Speech Processing

Introduction to automatic speech recognition & other speech classification tasks

Alberto Abad

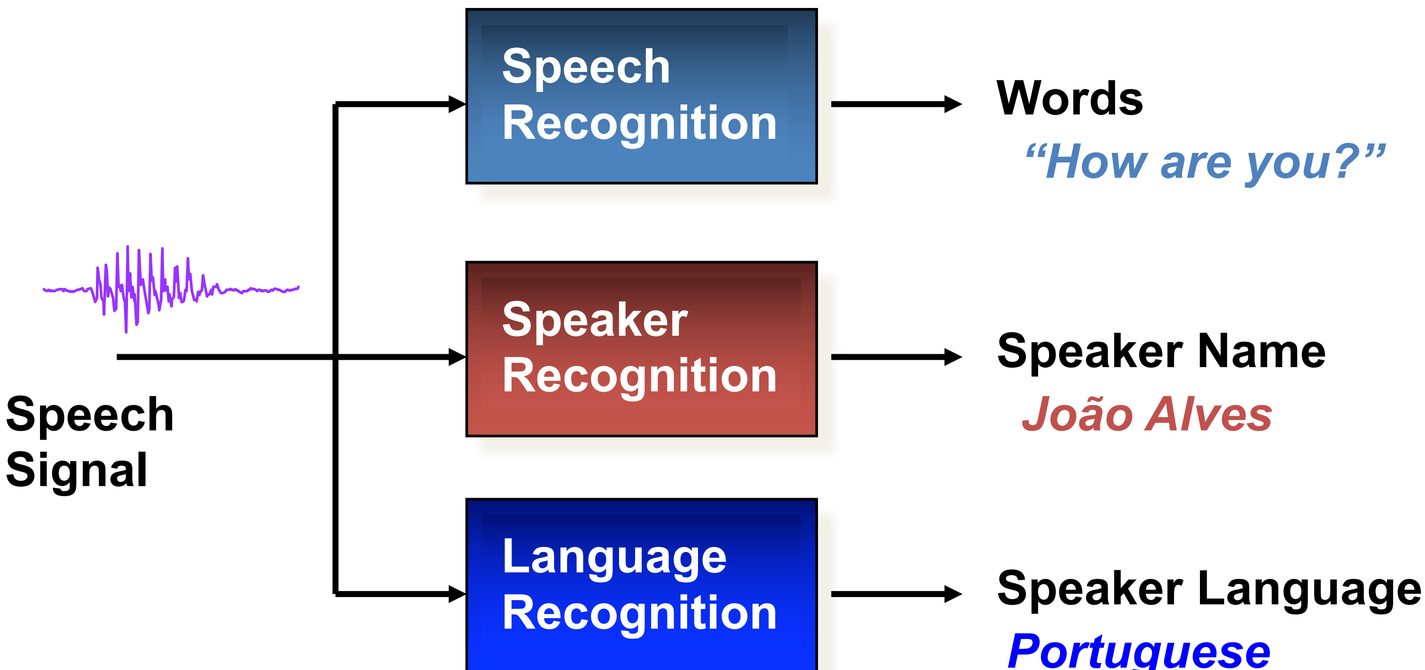
IST/INESC-ID

alberto.abad@tecnico.ulisboa.pt

Processamento de Língua Natural – MEIC IST, November, 2020

Introduction

Human Language Technologies



Speech processing: Speech coding, Speech enhancement, Audio segmentation, Text-to-speech synthesis, Automatic speech recognition, Speaker and language identification; Other speech pattern classification tasks

Text processing: Morphological analysis, Syntactic analysis, Semantic analysis, Discourse analysis, Named entity extraction, NL Generation, Information retrieval, Summarization, Question answering, Machine translation, Text analytics

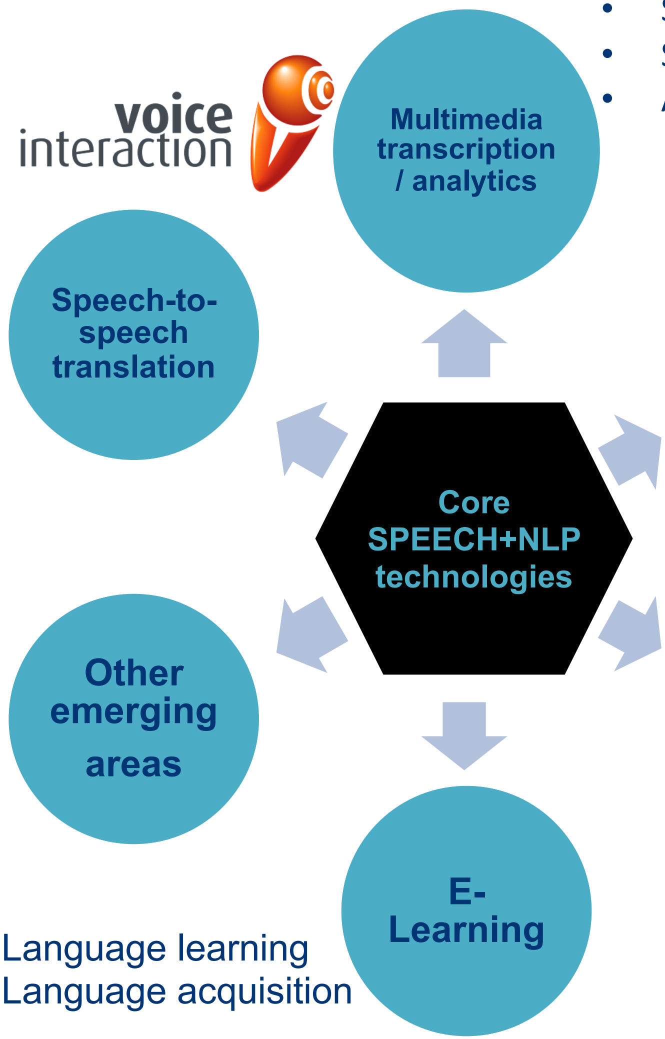
Spoken language processing Speech understanding, Speech synthesis from concepts, Spoken/multimodal dialog systems, Classification of multimedia documents, Summarization of spoken documents, Question answering on multimedia documents, Rich Transcription of multimedia documents, Speech-to-speech machine translation, Speech analytics

Introduction

Core application areas @ HLT.INESC-ID

- Rich transcription
- Speaker characterization
- Speech analytics/Text mining
- Audio & Music processing

- MT4 Microblogs
- Voice conversion
- Quality evaluation



- Smart homes
- Cultural heritage
- Chat bots

- Virtual therapists
- Monitoring neurodeg.
- Detection of pathological speech, depression, etc.
- Active ageing
- IE/IR for medical texts
- Sign language

- Security/Prosody
- Cyberbullying
- Brain interfaces

- Language learning
- Language acquisition

Introduction

Related disciplines

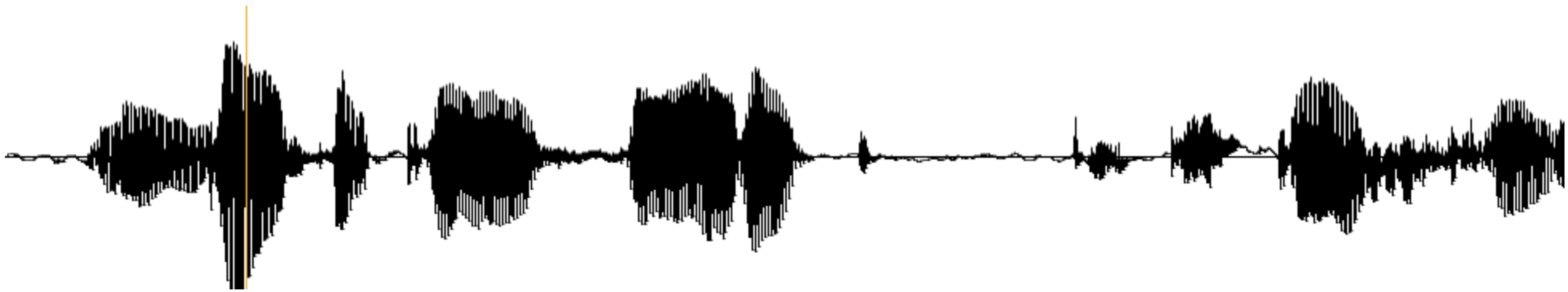
- Speech (and Language) processing is a challenging multidisciplinary research topic, that is related to areas, such as:
  - Digital signal processing
  - Speech sciences: linguistics, phonetics, sintaxis, etc.
  - Acoustics and physics
  - Natural language processing
  - Machine Learning
  - Human computer interaction
  - Artificial intelligence
  - Cognitive science
  - Perceptual psychology
  - …

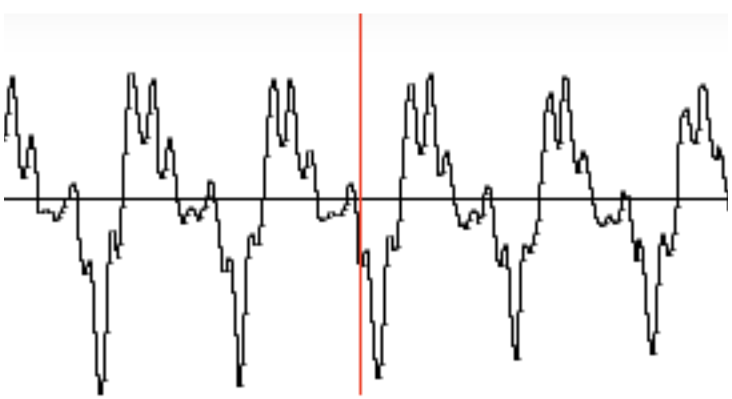
Outline

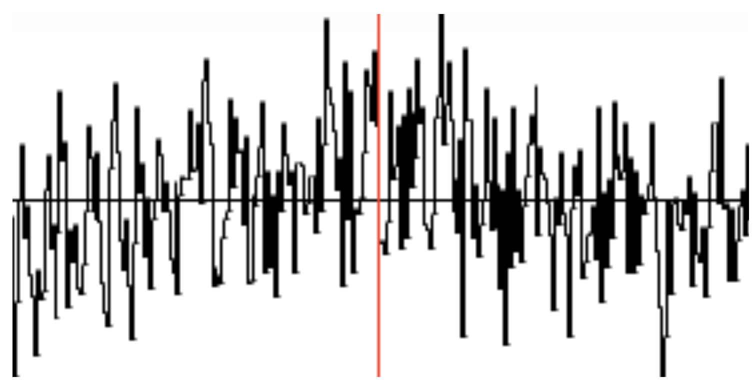
- Introduction to speech processing
- **Speech Pattern classification**
  - **Introduction to SPC**
  - Feature Extraction
    - Type of features
    - MFCCs
  - Machine learning
    - Speech common models
    - GMM
  - The “complex” task example: ASR
    - HMM
    - Examples
- ASR & Speech Pattern classification task examples
- Tools and references

