Biometric Person Recognition

Dr. Mathew Magimai Doss and Dr. Petr Motlicek

December 8, 2022

idiap RESEARCH INSTITUTE EPFL

Outline

Outline

Biometric Person Recognition (1)

Biometric characteristics can be categorized as

- Physiological only: **face**, **fingerprint**, **iris**, **hand geometry**, **palmprint**, DNA
- Behavioral only: handwriting, **signature**, typing rhythm, gait
- Both: **voice**

A few applications:

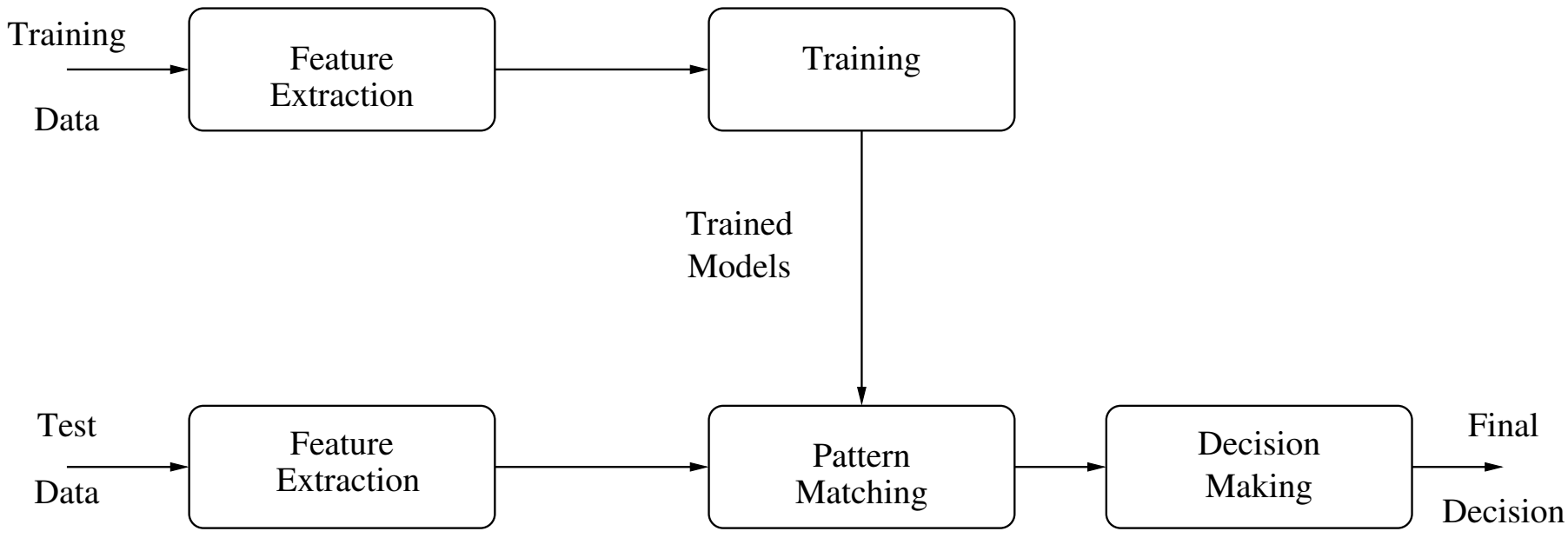
- Access control
- Forensics
- Surveillance

How precise these characteristics are? [1]

[1] Fingerprint FP $\sim$ 0.1%, FN $\sim$ 7%

Biometric Person Recognition (2)

Biometric person recognition can be seen as a pattern recognition problem



Types of person recognition tasks

- Person identification: identify a person from a finite set of persons (given a biometric signal).
  Pattern classification problem: Given a feature sequence $X = \{x_1, \cdots x_m, \cdots x_M\}$ (extracted from the biometric signal), a person I identified (classified) as person I_c if

$$P(I = I_c|X) \geq P(I = I_\ell|X), \quad \forall \ell \neq c$$

- Person verification: verify a claimed identity of a person.
  Hypothesis testing problem: Given a feature sequence X, and a claimed identity I_c, estimate the probability that the person I is indeed I_c and not another person:

$$\frac{P(I = I_c|X)}{P(I = \overline{I_c}|X)} \geq \Delta$$

Similarity Measures (1)

Given a biometric signal/feature sequence X the probability that it belongs to a person (rather than some other person) is given by:

$$P(I_c|X) = \frac{P(X|I_c)P(I_c)}{P(X)} = \frac{P(X|I_c)P(I_c)}{\sum_{i=1}^{I} P(X|I_i)P(I_i)}$$

Ideally, the sum in the denominator should include all possible persons.

Person Identification:

In this case, after training, $P(X)$ is a constant, and person I will be identified as person I_c if:

$$P(X|I_c) \geq P(X|I_i), \ \forall i \neq c$$

Similarity Measures (2)

Person Verification:

Person verification, however, is a form of hypothesis test. In this case, we will verify the hypothesis that a person I is indeed the putative person I_c if:

$$P(I_c|X) > P(\overline{I_c}|X)$$

where $\overline{I_c}$ represents the set of all possible rival persons, and the right hand side is the probability of the person being anyone except I_c.

Typically, this is stated with some margin or threshold, i.e., a person I is taken to be the person I_c if:

$$\frac{P(I_c|X)}{P(\overline{I_c}|X)} > \delta$$

where δ is a threshold > 1

Similarity Measures (3)

$$P(\overline{I_c}|X) = P(I_1 \text{ or } I_2 \text{ or } \ldots \text{ or } I_{i\neq c}|X) = \sum_{i\neq c} P(I_i|X)$$

if events I_i are independent (which is the case) and collectively exhaustive (which will often be wrong).
Consequently,

$$I = I_c \quad \text{if} \qquad \frac{P(X|I_c)}{P(X|\overline{I_c})} = \frac{P(X|I_c)}{\sum_{i\neq c} P(X|I_i)} > \delta$$

defined as the **likelihood ratio criterion**, and where the sum over i incorporates all the possible persons.
Alternatively:

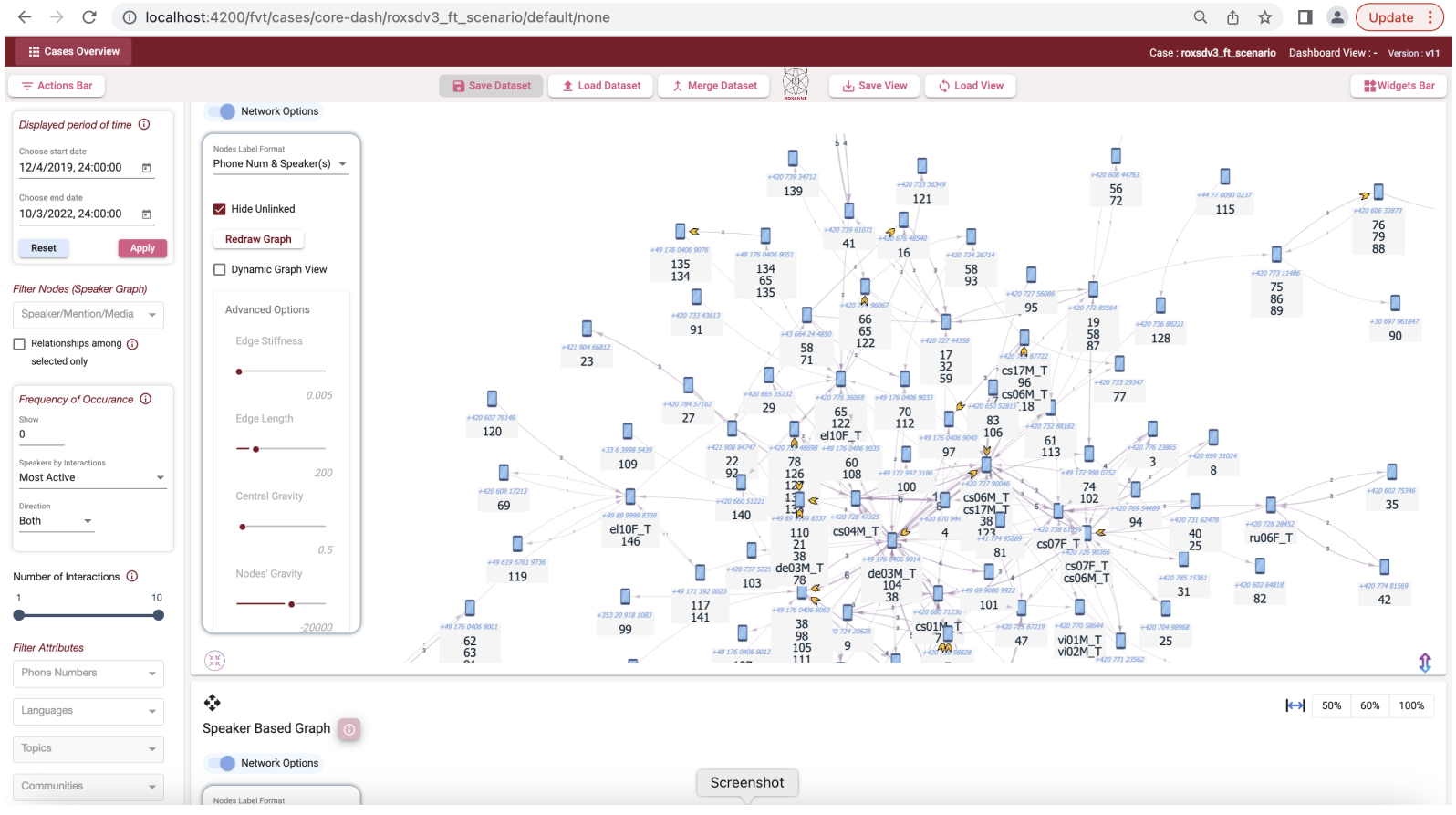
$$I = I_c \quad \text{if} \qquad \log P(X|I_c) - \log P(X|\overline{I_c}) > \Delta$$

where $\Delta = \log \delta$.

Typical applications (1) - person ID

- identify the person from the closed set
- Identify the person from the open set
- Cluster persons into a finite set of clusters?

Example of a investigation platform built at Idiap:



Typical applications (2) - person verif.

- Verification of an user of given app.
- Comparison of suspect with an offender for forensics (law enforcement)

Example of a forenscics platform :

Design of person verification system

Feature representation x_m:

- Cepstral features (typically high order cepstral coefficients and their temporal derivatives): $C_0 - C_{20} + \Delta + \Delta\Delta$ and optionally $\log F_0 + \Delta + \Delta\Delta$
- Log filter bank energies $+\Delta + \Delta\Delta$ (when using neural networks)

1. Training of a $P(X|I_c)$ estimator for each speaker on their respective speech data. (*also referred to as speaker enrollment*)
2. Training of a good **normalization factor** $P(X|\overline{I_c})$ estimator
3. Optimal setting of the **decision threshold**.
   Typically, by assuming a Gaussian distribution of likelihoods $P(X|I_c)$ and $P(X|\overline{I_c})$ for a specific training and test set. The variability of these distributions also shows the importance of using a similarity measure based on a likelihood ratio measure.

Outline

Estimation of $P(X|I_c)$ (1)

1. **Bag of Features**: With independence assumption, treats the feature sequence X as a collection (i.e. features in the sequence can be treated in any order).

$$P(X|I_c) = \left(\prod_{m=1}^{M} P(x_m|I_c)\right)^{\frac{1}{N}}$$

The likelihood $P(x_m|I_c)$ can be estimated by modelling the feature distribution using

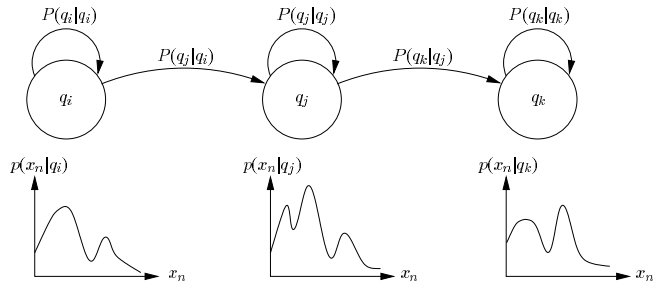
   i. K-Means clustering
   ii. Training models based on long-term statistics such as the mean and variance calculated on a sufficiently large collection of features, such as Gaussian mixture model (GMM).
   iii. Adapting a pre-trained **universal backgroundGMM** (*see the* $P(X|\overline{I_c})$ *estimation part*)

Estimation of $P(X|I_c)$ (2)

