

Digitization of Speech

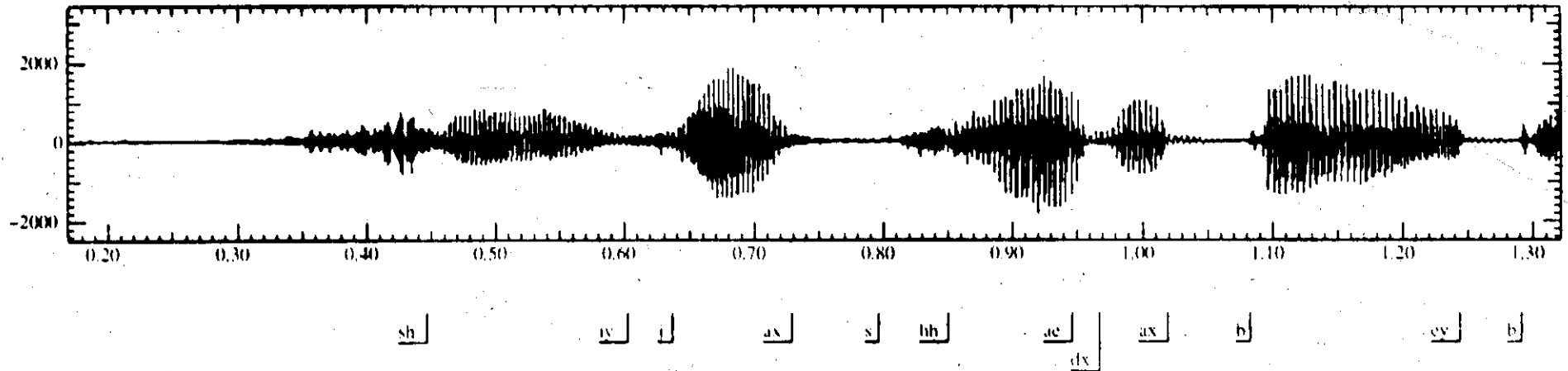
Gerald Penn

CSC 401
University of Toronto

<http://www.cs.toronto.edu/~gpenn/csc401>

The physical speech signal (1)

Jurafsky and Martin



She just had a baby (Switchboard Corpus). The x -axis is time; the y -axis is amplitude.

How to Digitize Speech

Speech is a longitudinal pressure wave (although we often represent it transversally).

Speech recognizers must first:

1. Sample this. Sampling rate is typically between 6 and 40kHz.
 - Often 16 kHz per channel
 - Telephone speech: 8 kHz
 - “CD-quality:”: 40.1 kHz
 - The human ear can distinguish pressure waves between 20 Hz and 20 kHz as sound, but *Nyquist's Theorem* says that the sampling frequency must be twice that of the maximum frequency that we wish to faithfully preserve.

How to Digitize Speech

2. Quantize the samples. Place bins at intervals along the y-axis, and indicate in which bin the pressure is measured at each sample time step.
 - This technique is called *pulse code modulation*
 - The number of bins determines the *sample size* — often 16 bits.
 - But long-term characteristics of speech do not yield a uniform distribution across y-bins unless we distort them — bigger bins near peaks of signal, smaller, better resolved bins near x-axis.

“Compressing”

Distortion of y-bins to improve fidelity of signal relative to a fixed signal size.

Two compressing methods are common in telephony: A-law (European digital), and μ -law (North America and Japan).

