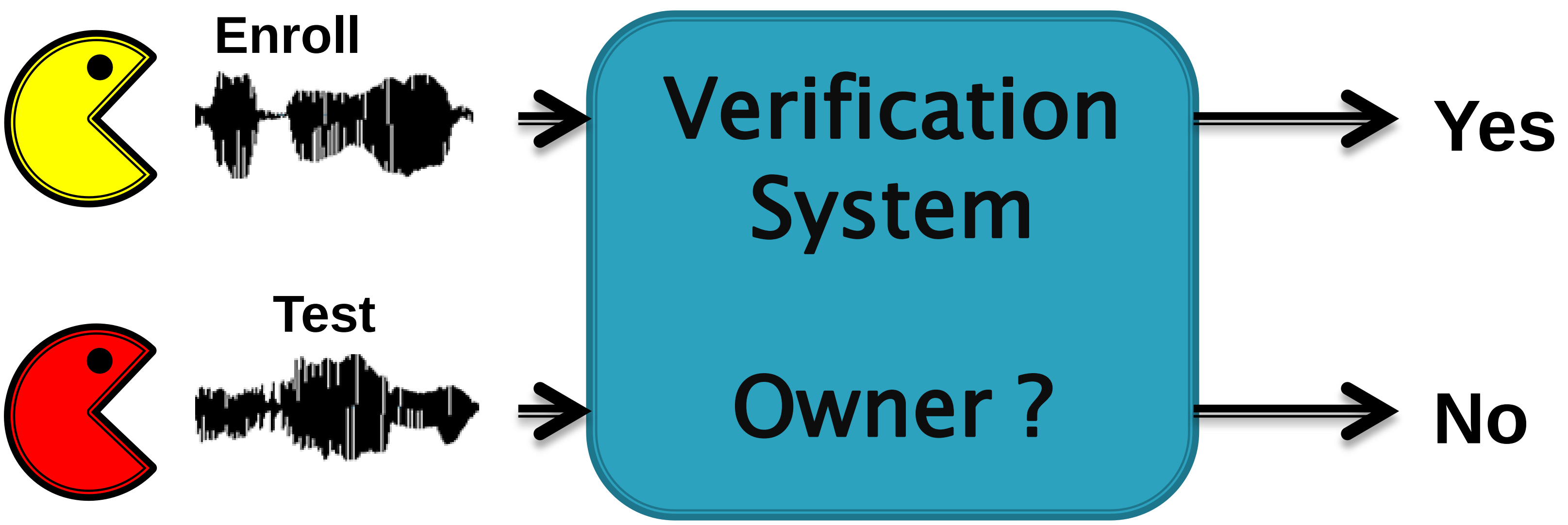
Speaker Verification

Instructor: Tom Ko

ML tasks related to speaker

- Speaker identification
  - Classify a speaker within a closed set
  - E.g. Checking the attendance of company members
- Speaker verification
  - Determine if a test speaker matches an enrolled speaker
  - E.g. Unlocking your mobile phone
- Speaker diarization
  - Determine "who spoke when" in a continuous audio
  - E.g. Meeting memo

Speaker Verification (SV)



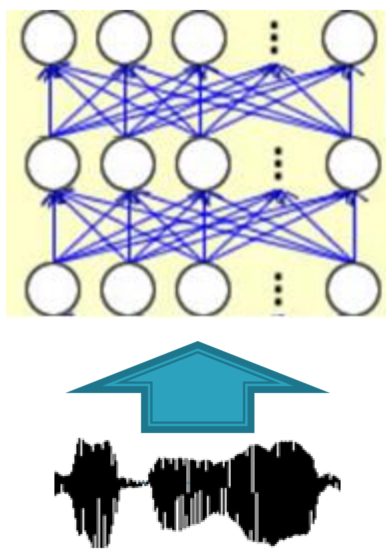
Identification vs. verification

- Which task is more difficult?

Identification vs. verification

- Which task is more difficult?

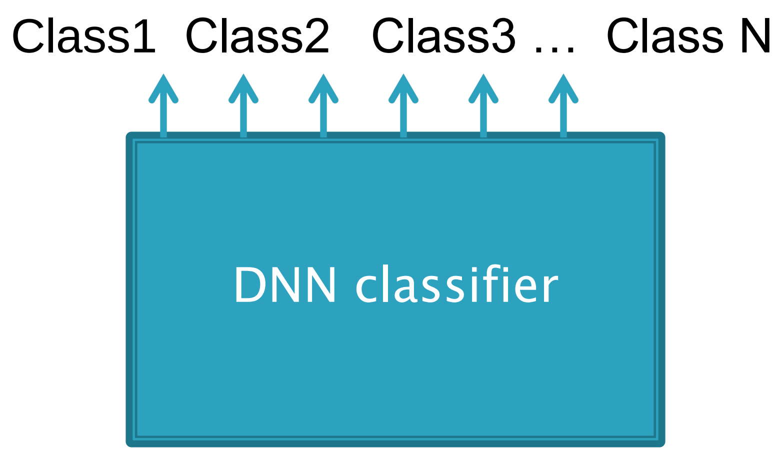
Spk1 Spk2 Spk3 … Spk N



Are these two signals generated by the same speaker?

