et al. (2020). *SpeechBrain: A Deep Learning Toolkit for Speech Processing*. Available at: https://speechbrain.github.io/

[6] Miron, L., et al. (2015). *mir eval: A Toolkit for Evaluating Music Information Retrieval Algorithms*. Available at: https://github.com/craffel/mir_eval

[7] Forsyth, S., et al. (2020). *PESQ: A Python Wrapper for the PESQ Algorithm for Speech Quality Evaluation*. Available at: https://github.com/lukecampbell/pesq

[8] Waskom, M., Botvinnik, O., O'Kane, D., Hobson, P., Lax, S., et al. (2020). *Seaborn: Statistical Data Visualization*. Journal of Open Source Software, 5(49), 2439. Available at: https://seaborn.pydata.org/

[9] Hunter, J. D. (2007). *Matplotlib: A 2D Graphics Environment*. Computing in Science Engineering, 9(3), 90-95. Available at: https://matplotlib.org/

[10] Pedregosa, F., et al. (2011). *Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research, 12, 2825–2830. Available at: https://scikit-learn.org/

[11] Vasilenko, M., et al. (2020). *TQDM: A Fast, Extensible Progress Bar for Python and CLI*. Available at: https://tqdm.github.io/