Audio, Data Intelligence and Speech

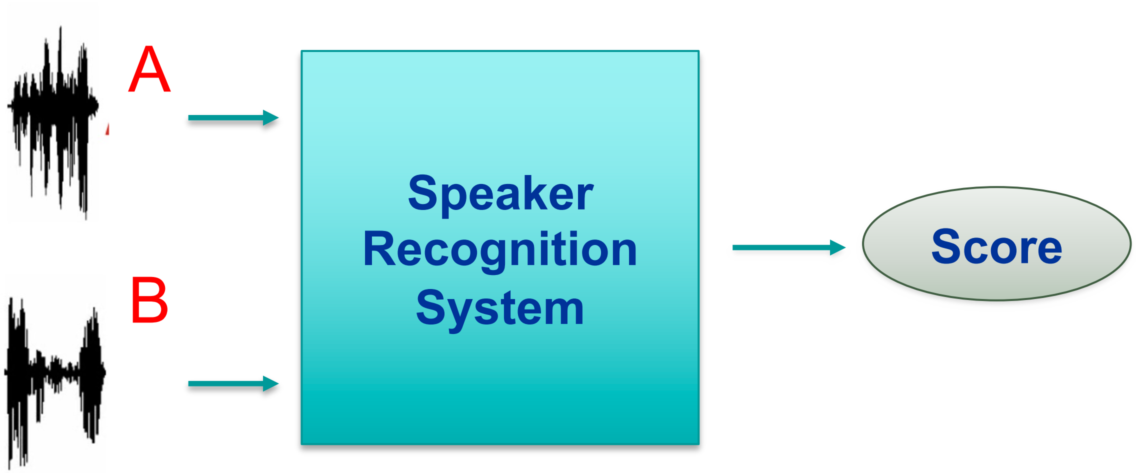
Universidad Autónoma de Madrid

Práctica 2: Reconocimiento de Locutor en VoxCeleb

Escuela Politécnica Superior

Speaker Recognition

- Speaker recognition is the task of identifying a person from his/her voice
  - Speaker verification:
    - Do both voice fragments come from the same person?

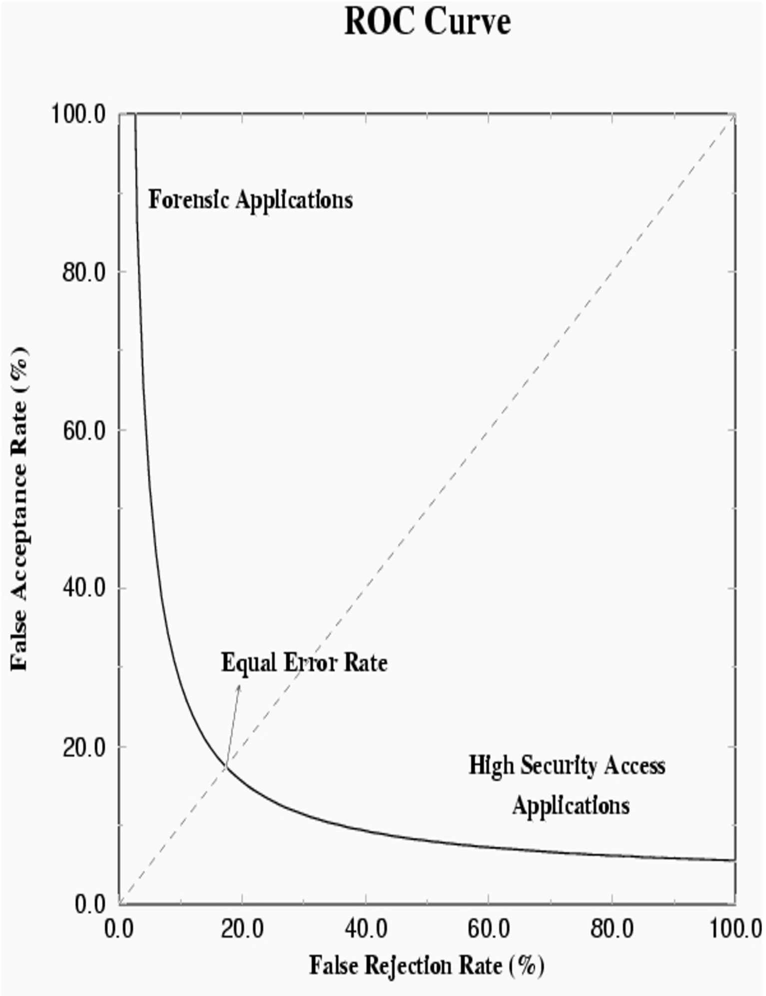


> threshold: A and B are the same speaker

< threshold: A and B are different speakers

Results and metrics

- When deciding if two audio fragments belong to the same person, there are two types of errors:
  - False Acceptance (FA)
  - False Reject (FR)
- Threshold is defined according to the scenario
- **EER (%): Equal Error Rate**
  - Error when FAR and FRR is equal



DNN Embeddings

X-VECTORS: ROBUST DNN EMBEDDINGS FOR SPEAKER RECOGNITION

David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, Sanjeev Khudanpur

Center for Language and Speech Processing & Human Language Technology Center of Excellence
The Johns Hopkins University, Baltimore, MD 21218, USA

- X-Vector: An embedding, output of one of the last layers, which is used as a model of the speaker/utterance
- The score can be obtained as the cosine distance of both vectors (speaker model and target audio, trial)

