

# Speech

Many slides courtesy of  
Dan Klein, Stuart Russell,  
or Andrew Moore

**CS 5300 / CS 6300**  
**Artificial Intelligence**  
**Spring 2010**

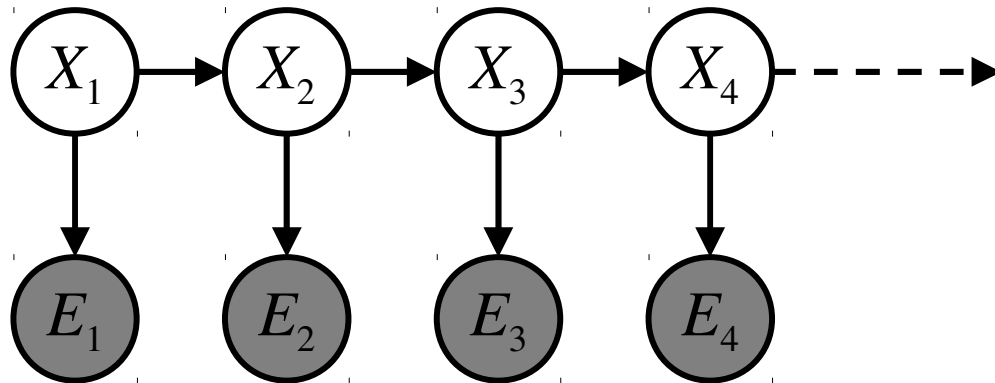
Hal Daumé III  
hal@cs.utah.edu

[www.cs.utah.edu/~hal/courses/2010S\\_AI](http://www.cs.utah.edu/~hal/courses/2010S_AI)

# Announcements

- Project 5
- Contest

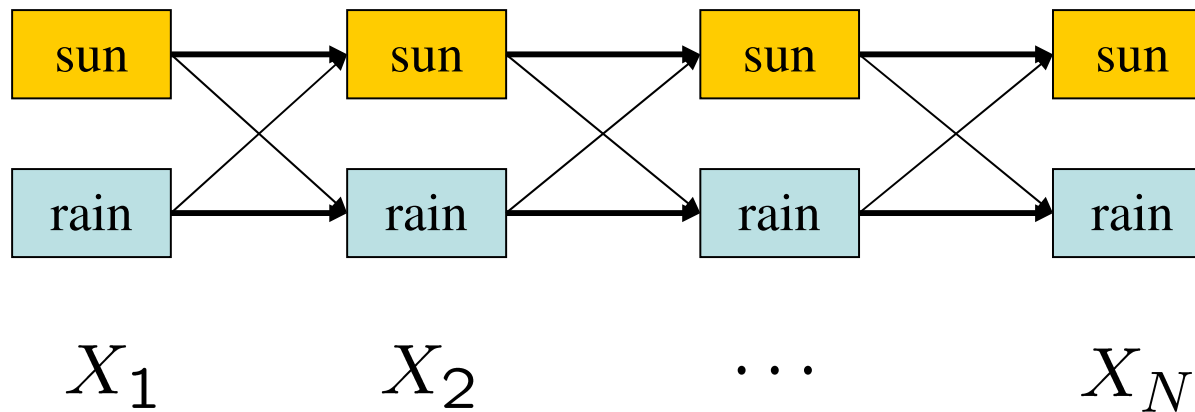
# Hidden Markov Models



- An HMM is
  - Initial distribution:  $P(X_1)$
  - Transitions:  $P(X|X_{-1})$
  - Emissions:  $P(E|X)$
  
- Query: most likely seq:  $\arg \max_{x_{1:t}} P(x_{1:t}|e_{1:t})$

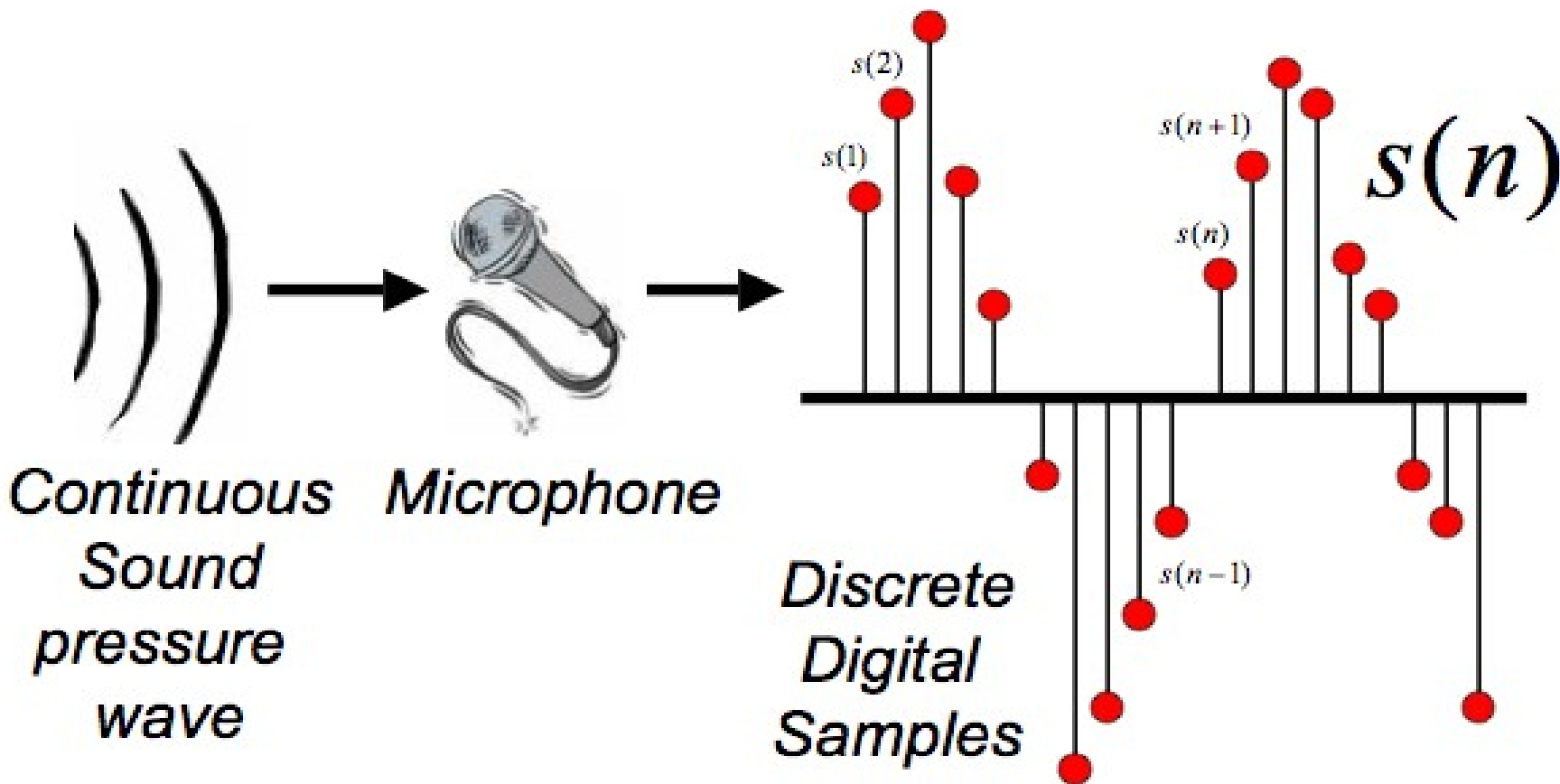
# State Path Trellis

- State trellis: graph of states and transitions over time



- Each arc represents some transition
- Each arc has weight
- Each path is a sequence of states
- The product of weights on a path is the seq's probability
- Can think of the Forward (and now Viterbi) algorithms as computing sums of all paths (best paths) in this graph