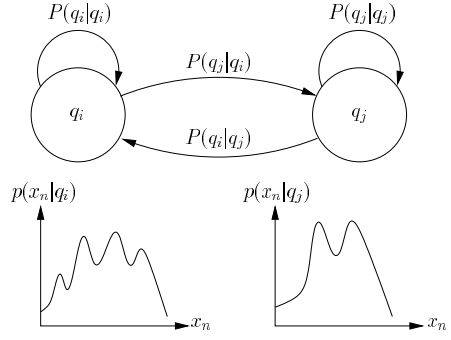
2. **Sequential model**: Here it is assumed that the feature sequence X has been generated by a sequence of states $Q = \{q_1, \cdots q_m, \cdots q_M\}$ (structured model), e.g. an HMM.

$$P(X|I_c) = (\prod_{m=1}^{M} P(\mathrm{x}_m|q_m, I_c)^{\frac{1}{M}}$$

   i. Constrained topology, such as left-to-right (**Text-dependent**)



   ii. Unconstrained topology, such as fully connected (ergodic) HMM where all states are connected to each other



Estimation of $P(X|\overline{I_c})$ (1)

There are different approaches to model the "normalization factor" $P(X|\overline{I_c})$ estimator.

1. We assume that the set of reference persons already enrolled in the database is sufficiently representative of all possible persons:

$$\log P(X|\overline{I_c}) \approx \sum_{I_i \in R, i \neq c} \log P(X|I_i)$$

   where R represents the set of persons already enrolled in the system.

2. We assume that the sum in the denominator is dominated by the closest rival person:

$$\log P(X|\overline{I_c}) \approx \max_{I_i \in R, i \neq c} \log P(X|I_i)$$

Estimation of $P(X|\overline{I_c})$ (2)

3. "Cohort" model: a well chosen subset of reference persons on which $P(X|\overline{I_c})$ will be estimated:

$$\log P(X|\overline{I_c}) \approx \sum_{I_i \in R_c, i \neq c} \log P(X|I_i)$$

   where R_c represents the cohort associated with person I_c.

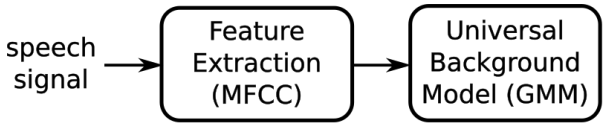
4. "Universal background model" (UBM): approximating $P(X|\overline{I_c})$ by training a Gaussian mixture model on a large set of "auxiliary" speakers data (i.e., speakers not part of the speaker verification system).