Problem of unseen class

- How does a DNN react to a sample from Class N+1?



Text-dependent vs. Text-independent

- If the words in the test utterance is a subset of the words in the enrollment utterance, it is text-dependent.
  - OK Google
  - Hey Cortana
  - 你好華為
- Otherwise, it is text-independent

Challenges in SV

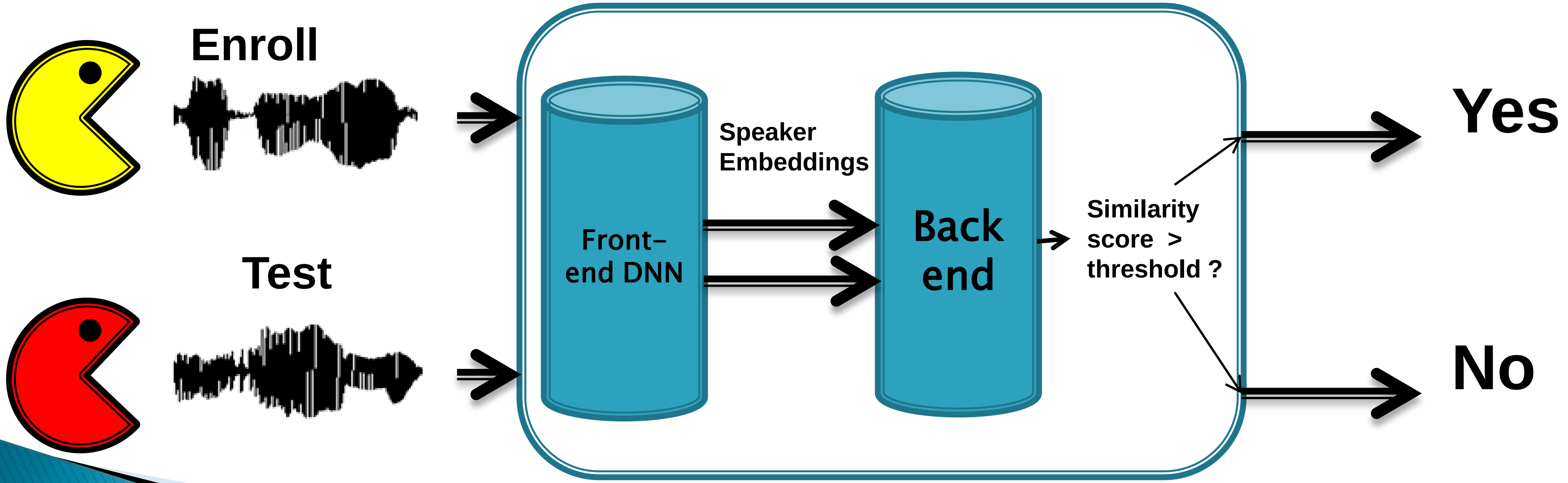
- You do not know any information of the users (the enrollment speakers are test speakers)
- The way to compute an fixed size vector (from varying length input) to represent an utterance.
- How to compute the similarity between two vectors.

Resources in SV

- Training set
  - It consists of utterances from a lot of training speakers
- Test set
  - It consists of enrollment utterances and test utterances
  - Your system has to decide if these utterances match or not.

The Speaker Embedding Approach

- Front-end DNN for speaker embedding extraction.
- Backend for similarity measure.



Evaluating the SV system

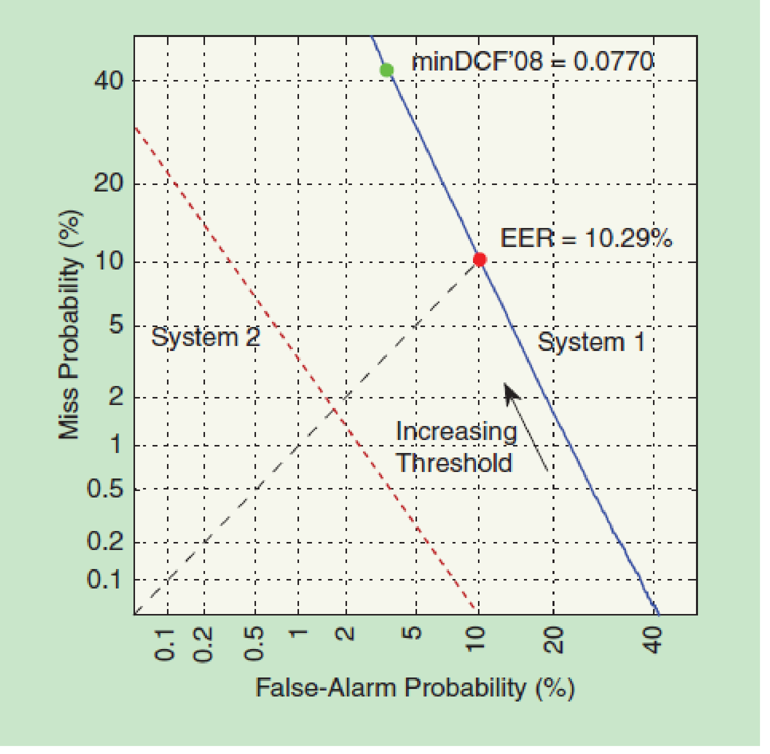
- Two types of error
  - False acceptance rate (FAR) – granting access to a bad guy
  - False rejection rate (FRR) – rejecting the owner
- $$\text{FAR} = \frac{\text{Number of wrong acceptance}}{\text{Number of wrong attempts}}$$
- $$\text{FRR} = \frac{\text{Number of rejecting the owner}}{\text{Number of owner attempts}}$$