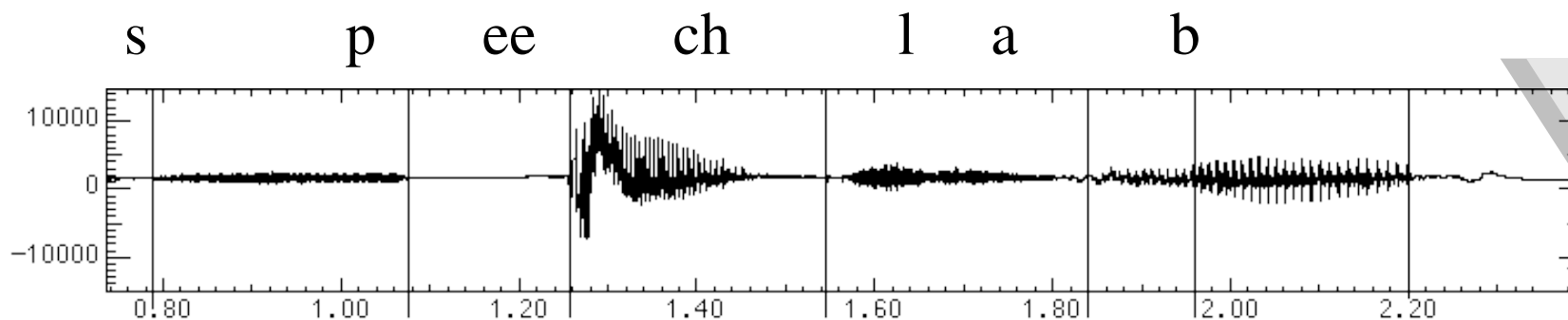
# Digitizing Speech



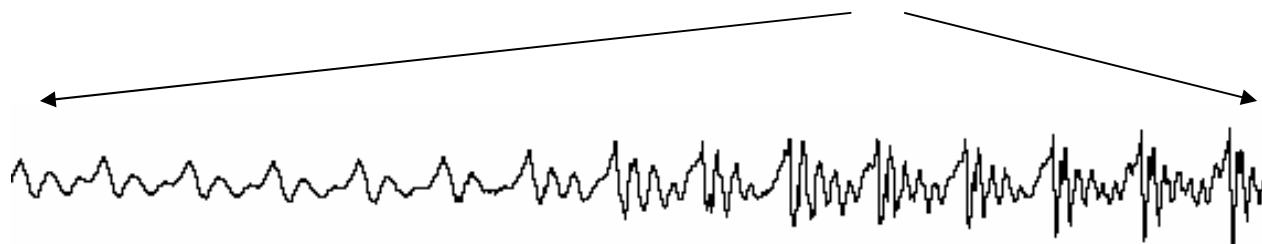
Thanks to Bryan Pellom for this slide!

# Speech in an Hour

- Speech input is an acoustic wave form



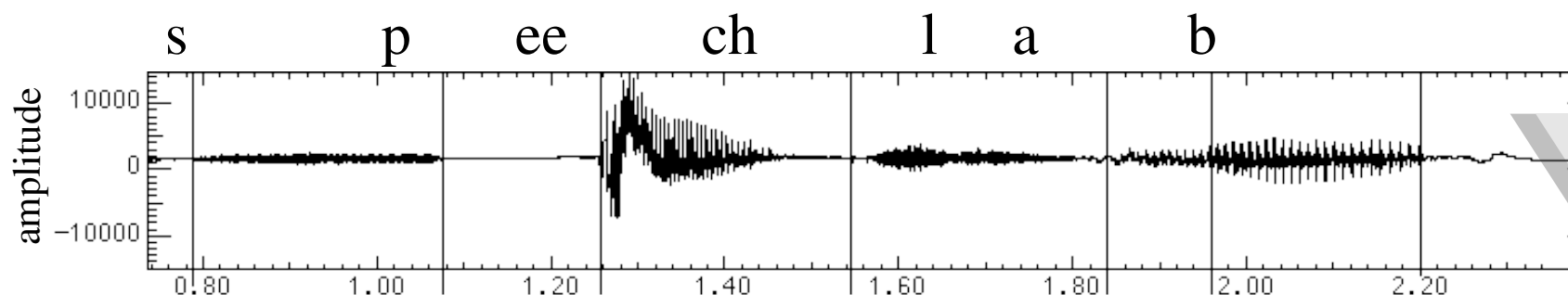
“l” to “a”  
transition:



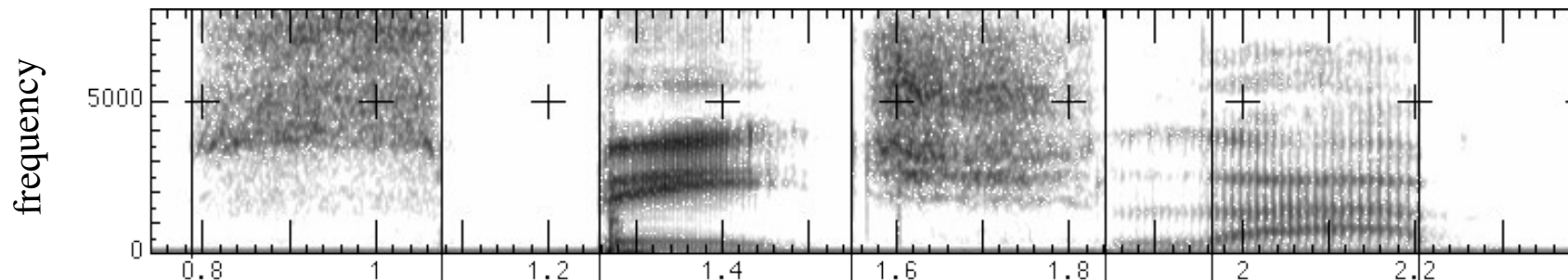
Graphs from Simon Arnfield’s web tutorial on speech, Sheffield:  
<http://www.psyc.leeds.ac.uk/research/cogn/speech/tutorial/>