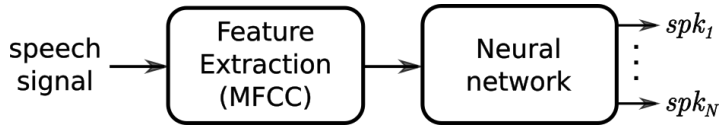
Speaker embeddings-based approach

i-vectors

x-vectors

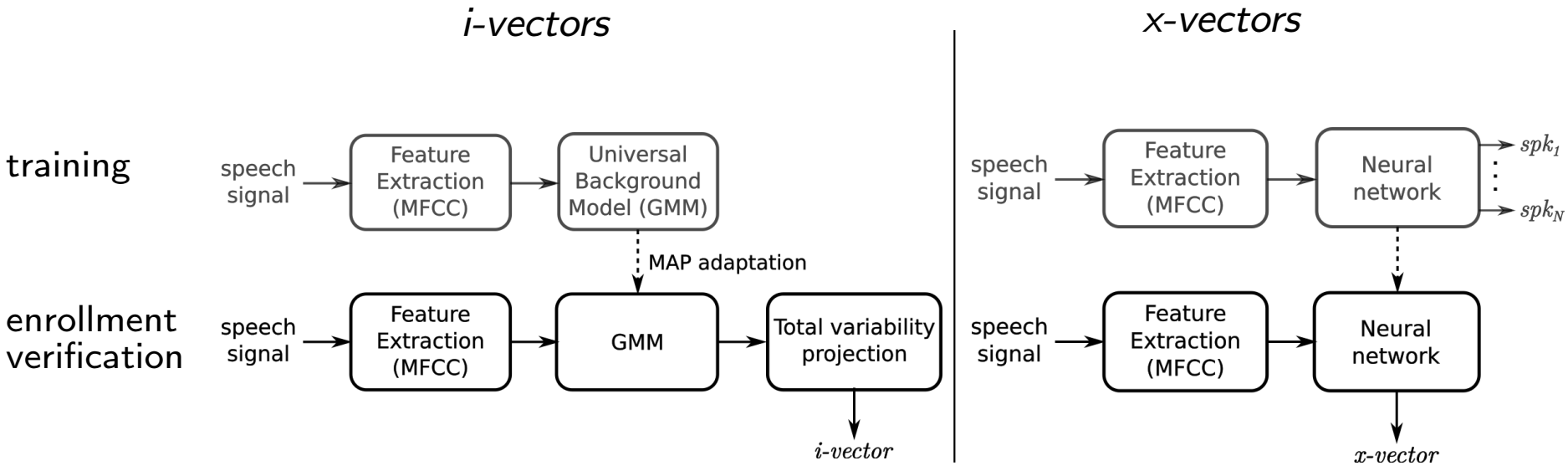
training





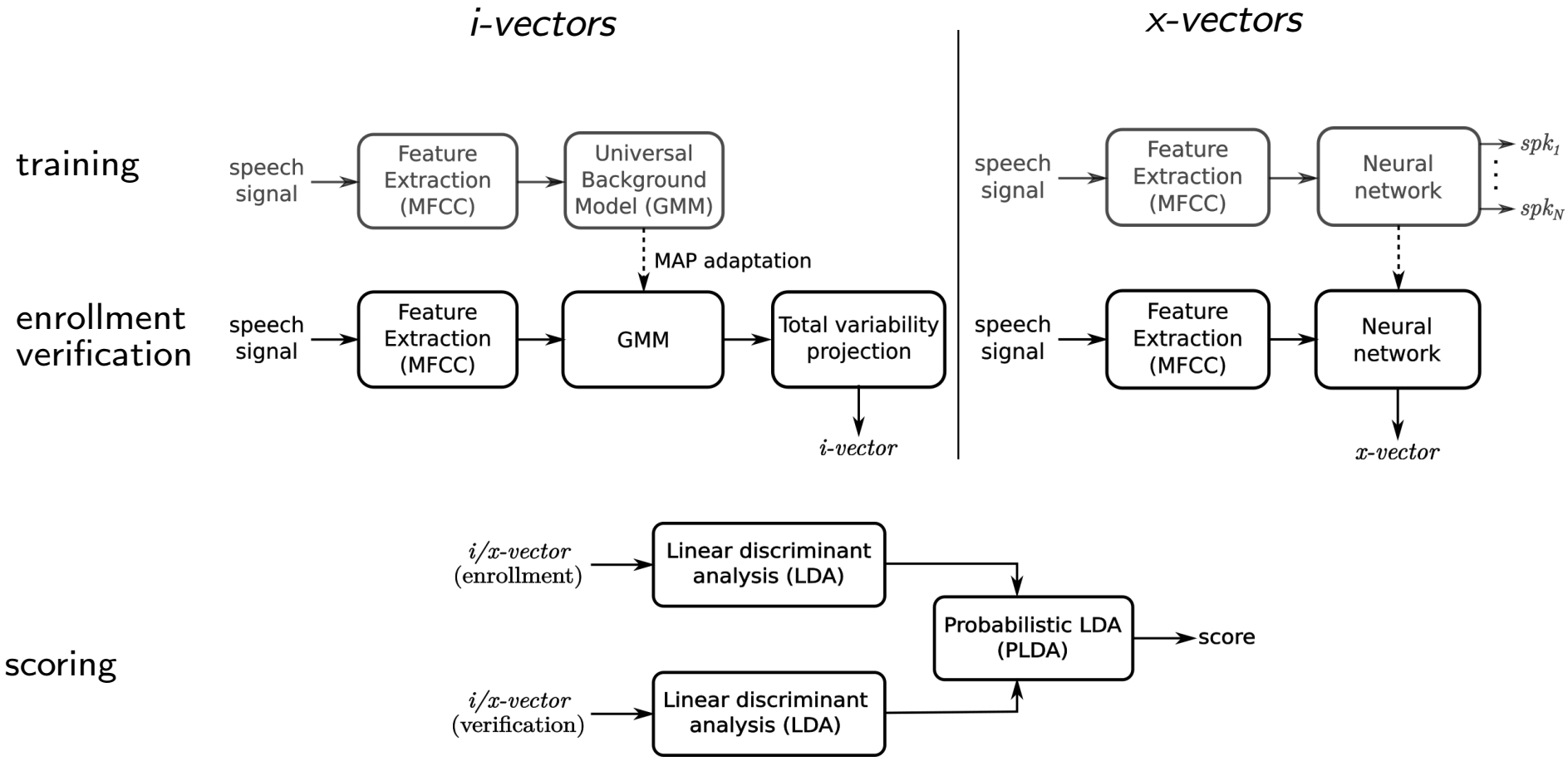
Najim Dehak *et al.* "Front-end factor analysis for speaker verification". IEEE Transactions on Audio, Speech, and Language Processing, 19(4):788–798, 2011.
David Snyder *et al.*, "X-vectors: Robust DNN embeddings for speaker recognition", ICASSP, 2018.

Speaker embeddings-based approach



Najim Dehak *et al.* "Front-end factor analysis for speaker verification". IEEE Transactions on Audio, Speech, and Language Processing, 19(4):788–798, 2011.
David Snyder *et al.*, "X-vectors: Robust DNN embeddings for speaker recognition", ICASSP, 2018.

Speaker embeddings-based approach



Najim Dehak *et al.* "Front-end factor analysis for speaker verification". IEEE Transactions on Audio, Speech, and Language Processing, 19(4):788–798, 2011.
David Snyder *et al.*, "X-vectors: Robust DNN embeddings for speaker recognition", ICASSP, 2018.

Speaker embeddings-based approach (d

1. Training
   - i-vector: Universal background model GMM (UBM-GMM) trained on auxiliary speakers data.
   - x-vector: speaker classification neural network (with a stats pooling layer) trained on auxiliary speakers data.
2. Speaker embedding extraction
   - i-vector: Enrollment and verification speaker embeddings are extracted by adapting the UBM-GMM on the feature vectors extracted from enrollment speech signal and verification speech signal and then applying factor analysis on the updated UBM-GMM parameters (total variability analysis), respectively.
   - x-vector: Enrollment and verification speaker embeddings are extracted by feeding the feature vectors extracted from enrollment speech signal and verification speech signal and taking output of an intermediate layer that is close to the output layer, respectively.
3. Scoring

   Applying linear discriminant analysis (LDA) on the enrollment speaker embeddings and verification speaker embeddings to reduce feature dimension, and comparing them using probabilistic linear discriminant analysis (PLDA). The output score is an estimate of log-likelihood ratio $\log P(X|I_c) - \log P(X|\overline{I_c})$.

Outline

Decision threshold Δ (1)

$$I = I_c \quad \text{if} \qquad \log P(X|I_c) - \log P(X|\overline{I_c}) > \Delta$$

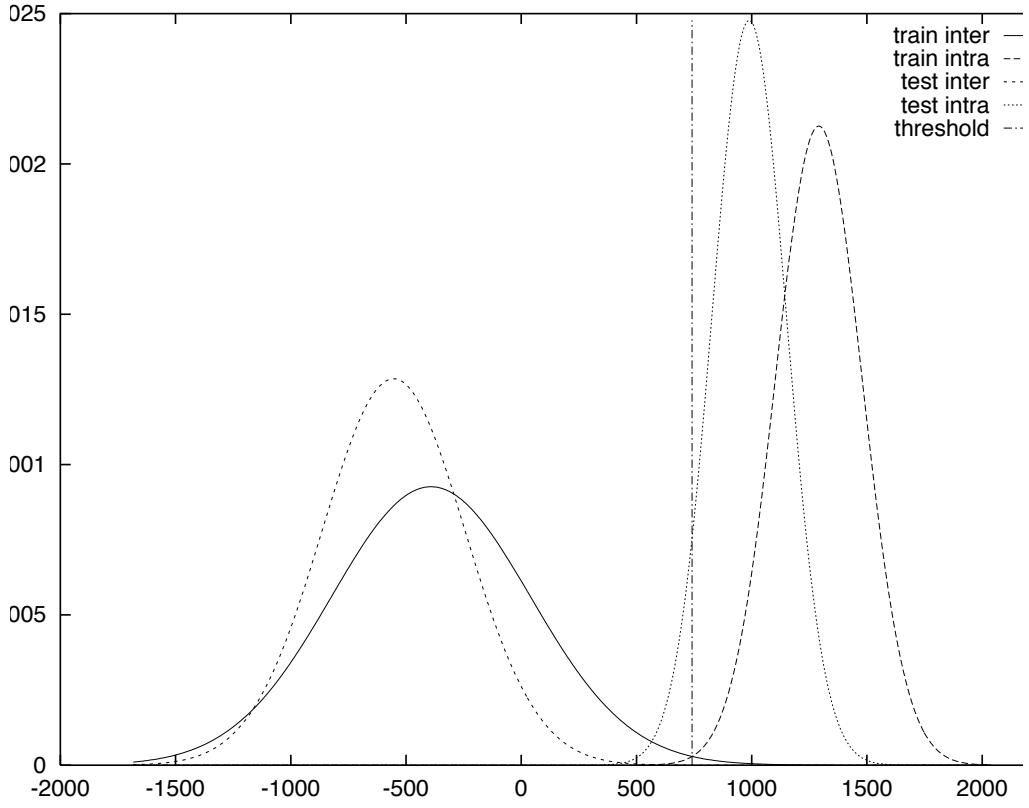
When making a decision the system can commit one of the two types of errors