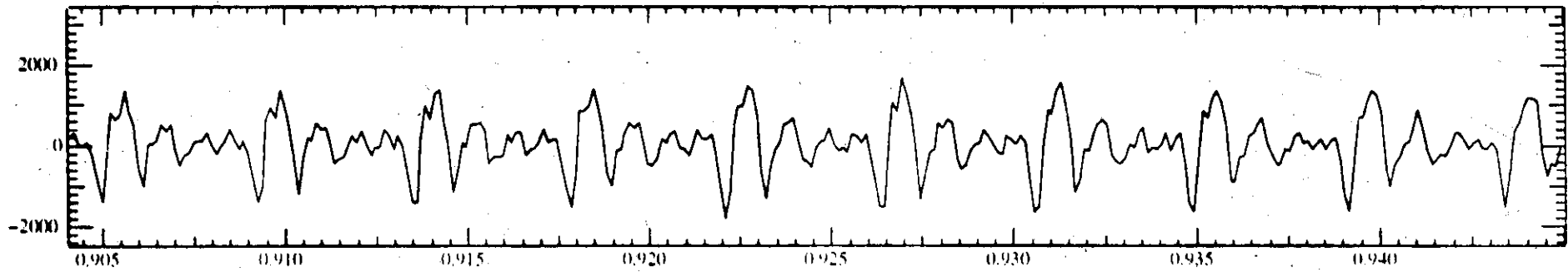
- A-law: $w(s) = \begin{cases} s & \text{if } |s| < \kappa A \\ \log s & \text{o.w.} \end{cases}$
- μ -law: $w(s) = \text{sgn}(s) A \frac{\log(1 + \mu/A|s|)}{\log(1 + \mu/A)}$

where:

- A is the maximum amplitude of the signal being quantized,
- κ is a compression parameter (in European telephony, $1/8756$), and
- μ is determined by the sample size (in North America, $\mu = 255$ because the sample size is 8 bits).

The physical speech signal (2)

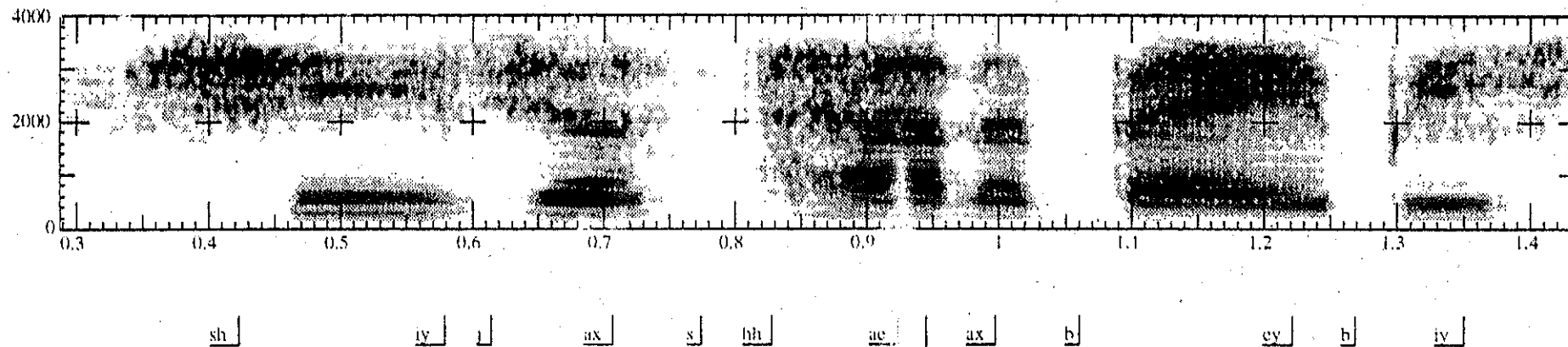
Jurafsky and Martin



Some of the [æ] of *had*. The *x*-axis is time; the *y*-axis is amplitude.

Spectrogram

Jurafsky and Martin



Spectrogram of *She just had a baby*. The x -axis is time; the y -axis is frequency; darkness is amplitude of frequency.

main body of the tongue. In contrast, the vowel /i/ as in "eve" is formed by raising the tongue toward the palate, thus causing a constriction at the front and increasing the opening at the back of the vocal tract. Thus, each vowel sound can be characterized by the vocal tract configuration (area function) that is used in its production. It is obvious that this is a rather imprecise characterization because of the inherent differences between the vocal tracts of speakers. An alternative representation is in terms of the resonance frequencies of the vocal tract. Again a great deal of variability is to be expected among speakers producing the same vowel. Peterson and Barney [11] measured the formant (resonance) frequencies (using a sound spectrograph) of vowels that were perceived to be equivalent. Their results are shown in Fig. 3.4 which is a plot of second