Introduction to SPC

Speech signal in the time domain

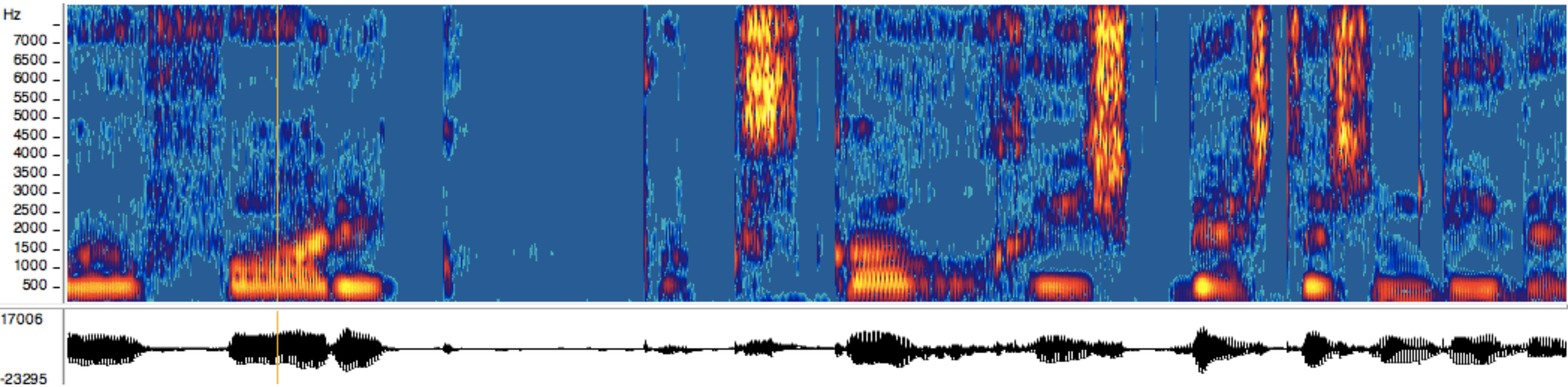


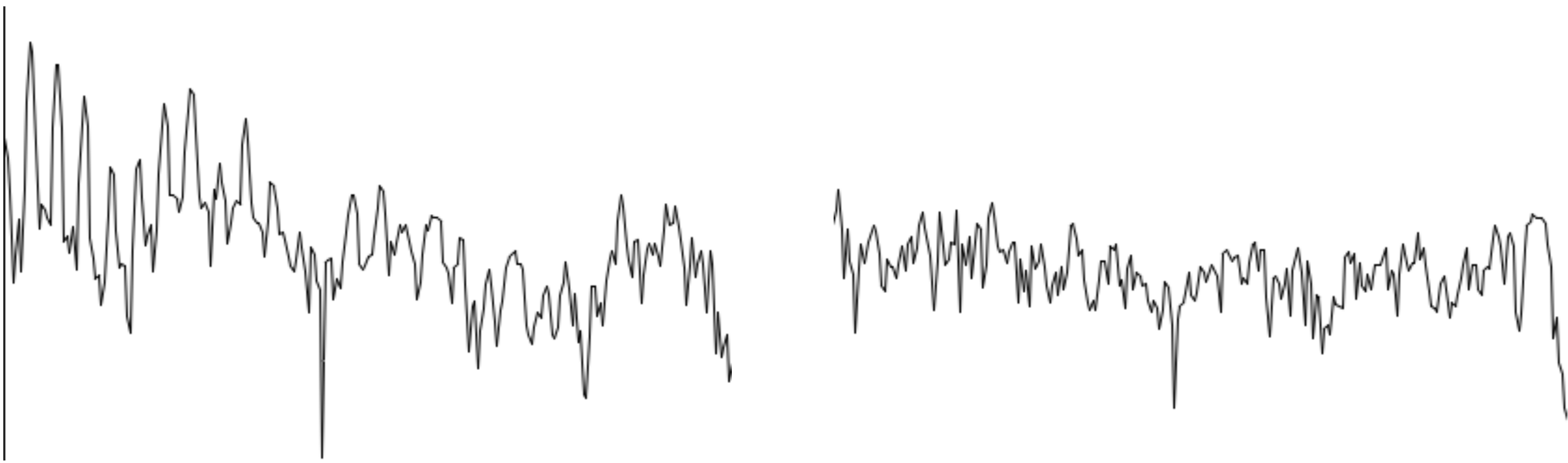




Introduction to SPC

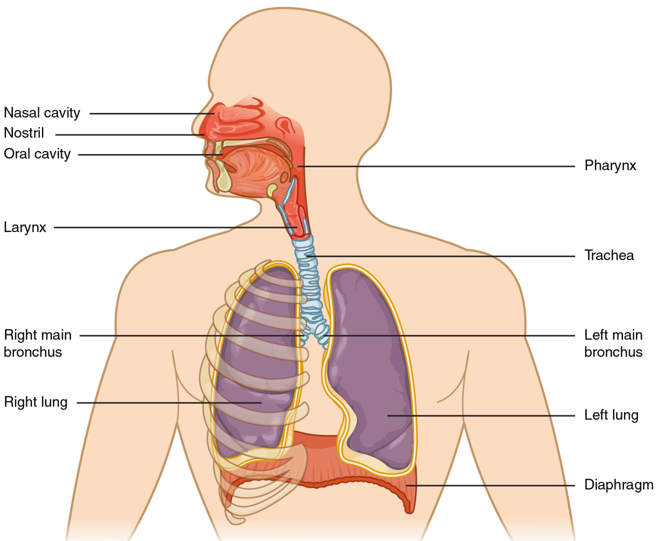
Speech signal time / frequency representation

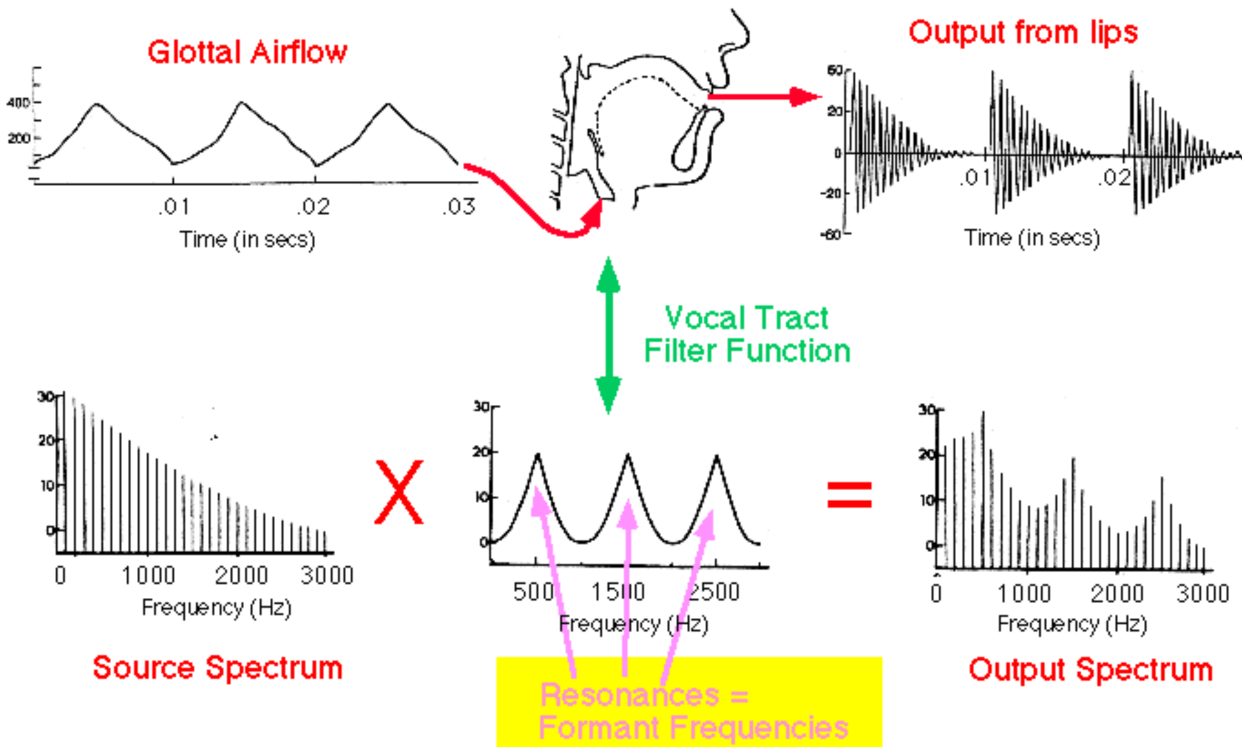




Introduction to SPC

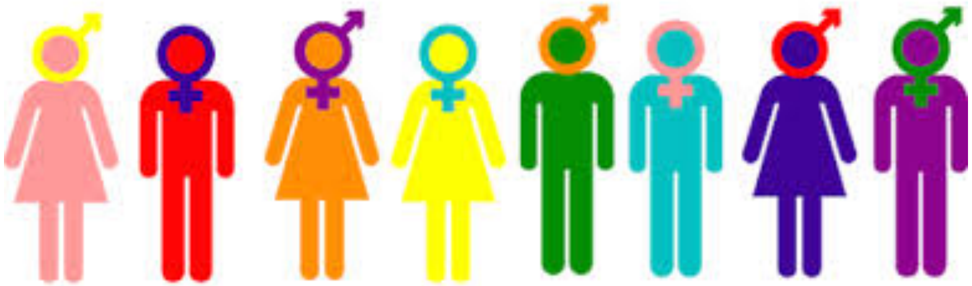
Speech signal: Physiology & Source/filter model





Introduction to SPC

- Speech carries a lot of information:
  - Of course information related to the message (LINGUISTIC?)...
  - ... but also, speaker traits (NON-LINGUISTIC/PARA-LINGUISTIC?):
    - Gender; Age; Language/accent; ID; Personality; Education; Intoxication; Sleepiness; Friendliness; Mood; Physical Stress; Cognitive Load; Emotion; Pathologies?