- False acceptance (FA): An impostor claim is accepted.
- False rejection (FR): A true claim is rejected.

The choice of the decision threshold gives a tradeoff between these errors. Furthermore, the cost associated to each of these errors is not usually equal.
For instance, in a banking application it is better to have 0% FA rate at the cost of having a FR rate higher than 0%. However, too high FR rate can be a cause of dissatisfication to customers.

Decision threshold Δ (2)

Example of Gaussian approximations of the distributions of $P(X|\overline{I_c})$ (the left two Gaussians, respectively for training and test set) and $P(X|I_c)$ (right Gaussians). The vertical line (750) represents the decision threshold corresponding to the Equal Error Rate (EER) as estimated on the training data. As shown here, the position of the Gaussian can vary from training to test data, depending on the variability of environment and person characteristics. Means and variances were computed on a set of real data corresponding to a specific person I_c and a given set of impostors.



Decision threshold Δ (3)

- Theory: The decision threshold $\Delta > 0$ ideally should be depending upon the operating conditions, e.g., which type of error is more tolerable.
- Practice: The decision threshold is also dependent upon factors such as actual person, model of the person, recording environment, quality of the signal. As a result, the decision threshold can be different for different person, i.e., Δ_c.

The main reason being mismatch between observation X and the trained model due to

- Model limitations, e.g. flexibility to model the distribution, choice for estimating $P(X|\overline{I_c})$
- Insufficient training data/lack of coverage
- Unseen test conditions, e.g. environment (indoor conditions, outdoor conditions), signal quality (change in the type of microphone or camera or resolution of image), behavioral changes (emotions, facial expressions, change in voice characteristics due to cold).

Score normalization (1)

Let,

$$S_{I_c}(X) = \log P(X|I_c) - \log P(X|\overline{I_c})$$

for a test feature sequence X.
To handle the score $S_{I_c}(X)$ variability and to make "person independent" decision threshold (Δ) tuning easier different **score normalization** methods have been proposed, such as,

- Z-norm: For each person model I_c normalize the impostor score distribution

$$\frac{S_{I_c}(X) - \mu_{I_c}}{\sigma_{I_c}}$$

where, μ_{I_c} and σ_{I_c} are the mean and standard deviation of impostor score distribution estimated by using "pseudo" impostors data.
pros: μ_{I_c} and σ_{I_c} can be estimated offline during training.
cons: pseudo impostor data and test data may not match.

Score normalization (2)

- T-norm: For a given test feature sequence X normalize the impostor score distribution

$$\frac{S_{I_c}(X) - \mu_X}{\sigma_X}$$

where, μ_X and σ_X are the mean and standard deviation of the impostor score distribution obtained by matching the test feature sequence X using person models other than I_c.
pros: no pseudo impostor data required and can deal with variability in the test feature sequence.
cons: requires more resources during testing to estimate μ_X and σ_X.

The choice to normalize impostor score distribution is dictated by two reasons (a) easy availability of pseudo impostors, and (b) impostor distribution represents the largest part of score distribution variance.

Outline

Evaluation measures (1)

Given a set of decisions $D_1, \cdots D_k, \cdots D_K$ made by the system (using a person independent threshold Δ) and their respective "true" decisions/labels $T_1, \cdots T_k, \cdots T_K$ for a set of test feature sequences $X_1, \cdots X_k, \cdots X_K$

1. Set number of FA (nFA), number of FR (nFR), number of true claims (nTC), and number of impostor claims (nIC) to 0.
2. For each test feature sequence $k = 1, \cdots K$

$$
\begin{aligned}
nFA &= nFA + 1 \quad \text{if } (D_k = \textit{accept} \mid T_k = \textit{impostor claim}) \\
nFR &= nFR + 1 \quad \text{if } (D_k = \textit{reject} \mid T_k = \textit{true claim}) \\
nIC &= nIC + 1 \quad \text{if } (T_k = \textit{impostor claim}) \\
nTC &= nTC + 1 \quad \text{if } (T_k = \textit{true claim})
\end{aligned}
$$

3.

$$
\begin{aligned}
\text{FA rate (in \%)} &= P_{FA}(\Delta) = \frac{nFA}{nIC} * 100 \\
\text{FR rate (in \%)} &= P_{FR}(\Delta) = \frac{nFR}{nTC} * 100
\end{aligned}
$$

Half total error rate (HTER): average of FA rate and FR rate.