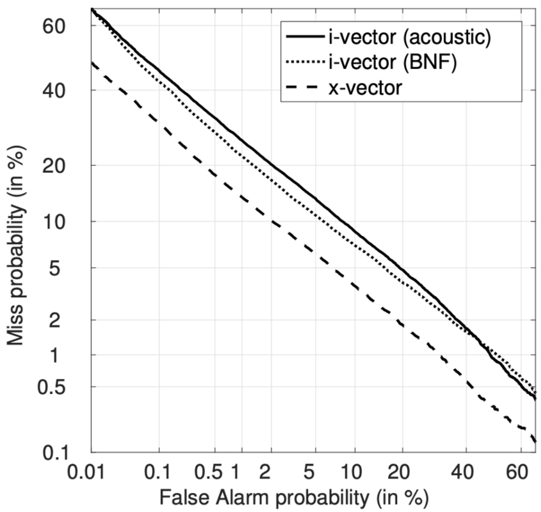


Fig. 1. DET curve for the Cantonese portion of NIST SRE16 using Section 4.5 systems.

ResNet

- “The Deeper the better”
  - When it comes to convolutional networks (CNN)
- VGG had a problem when layers were added: vanishing gradients
- ResNets fix this problem by adding residual conections

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		[3×3, 64; 3×3, 64] ×2	[3×3, 64; 3×3, 64] ×3	[1×1, 64; 3×3, 64; 1×1, 256] ×3	[1×1, 64; 3×3, 64; 1×1, 256] ×3	[1×1, 64; 3×3, 64; 1×1, 256] ×3
conv3_x	28×28	[3×3, 128; 3×3, 128] ×2	[3×3, 128; 3×3, 128] ×4	[1×1, 128; 3×3, 128; 1×1, 512] ×4	[1×1, 128; 3×3, 128; 1×1, 512] ×4	[1×1, 128; 3×3, 128; 1×1, 512] ×8
conv4_x	14×14	[3×3, 256; 3×3, 256] ×2	[3×3, 256; 3×3, 256] ×6	[1×1, 256; 3×3, 256; 1×1, 1024] ×6	[1×1, 256; 3×3, 256; 1×1, 1024] ×23	[1×1, 256; 3×3, 256; 1×1, 1024] ×36
conv5_x	7×7	[3×3, 512; 3×3, 512] ×2	[3×3, 512; 3×3, 512] ×3	[1×1, 512; 3×3, 512; 1×1, 2048] ×3	[1×1, 512; 3×3, 512; 1×1, 2048] ×3	[1×1, 512; 3×3, 512; 1×1, 2048] ×3
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

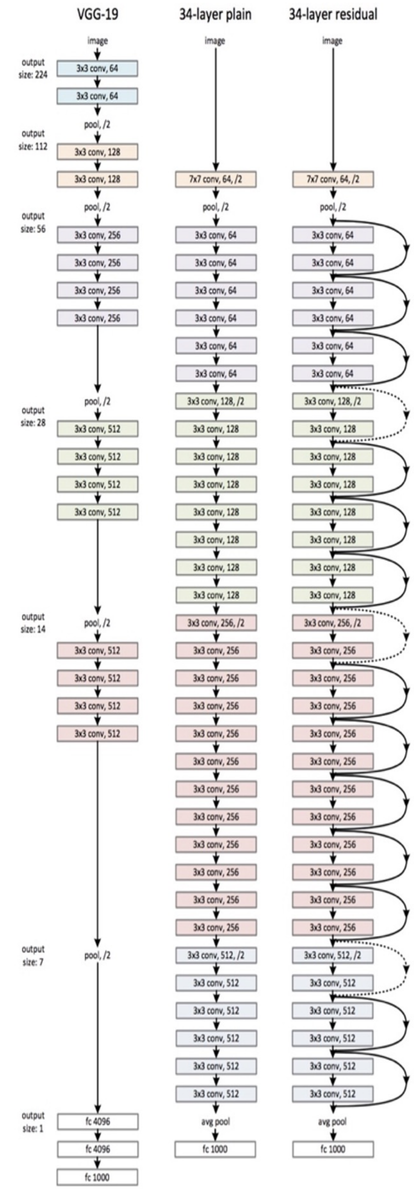


Figure 3. Example network architectures for ImageNet. **Left**: the VGG-19 model [41] (19.6 billion FLOPs) as a reference. **Middle**: a plain network with 34 parameter layers (3.6 billion FLOPs). **Right**: a residual network with 34 parameter layers (3.6 billion FLOPs). The dotted shortcuts increase dimensions. **Table 1** shows more details and other variants.

VoxCeleb

- Large Scale audio-visual dataset of human speech
- Extracted from interview videos from YouTube
- 7000+ Speakers
- 1 million+ utterances. In our case:
  - VoxCeleb1 (for Evaluation)
  - VoxCeleb2 (for Training)
    - We will use a subset of the original VoxCeleb2 dev database for speed (100 hours instead of 2000+ hours)

Overview

Overview - First session

- Preparation of the environment
  - Software (Google Colab and Pytorch)
  - Dataset

- Identifying key lines of code

Overview - Second session

- Loading previously trained model, evaluation and scoring
- Training a system from scratch
- "Improving" system changing the parameters
  - Is a smaller model better as we have a small subset of data, or we will get better performance adding layers and filters?

Evaluation

- Elaborate a report following, including the answers to the questions
  - Word / LaTex
- Submission:
  - PDF file
  - Until **<u>25th April (4pm)</u>**