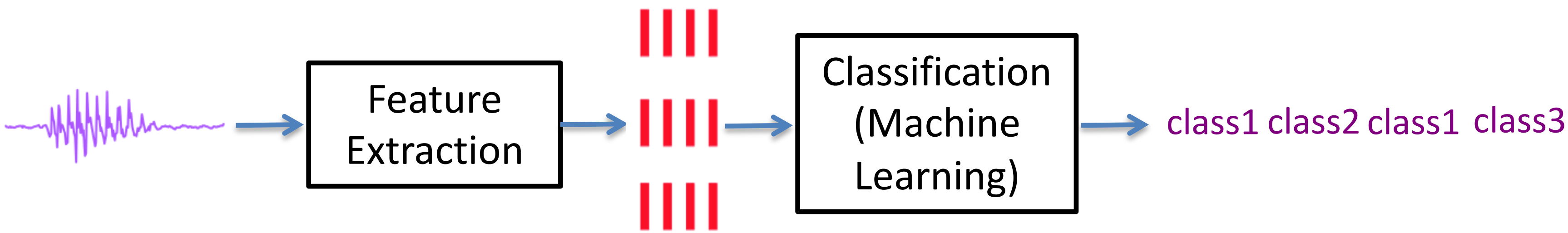


- If "Speech" is considered in a wider sense ("Audio") then more information is present:
  - Number of speakers; speakers role; speaker position; audio events; acoustic Scenes;

Introduction to SPC

Objectives

- The objective of ***speech pattern classification*** is to convert a speech input sequence into a sequence of class labels:



- The common blocks of any speech pattern classification task are the front-end/feature extraction and the back-end/classification:
  - The classifier module is “learnt” using data during the training phase and used to classify new unseen data during test
- Some examples are:
  - Automatic speech recognition; speech segmentation; speaker recognition; language recognition; speaker diarization; automatic document indexing; paralinguistic speaker trait recognition

Introduction to SPC

Challenges

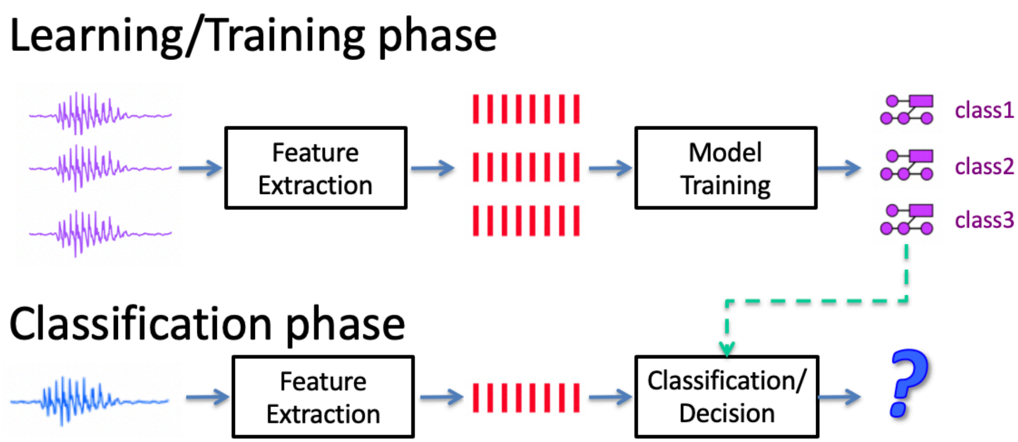
- Speech/audio variability → Samples belonging to the same "class" take extremely different forms due to:
  - Source variation: speaker, gender, accent, state, volume, etc.
  - Channel variation: mikes, acoustic environment, noise, reverberation, etc.
  - Other: Intrinsic nature of the classes, etc.

- From ML perspective, speech is a quite unique problem due to the nature of the input and class label outputs:
  - About the input → Time sequence
    - Very different length of the input wrt. output → Segmentation problem
    - Elasticity of the temporal dimension
    - Discriminative cues often distributed over a long temporal span
  - About the output → Output can be a sequence of class labels
    - Too much combinations → **Need structure!!!**

Introduction to SPC

The "simple" vs. the "complex" task

- The "simple" SPC task:



  - Static output:
    - No sequence of output labels
    - No segmentation problem → Audio segment corresponds to single class
  - No structured knowledge → Models correspond to output labels
  - Notice that:
    - Although being "simpler" from the ML perspective, they can be very hard
    - Can be classification/identification, verification or regression problems
    - Time-varying input still needs to be addressed

Outline

- Introduction to speech processing
- **Speech Pattern classification**
  - Introduction to SPC
  - **Feature Extraction**
    - Type of features
    - MFCCs
  - Machine learning
    - Speech common models
    - GMM
  - The “complex” task example: ASR
    - HMM
    - Examples
- ASR & Speech Pattern classification task examples
- Tools and references

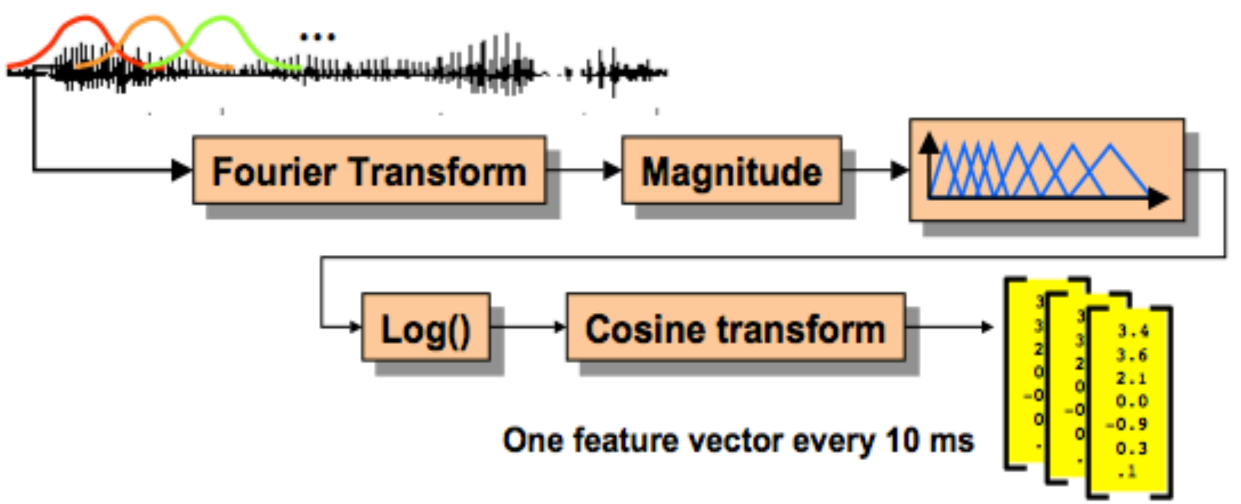
Features for SPC

- Desirable attributes of features for automatic methods:
  - **Informative**
    - Similar(dissimilar) sounds have similar (dissimilar) features
    - Provides discriminative information wrt the target task
      - Discard stuff irrelevant (ie. pitch in Portuguese ASR)
      - Pattern recognition techniques are rarely independent of the problem domain → proper selection of features affects performance
  - **Practical**
    - Occurs naturally and frequently in speech
    - Easy to measure
  - **Robust**
    - Not change over time
    - Not (very) affected by noise and channel

Classical spectral speech features

MFCC (Mel-frequency cepstral coefficients)

- Primary feature used in pattern recognition systems are cepstral feature vectors
- Some form of blind deconvolution is used to remove stationary channel effects
- Time differential cepstra (delta cepstra) are usually appended to cepstral features
- Typically 24-40 dimensional feature vectors are used



Slide after [1]

Outline

- Introduction to speech processing
- **Speech Pattern classification**
  - Introduction to SPC
  - Feature Extraction
    - Type of features
    - MFCCs
  - **Machine learning**
    - Speech common models
    - GMM
  - The “complex” task example: ASR
    - HMM
    - Examples
- ASR & Speech Pattern classification task examples
- Tools and references

Introduction to ML

- Assume we have a training set D={(x(i),y(i))} drawn from the distribution p(x,y), x€X y€Y

- The goal of learning is to find a decision function f: X → Y that correctly predicts the output of future input from the same distribution:

$$f(x) = \mathit{argmax}_y \, d_y(x)$$

- ML methods differ on:
  - Type of "discriminant function" (the model)
  - Type of "loss function" (the training objective)
  - How training data is used:
    - Supervised – all training samples are labeled
    - Semi-supervised – both labeled and unlabeled
    - Unsupervised – all training samples are unlabeled

Statistical models is speech pattern classification problems

- Several types of models have been used in different speech pattern recognition tasks, including:
  - K-NN – K nearest neighbor
  - MLP – Multi-layer perceptron
  - SVM – Support Vector Machines
  - DNN – Deep neural networks
  - etc.
- Traditionally, the most common model *has been* the **Gaussian Mixture Model (GMM)**

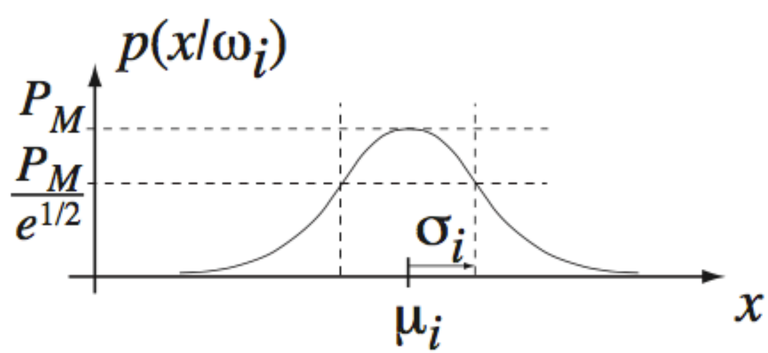
Gaussian mixture models (GMM)