# Spectral Analysis

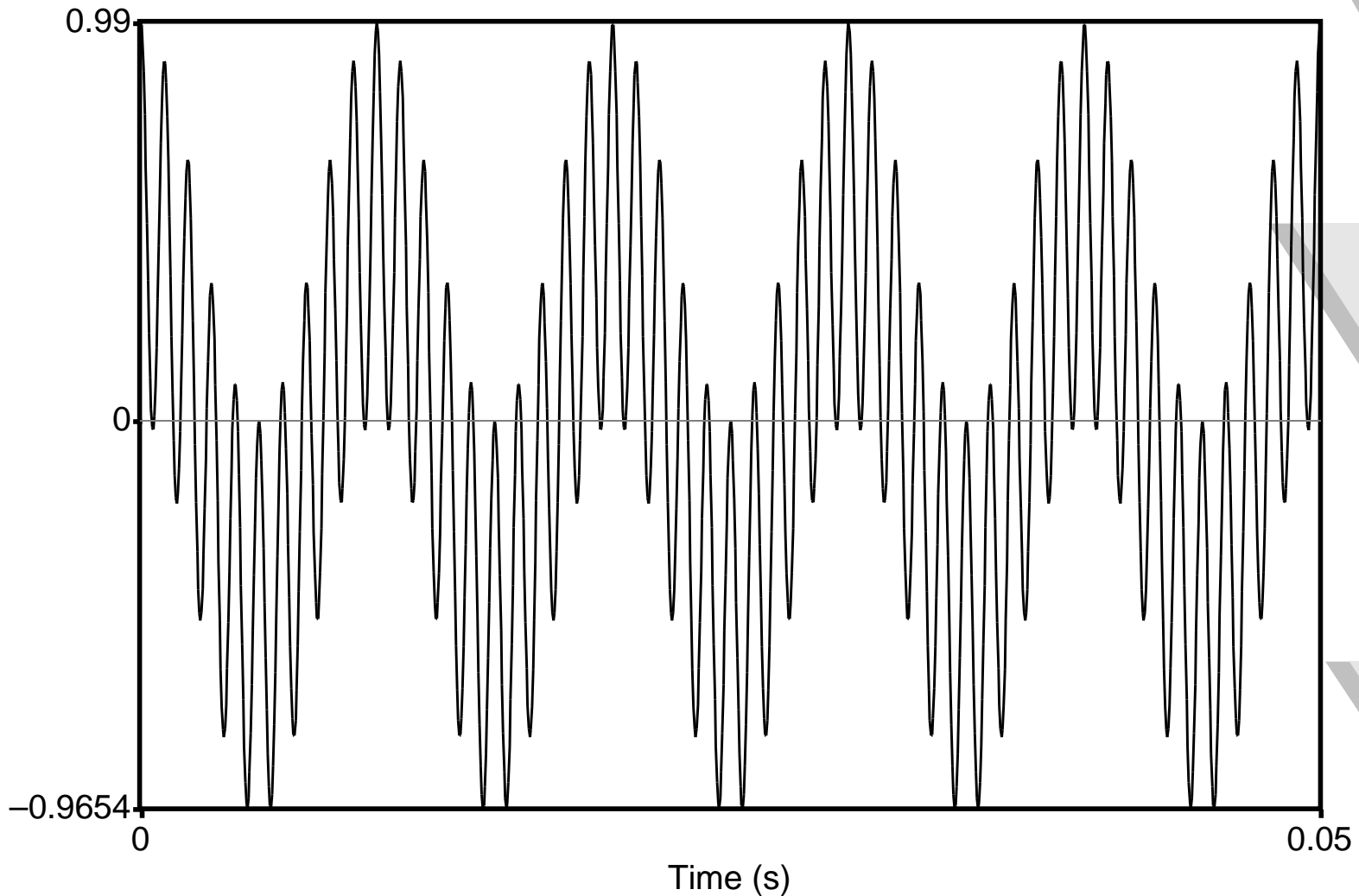
- Frequency gives pitch; amplitude gives volume
- sampling at ~8 kHz phone, ~16 kHz mic (kHz=1000 cycles/sec)



- Fourier transform of wave displayed as a spectrogram
- darkness indicates energy at each frequency

