Speaker Verification Project – Short Technical Notes

These notes summarize **key concepts, definitions, design choices, and parameter selections** used in the project. They are intended for **quick revision, PPT explanation, and viva/interview preparation**.

---

1. Speaker Verification (SV)

Definition: Speaker Verification is a biometric task that determines whether two speech samples belong to the **same speaker** or **different speakers**.

Why verification (not classification)?

- Real-world systems must handle **unseen speakers**
- Decisions are based on **similarity**, not class labels
- Naturally modeled using **Siamese / metric learning** frameworks

---

2. Siamese Network

Definition: A Siamese Network consists of **two identical neural networks with shared weights**, processing two inputs in parallel.

Why Siamese for this project?

- Learns **speaker embeddings** instead of fixed classes
- Enables **distance-based comparison**
- Generalizes well to new speakers

Output:

- Fixed-dimensional embedding vector (128-D)

---

3. Audio Preprocessing

3.1 Resampling to 16 kHz

Definition: Converting all audio signals to a common sampling rate of **16,000 Hz**.

Why 16 kHz?

- Human speech information is mostly below 8 kHz
- Industry standard for speech models
- Reduces computational cost

3.2 Silence Trimming

Definition: Removal of low-energy non-speech regions.

Why?

- Silence contains no speaker-discriminative information
- Improves model focus on speech
- Reduces noise in embeddings

3.3 Normalization

Definition: Scaling waveform amplitudes to a standard range.

Why?

- Removes loudness variability
- Stabilizes training
- Prevents dominance of high-energy samples

---

4. Log-Mel Spectrogram

Definition: A time–frequency representation where:

- Frequency axis is mapped to the **Mel scale**
- Amplitudes are converted to **log scale**

Why Log-Mel?

- Aligns with human auditory perception
- Compresses dynamic range
- Standard input for speech models

Key Parameters

- **Number of Mel bins (80):**

• Trade-off between frequency resolution and computation

• Widely used in speaker verification literature

• **FFT size (400):**

• ≈ 25 ms window at 16 kHz

• Captures phonetic information

• **Hop length (160):**

• ≈ 10 ms stride

• Preserves temporal continuity

---

5. Padding / Cropping (400 frames)

Why fixed length?

• CNNs require uniform tensor shapes
• Enables batch processing

Why 400 frames?

• Covers sufficient speech content (\~4 seconds)
• Balances temporal context and memory usage

---

6. Methodology 1 – CNN + GRU (From Scratch)

CNN Block

Role:

• Extract local spectral patterns (formants, harmonics)

Limitation:

• Trained from scratch → weak speaker representations

GRU Block (Gated Recurrent Unit)

Definition: A recurrent unit designed to model **temporal dependencies** with fewer parameters than LSTM.

Why GRU?

- Captures speech dynamics
- Lower risk of overfitting

Projection Head

Definition: A fully connected layer that maps features to a **128-D embedding space**.

Why 128-D?

- Common embedding size in biometric systems
- Good balance between discrimination and compactness

---

7. L2 Normalization

Definition: Scaling embeddings to unit length (on a hypersphere).

Why?

- Makes cosine similarity meaningful
- Prevents magnitude-based bias
- Standard practice in metric learning

---

8. Contrastive Loss

Definition: A metric learning loss that:

- Pulls same-speaker embeddings closer
- Pushes different-speaker embeddings apart by a margin
- D = distance between embeddings

y = label (1 = same, 0 = different)

m = margin

Loss:

$L = y * D^2 + (1 - y) * \max(0, m - D)^2$ - **Why Contrastive Loss?** - Directly optimizes similarity space - Suitable for Siamese networks

Limitation:

- Sensitive to weak embeddings
- Benefits from pretraining or hard negatives

---

9. Methodology 2 – Transfer Learning with VGGish

What changed?

- Accepted N_*Mels = 64. So, data is again feature*_extracted
- CNN replaced with **VGGish (speech-pretrained)**
- Siamese framework retained

Why VGGish?

- Trained on large-scale audio data
- Learns robust acoustic representations
- Improves generalization with limited data

---

10. Evaluation Metric – EER (Equal Error Rate)

Definition: The point where:

- False Acceptance Rate (FAR)
- False Rejection Rate (FRR)

are equal.

Why EER?

- Threshold-independent
- Standard metric in speaker verification

Interpretation:

- Lower EER → better system

---

11. Key Experimental Insight

- Siamese formulation was **correct from the start**
- Performance bottleneck was **representation learning**
- Pretrained speech models significantly reduce EER

---

12. Future Improvements

- ECAPA-TDNN backbone
- ArcFace / Angular Softmax loss
- Hard negative mining
- Larger speaker-disjoint evaluation sets

Final Takeaway

Speaker verification performance depends more on embedding quality than network depth. Transfer learning with speech-aware models is essential when data is limited.