Determining the threshold

- You can easily have a system with 0% FRR but 100% FAR – that means letting everyone in.
- Equal error rate (EER) = the point when FAR==FRR
- For a high security system, a higher or lower threshold should we prefer?

DET curve

- DET (detection error tradeoff) curve – plots FRR against FAR



Source: Hansen and Hasan, 2015

Evolution of speaker verification

- I-vector framework
  - GMM approaches
  - The most popular framework in the past decade
- D-vector
  - DNN approaches
  - Extract speaker-specific features from a DNN, averaging them and become the d-vector
- X-vector
  - Similar idea to the d-vector approach but do the averaging inside the DNN
  - Easy to implement and have a very good performance

I vector framework (GMM-UBM)

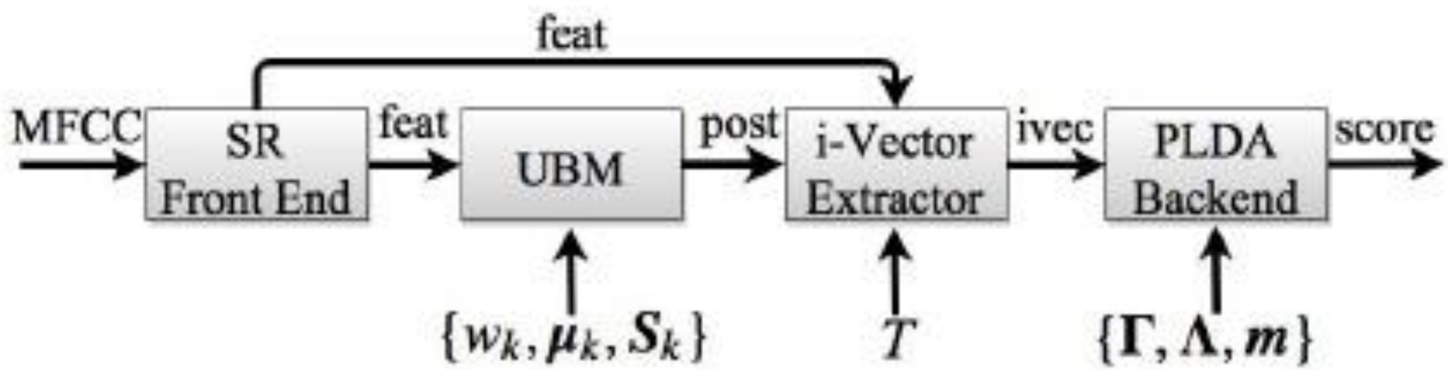


Fig. 1: GMM-based speaker recognition schema.

- The role of UBM is to compute the posteriors, posteriors indicate what phoneme the MFCC is.
- MFCC : 60-dim
- Posterior: 5000-dim
- Ivector:600-di

I vector framework (DNN-UBM)