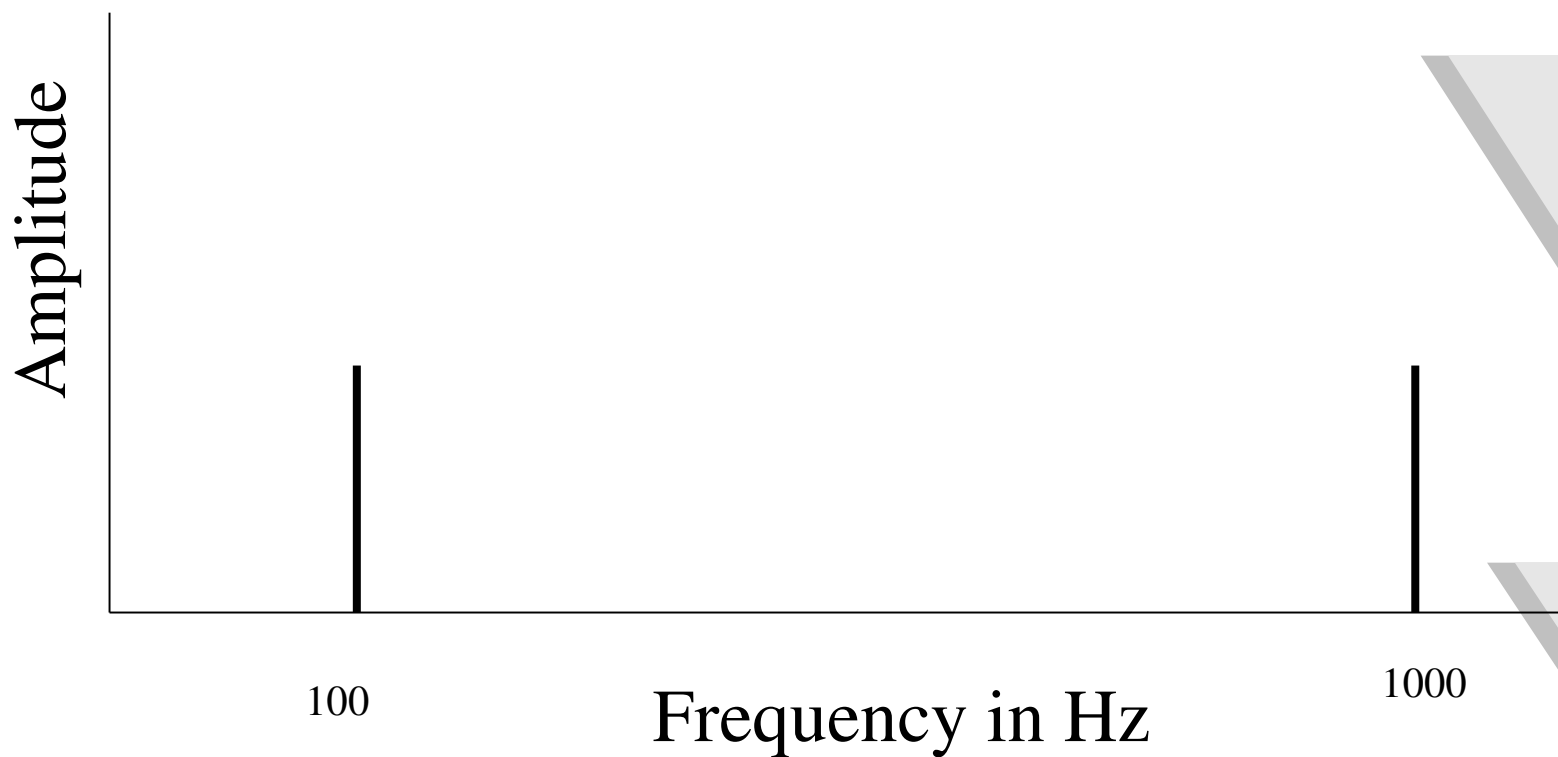


# Adding 100 Hz + 1000 Hz Waves

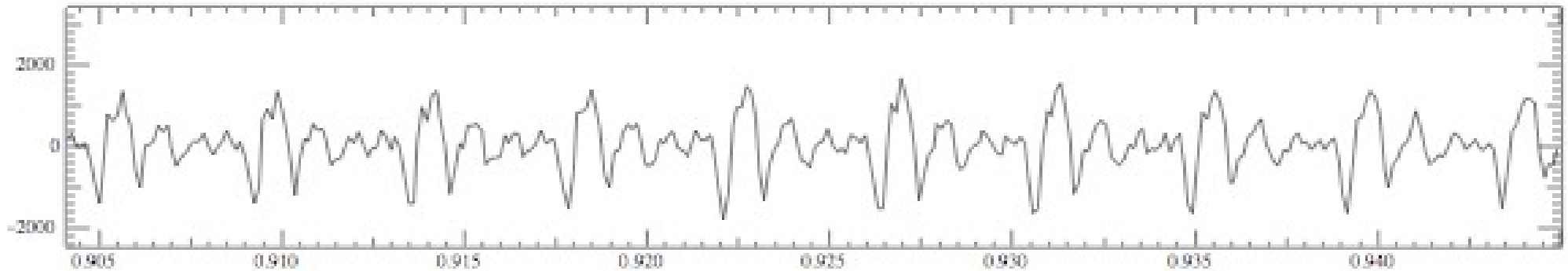


# Spectrum

Frequency components (100 and 1000 Hz) on x-axis



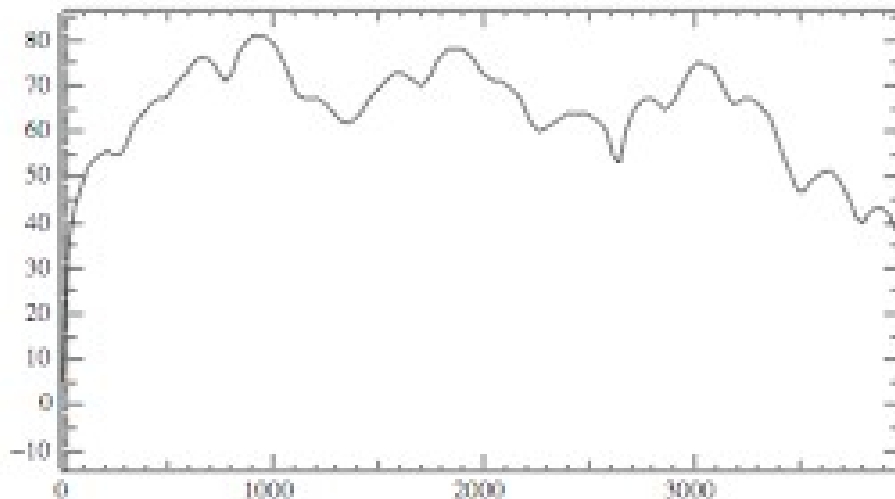
# Part of [ae] from “lab”



- Note complex wave repeating nine times in figure
- Plus smaller waves which repeats 4 times for every large pattern
- Large wave has frequency of 250 Hz (9 times in .036 seconds)
- Small wave roughly 4 times this, or roughly 1000 Hz
- Two little tiny waves on top of peak of 1000 Hz waves

# Back to Spectra

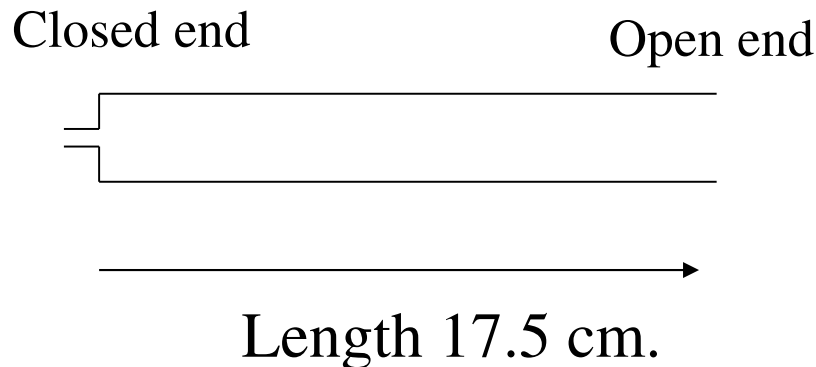
- Spectrum represents these freq components
- Computed by Fourier transform, algorithm which separates out each frequency component of wave.



- x-axis shows frequency, y-axis shows magnitude (in decibels, a log measure of amplitude)
- Peaks at 930 Hz, 1860 Hz, and 3020 Hz.

# Resonances of the vocal tract

- The human vocal tract as an open tube



- Air in a tube of a given length will tend to vibrate at resonance frequency of tube.
- Constraint: Pressure differential should be maximal at (closed) glottal end and minimal at (open) lip end.

