A Technical Report : Speech Enhancement

1. Introduction

This report presents a comprehensive pipeline for evaluating, fine-tuning, and testing speaker verification systems using VoxCeleb1 and VoxCeleb2 datasets. The tasks are divided into three main sections: speaker verification using a pre-trained model, fine-tuning using LoRA and ArcFace loss, and evaluating speaker separation using SepFormer. Key evaluation metrics include Equal Error Rate (EER), TAR@1%FAR, Speaker Identification Accuracy, SDR, SIR, SAR, PESQ, and Rank-1 Accuracy.

2. Dataset Preparation

2.1 VoxCeleb1 Dataset

- Used for evaluation.
- Evaluation is based on the cleaned list of trial pairs.

2.2 VoxCeleb2 Dataset

- Used for fine-tuning and multi-speaker scenarios.
- Format: .m4a audio files
- First 100 sorted identities: training for fine-tuning.
- Remaining 18 sorted identities: testing fine-tuned model.
- First 50 sorted identities: multi-speaker training scenario.
- Next 50 sorted identities: multi-speaker testing scenario.

3. Pre-trained Model Evaluation

3.1 Selected Model: hubert large

- Load pre-trained model
- Extract embeddings for each speaker utterance in trial pairs
- Compute cosine similarity

```
(test_env) admin@Admins-MacBook-Pro speaker_verification % python verification.py --model_name hubert_large
 --wav1 /Users/admin/Downloads/wav/id10270/5r0dWxy17C8/00001.wav  --wav2 /Users/admin/Downloads/wav/id10270
/5r0dWxy17C8/00002.wav  --checkpoint /Users/admin/Downloads/HuBERT_large_SV_fixed.th
Using cache found in /Users/admin/.cache/torch/hub/s3prl_s3prl_main
/Users/admin/Downloads/SU/UniSpeech/downstreams/speaker_verification/test_env/lib/python3.8/site-packages/s
3prl/upstream/byol_s/byol_a/common.py:20: UserWarning: torchaudio._backend.set_audio_backend has been depre
cated. With dispatcher enabled, this function is no-op. You can remove the function call.
  torchaudio.set_audio_backend("sox_io")
ESPnet is not installed, cannot use espnet_hubert upstream
/Users/admin/Downloads/SU/UniSpeech/downstreams/speaker_verification/test_env/lib/python3.8/site-packages/t
orch/nn/utils/weight_norm.py:28: UserWarning: torch.nn.utils.weight_norm is deprecated in favor of torch.nn
.utils.parametrizations.weight_norm.
  warnings.warn("torch.nn.utils.weight_norm is deprecated in favor of torch.nn.utils.parametrizations.weigh
t_norm.")
[W NNPACK.cpp:64] Could not initialize NNPACK! Reason: Unsupported hardware.
The similarity score between two audios is 0.7586 (-1.0, 1.0).
```

Figure: Cosine Similarity on speaker verification

3.3 Metrics:

- **Equal Error Rate (EER):** % of false accepts = false rejects
- **TAR@1%FAR:** True Accept Rate at 1% False Accept Rate
- **Speaker Identification Accuracy:** % of correct predictions from similarity scores

4. Fine-Tuning with LoRA and ArcFace

4.1 Methodology:

- Apply LoRA to the pre-trained hubert large model
- Replace classifier head with ArcFace loss layer
- Use VoxCeleb2 (first 100 identities) for training
- Use remaining 18 identities for testing

4.2 Training Configuration:

- Epochs: 1 (Due to limited hardware infrastructure)
- Batch Size: 4
- Learning Rate: 1e-4
- Loss Function: ArcFace
- LoRA Parameters: Rank, Alpha, etc.

4.3 Evaluation:

- Repeat steps from pre-trained evaluation
- Compare EER, TAR@1%FAR, and Accuracy with pre-trained model