Gaussian models

- Easiest way to model distributions is via parametric model
  - assume known form, estimate a few parameters
- Gaussian model is simple and useful. In 1D

$$p(x \mid \theta_i) = \frac{1}{\sigma_i \sqrt{2\pi}} \exp\left[-\frac{1}{2} \left(\frac{x - \mu_i}{\sigma_i} \right)^2 \right]$$

- Parameters mean μ_i and variance σ_i → fit



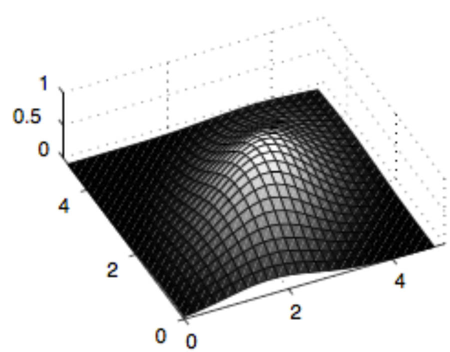
Slide after [1]

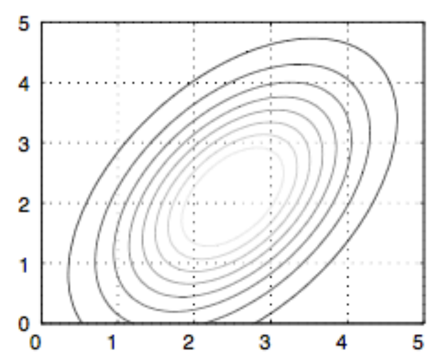
Gaussian mixture models (GMM)

Gaussians in d dimensions

$$p(\mathbf{x} \mid \theta_i) = \frac{1}{(2\pi)^{d/2}|\Sigma_i|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}-\mu_i)^T \Sigma_i^{-1}(\mathbf{x}-\mu_i)\right]$$

Described by a d-dimensional mean μ_i
and a $d \times d$ covariance matrix Σ_i



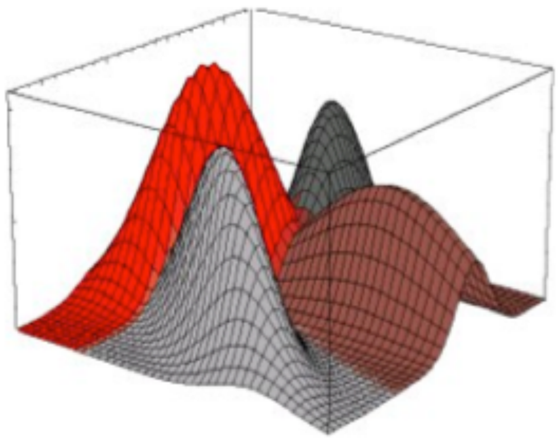


Slide after [1]

Gaussian mixture models (GMM)

Gaussian mixture models

- Single Gaussians cannot model
  - distributions with multiple modes
  - distributions with nonlinear correlations
- What about a weighted sum?



$$p(x) \approx \sum_k c_k p(x \mid \theta_k)$$

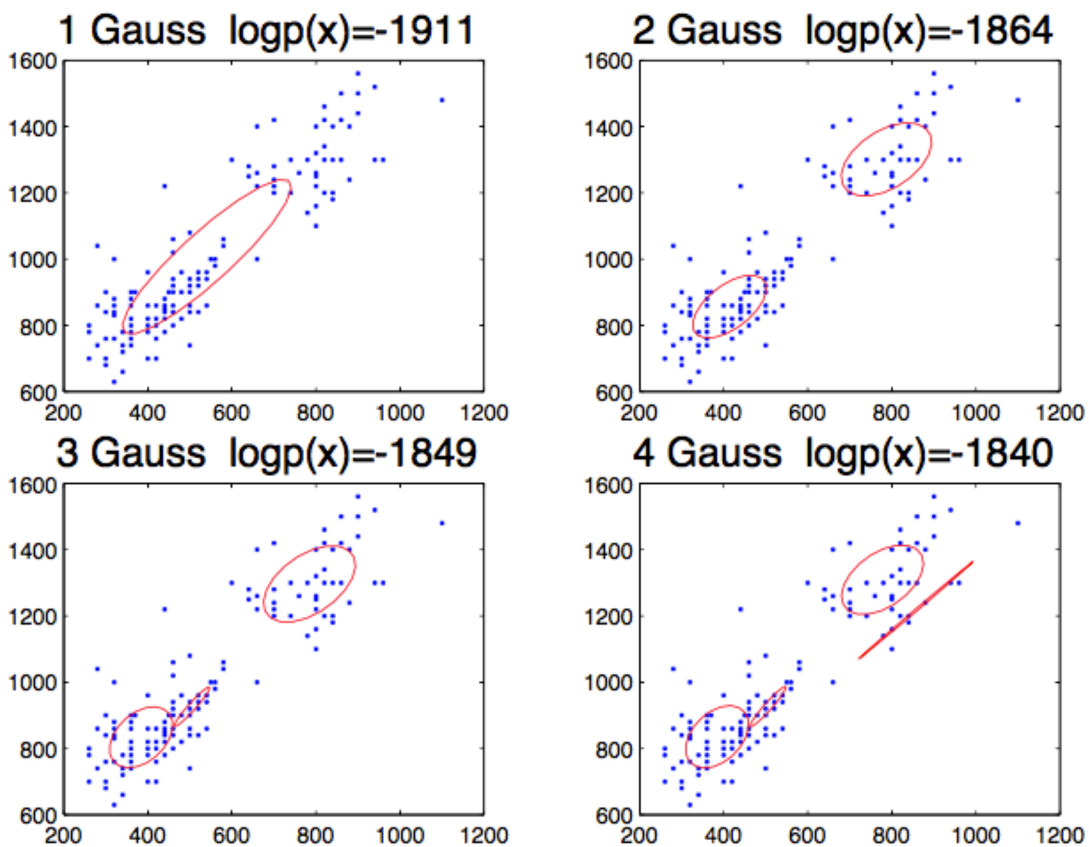
  - where $\{c_k\}$ is a set of weights and $\{p(x \mid \theta_k)\}$ is a set of Gaussian components
  - can fit anything given enough components
- Interpretation: each observation is generated by one of the Gaussians, chosen with probability $c_k = p(\theta_k)$

Slide after [1]

Gaussian mixture models (GMM)

GMM examples

Vowel data fit with different mixture counts



Slide after [1]

Outline

- Introduction to speech processing

- **Speech Pattern classification**
  - Introduction to SPC
  - Feature Extraction
    - Type of features
    - MFCCs
  - Machine learning
    - Speech common models
    - GMM
  - **The “complex” task example: ASR**
    - HMM
    - Examples

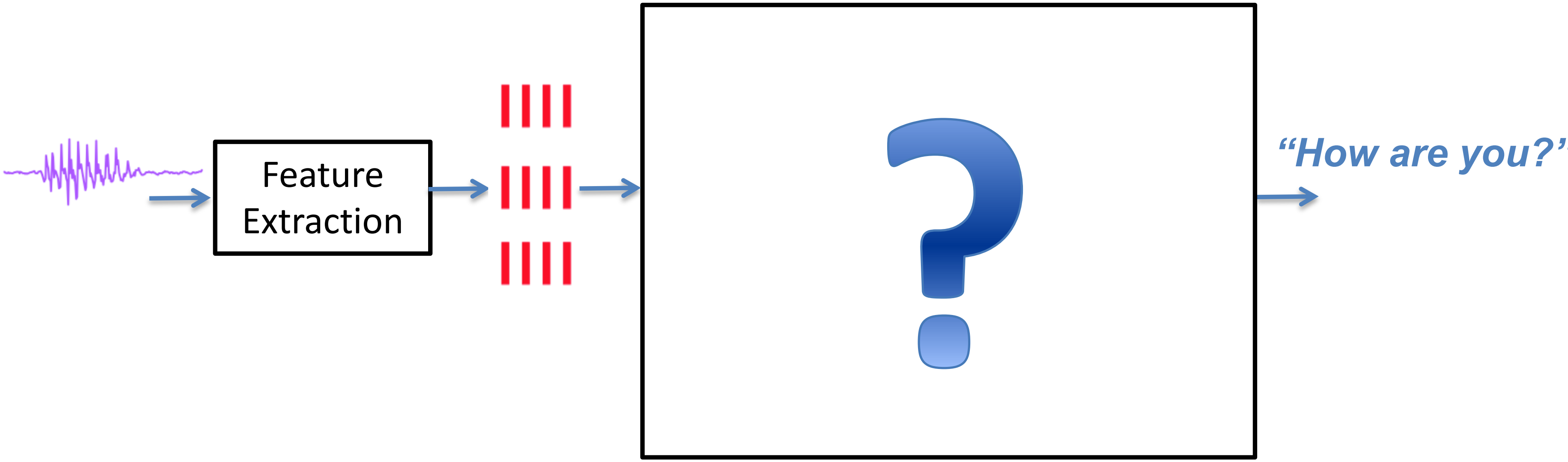
- ASR & Speech Pattern classification task examples

- Tools and references

Automatic Speech Recognition (ASR)