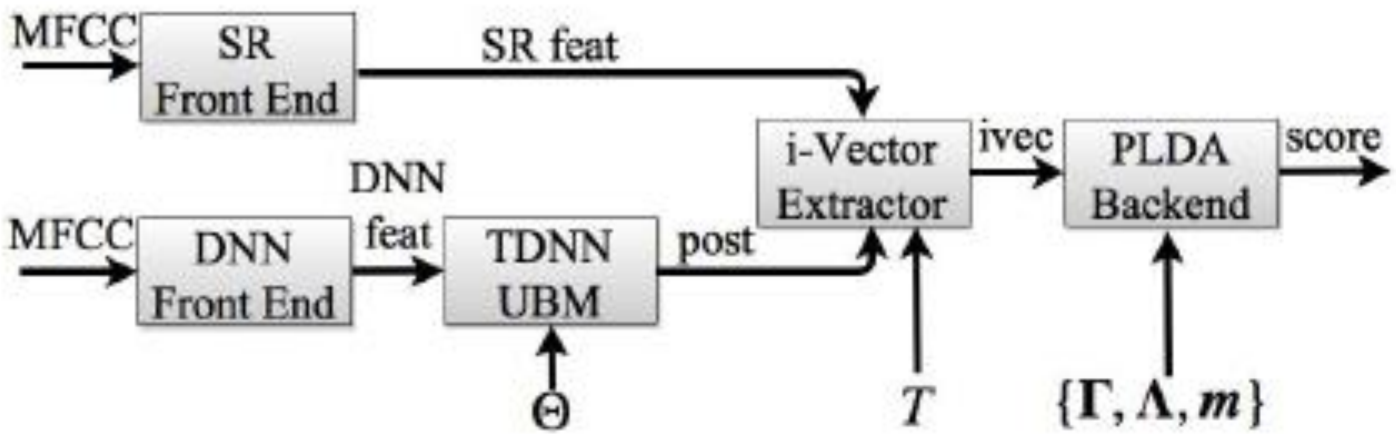
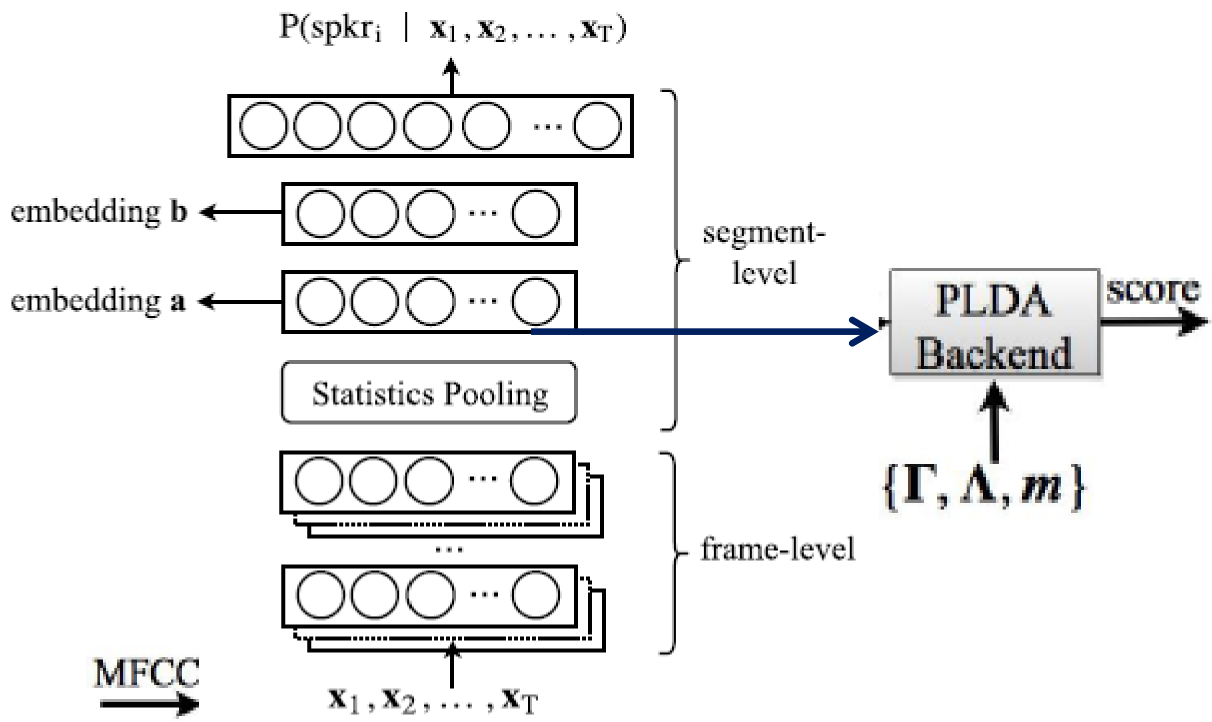