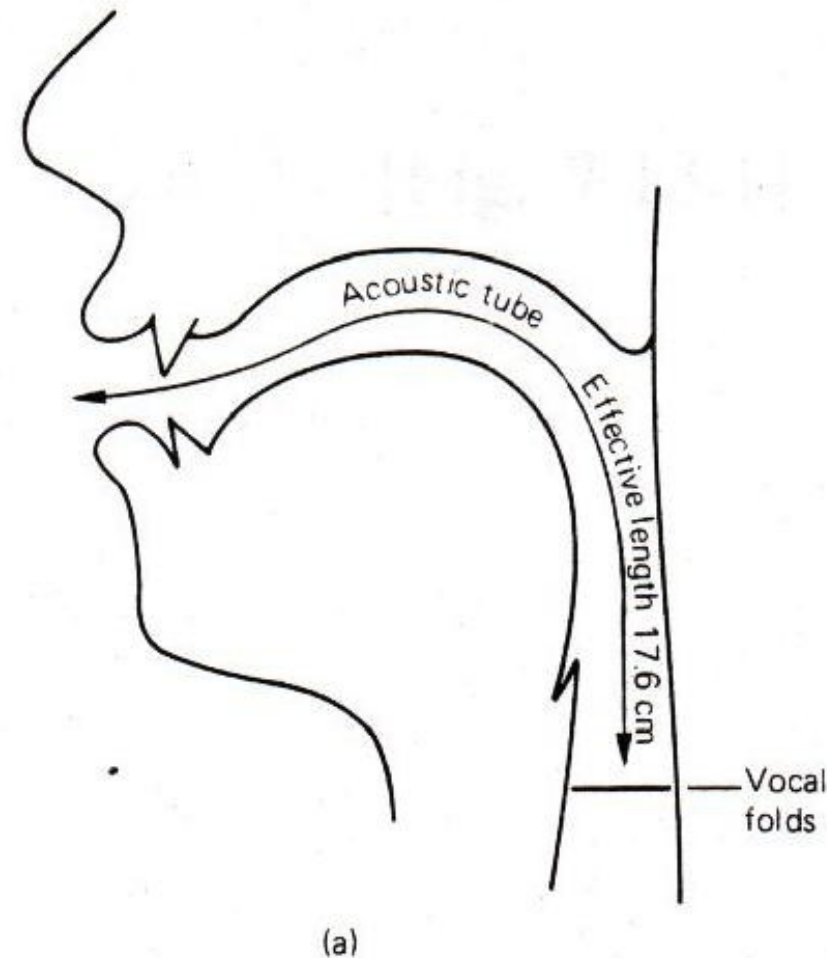
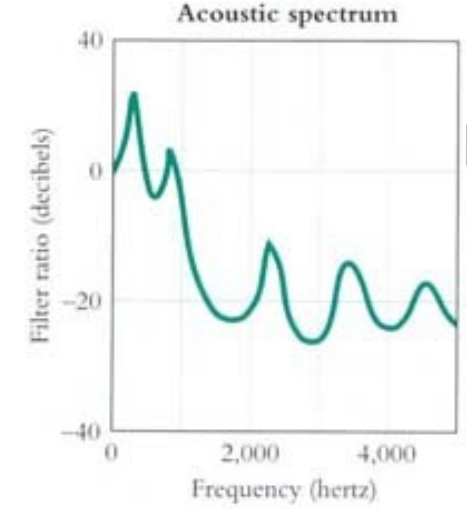
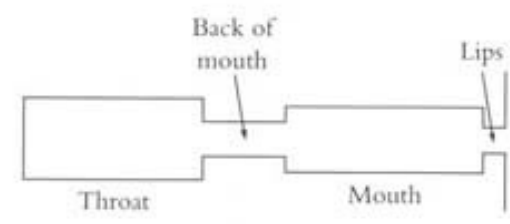
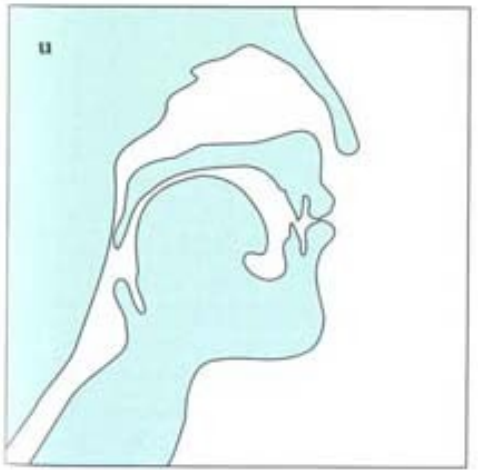
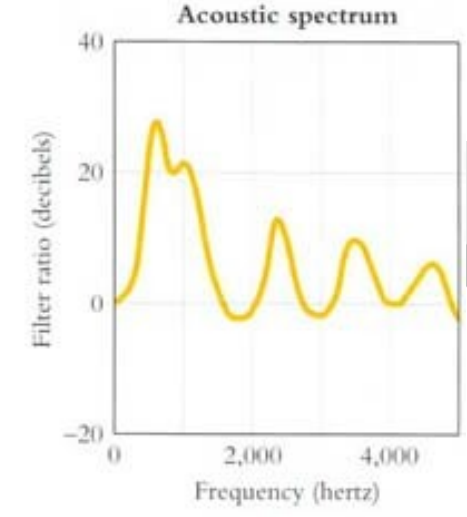
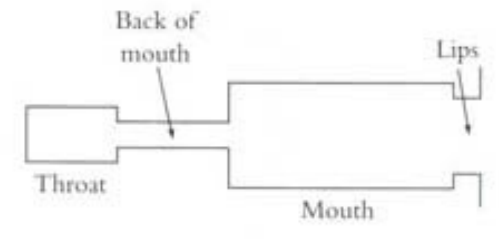
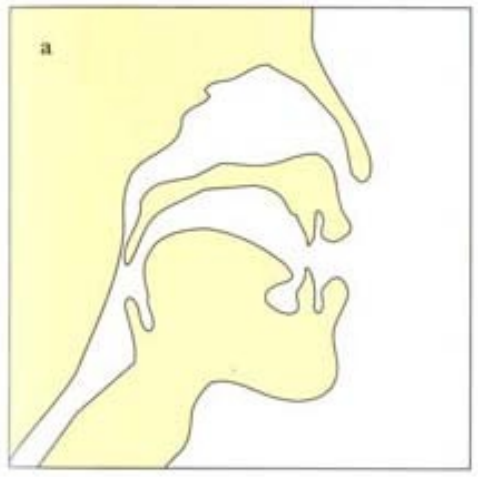
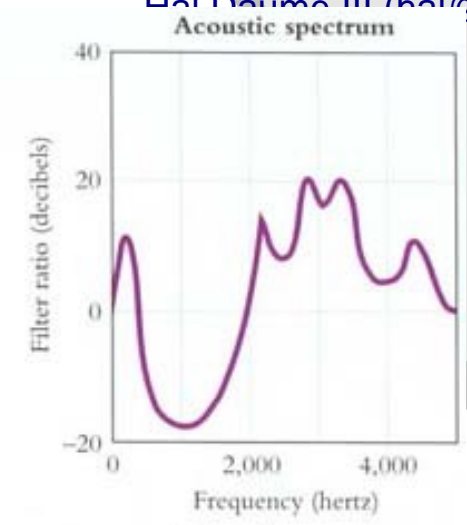
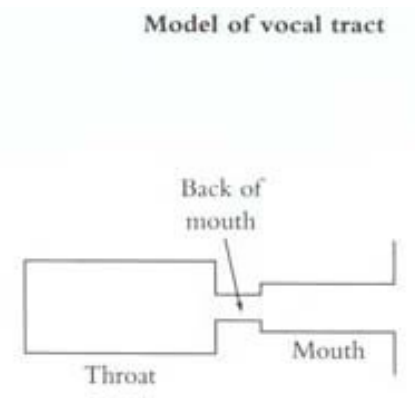
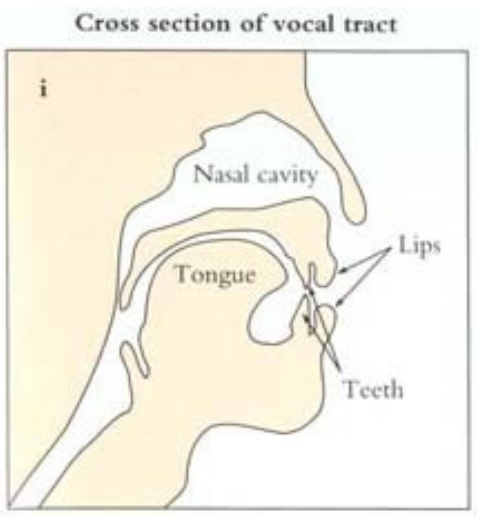
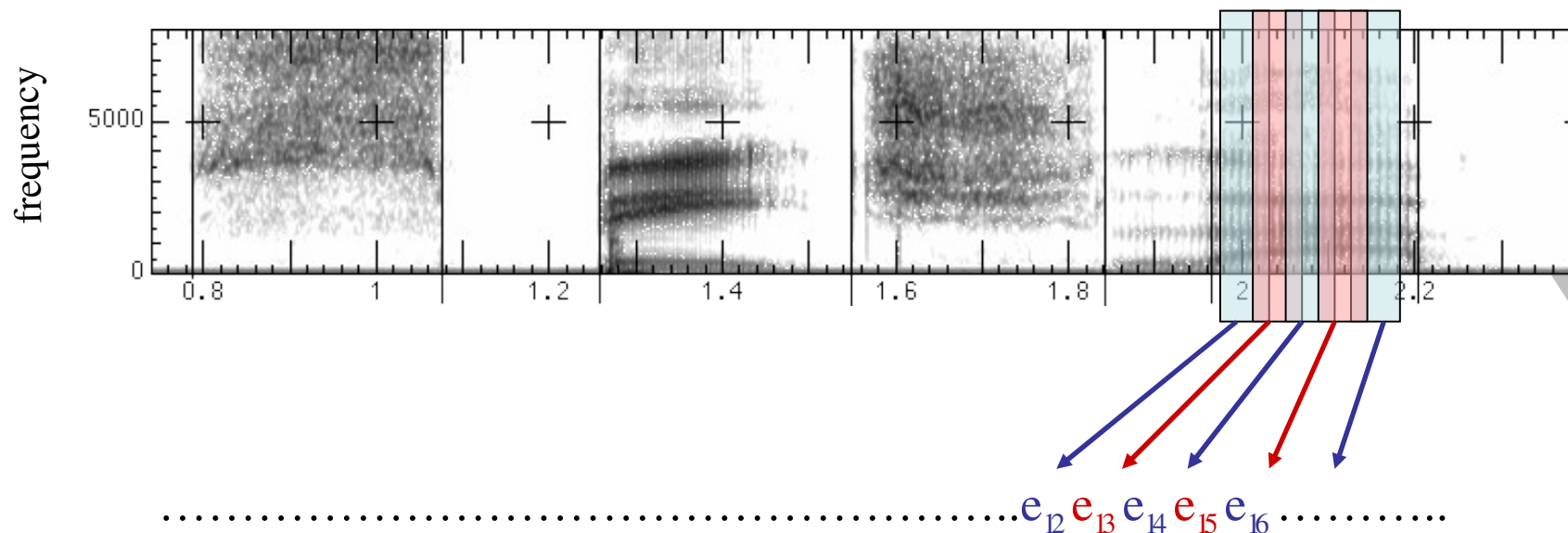


Figure from W. Barry  
Speech Science slides



# Acoustic Feature Sequence

- Time slices are translated into acoustic feature vectors (~39 real numbers per slice)