The “complex“ task

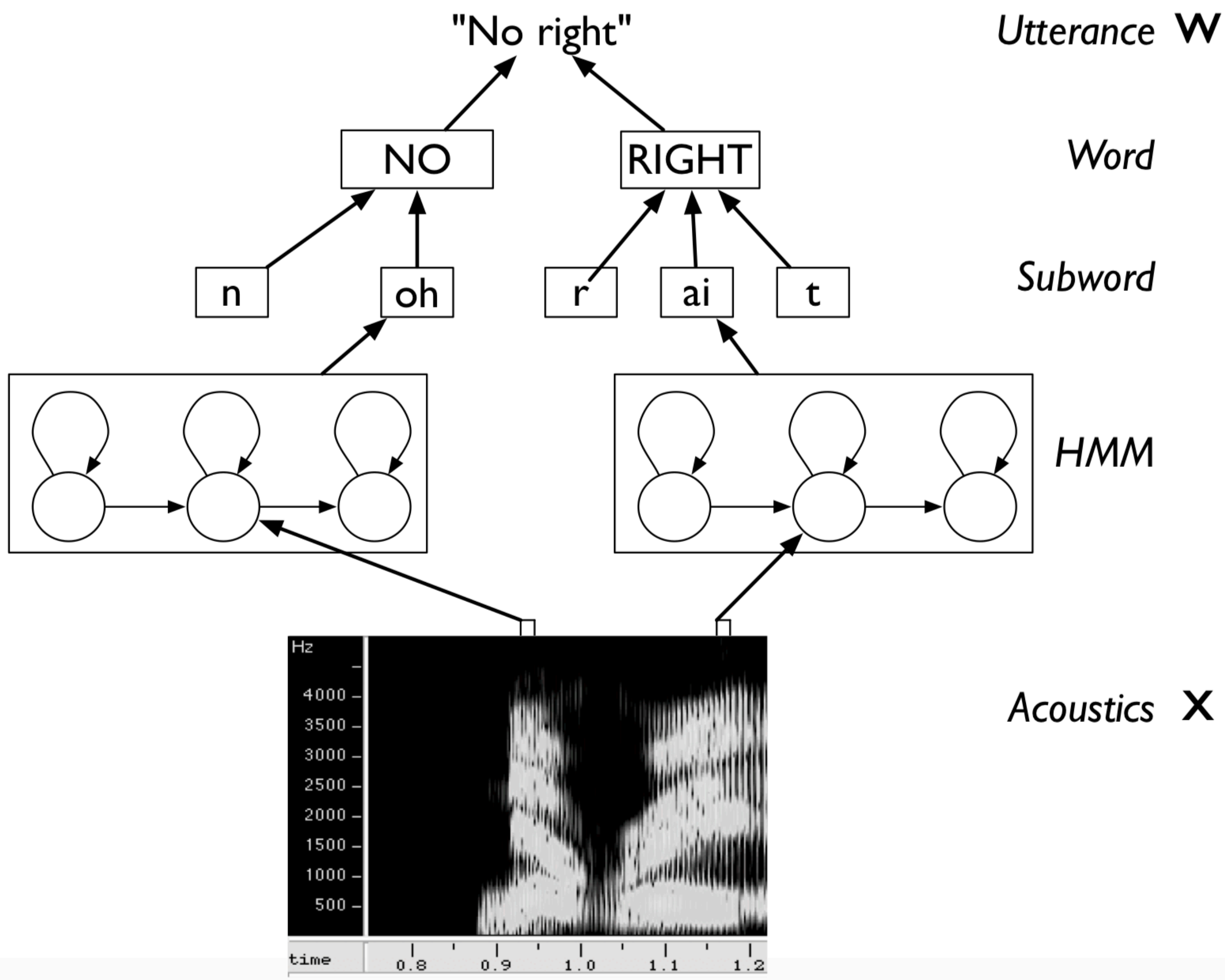
- **Goal** Given a sequence of observations determine which is the most likely sequence of words



- Already decades of research on ASR (and other SLT related topics) → **Very challenging!!!**
- **Related sub-tasks:** Isolated ASR, Continuous ASR, KWS, LVCSR, STD/Search on Speech, etc.

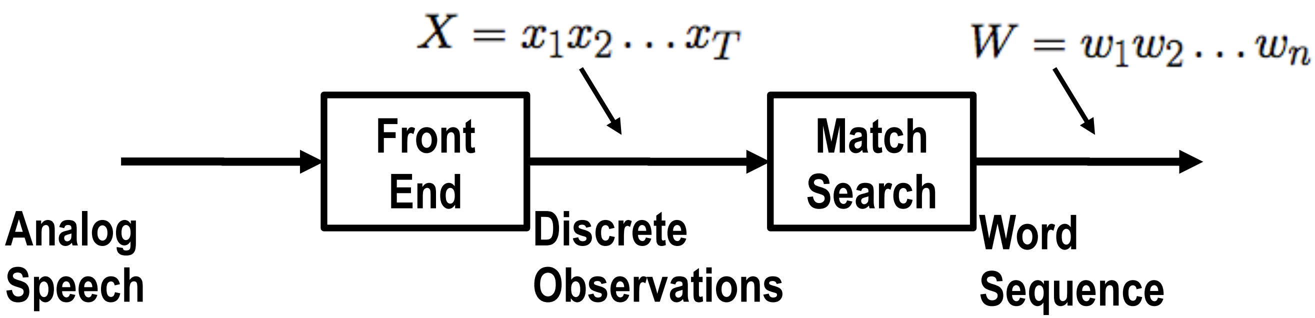
Automatic Speech Recognition

Hierarchical speech modeling



The goal of speech recognition

- Given the feature vector sequence $X = x_1 x_2 \ldots x_T$ the goal of speech recognition is to find the word sequence $W = w_1 w_2 \ldots w_n$ that has the maximum a posteriori $P(W|X)$



- Bayes Rule:

$$\hat{W} = \arg\max_{w} P(W|X) = \arg\max_{w} \frac{P(W)P(X|W)}{P(X)}$$

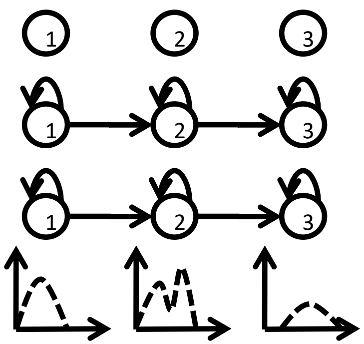
$$\hat{W} = \arg\max_{w} \underbrace{P(W)}_{\text{LM}} \underbrace{P(X|W)}_{\text{AM}}$$

Automatic Speech Recognition

Acoustic Model: Hidden Markov Model

- Hidden Markov Model (HMM) is a powerful statistical method of characterizing the observed data samples of a discrete-time series (speech), specified by:

- states 1…N
- transition probabilities $a_{i,j}=p(s_t=j|s_{t-1}=i)$
- emission distributions $b_j(x_t)=p(x_t|s_t=j)$



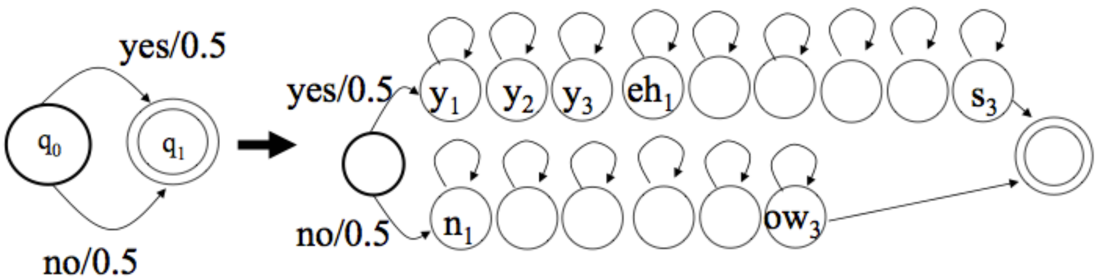
$\Phi = (A,B,\pi)$

$$P(X|\Phi) = \sum_{allS_q} P(X|S_q,\Phi)P(S_q|\Phi) = $$
$$= \sum_{allS_q} \prod_t P(X_t|s_{q,t})P(s_{q,t}|s_{q,t-1})$$

- The HMM assumptions:
  1. **Markov process:** $P(S_{q,t}|S_{q,t-1},S_{q,t-2},\ldots,S_{q,1}) = P(S_{q,t}|S_{q,t-1})$
  2. **Observation independence:** $P(X_t|S_{q,t},S_{q,t-1},S_{q,t-2},\ldots,S_{q,1}, X_{t-1},X_{t-2},\ldots,X_1) = P(X_t|S_{q,t})$
  - Under these assumptions, there are good algorithms to use HMMs: Forward, Viterbi, Baum-Welch

Language Model: CFG vs n-grams

- CFGs
  + well-adapted to simple phrases (eg. digits)
  -- complex phrases
  -- "wrong" phrases not allowed



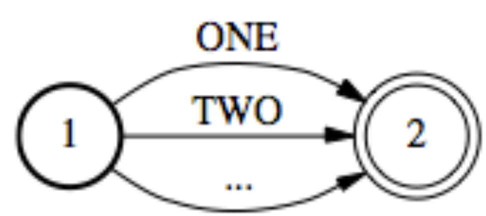
- Statistical LM → **n-grams**
  + P(W) depends on n-1 previous words (tractable)
  + "wrong" phrases possible
  -- need large amounts of texts to estimate probabilities
  -- OOVs, no long term, no linguistic knowledge

$$\boxed{P(w_1, w_2, \cdots, w_n) = P(w_1) \prod_{n=2}^{N} P(w_n | w_{n-1})}$$

Automatic Speech Recognition

Isolated Word Recognition with HMMs

- For every word W we define a HMM model $\Phi_{W:}$
  - A reasonable number of states is 3 states per phoneme
- Training
  - for each class (word) W, collect all training samples X with that label (manual)
  - to train Φ_W, run Baum-Welch on this data
- Decoding
  - Calculate (Viterbi or Forward) $P(X|\Phi_W)$ for every W and pick the best
- Decoding (better)
  - Merge each word HMM in a single big HMM (all words in parallel)
  - use Viterbi to find the best sequence (backtrace to obtain words)

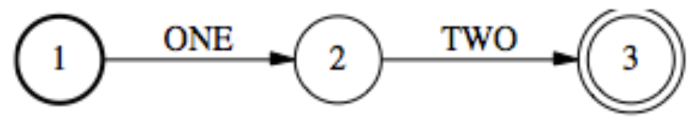


Automatic Speech Recognition

Continuous Speech Recognition with HMMs

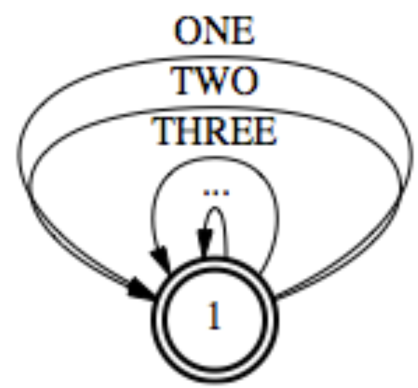
Example Digit string

- We can not build an HMM for each digit sequence
  - define word HMMs for every digit