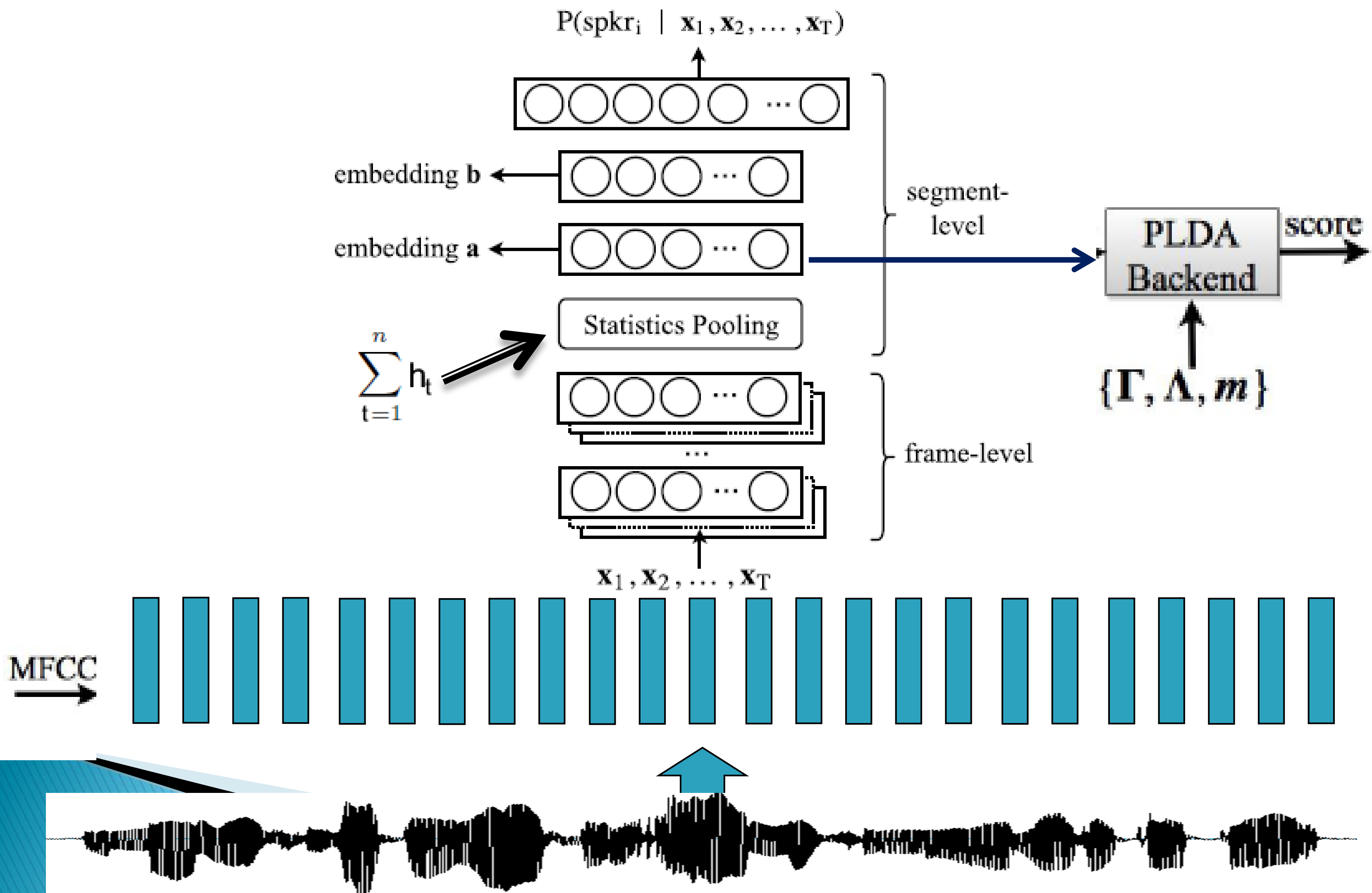
Fig. 2: TDNN-based speaker recognition schema.

X-vector Framework



" Deep Neural Network Embeddings for Text-Independent Speaker Verification", David Snyder, Daniel Garcia-Romero, Daniel Povey and Sanjeev Khudanpur, Interspeech 2017

X-vector Framework



X-vector Framework

Table 1: *EER(%) on NIST SRE10*

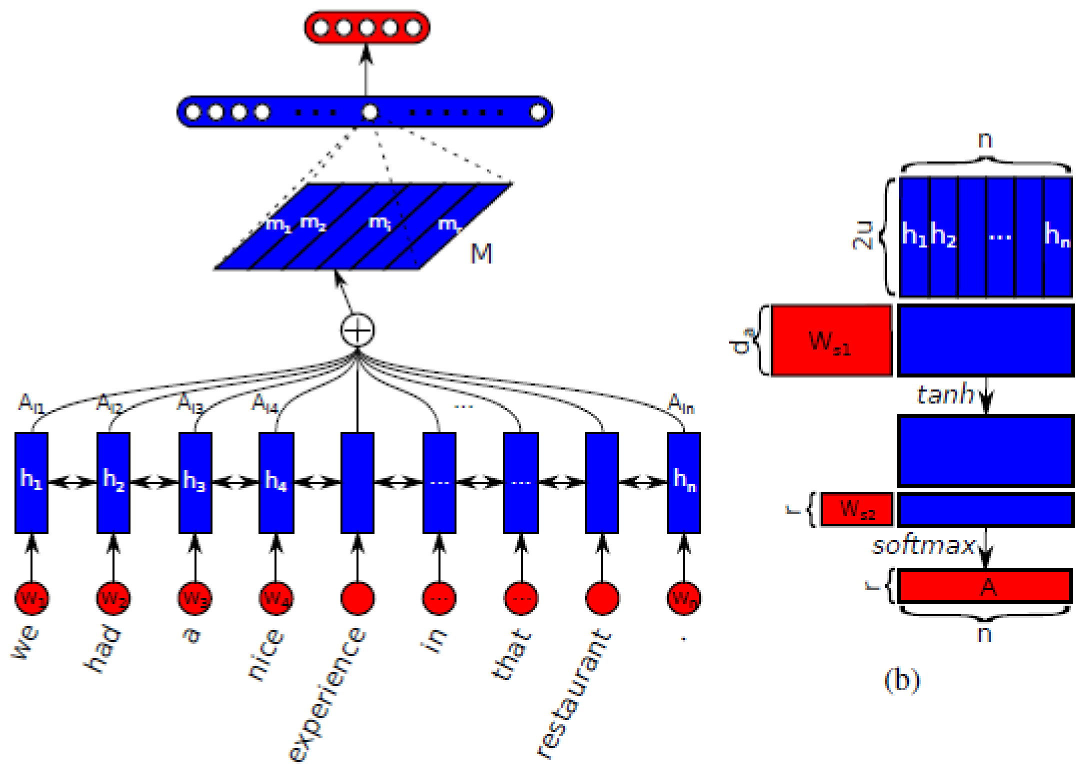
	5s	10s	20s	60s	full
ivector	9.1	6.0	3.9	2.3	1.9
embeddings	7.6	5.0	3.8	2.9	2.6