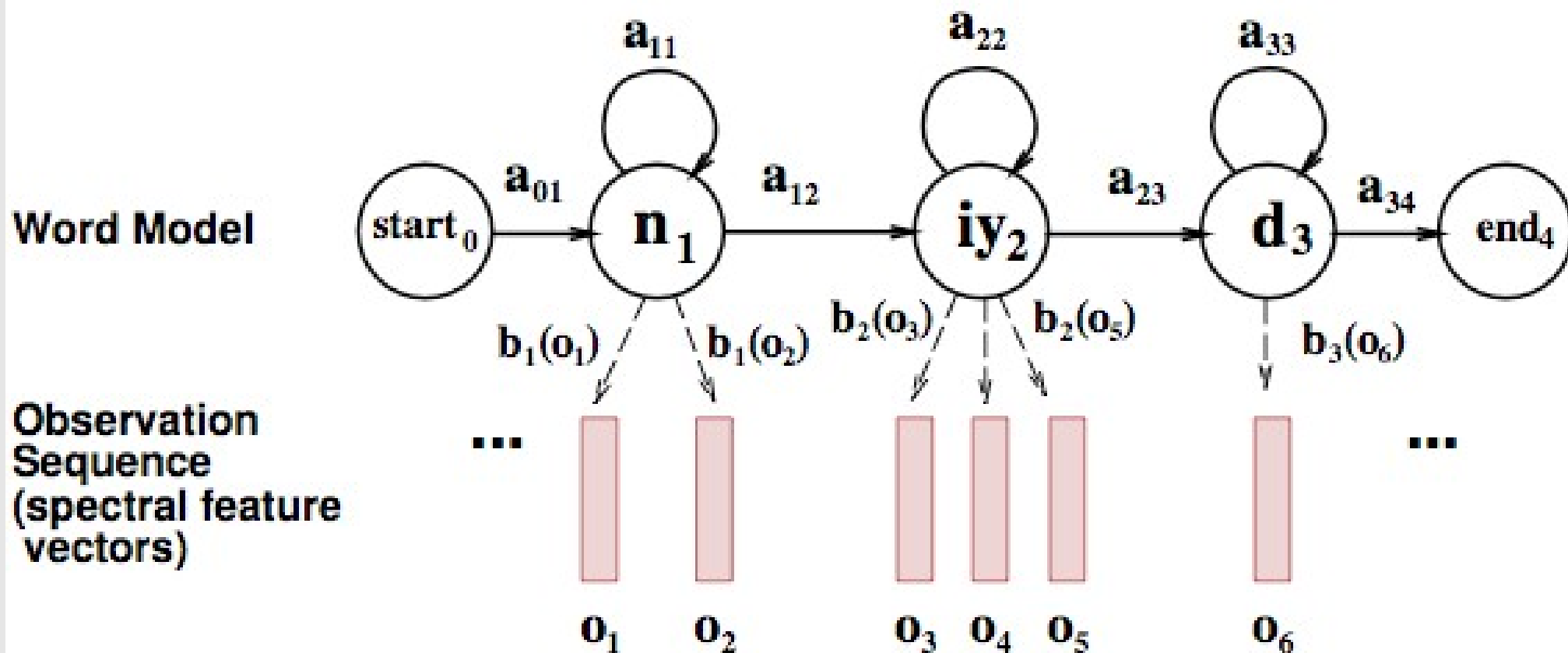


- These are the observations, now we need the hidden states  $X$

# State Space

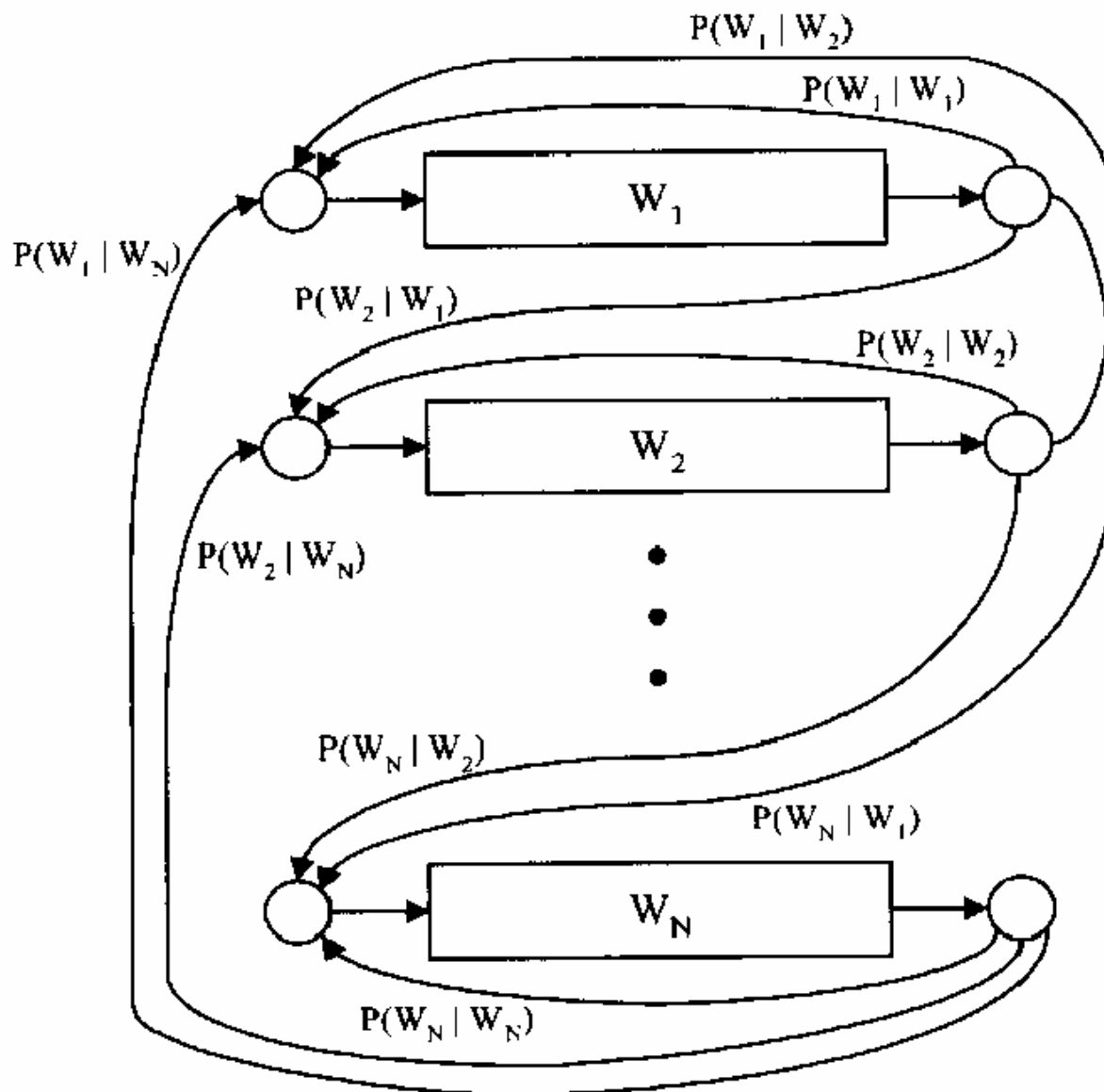
- $P(E|X)$  encodes which acoustic vectors are appropriate for each phoneme (each kind of sound)
- $P(X|X')$  encodes how sounds can be strung together
- We will have one state for each sound in each word
- From some state  $x$ , can only:
  - Stay in the same state (e.g. speaking slowly)
  - Move to the next position in the word
  - At the end of the word, move to the start of the next word
- We build a little state graph for each word and chain them together to form our state space  $X$

# HMMs for Speech



# Markov Process with Bigrams

Figure from Huang et al page 618



# Decoding

- While there are some practical issues, finding the words given the acoustics is an HMM inference problem
- We want to know which state sequence  $x_{1:T}$  is most likely given the evidence  $e_{1:T}$ :

$$\begin{aligned}x_{1:T}^* &= \arg \max_{x_{1:T}} P(x_{1:T} | e_{1:T}) \\ &= \arg \max_{x_{1:T}} P(x_{1:T}, e_{1:T})\end{aligned}$$

- From the sequence  $x$ , we can simply read off the words

# Training (aka “preview of ML”)

- Two key components of a speech HMM:
  - Acoustic model:  $p(E | X)$
  - Language model:  $p(X | X')$
  - Where do these come from?
  