- Training
  - glue the sequence HMM, update counts for each word HMM

- Decoding
  - build the (big) HMM (graph) that represents all digit strings
  - Apply Viterbi



Automatic Speech Recognition

LVCSR with HMMs

- Basic principals stand, but:
  - Acoustic Models (AM)
    - basic units are sub-word (context-dependent) units:
      - need for a **Pronunciation Model**
    - need to increase AM complexity
  - Language Model (LM)
    - can not use simple grammar/rules FSA
    - use probabilistic models → **n-grams**
  - Decoding
    - increased complexity affects to the size of the search space (graph)
    - direct Viterbi over the whole graph is not impossible

Automatic Speech Recognition

Recent evolution and current trends

- Research in ASR has produced very significant outcomes during last decades (but it is still an open problem).
- Currently, there are two main current trends to tackle the problem:
  1. **Hierarchical modelling of speech**
     - Speech modelling problem is structured in sub-problems
     - This is the conventional approach until ~2012
     - Today still very relevant in certain tasks/conditions
  2. **end2end**
     - Direct mapping from acoustics to words/characters
     - Different flavours from 2012 (CTC, encoder-decoder, etc.)
     - State of the art (in very large data)

Outline

- Introduction to speech processing
- Speech Pattern classification
  - Introduction to SPC
  - Feature Extraction
    - Type of features
    - MFCCs
  - Machine learning
    - Speech common models
    - GMM
  - The “complex” task example: ASR
    - HMM
    - Examples
- **ASR & Speech Pattern classification task examples**
- Tools and references

Examples

Relevant ASR benchmarks

NIST Evaluations

NIST Speech-To-Text transcription (STT)

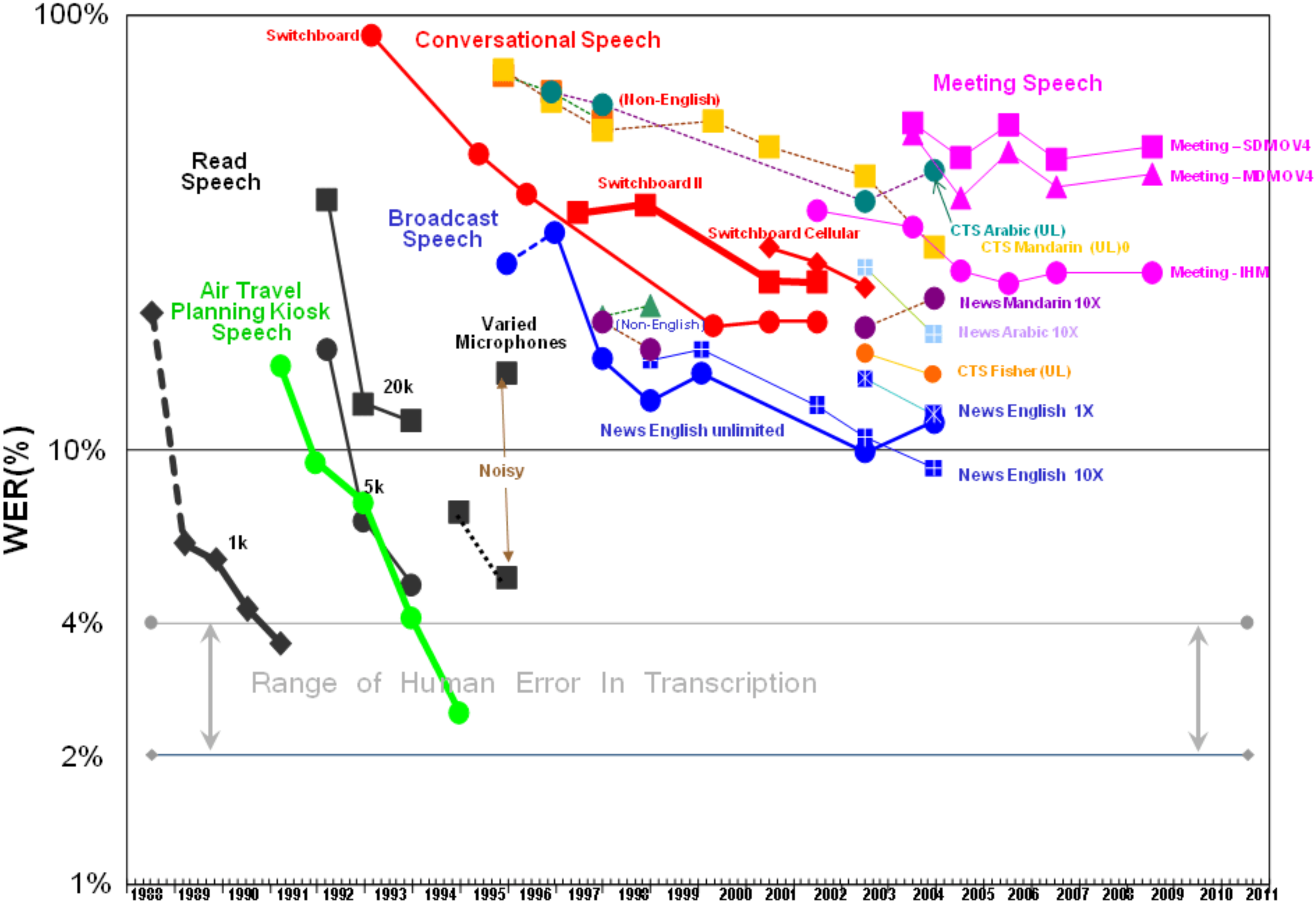
https://www.nist.gov/itl/iad/mig/rich-transcription-evaluation

ASR EVALUATION METRIC: Word error rate (WER)

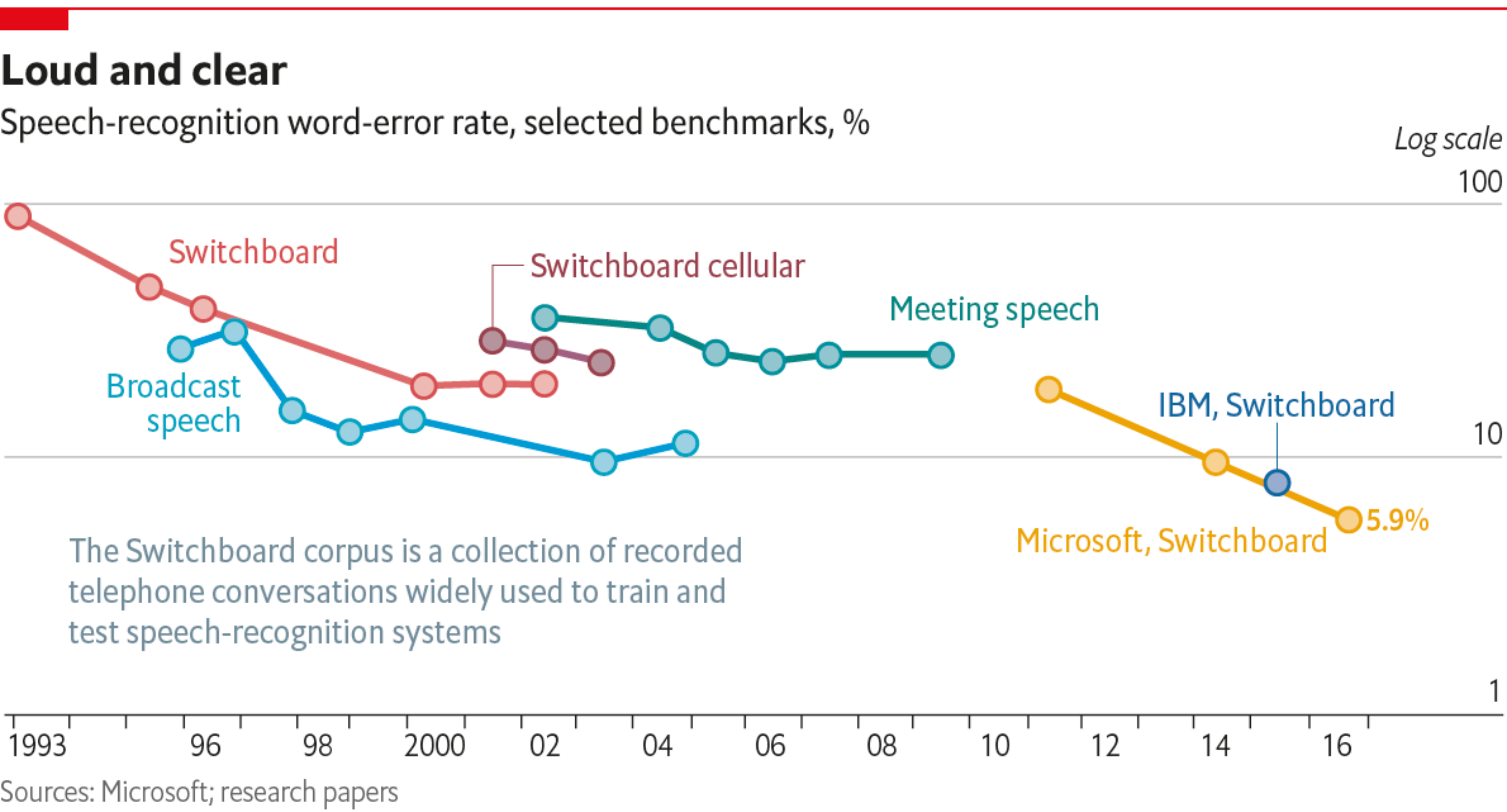
```
Scores: (#C #S #D #I) 9 3 1 2
REF:   was an engineer SO I   i was always with **** **** MEN UM   and they
HYP:   was an engineer ** AND i was always with THEM THEY ALL THAT and they
Eval:                  D  S                      I    I    S   S
```

$$\boxed{WER = \frac{I + S + D}{N}}$$

NIST STT Benchmark Test History – May. '09



ASR Benchmark history (more recent)