- X-vector is better than I-vector for short utterance
- For long utterance, X-vectors get worse, why ?

Motivation

- Problems: Not every words carry the discriminative information of speakers but the pooling layer give the same weight to every frames in the utterance.
- Solution: Self-attention

Self-attention

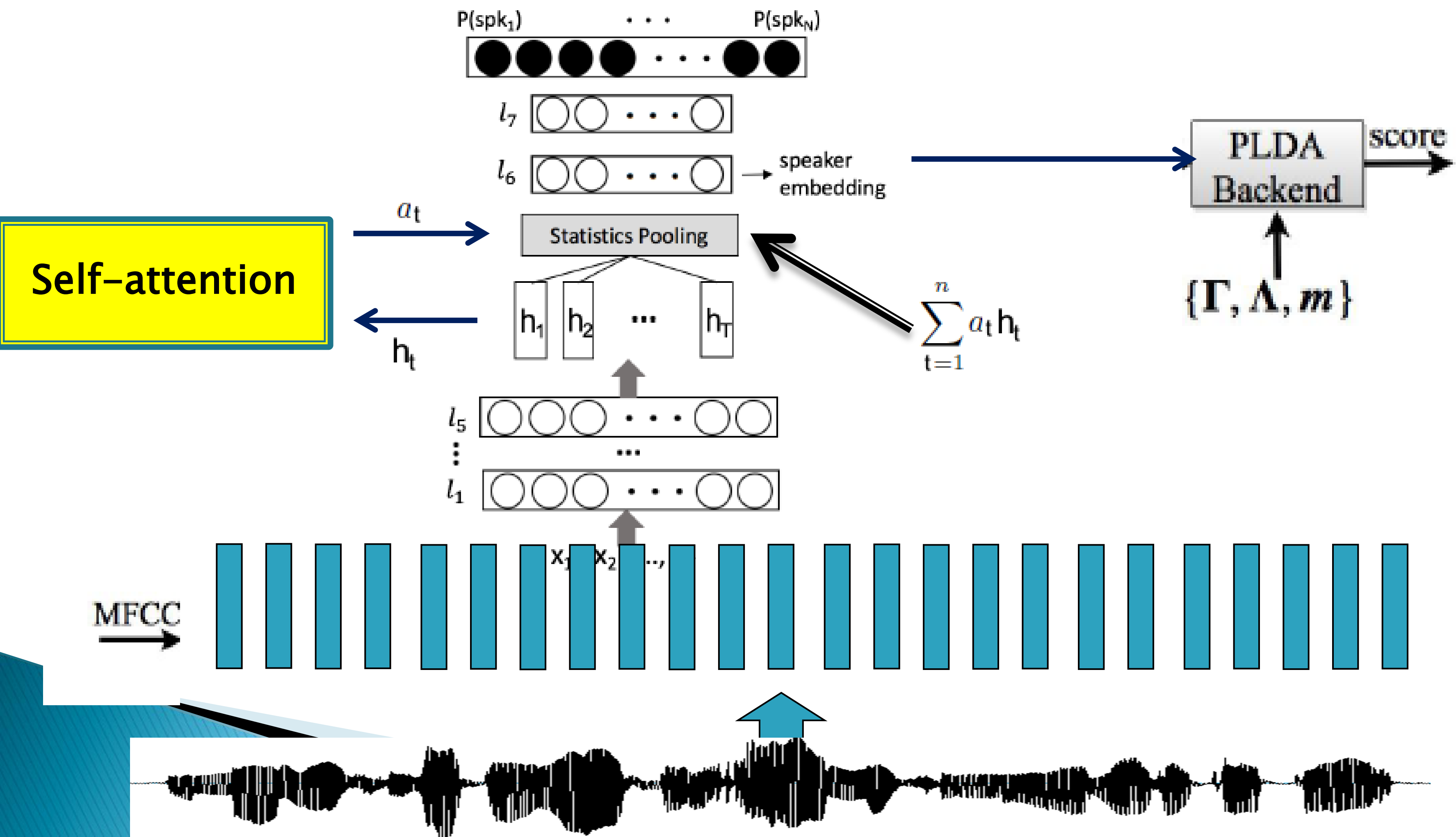


(a)

(b)

Z. Lin, M. Feng, C. N. d. Santos, M. Yu, B. Xiang, B. Zhou,and Y. Bengio,
"A structured self-attentive sentence embedding,"
Proceedings of the International Conference on Learning Representations, 2017.

