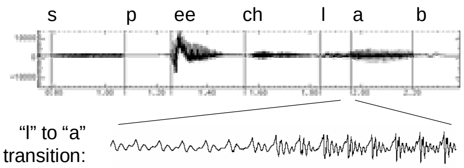
Speech Recognition: Acoustic Waves

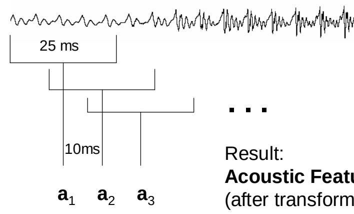
- Human speech generates a wave
  - like a loudspeaker moving
- A wave for the words “speech lab” looks like:



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

Acoustic Sampling

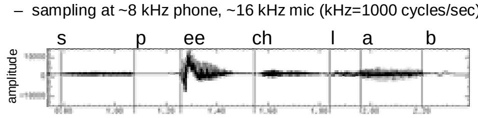
- 10 ms frame (ms = millisecond = 1/1000 second)
- ~25 ms window around frame [wide band] to allow/smooth signal processing – it let’s you see formants



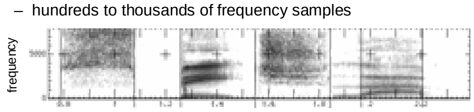
Result:
Acoustic Feature Vectors
(after transformation, numbers in roughly $\mathbf{R}^{14}$)

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
  - hundreds to thousands of frequency samples



The Speech Recognition Problem

- The **Recognition Problem: Noisy channel model**
  - We started out with English words, they were encoded as an audio signal, and we now wish to decode.
  - Find most likely sequence **w** of “words” given the sequence of acoustic observation vectors **a**
  - Use Bayes’ law to create a **generative model** and then decode
  - $\mathrm{ArgMax}_{\mathbf{w}}\ P(\mathbf{w}|\mathbf{a}) = \mathrm{ArgMax}_{\mathbf{w}}\ P(\mathbf{a}|\mathbf{w})\, P(\mathbf{w}) / P(\mathbf{a})$
    $= \mathrm{ArgMax}_{\mathbf{w}}\ P(\mathbf{a}|\mathbf{w})\, P(\mathbf{w})$
- **Acoustic Model:** P(**a**|**w**)
- **Language Model:** P(**w**)

A probabilistic theory of a language

Probabilistic Language Models

- Assign probability P(**w**) to word sequence **w** = $w_1, w_2, \ldots, w_k$
- Can’t directly compute probability of long sequence – one needs to decompose it
- Chain rule provides a **history-based** model:
  $P(w_1, w_2, \ldots, w_k) =$
  $P(w_1)\, P(w_2|w_1)\, P(w_3|w_1,w_2) \cdots P(w_k|w_1,\ldots,w_{k-1})$
- **Cluster** histories to reduce number of parameters
- E.g., just based on the last word (1st order Markov model):
  $P(w_1, w_2, \ldots, w_k) = P(w_1|\langle s\rangle)\, P(w_2|w_1)\, P(w_3|w_2) \cdots P(w_k|w_{k-1})$
- How do we estimate these probabilities?
  - We count word sequences in corpora
  - We “smooth” probabilities so as to allow unseen sequences

N -gram Language Modeling

- *n*-gram assumption clusters based on last *n*-1 words
  - $P(w_j|w_1,\ldots,w_{j-1}) \approx P(w_j|w_{j-n-1},\ldots,w_{j-2},w_{j-1})$
  - unigrams ~ $P(w_j)$
  - bigrams ~ $P(w_j|w_{j-1})$
  - trigrams ~ $P(w_j|w_{j-2},w_{j-1})$
- Trigrams often interpolated with bigram and unigram:

$$\hat{P}(w_3 \mid w_1, w_2) = \lambda_3 \frac{F(w_3 \mid w_1, w_2)}{\sum_k F(w_k \mid w_1, w_2)} + \lambda_2 \frac{F(w_3 \mid w_2)}{\sum_k F(w_k \mid w_2)} + \lambda_1 \frac{F(w_3)}{\sum_k F(w_k)}$$

  - the λ_i typically estimated by maximum likelihood estimation on held out data (F(.|.) are relative frequencies)
  - many other interpolations exist (another standard is a non-linear **backoff**)