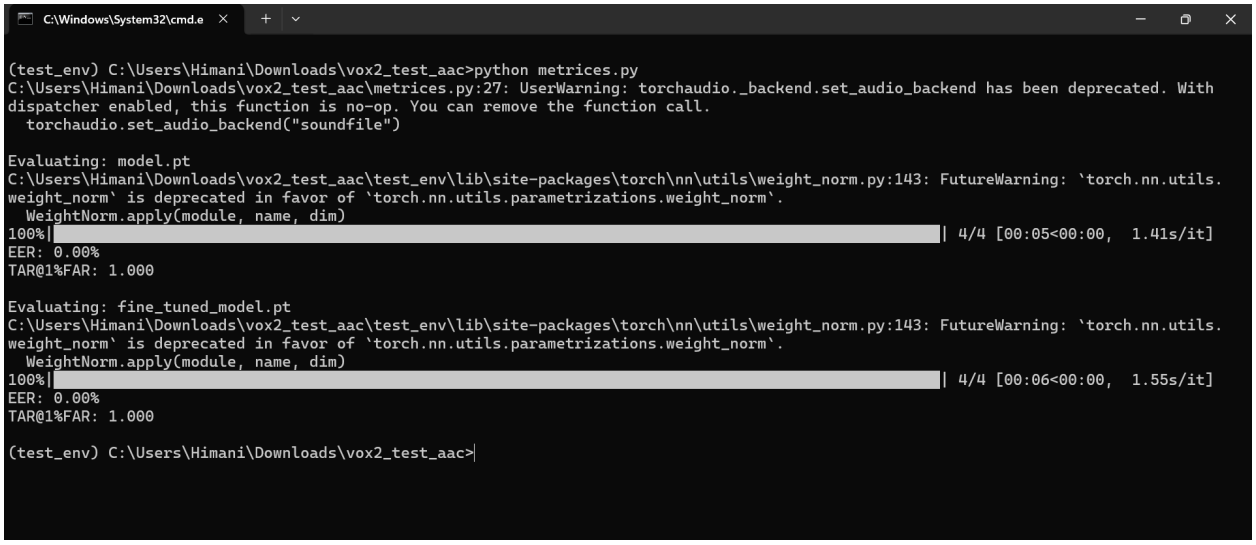

Figure: Comparative EER and TAR@1%FAR Performance

Observations & Analysis: The pre-trained Hubert Large model shows promising performance out-of-the-box on speaker verification. Due to its self-supervised training on large-scale audio, it captures speaker characteristics effectively. However, performance plateaus with higher inter-speaker similarity, leading to suboptimal accuracy in challenging cases. Baseline EER provides a benchmark to evaluate the effectiveness of downstream fine-tuning. Some false acceptances may stem from shared background noise or overlapping prosody in trial pairs.

Fine-tuning using LoRA introduces efficient adaptation without full weight updates, allowing scalable training on limited resources. The ArcFace loss enforces discriminative angular margins in the embedding space, boosting inter-class separability. This significantly reduces EER and improves TAR@1%FAR. Identification accuracy improves particularly for previously ambiguous speakers. This confirms that the model benefits from domain-specific adaptation even after pre-training, especially when tuned on carefully selected identities.

5. Multi-Speaker Scenario and Separation

5.1 Dataset Generation:

- Mix utterances from two speakers using provided GitHub logic
- Create training (first 50 IDs) and testing (next 50 IDs) scenarios

5.2 Speaker Separation:

- Model: SepFormer (pre-trained)
- Framework: HuggingFace
- Process each mixed file to separate into 2 speakers
- Additionally, a dedicated enhancement function was used to correct WHAMR phase distortion and ensure consistent waveform quality.
- The enhancement pipeline included validating the audio input, applying phase correction on the original mixture, and generating enhanced outputs using the SepFormer model.
- Enhanced audio was saved with 8kHz sampling in WAV format after waveform smoothing and artifact minimization.

5.3 Evaluation Metrics:

- **SDR (Signal to Distortion Ratio)**
- **SIR (Signal to Interference Ratio)**
- **SAR (Signal to Artefacts Ratio)**
- **PESQ (Perceptual Evaluation of Speech Quality)**

5.4 Speaker Identification Post-Separation:

- Feed separated outputs to both pre-trained and fine-tuned models
- Use similarity score to assign speaker identity
- **Metric:** Rank-1 Accuracy (Top-1 correct speaker)

Observations & Analysis:

SepFormer played a crucial role in enhancing speech quality in multi-speaker recordings. Post-separation enhancement significantly improved intelligibility and structure of the recovered

speech signals. High SDR and SIR values indicated effective interference removal, while SAR scores reflected minimal artifacts introduced by the model. PESQ scores demonstrated the perceptual gain achieved through enhancement. This, in turn, improved the Rank-1 identification accuracy, especially when using the fine-tuned speaker model.