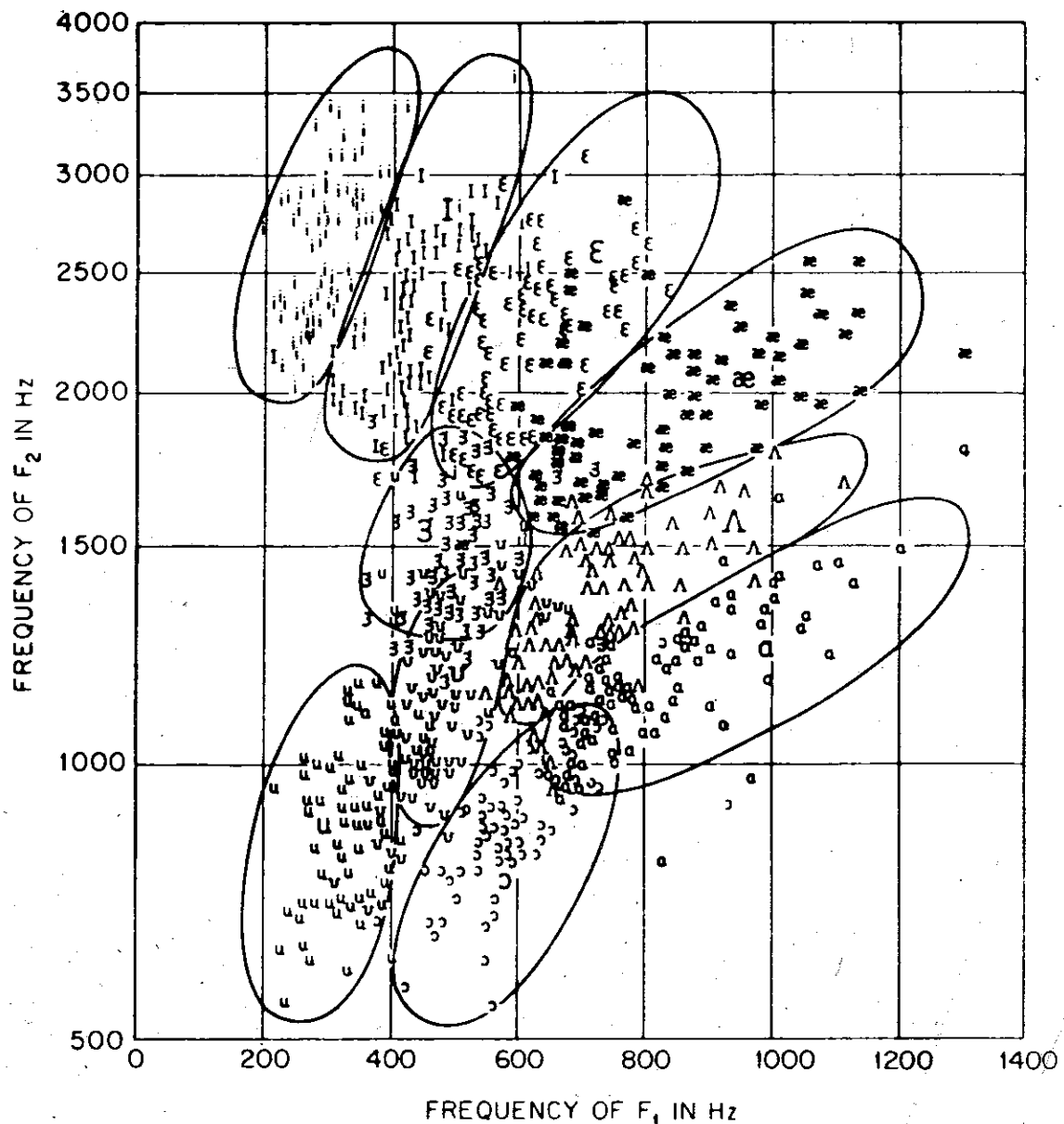


Fig. 3.4 Plot of second formant frequency versus first formant frequency for vowels by a wide range of speakers. (After Peterson and Barney [11].)

