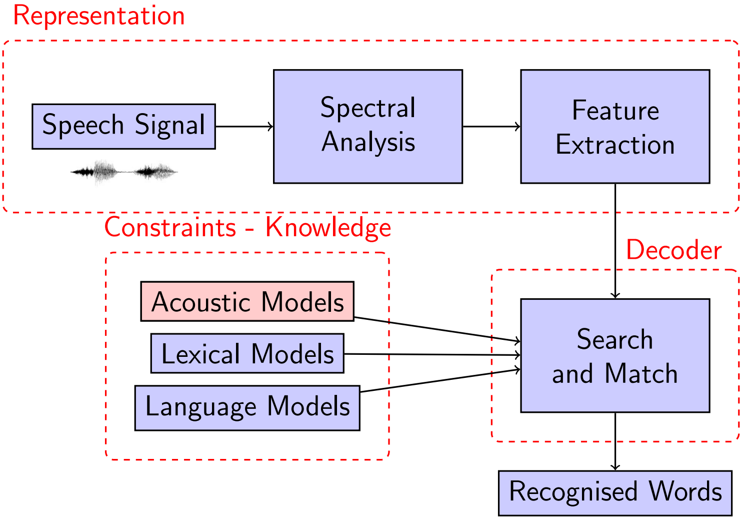
DT2118
Speech and Speaker Recognition
Lecture 05: Acoustic and Lexical Modelling

Giampiero Salvi

KTH/CSC/TMH giampi@kth.se

VT2016

Components of ASR System



Outline

Acoustic Models

- Limitations
- Practical Issues

Lexical Models

Evaluation

A probabilistic perspective: Bayes' rule

$$P(\text{words}|\text{sounds}) = \frac{P(\text{sounds}|\text{words})P(\text{words})}{P(\text{sounds})}$$

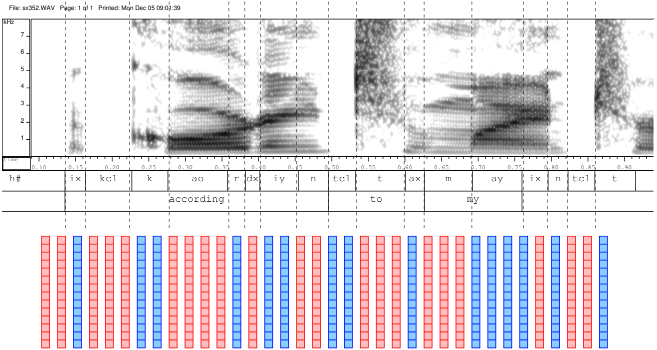
- $P(\text{sounds}|\text{words})$ can be estimated from training data and transcriptions
- $P(\text{words})$: *a priori* probability of the words (Language Model)
- $P(\text{sounds})$: *a priori* probability of the sounds (constant, can be ignored)

Probabilistic Modelling

Problem: How do we model $P(\text{sounds}|\text{words})$?

Probabilistic Modelling

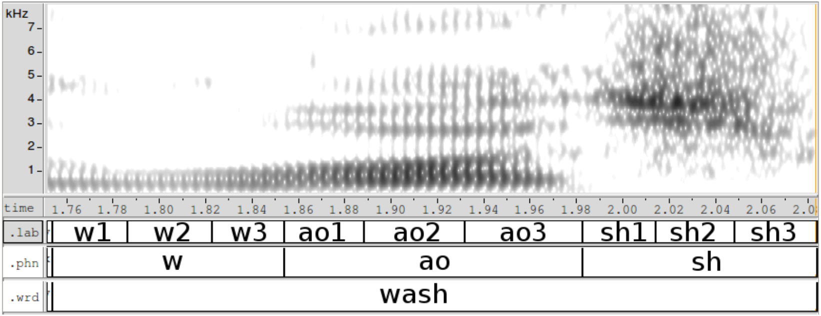
Problem: How do we model $P(\text{sounds}|\text{words})$?