Evaluation measures (2)

By varying Δ and estimating $P_{FA}(\Delta)$ and $P_{FR}(\Delta)$ different systems (based on different approaches/methods) can be evaluated and compared in terms of one of the following measures:

1. Equal error rate (EER)

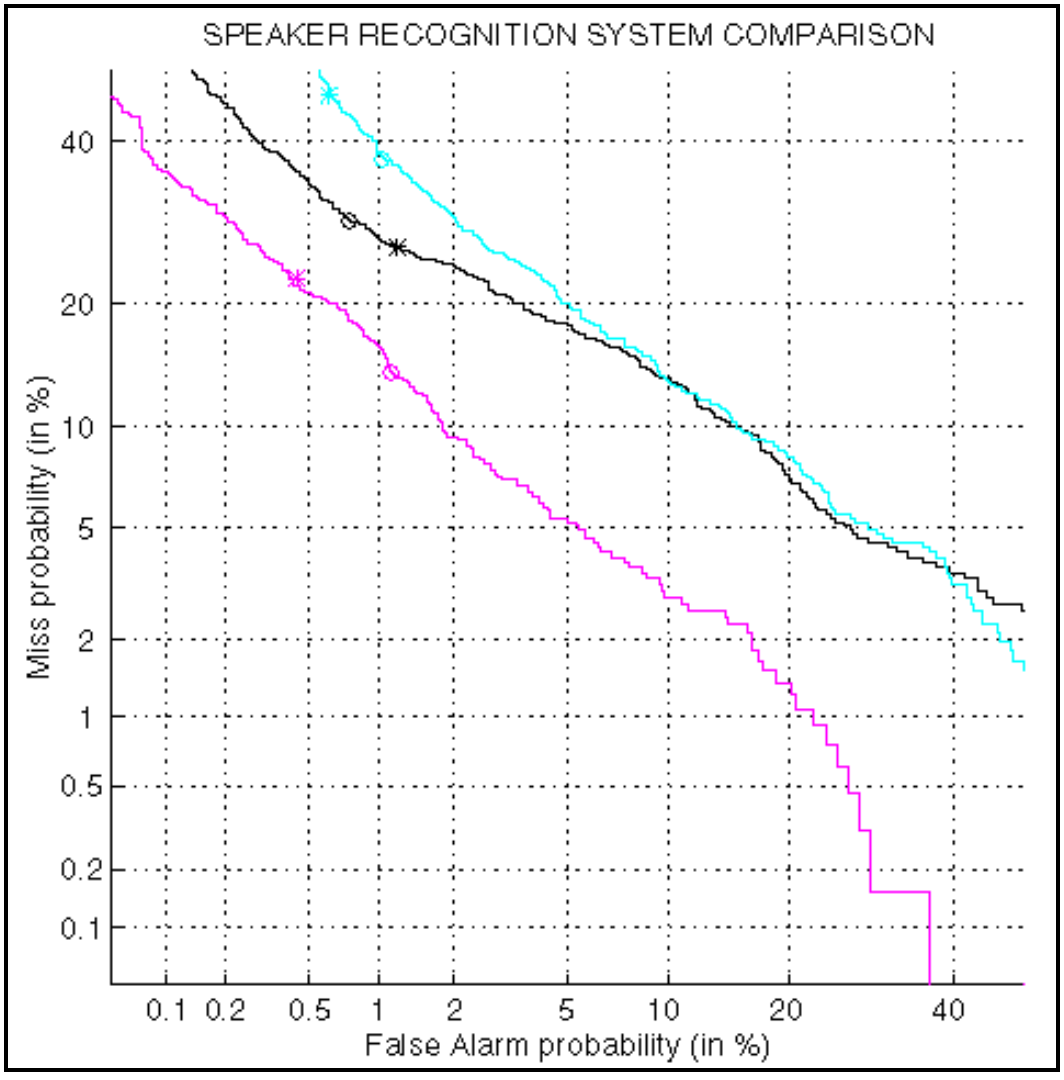
$$\Delta_{EER} = \arg\limits_{\Delta} \{P_{FA}(\Delta) = P_{FR}(\Delta)\}$$

$$P_{FA}(\Delta_{EER}) = P_{FR}(\Delta_{EER}) = EER$$

2. Receiver operating characteristics (ROC): Plotting $P_{FA}(\Delta)$ versus $P_{TP}(\Delta)$ on a linear scale as a function of Δ.
3. Detection error trade off (DET) curve: Plotting $P_{FR}(\Delta)$ versus $P_{FA}(\Delta)$ on a normal deviate (log) scale as a function of Δ. Helps in distinguishing different well performing systems.
4. Decision cost function (DCF): Given a system operational cost $\alpha \leq 1$ associated with one of the errors, say, FA

$$DCF(\alpha, \Delta) = \alpha * P_{FA}(\Delta) + (1 - \alpha) * P_{FR}(\Delta)$$

DET curve example



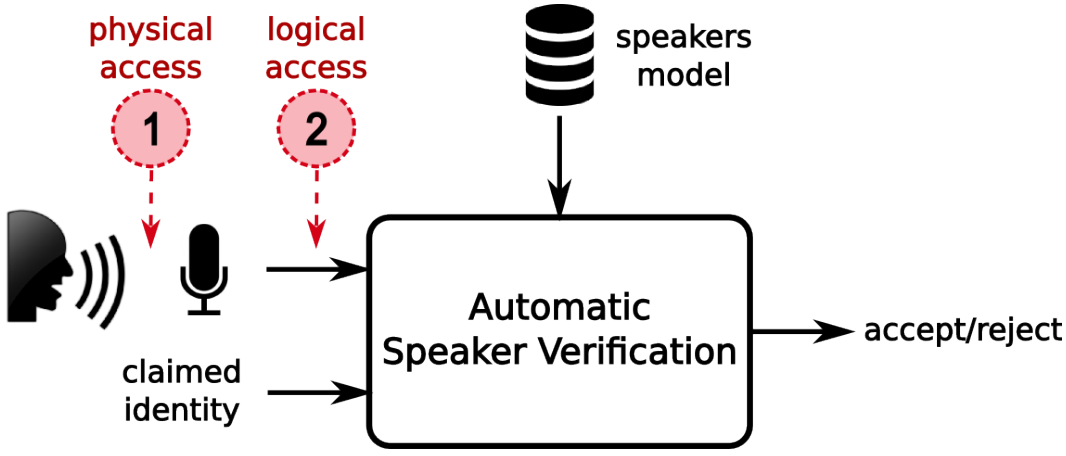
Source: A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki, "The DET curve in assessment of detection performance", in Proc. of Eurospeech, 1997.

1. Different systems can be compared at different operating points (i.e., decision thresholds)
2. Closer is the curve to the origin better is the speaker verification system performance
3. Point of intersection of the curve with the diagonal line from the origin yields EER estimate.