frame the probability of [p] is .8, the probability of [b] is .1, the probability of [f] is .02, etc.”); for a Gaussian model the probabilities are slightly different. Finally, in the **decoding** stage, we take a dictionary of word pronunciations and a language model (probabilistic grammar) and use a Viterbi or A* **decoder** to find the sequence of words which has the highest probability given the acoustic events.

DECODER

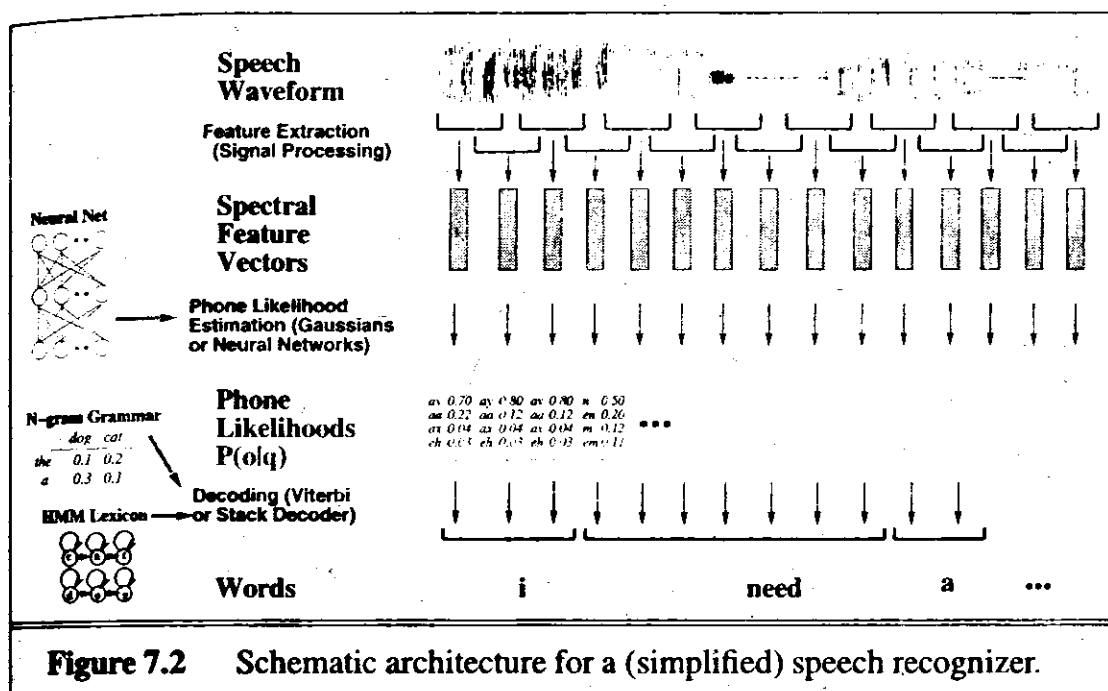


Figure 7.2 Schematic architecture for a (simplified) speech recognizer.

7.2 OVERVIEW OF HIDDEN MARKOV MODELS

In Chapter 5 we used **weighted finite-state automata** or **Markov chains** to model the pronunciation of words. The automata consisted of a sequence of states $q = (q_0 q_1 q_2 \dots q_n)$, each corresponding to a phone, and a set of transition probabilities between states, a_{01}, a_{12}, a_{13} , encoding the probability of one phone following another. We represented the states as nodes, and the transition probabilities as edges between nodes; an edge existed between two nodes if there was a non-zero transition probability between the two nodes. We also saw that we could use the **forward** algorithm to compute the **likelihood** of a sequence of observed phones $o = (o_1 o_2 o_3 \dots o_n)$. Figure 7.3