X-vector Framework



Evaluation Set: SRE16

- Training data: 2000 hours of English
- Evaluation data: Cantonese
- Training speakers ~4400
- Enrollment speakers ~1000
- Test speakers ~9000

Results with different test duration

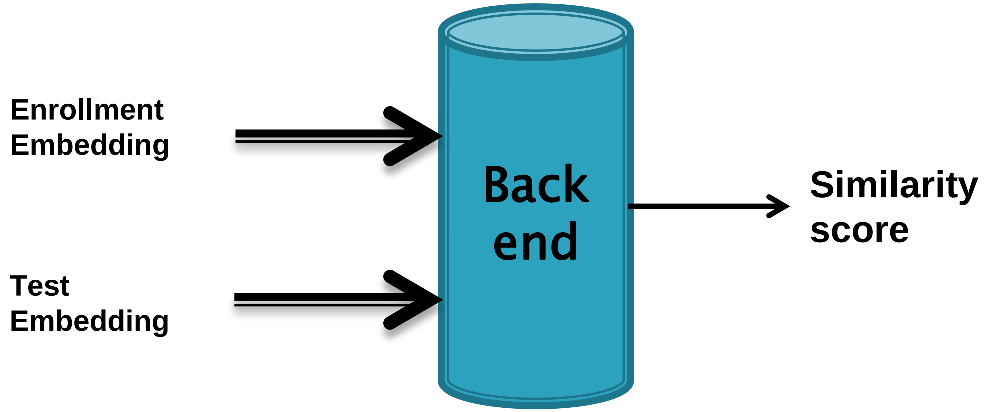
Table 2: *EER(%) on SRE16*

	baseline	attn-1	attn-2	attn-5
Cantonese				
10s-20s	6.95	6.84	6.22	5.91
20s-40s	5.37	5.29	4.73	4.52
40s-60s	4.39	3.98	3.91	3.83

" Self-Attentive Speaker Embeddings for Text-Independent Speaker Verification",
Yingke Zhu, **Tom Ko**, David Snyder, Brian Mak, Daniel Povey, Interspeech 2018

Backend

- Cosine similarity
- PLDA (Probabilistic Linear Discriminant Analysis)



Cosine similarity

- The cosine and the dot product
- Dot product favors long vectors
  - $\boldsymbol{w} \cdot \boldsymbol{x} = |\boldsymbol{w}|\ |\boldsymbol{x}| \cos\theta$

- The cosine of the angle between the two vectors, which is the normalized dot product, is the most common similarity metric.

PLDA (Probabilistic Linear Discriminant Analysis)

- It assumes that the j-th sample of i-th person is generated from

$$\mathbf{x}_{ij} = \mu + \mathbf{F}\mathbf{h}_i + \mathbf{G}\mathbf{w}_{ij} + \epsilon_{ij}$$

where F describes the between-person variation

G describes the within-person variation

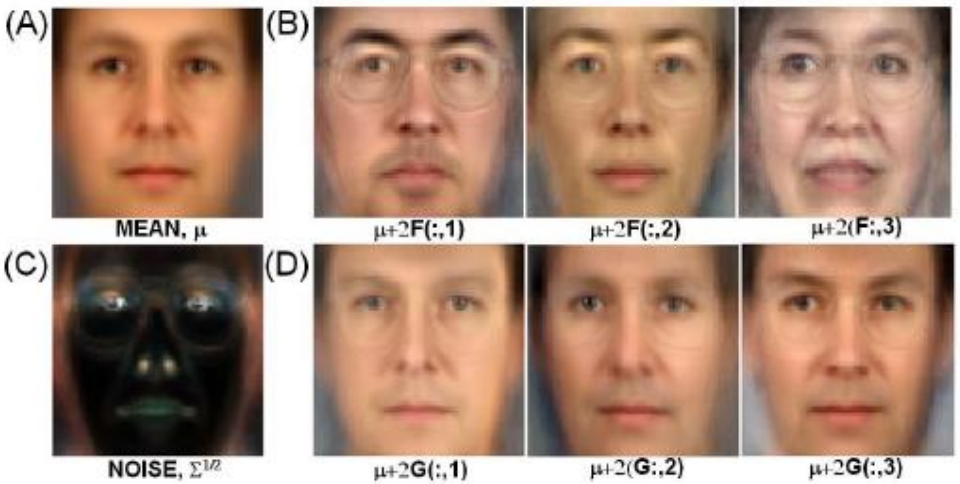
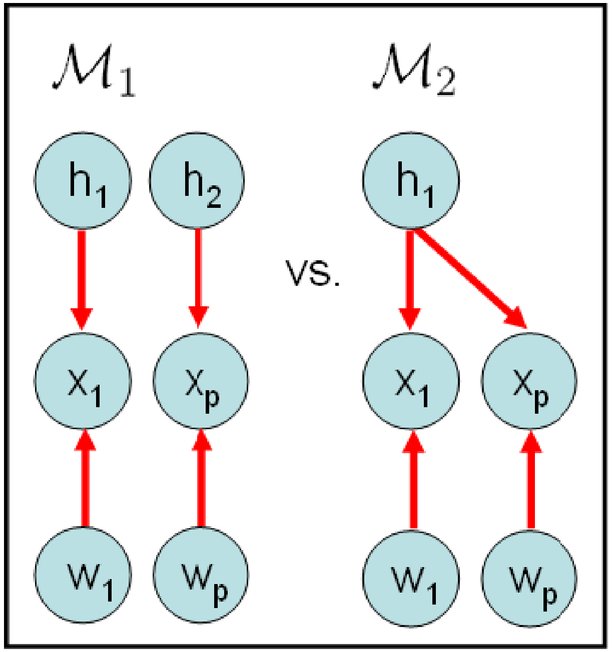