Examples

Relevant Paralinguistic/Non-linguistic challenges

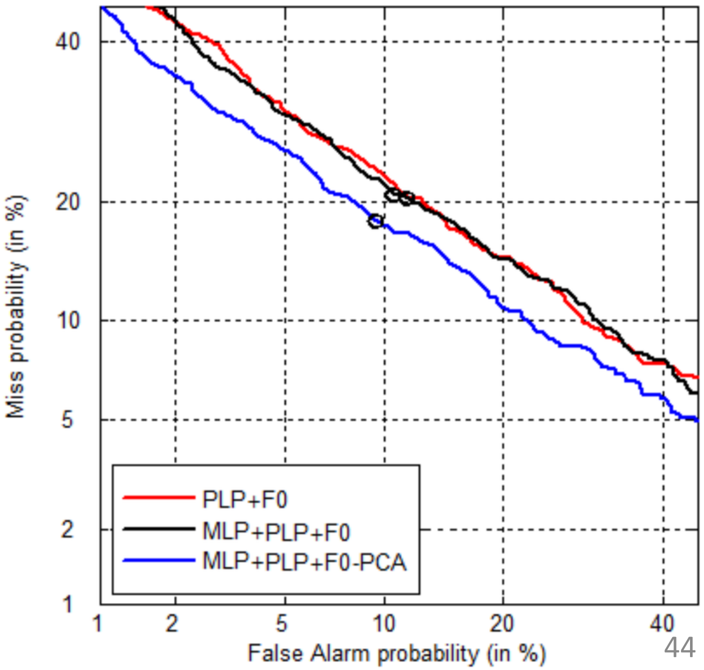
NIST Evaluations

NIST Speaker Recognition Evaluation (SRE)

http://www.nist.gov/itl/iad/mig/sre.cfm

SRE Metric: DET curves

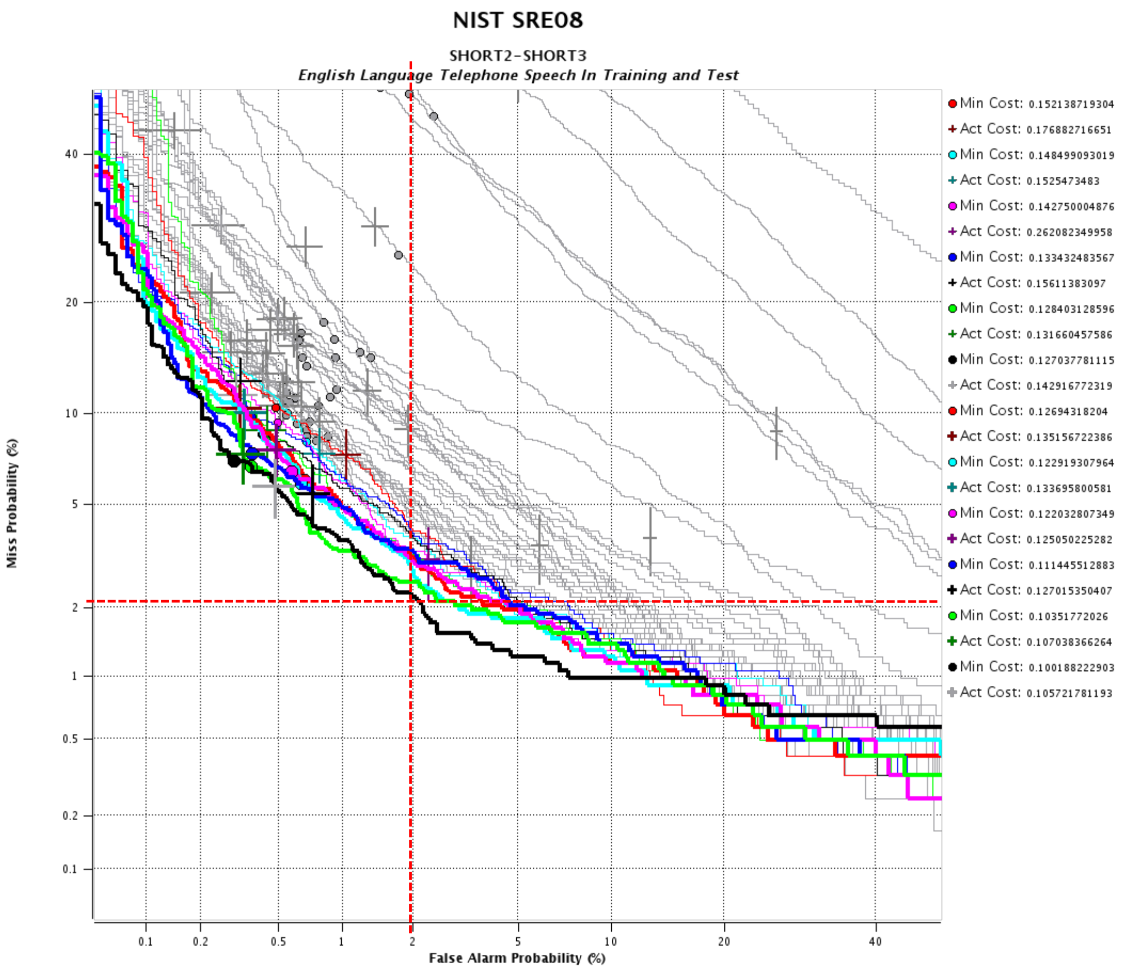
- For a large set of trials, plot of false alarm vs miss rate at different operation points