Every feature vector (observation at time t) is a continuous stochastic variable (e.g. MFCC)

Stationarity

- we need to model short segments independently
- the fundamental unit can not be the word, but must be shorter
- usually we model three segments for each phoneme



Local probabilities (frame-wise)

If segment sufficiently short

$$P(\text{sounds}|\text{segment})$$

can be modelled with standard probability distributions

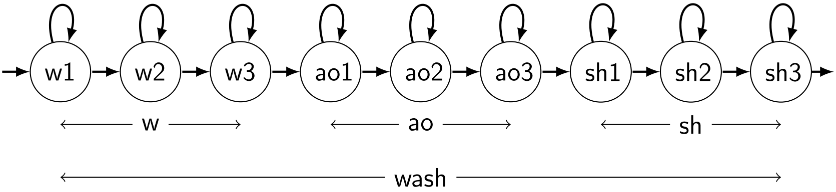
$$\phi(o, s_a) = P(o|s_a)$$

Usually Gaussian or Gaussian Mixture but also discrete distributions

Global Probabilities (utterance)

Problem: How do we combine the different $P(\text{sounds}|\text{segment})$ to form $P(\text{sounds}|\text{words})$?

Answer: Hidden Markov Model (HMM)

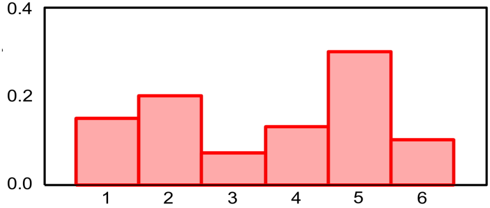


State to output probability model

- Discrete HMMs (DHMMs)
  - vector quantisation
- Continuous HMMs
  - Single Gaussian $\phi_j(x_n) = N(x_n|\mu_j, \Sigma_j)$
  - Gaussian Mixture
- Semi-continuous HMMs (SCHMMs)

Discrete HMMs

- quantise feature vectors
- observation: sequence of discrete symbols
- $\phi_j(x_n)$ simple discrete probability distribution
- problem: quantisation error



Discrete HMMs: learn $\phi_j(x_n)$

Remember that

$$\gamma_n(i,j) = P(z_{n-1} = s_i, z_n = s_j | X, \theta)$$

then

$$\xi_n(j) = P(z_n = s_j | X, \theta) = \sum_{i=1}^{M} \gamma_n(i,j)$$

Update rule:

$$\phi_j(x_n = k) = \frac{E[x_n = k, z_n = s_j]}{E[z_n = s_j]} = \frac{\sum_{n:(x_n=k)} \xi_n(j)}{\sum_{n=1}^{N} \xi_n(j)}$$

HMMs with Gaussian Emission Probability

$$\phi_j(x_n) = N(x_n|\mu_j, \Sigma_j)$$

Update rules:

$$\mu_j = \frac{\sum_{n=1}^{N} \xi_n(j) x_n}{\sum_{n=1}^{N} \xi_n(j)}$$

$$\Sigma_j = \frac{\sum_{n=1}^{N} \xi_n(j) \left(x_n - \mu_j\right) \left(x_n - \mu_j\right)^T}{\sum_{n=1}^{N} \xi_n(j)}$$

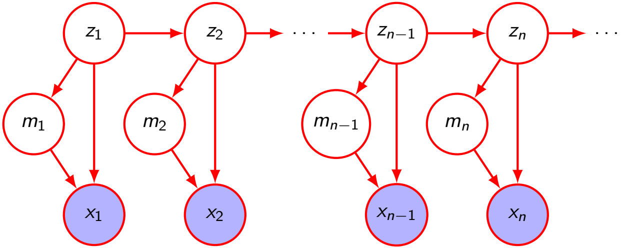
HMMs with Mixture Emission Probability

Often the Emission probability is modelled as a Mixture of Gaussians

$$\phi_j(x_n) = \sum_{k=1}^{K} w_{jk} N(x_n | \mu_{jk}, \Sigma_{jk})$$

$$\sum_{k=1}^{M} w_{jk} = 1$$

HMMs with Mixture Emission Probability



Emission:

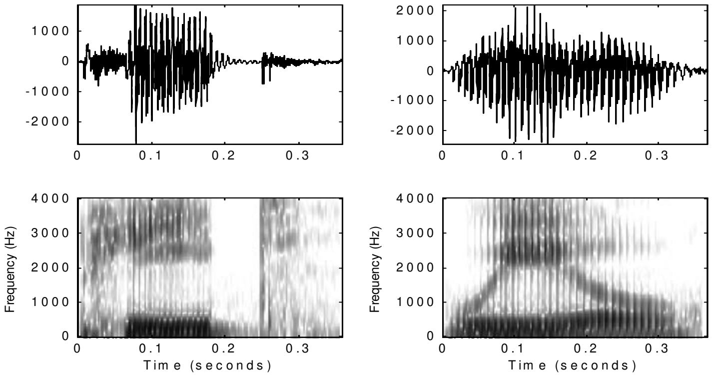
$$
\begin{aligned}
p(x_n|z_n, m_n) &= \mathcal{N}(x_n; \mu_{z_n,m_n}, \Sigma_{z_n,m_n}) \\
p(m_n|z_n) &= W(m_n, z_n)
\end{aligned}
$$

Semi-Continuous HMMs

- All Gaussian distributions in a pool of pdfs
- each $\phi_j(x_n)$ is a discrete probability distribution over the pool of Gaussians
- similar to quantisation, but probabilistic
- used for sharing parameters

Modelling Coarticulation

Example peat /piːt/ vs wheel /wiːl/



Modelling Coarticulation

Context dependent models (CD-HMMs)

- Duplicate each phoneme model depending on left and right context:
- from "a" monophone model
- to "d−a+f", "d−a+g", "l−a+s"... triphone models
- If there are $N = 50$ phonemes in the language, there are $N^3 = 125000$ potential triphones
- many of them are not exploited by the language

Amount of parameters

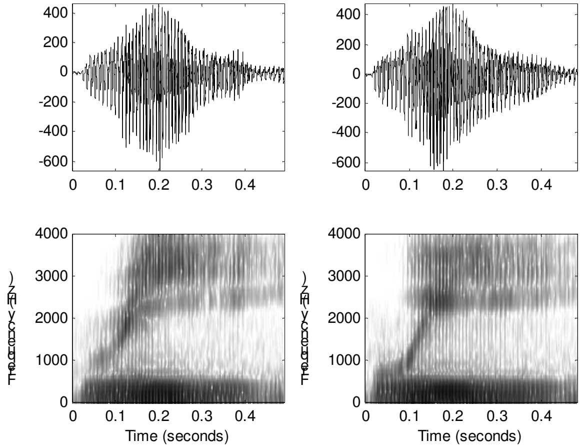
Example:

- a large vocabulary recogniser may have 60000 triphone models
- each model has 3 states
- each state may have 32 mixture components with $1 + 39 \times 2$ parameters each (weight, means, variances): $39 \times 32 \times 2 + 32 = 2528$

Totally it is $60000 \times 3 \times 2528 = 455$ million parameters!

Similar Coarticulation

/riː/ vs /wiː/



Tying to reduce complexity

Example: similar triphones d−a+m and t−a+m

- same right context, similar left context
- 3rd state is expected to be very similar
- 2nd state may also be similar

States (and their parameters) can be shared between models

+ reduce complexity
+ more data to estimate each parameter
− fine detail may be lost

Tying to reduce complexity

Example: similar triphones d−a+m and t−a+m

- same right context, similar left context
- 3rd state is expected to be very similar
- 2nd state may also be similar

