Part 1: Implementation Process

Dataset Preparation

The ASVspoof 2019 dataset was downloaded from Kaggle. The dataset consists of Logical Access (LA) and Physical Access (PA) scenarios, where our focus was on LA. The dataset was extracted and organized as per the provided directory structure.

https://www.kaggle.com/datasets/awsaf49/asvpoof-2019-dataset

1.1 Challenges Encountered & Solutions

- Pretrained Model Download Issues:
    - Faced permission errors when fetching the Kaldi X-vector model.
    - Solution: Switched to using SpeechBrain's pretrained X-vector model, which supports PyTorch.
- Dataset Handling & Feature Extraction:
    - Required proper parsing of ASVspoof 2019 protocol files.
    - Solution: Developed a function to extract (utterance_id, label) pairs and match them with audio files.
    - Feature Extraction Issues:
        - Used Log-Mel Spectrograms and MFCC features.
        - Stored features in .ark format using Kaldiio.
- X-Vector Extraction Issues:
    - Original Kaldi models were not accessible.
    - Solution: Implemented SpeechBrain-based X-vector extraction.
- Model Training Difficulties:
    - Initial training showed high EER, indicating poor discrimination.
    - Solution: Fine-tuned training parameters, used AM-Softmax loss, and experimented with learning rates.

1.2 Assumptions Made

- The dataset contains balanced bonafide vs. spoofed samples.
- X-vectors extracted by SpeechBrain retain sufficient speaker information.
- Model generalizes well to unseen spoofing attacks.

Part 2: Model Selection & Analysis

2.1 Why This Model?

- X-Vectors:
  - Pretrained on large speaker recognition datasets.
  - Extracts discriminative speaker embeddings.
- LightCNN & SeNet:
  - Both are effective in classification tasks.
  - CNN-based architectures efficiently model spectro-temporal features.

2.2 How the Model Works?

1. Feature Extraction:
   - Audio is converted into Log-Mel Spectrograms & MFCC features.
   - Features stored in .ark format.
2. X-Vector Extraction:
   - SpeechBrain’s X-vector model generates 512-dimensional embeddings.
3. Model Training:
   - LightCNN & SeNet trained on extracted X-vectors.
   - Uses Cross-Entropy Loss or AM-Softmax Loss.
4. Evaluation:
   - Computes Equal Error Rate (EER).
   - Generates DET curves for performance analysis.

2.3 Performance Results

- EER Score: 9.87% (to be updated)
- Threshold for Best Performance: 0.0019
- Observations:
  - LightCNN showed better generalization than SeNet.
  - AM-Softmax loss improved performance.

2.4 Strengths & Weaknesses

Strengths:

- X-vectors retain high speaker variability.
- LightCNN and SeNet capture subtle spoofing cues.
- SpeechBrain simplifies feature extraction.

Weaknesses:

- Some spoofing attacks remain undetected.
- Performance on unseen attacks needs improvement.
- Sensitive to data augmentation choices.

Part 3: Reflection

3.1 Significant Challenges in Implementation

- Adapting X-vector extraction without Kaldi.
- Optimizing model parameters to reduce EER.
- Handling large-scale feature extraction efficiently.

3.2 Real-World vs. Research Performance

- Real-World:
  - Needs robustness against new spoofing techniques.
  - Likely requires more diverse training data.
- Research Dataset:
  - Limited to predefined spoofing attacks.
  - Easier to achieve high accuracy.

3.3 Additional Data/Resources for Improvement

- Augment dataset with more spoofing techniques.
- Use adversarial training for better generalization.
- Implement PLDA scoring for improved classification.

3.4 Deploying in Production

- Pipeline: Convert audio → Extract X-vector → Classify.
- Deployment Considerations:
  - Optimize model size for real-time detection.
  - Implement an adaptive learning approach to handle new spoofing attacks.