Outline

Presentation attacks

Forged or altered speech samples

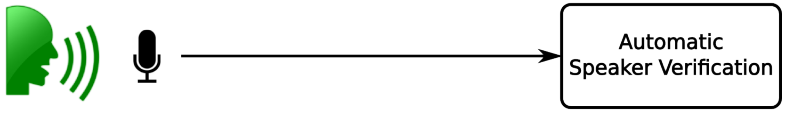


- Physical access: attacker needs a replay device
- Logical access: attacker needs to hack and inject the fake sample into the system

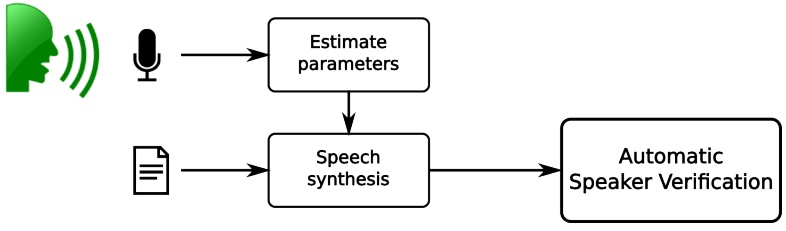
Presentation attacks

Bona fide sample

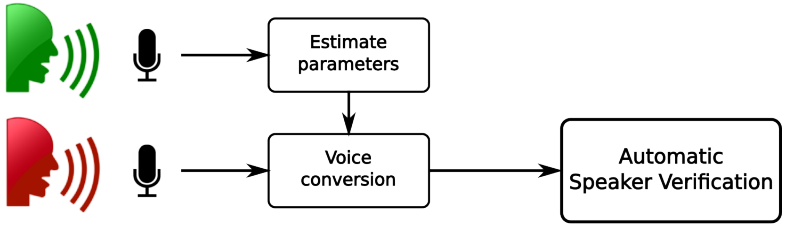
Replay



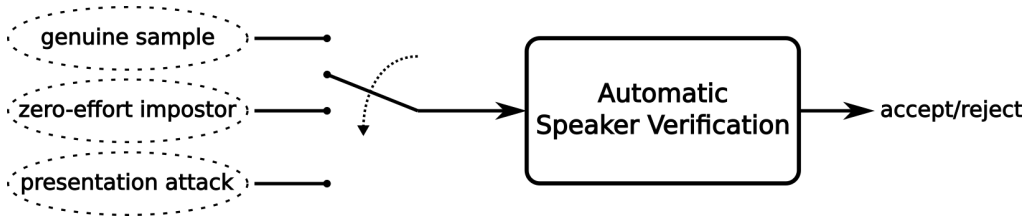
Speech synthesis



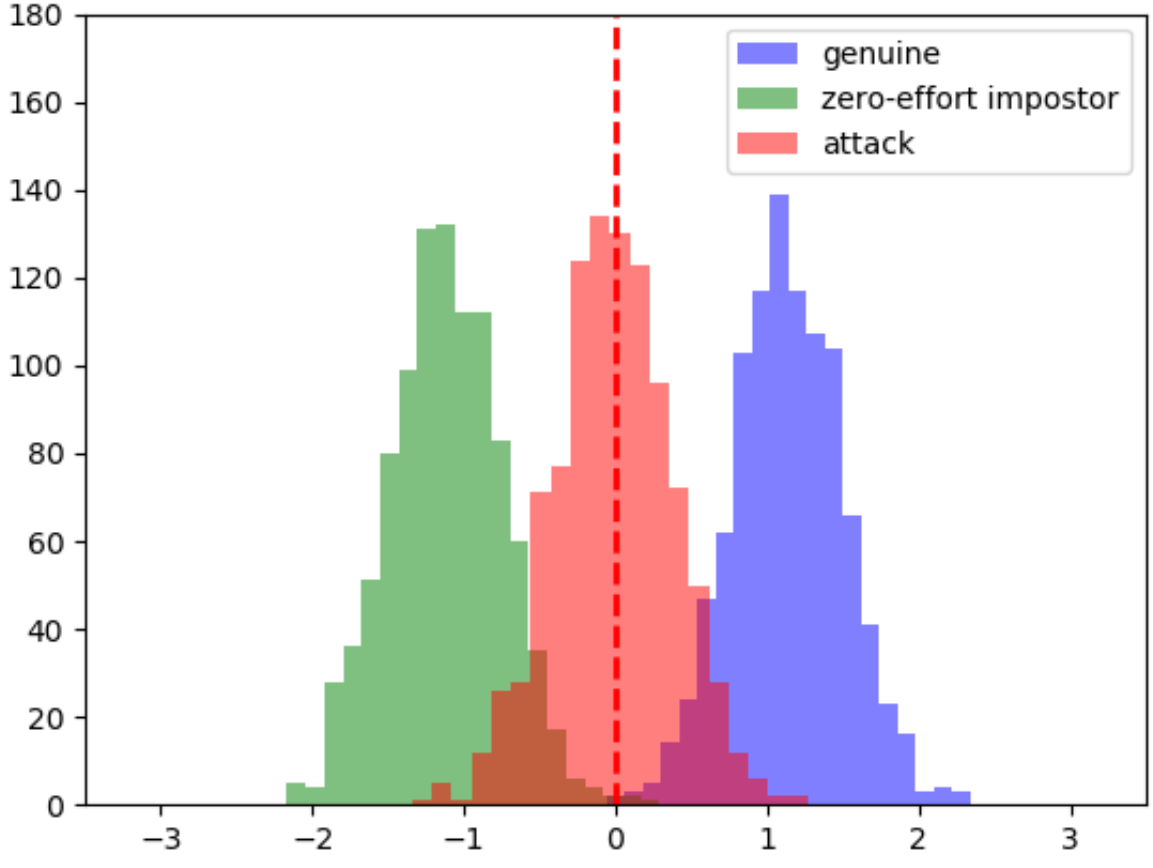
Voice conversion



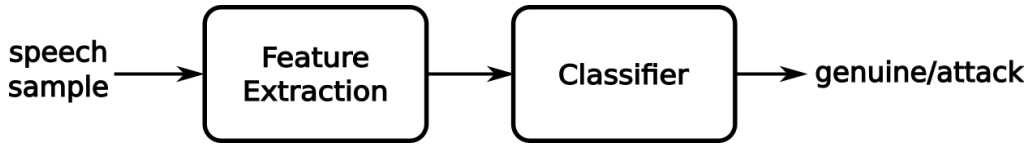
Speaker verification system vulnerability



Zero-effort-impostor refers to conventional impostor in speaker verification system.