NIST SRE Results

NIST SRE 2008 core condition

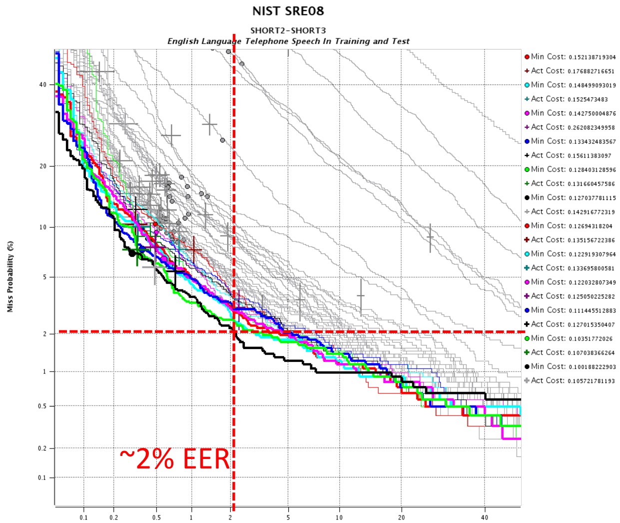
Tel-tel + only English sub-conditions



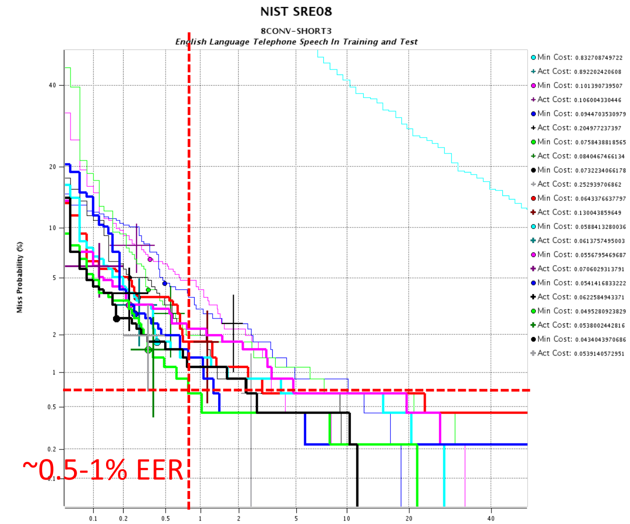
NIST SRE Results

NIST SRE 2008: Importance of speech length

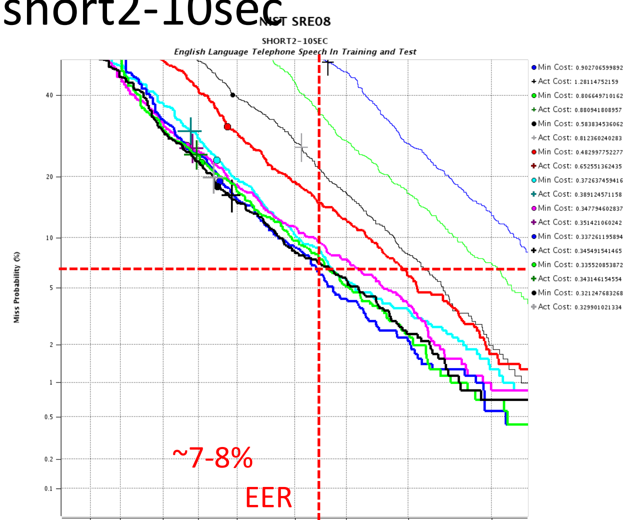
short2-short3



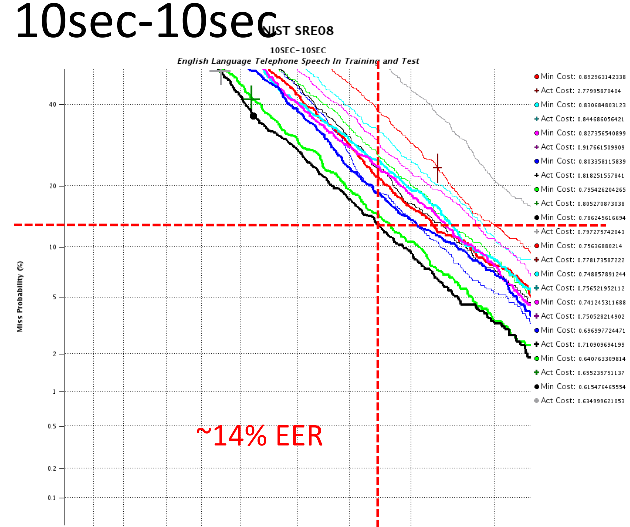
8conv-short3



short2-10sec



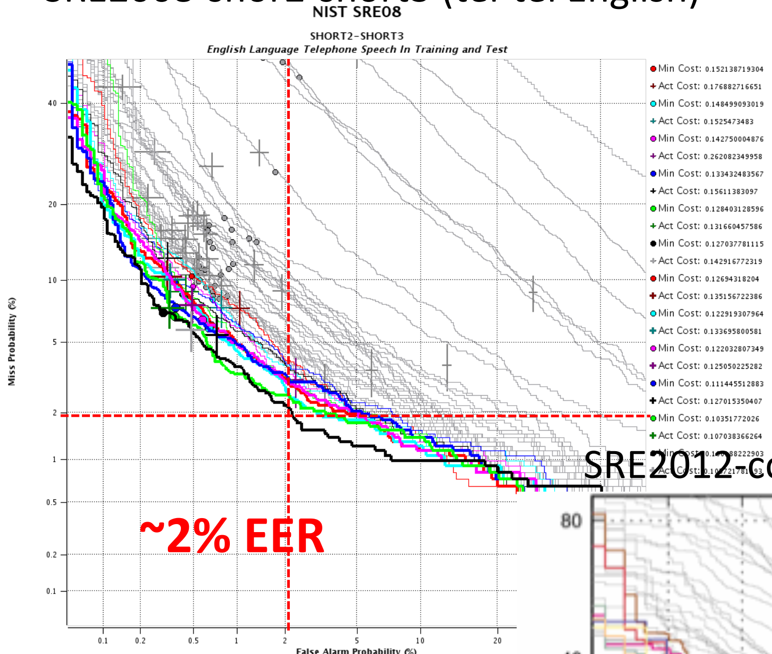
10sec-10sec



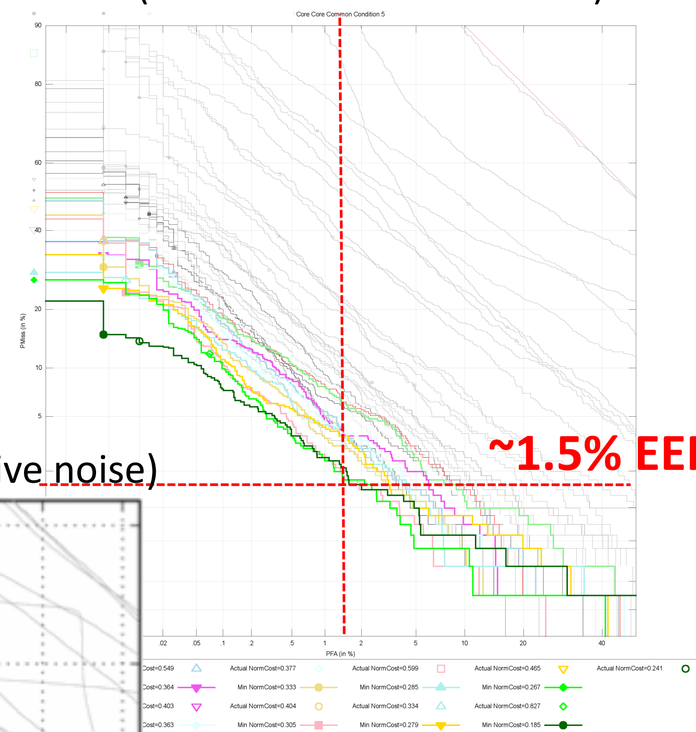
NIST SRE Results

Recent years evolution

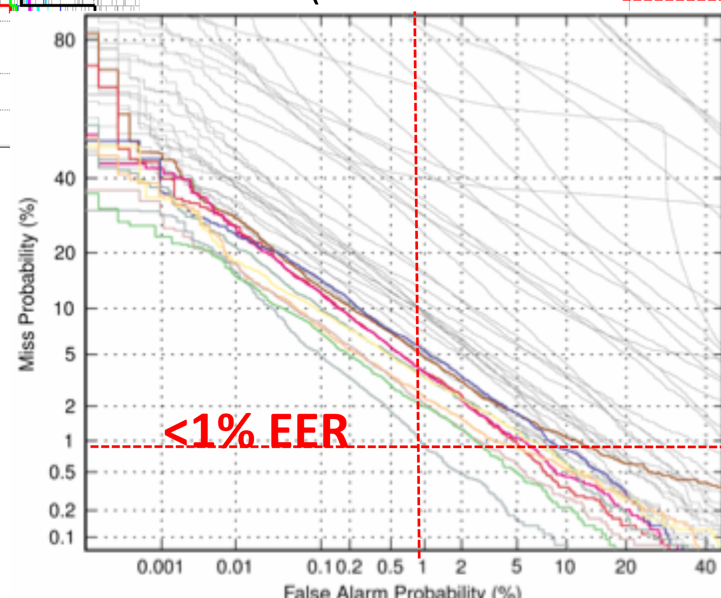
SRE2008-shor2-short3 (tel-tel English)



SRE2010-cc5 (tel-tel normal vocal effort)



SRE2012-cc2 (tel-tel no additive noise)



Examples

Relevant Paralinguistic/Non-linguistic challenges

COMPARE (Computational Paralinguistic Evaluation) Challenge Series

http://compare.openaudio.eu/

INTERSPEECH 2016 Computational Paralinguistics Challenge

Deception Sub-Challenge

Sincerity Sub-Challenge

Native Language Sub-Challenge

COMPARE2016

Results of the BEST system in the TEST set

	DEV [UAR %]	TEST [UAR %]
ComPaRe 2016 Official Baseline	45.1%	47.5%
INESC-ID ComPaRe 2016 system	**84.6%**	**81.3%**

COMPARE 2016
quiz

1 2 3 4 5 6 7

Arabic French German Italian Japanese Mandarin Chinese Spanish

Summary

- Speech processing has been the focus of extensive research during the last decades.
- As a result, there is a significant amount of very successful technologies in the market, such as Automatic Speech Recognition (ASR).
- ASR is a particularly difficult case of *speech pattern classification* due to the sequence to sequence nature of the task and the variability of speech:
  - Nevertheless, impressive results are attained nowadays in part thanks to the very positive impact of deep learning.
  - Still, the task presents some open challenges and problems.
- In general, speech processing is becoming mature enough to foresee novel areas of application.

Outline

- Introduction to speech processing

- Speech Pattern classification
  - Introduction to SPC
  - Feature Extraction
    - Type of features
    - MFCCs
  - Machine learning
    - Speech common models
    - GMM
  - The “complex” task example: ASR
    - HMM
    - Examples

- ASR & Speech Pattern classification task examples

- **Tools and references**

Tools for Feature Extraction: HTK

HTK http://htk.eng.cam.ac.uk

- HMM toolkit primarily used for ASR
  - I has been one of the most important publicly available ASR toolkits for many years
  - Provides source code written in C (Linux/Windows)
    - It does not allow re-distribution
  - Well-documented
- Contains several tools, including **HCopy** , the tool that allows for feature extraction
  - **HCopy** permits computation of the most relevant classical ASR features and typical pre-/post- processing:
    - LPC, FBE, MFCC, PLP
    - Energy, Delta, double-delta, CMVN, VTLN
  - It can read several audio input formats

Tools for Feature Extraction: openSMILE

openSMILE - Open-Source Audio Feature Extractor

SMILE - Speech & Music Interpretation by Large-space Extraction

http://audeering.com/research/opensmile/

- It is a extremely popular and versatile feature extraction tool in the area of paralinguistics:
  - Baseline in ComParE evaluations
- Open-source multi-platform (written in C++)
  - It permits stand-alone tool usage or library access
- Well-documented http://www.audeering.com/research-and-open-source/files/openSMILE-book-latest.pdf
- Popular I/O file formats are supported:
  - HTK, Comma separated value (CSV) text, WEKA, LibSVM

Other public toolboxes for FE

- PRAAT
  - http://www.fon.hum.uva.nl/praat/
  - Phonetics & linguistic oriented
- MIR toolbox
  https://www.jyu.fi/hum/laitokset/musiikki/en/research/coe/materials/mirtoolbox
  - MATLAB code
  - Music oriented, but it also contains speech features
- YAAFE – Yet another audio feature extraction
  http://yaafe.sourceforge.net/
  - Python and MATLAB bindings
  - Collection of audio features

Tools for (speech) data modeling

GMM

- SPEAR: A Speaker Recognition Toolkit based on Bob (Python) https://pythonhosted.org/bob.bio.spear/
- MATLAB - Statistics and Machine Learning Toolbox http://www.mathworks.com/help/stats/fitgmdist.html

SVM

- LIBSVM -- A Library for Support Vector Machines https://www.csie.ntu.edu.tw/~cjlin/libsvm/

NEURAL NETWORKS

- Neural Network Toolbox http://www.mathworks.com/help/nnet/index.html
- QuickNet http://www1.icsi.berkeley.edu/Speech/qn.html

DNNs

- Theano, TensorFlow, CNTK, Keras, PyTorch

DATA MINING TOOLBOXES

- Weka 3: Data Mining Software in Java http://www.cs.waikato.ac.nz/ml/weka/
- SciKit learn (Python) http://scikit-learn.org/stable/

Tools for ASR development

HTK http://htk.eng.cam.ac.uk

- I has been one of the most important publicly available ASR toolkits for many years
- Provides source code written in C (Linux/Windows)
- Well-documented

KALDI http://kaldi-asr.org

- Provides current state of the art methods (DNNs)
- Many recipes ready to be used

Tools for LM training

- SRILM Toolkit: www.speech.sri.com/projects/srilm
- CMU-Cambridge Statistical LM toolkit: http://mi.eng.cam.ac.uk/~prc14/toolkit.html

References

- These are some presentations that were used for this lecture:

[1] Michael Mandel, “Lecture 3: Machine learning, classification, and generative models”
http://www.ee.columbia.edu/~dpwe/e6820/lectures/L03-ml.pdf

[2] Douglas A. Reynolds, “Overview of Automatic Speaker Recognition”
http://www.fit.vutbr.cz/study/courses/SRE/public/prednasky/2009-10/07_spkid_doug/sid_tutorial.pdf

[3] Lawrence Rabiner, “Digital Speech Processing— Lecture 1 Introduction to Digital Speech Processing"
http://www.ece.ucsb.edu/Faculty/Rabiner/ece259/digital%20speech%20processing%20course/lectures_new/Lecture%201_winter_2012_robot_video.pdf

[4] Steve Renals, “AUTOMATIC SPEECH RECOGNITION (ASR) 2018-19: LECTURES“,
http://www.inf.ed.ac.uk/teaching/courses/asr/lectures-2019.html

- These are some recommended tutorial-like reading in the topic of ASR:

  – **G&Y**: MJF Gales and SJ Young (2007). The Application of Hidden Markov Models in Speech Recognition, *Foundations and Trends in Signal Processing*, **1** (3), 195-304.
  – G Hinton et al (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups, IEEE Signal Processing Magazine, **29**(6):82-97.