- Can we estimate these models from data:
  - $p(E | X)$  might be estimated from transcribed speech
  - $p(X | X')$  might be estimated from large amounts of raw text

# n-gram Language Models

- Assign a probability to a sequences of words

$$\begin{aligned} p(w_1, w_2, \dots, w_I) &= \prod_{i=1}^I p(w_i | w_1, \dots, w_{i-1}) \\ &\approx \prod_{i=1}^I p(w_i | w_{i-k}, \dots, w_{i-1}) \end{aligned}$$

- If I gave you a copy of the web, how would you estimate these probabilities?

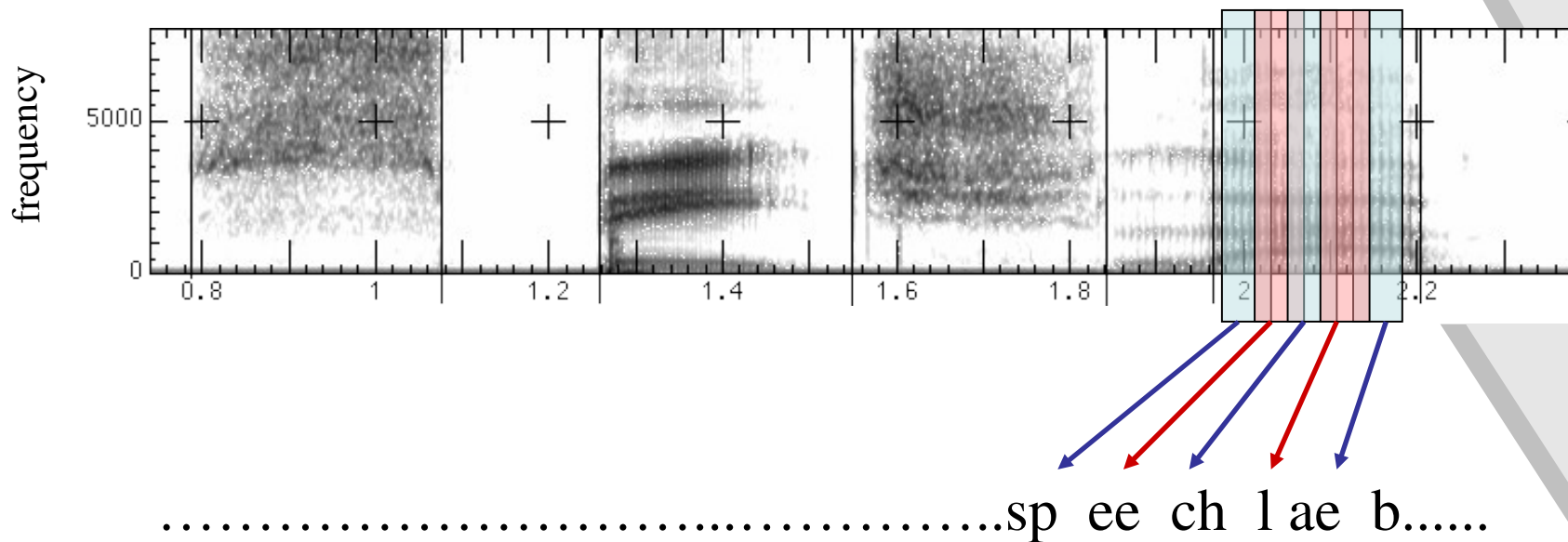
Need to “smooth” estimates intelligently to avoid zero probability  $n$ -grams.

Language modeling is the art of good smoothing.

See [Goodman 1998], [Teh 2007]

# Acoustic models

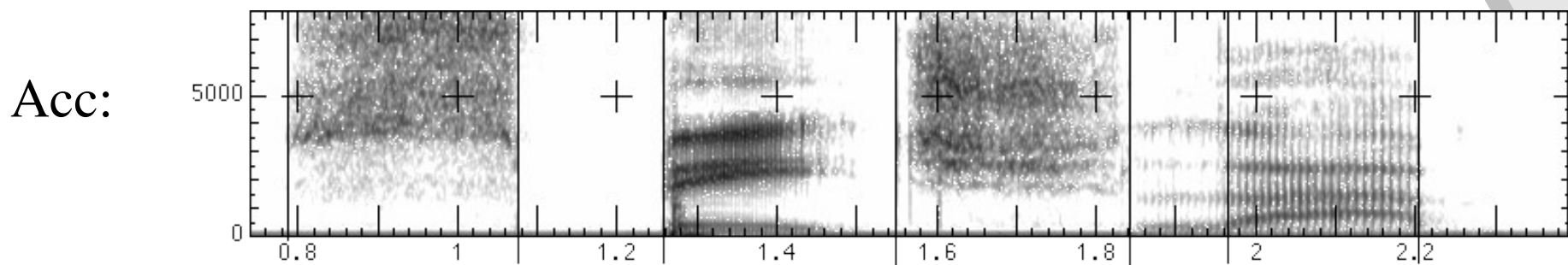
- What if I gave you data like:



- How would you estimate  $p(E|X)$ ?
- What's wrong with this approach?

# Acoustic models II

- What does our data really look like:

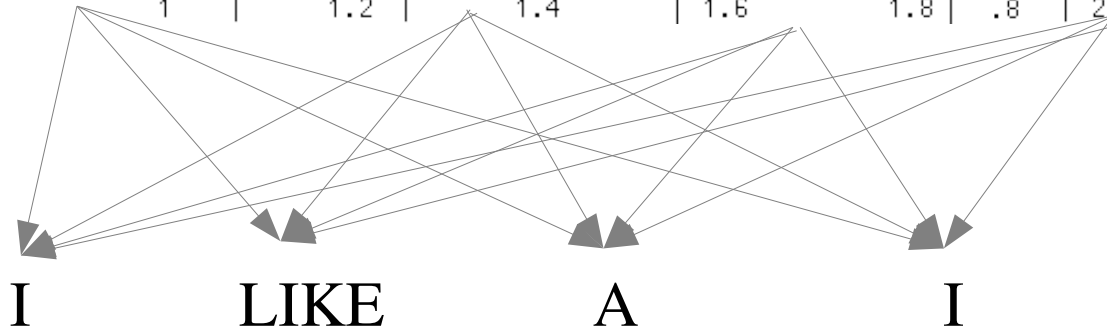
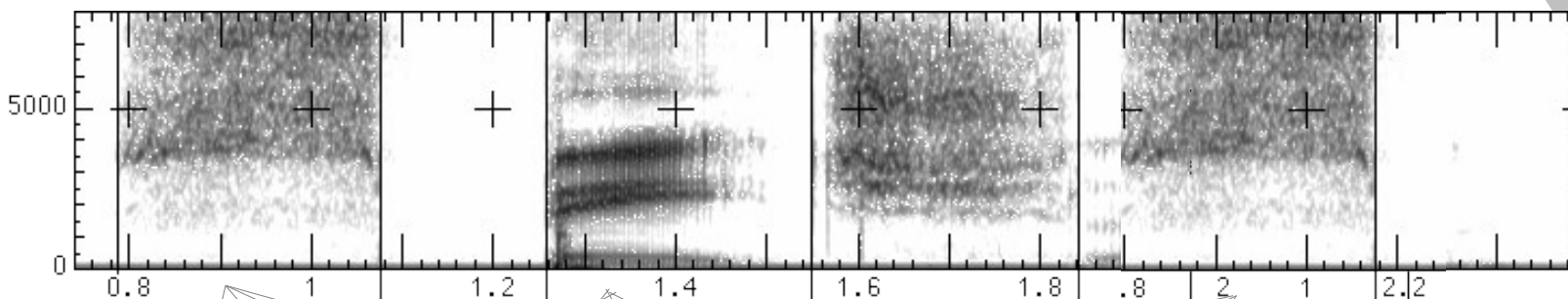


W: yesterday I went to visit the speech lab

- We'd like to know *alignments* between transcript and waveform
- Suppose someone gave us a good speech recognizer.... could we figure out alignments from that?