States (and their parameters) can be shared between models

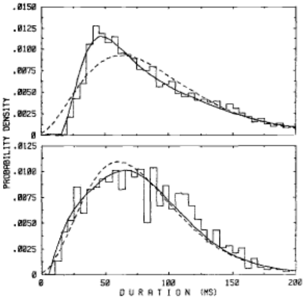
\+ reduce complexity

\+ more data to estimate each parameter

− fine detail may be lost

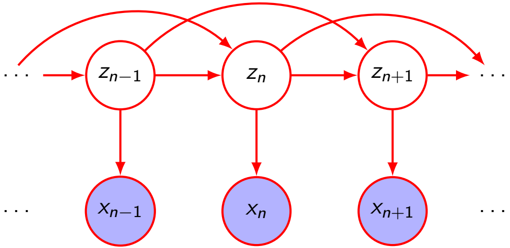
done with CART tree methodology

HMM Limitations: Duration modelling

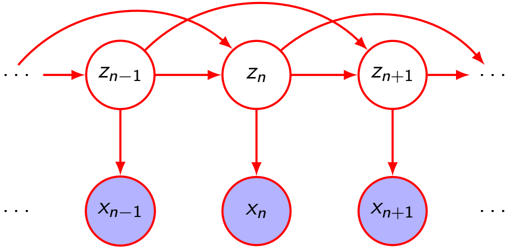


- $P(d_i = n) = a_{ii}^n(1 - a_{ii})$
- Several solutions proposed, but modest improvements

HMM Limitations: First Order Assumption

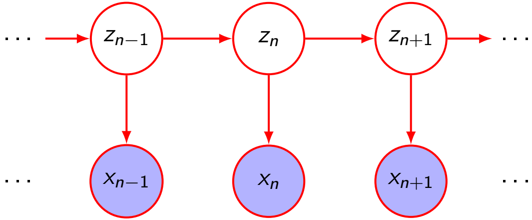


HMM Limitations: First Order Assumption

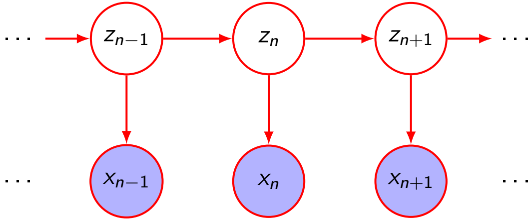


but: increasing order gives modest improvements

HMM Limitations: Conditional Independence Assumption



HMM Limitations: Conditional Independence Assumption



use dynamic features!

Dynamic Features

Concatenate static MFCCs (or LPCs) to Δ and $\Delta\Delta$ vectors.

Δ_n computed as weighted sum of $d_k(n)$

$$\Delta_n = \frac{\sum_{k=1}^{K} w_k d_k(n)}{\sum_{k=1}^{K} w_k}$$

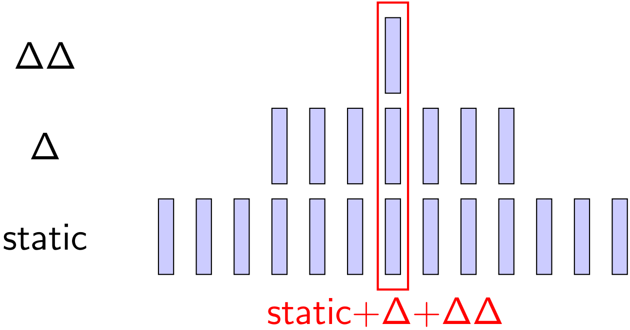
$d_k(n)$: finite differences centered around n with interval $2k$:

$$d_k(n) = \frac{c_{n+k} - c_{n-k}}{2k}$$

Similarly for $\Delta\Delta_n$

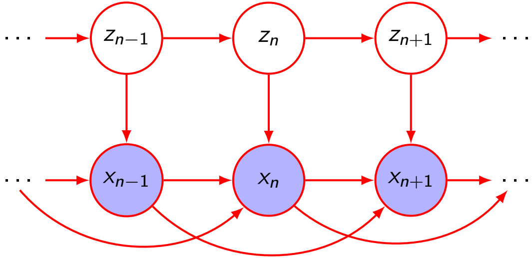
Dynamic Features: Common values

- In HTK $w_k = 2k^2$
- Usually k goes from 1 to 3
- to compute static+Δ+$\Delta\Delta$ we need 13 consecutive static vectors (around 130 msec).