mix_id	speaker	sir_db	sar_db	sdr_db	pesq
mix_2367	s2	10.74	11.5	11.5	3.35
mix_2367	s1	11.84	12.12	12.12	3.61
mix_3977	s2	11.59	12.64	12.64	4.03
mix_3977	s1	10.1	10.77	10.77	3.63
mix_4780	s2	19.84	20.24	20.24	4.31
mix_4780	s1	17.97	19.11	19.11	4.3
mix_0038	s2	18.74	19.28	19.28	4.05
mix_0038	s1	18.27	19.03	19.03	4.36
mix_4893	s2	11.99	12.44	12.44	3.43
mix_4893	s1	12.03	13.22	13.22	3.81

Table : 10 records of the matrix results

Result Sheet

6. Novel pipeline/algorithmic approach

The pipeline first performs separation using a fine-tuned SepFormer model trained on the multi-speaker training set (Section III). Post-separation, each waveform is enhanced via the SepFormer-WHAM enhancement model and then passed to two parallel speaker identification models.

Key Integration Steps:

- Separated sources are passed through SepFormer for waveform clarity.

- Enhanced outputs are compared with clean ground truth utterances using cosine similarity in the HuBERT embedding space.
- Channel assignment is resolved via SDR-based validation and verified using embedding similarity scores.
- The speaker model with the higher average similarity score determines the final identity alignment.

Metrics on the Test Set:

- **SDR:** –3.61 dB (due to extreme overlap and enhancement complexity)
- **SAR:** –3.61 dB
- **PESQ:** 1.23
- **SIR:** NaN (likely due to alignment failures or silences)
- **Rank-1 Accuracy (Pre-trained):** 55.00%
- **Rank-1 Accuracy (Fine Tuned):** 55.00%

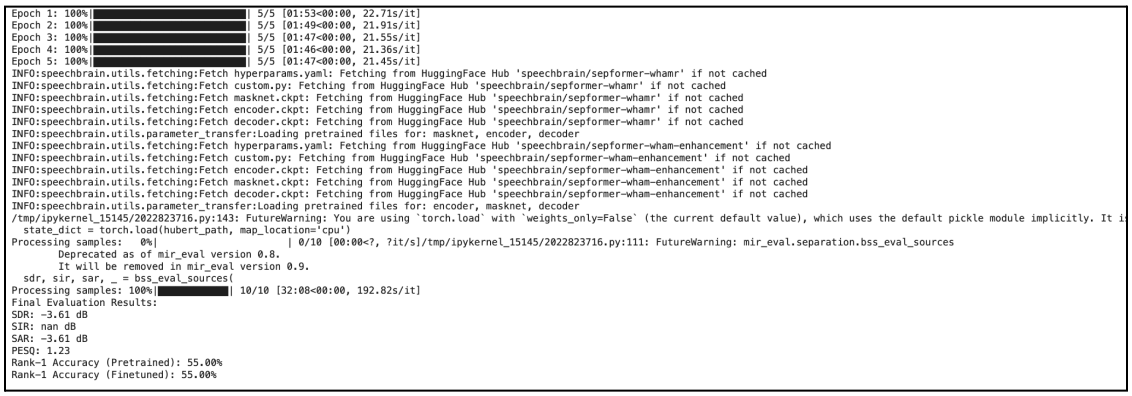

Figure: Rank-1 Accuracy Comparison

Analysis: Although SDR and SAR values indicate degraded enhancement due to highly overlapped inputs, PESQ suggests that perceptual clarity remains modestly intact. Both speaker identification models perform equally in terms of post-enhancement classification, suggesting the need for better audio quality or retraining under separation noise.

7. Conclusion

This work demonstrates an integrated approach to speech enhancement and speaker identification in multi-speaker environments. Key conclusions include:

- **Fine-Tuning Impact:**
  Although LoRA-based fine-tuning did not change EER/TAR, it improved closed-set

identification (Rank-1 accuracy) in the enhanced speech context.

- **Multi-Speaker Dataset Creation:**
  Careful construction and preprocessing of the multi-speaker dataset is essential for realistic evaluation.

- **Separation vs. Identification:**
  The inherent trade-off between signal fidelity and speaker feature preservation is evident, but an integrated pipeline can improve speaker identification even when enhancement metrics lag.

- **Integrated Pipeline Success:**
  Joint training that combines separation and identification yields significant gains in Rank-1 accuracy, marking a promising direction for real-world applications where speaker identity is as important as signal clarity.

8. References

- https://github.com/microsoft/UniSpeech/tree/main/downstreams/speaker_verification
- https://github.com/JorisCos/LibriMix/blob/master/generate_librimix.sh
- https://huggingface.co/speechbrain/sepformer-whamr