# Expectation Maximization

- A general framework to do parameter estimation in the presence of hidden variables
- Repeat ad infinitum:
  - E-step: make probabilistic guesses at latent variables
  - M-step: fit parameters according to these guesses

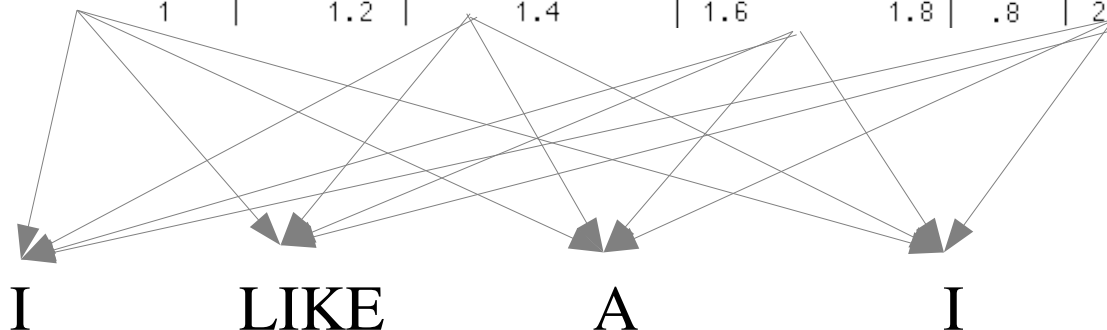
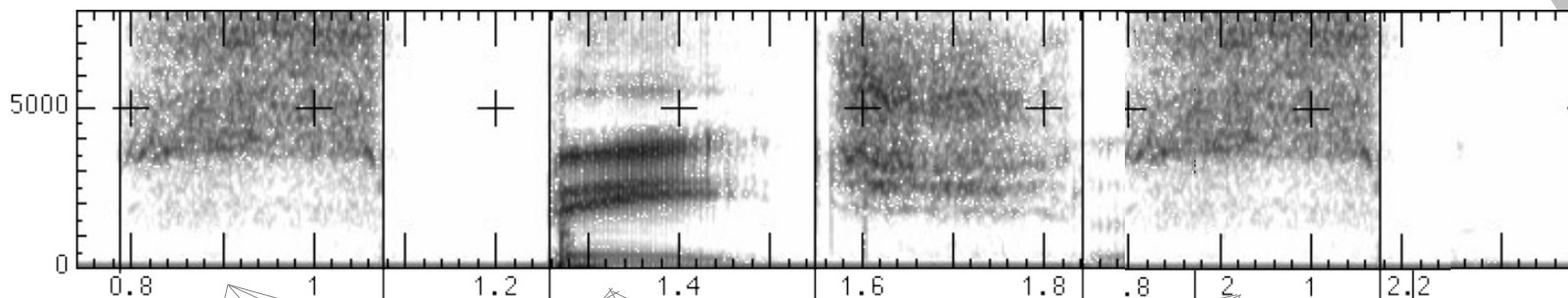


W:

I LIKE A LIKE I

# Expectation Maximization

e	$p(e   \text{"I"})$	$p(e   \text{"LIKE"})$	$p(e   \text{"A"})$
	0.33 → 2	0.33 → 1	0.33 → 1
	0.33 → 1	0.33 → 1	0.33 → 1
	0.33 → 1	0.33 → 1	0.33 → 1

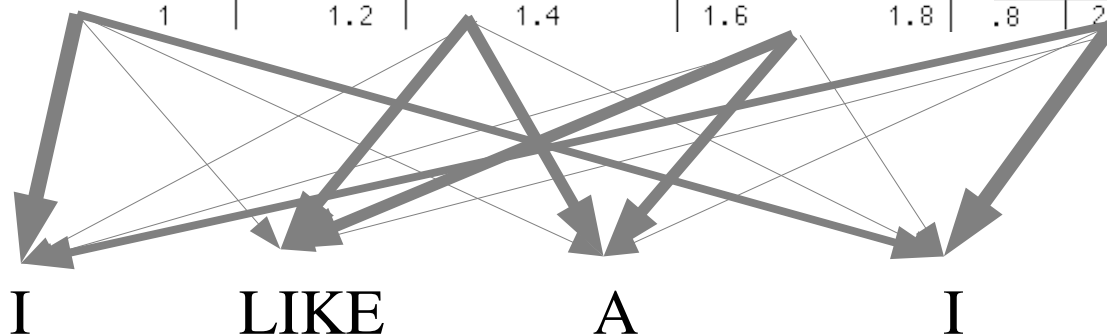
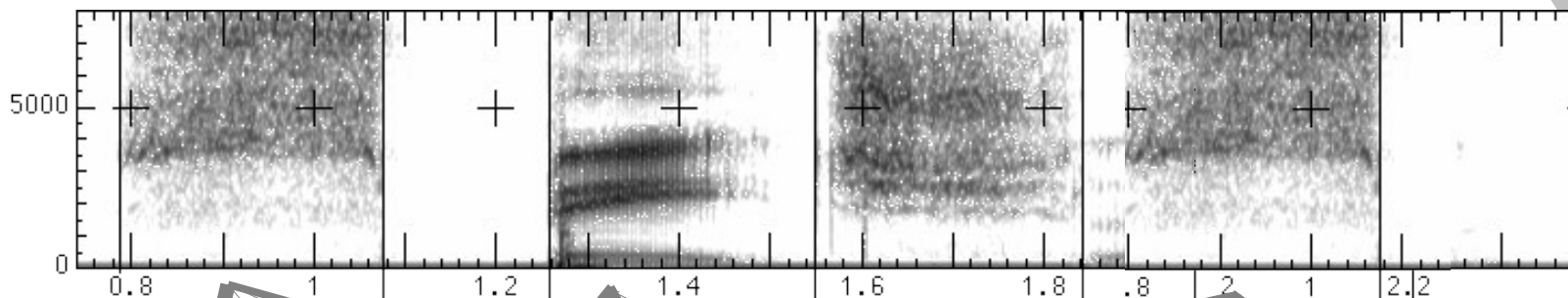


W:

I LIKE A I

# Expectation Maximization

e	$p(e   \text{"I"})$	$p(e   \text{"LIKE"})$	$p(e   \text{"A"})$
	0.5 → 4	0.33 → 1	0.33 → 1
	0.25 → 1	0.33 → 2	0.33 → 2
	0.25 → 1	0.33 → 2	0.33 → 2



W:

I LIKE A I



# Summary

- HMMs allow us to “separate” two models:
  - acoustic model (how does what I want to say sound?)
  - language model (what do I want to say)
- Speech recognition is “just” decoding in an HMM/DBN
  - Plus a heck of a lot of engineering
- Expectation maximization lets us estimate parameters in models with hidden variables
- Most research today focuses on language modeling