Figure 1. Components of PLDA Model. (A) Mean face (B) Three directions in between-individual subspace. Each image looks like a different person. (C) Per-pixel noise covariance (D) Three directions in within-individual subspace. Each images looks like the same person under minor pose and lighting changes.

PLDA on verification task

- For verification, we need to decide if two samples are generated from the same person.
- Two models:
  - M0 – two samples not match
  - M1 – two samples match
- Given the observed data X, we calculate posterior probability

$$Pr(\mathcal{M}_q|\mathbf{x}) = \frac{Pr(\mathbf{x}|\mathcal{M}_q)Pr(\mathcal{M}_q)}{\sum_{r=0}^{R} Pr(\mathbf{x}|\mathcal{M}_r)Pr(\mathcal{M}_r)}$$

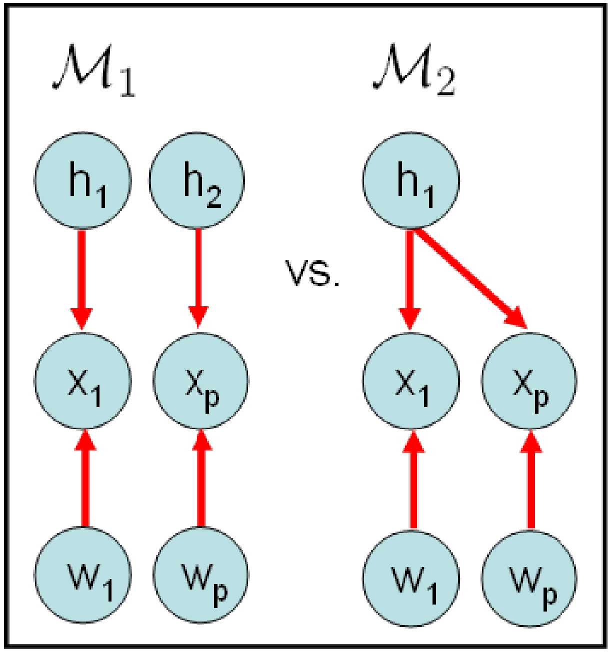


PLDA on verification task

$$Pr(\mathbf{x}_{1,\,p}|\mathcal{M}_1) = Pr(\mathbf{x}_1|\mathcal{M}_1)Pr(\mathbf{x}_p|\mathcal{M}_1)$$

$$Pr(\mathbf{x}_1|\mathcal{M}_1)=\int\int Pr(\mathbf{x}_1|\mathbf{h}_1,\mathbf{w}_1)Pr(\mathbf{w}_1)d\mathbf{w}_1 Pr(\mathbf{h}_1)d\mathbf{h}_1$$

$$Pr(\mathbf{x}_p|\mathcal{M}_1)=\int\int Pr(\mathbf{x}_p|\mathbf{h}_2,\mathbf{w}_p)Pr(\mathbf{w}_p)d\mathbf{w}_p Pr(\mathbf{h}_2)d\mathbf{h}_2$$



$$Pr(\mathbf{x}_{1,p}|\mathcal{M}_2) = \int \left[\int Pr(\mathbf{x}_1|\mathbf{h}_1,\mathbf{w}_1)Pr(\mathbf{w}_1)d\mathbf{w}_1 \int Pr(\mathbf{x}_p|\mathbf{h}_1,\mathbf{w}_p)Pr(\mathbf{w}_p)d\mathbf{w}_p\right].Pr(\mathbf{h}_1)d\mathbf{h}_1$$

Reading list

- Try to study the following mathematical tools on your own
  - PCA (Principal component analysis)
  - LDA (Linear discriminant analysis)
  - Factor analysis