HMM Limitations: Conditional Independence Assumption

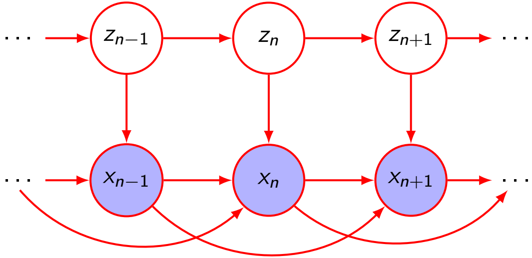
Autoregressive HMM [1]



[1] M. Shannon and W. Byrne. "Autoregressive HMMs for speech synthesis". In: *Proc. Interspeech*. Brighton, U.K., 2009

HMM Limitations: Conditional Independence Assumption

Autoregressive HMM [1]



Also interesting results with Time Delay Neural Networks (TDNN)

[1] M. Shannon and W. Byrne. "Autoregressive HMMs for speech synthesis". In: *Proc. Interspeech*. Brighton, U.K., 2009

HMMs: Practical Issues

- Initialisation
- Training Criteria
- Probability Representations

Initialisation

Important in order to reach a high local maximum

- Discrete HMM
  - Initial zero probability remains zero
  - Uniform distribution works reasonably well
- Continuous HMM methods
  - k-means clustering
  - Proceed from discrete HMM to semi-continuous to continuous
  - Start training single Gaussian models.
- Use previously segmented data or “flat start” (equal distribution for all states in the training data)

Training Criteria

- Maximum Likelihood Estimation (MLE)
  - Sensitive to inaccurate Markov assumptions
  - Maximises model likelihood rather than discrimination between models
- Minimum Classification Error (MCE) and Maximum Mutual Information Estimation (MMIE) might work better
- Maximum A Posteriori (MAP) if we have prior knowledge
  - for adaptation and small training data

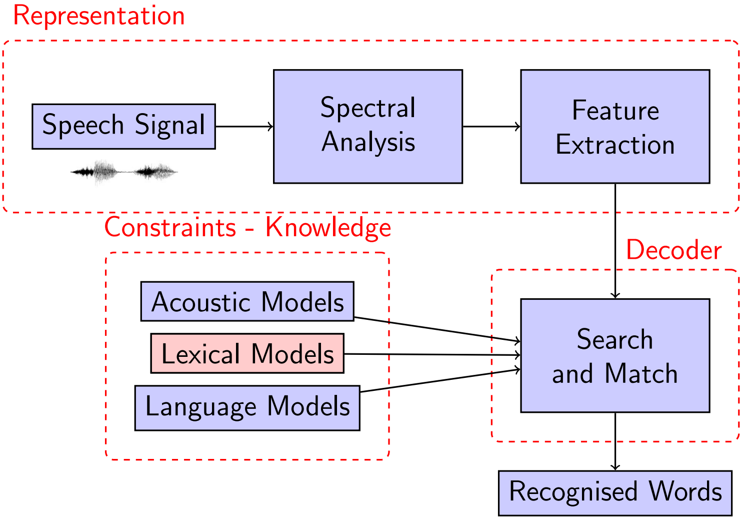
Probability Representations

Problem: the probabilities become very small (underflow problem)

- Viterbi decoding (only multiplication): use logarithm
- Forward-backward (multiplication and addition): difficult
- Solution 1: scale by $\left(\sum_{i=1}^{M} \alpha_n(i)\right)^{-1}$
- Solution 2: use logarithm and look-up table to speed up $\log(p_1 + p_2)$

Outline

Components of ASR System



Lexical Models

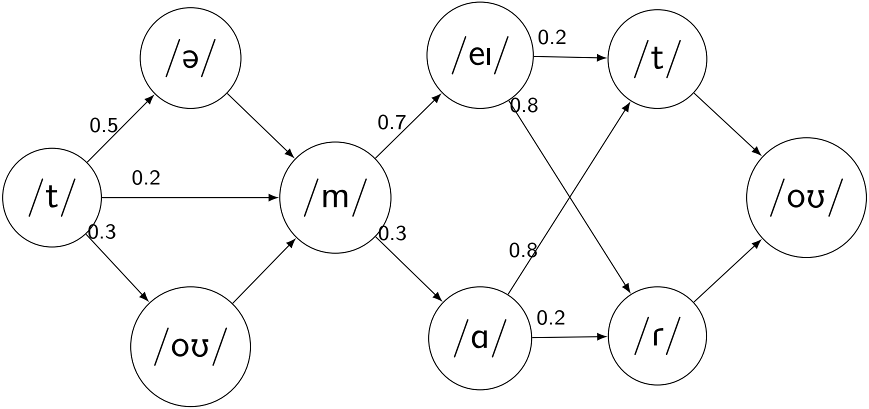
- in general specify sequence of phoneme for each word
- example:

"dictionary"	IPA	X-SAMPA
UK:	/d ɪ k ʃ ə n (ə) ɹ i/	/d I k S @ n (@) r i/
USA:	/d ɪ k ʃ ə n ɛ ɹ i/	/d I k S @ n E r i/

- expensive resources
- include multiple pronunciations
- phonological rules (assimilation, deletion)

Pronunciation Network

Example: tomato



Assimilation

did you	/d ɪ dʒ j ə/
set you	/s ɛ tʃ ɜ/
last year	/l æ s tʃ iː ɹ/
because you've	/b iː k ə ʒ uː v/

Deletion

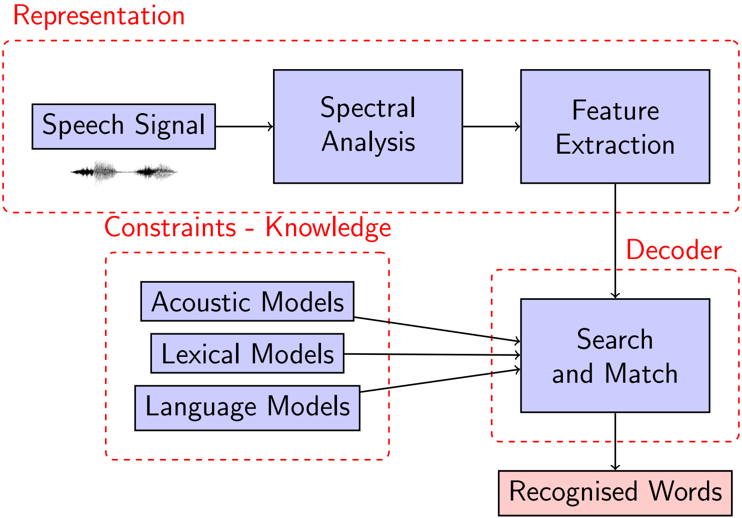
find him	/f a ɪ n ɪ m/
around this	/ə ɹ aʊ n ɪ s/
let me in	/l ɛ m iː n/

Out of Vocabulary Words

- Proper names often not in lexicon
- derive pronunciation automatically
- English has very complex grapheme-to-phoneme rules
- attempts to derive pronunciation from speech recordings

Outline

Components of ASR System



ASR Evaluation

- recognition results are sequences of words
- evaluation is non-trivial
- need to realign the recognised sequence to the transcription
- example:

ref:	I really wanted to see you
rec:	I wanted badly to meet you

- possible to use detailed time alignment
- usually only symbolic level is used
- dynamic programming

Word Accuracy and Word Error Rate (WER)

$$A = 100\frac{N - S - D - I}{N}$$

Where

- N: total number of reference words
- S: substitutions
- D: deletions
- I: insertions

$$\mathrm{WER} = 100 - A$$

Word Accuracy: example

Ref/Rec	I	wanted	badly	to	meet	you
I	corr					
really	del					
wanted		corr				
to			ins	corr		
see					sub	
you						corr

6 words, 1 substitution, 1 insertion, 1 deletion

$$A = 100\frac{6-1-1-1}{6} = 50\%$$

requires dynamic programming

Effects of Sampling Rate on WER

Sampling Rate (kHz)	Relative Error Reduction (%)
8	baseline
11	+10
16	+10
22	+0

(from Huang, Acero and Hon)

Effects of Feaures on WER

Feature Set	Relative Error Reduction (%)
13th order LPC cepstrum	baseline
13th order MFCC	$+10$
16th order MFCC	$+0$
with Δ and $\Delta\Delta$	$+20$
with $\Delta\Delta\Delta$	$+0$

(from Huang, Acero and Hon)

Effect of Modelling Context

Units	Relative Error Reduction (%)
Context-independent phone	baseline
Context-dependent phone	+25
Clustered triphone	+15
Senone	+24

(from Huang, Acero and Hon)