Presentation attack detection (PAD)



Challenge: we do not have a good prior knowledge about the "task specific" information present in the speech signal.
Features:

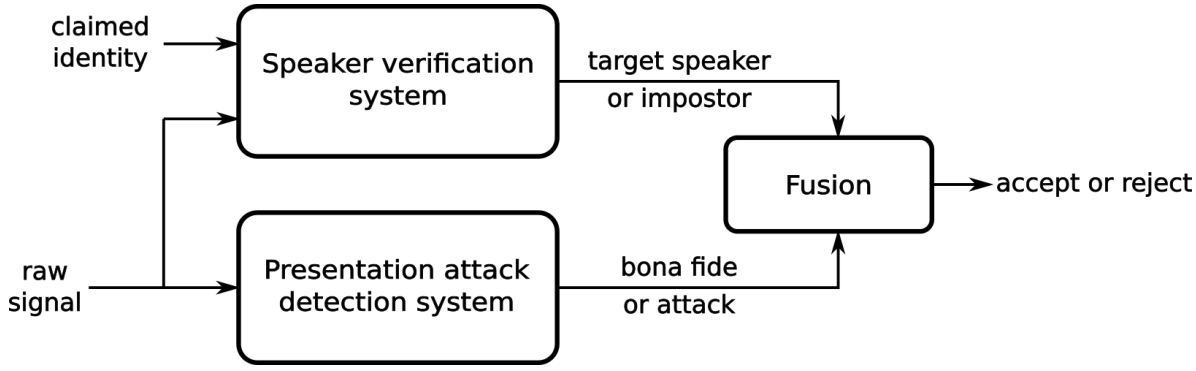
- magnitude spectrum-based features (e.g, cepstral features, filter-bank energies);
- phase spectrum-based features (e.g., group delay, relative phase-shift);
- spectro-temporal features (e.g., modulation spectrum).

Classifiers: Gaussian mixture models, neural networks, support vector machine, logistic regression.

PAD system evaluation

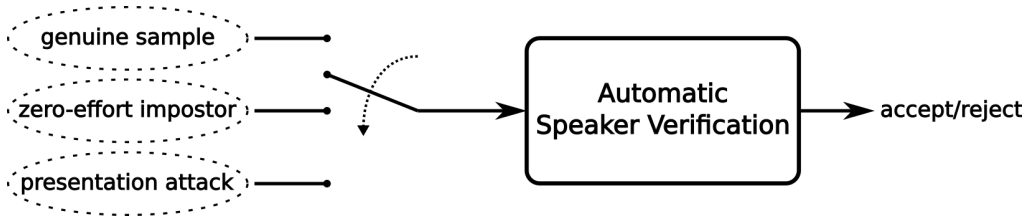
- Types of errors: False acceptance (attack classified as genuine) and False rejection (genuine classified as attack)
- Evaluation measures used for speaker verification system evaluation such as, HTER, EER, ROC, DET, DCF can be employed

Independent development and evaluation of speaker verification system and presentation attack detection system is not sufficient. Challenge: how to combine the two systems? (open research question)

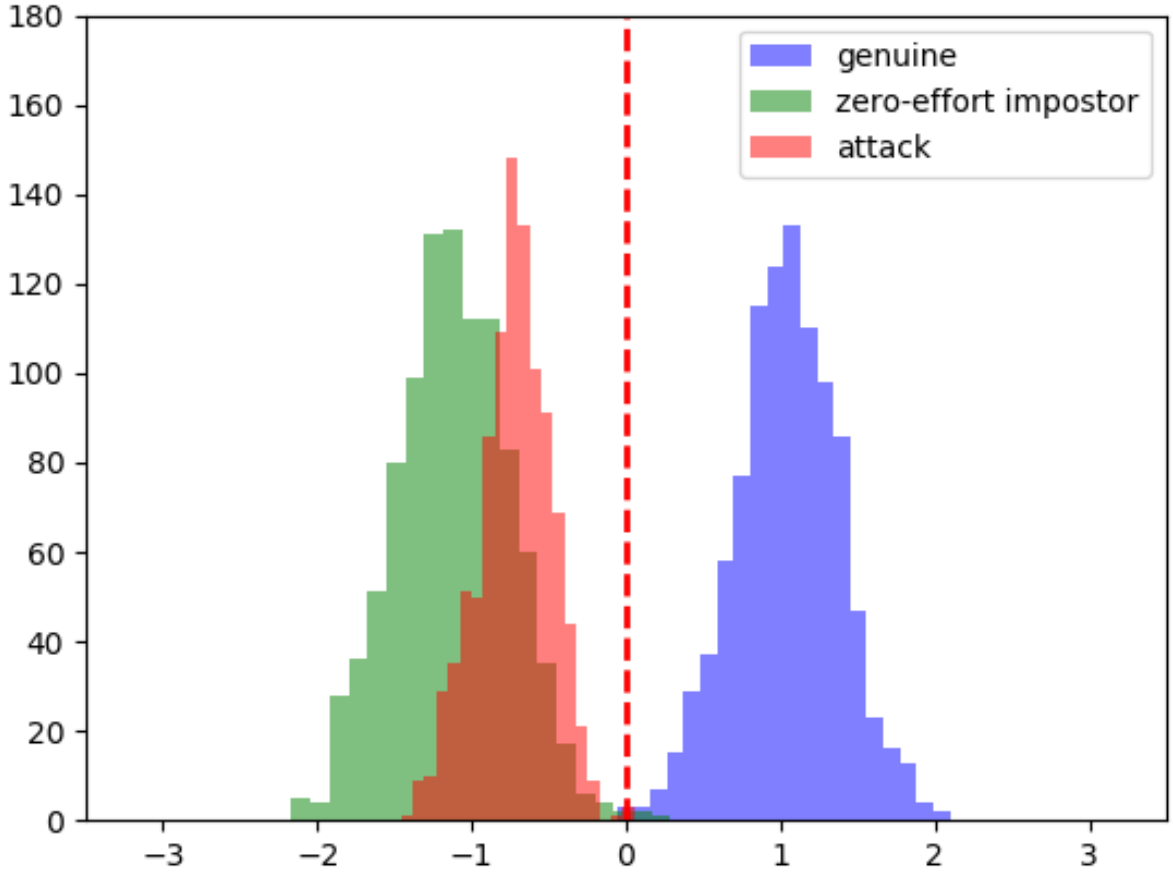


One solution: Fuse the decisions of the two systems using AND logic.

Vulnerability after integrating PAD



Zero-effort-impostor refers to conventional impostor in speaker verification system.



Thank you for your attention!

Dr. Mathew Magimai Doss and Dr. Petr Motlicek

Idiap Research Institute, Martigny, Switzerland