speech recognition: acoustic waves

- human speech generates a wave
  - like a loudspeaker moving

- a wave for the words "speech lab" looks like:

  "i" to "a" transition:

  [image]

  graphs from simon amber's web tutorial on speech, sheffield:
  http://www.pspc.leeds.ac.uk/research/logicspeech/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

  [image]

  result:
  acoustic feature vectors
  (after transformation, numbers in roughly \( \mathbb{R}^d \))

spectral analysis

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

  [image]

- 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
  - \( \arg\max_{w} P(w | a) = \arg\max_{w} P(a | w) P(w) / P(a) \)
  - 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:
    \[
    P(w_1, w_2, \ldots, w_k) = P(w_k | w_{k-1}, \ldots, w_1) P(w_{k-1} | w_{k-2}, \ldots, w_1) \ldots P(w_2 | w_1)
    \]
  - cluster histories to reduce number of parameters
    - e.g., just based on the last word (1st order markov model):
      \[
      P(w_1, w_2, \ldots, w_k) = P(w_k | w_1 < s) P(w_2 | w_1) P(w_3 | w_2) \ldots P(w_k | w_{k-1})
      \]

- 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
  - \( P(w_{k} | w_{k-1}, \ldots, w_{k-n+1}) = P(w_{k-1}, \ldots, w_{k-n+1}) P(w_{k-n+1}) \)
  - unigrams \( P(w_1) \)
  - bigrams \( P(w_2 | w_1) \)
  - trigrams \( P(w_3 | w_2, w_1) \)

- trigrams often interpolated with bigram and unigram:
  \[
  \hat{P}(w_k | w_{k-1}, w_{k-2}) = \lambda_1 \frac{F(w_k | w_{k-1}, w_{k-2})}{\sum_{w_{k+1}} F(w_{k+1} | w_{k-1}, w_{k-2})} + \lambda_2 \frac{F(w_k | w_{k-1})}{\sum_{w_{k+1}} F(w_{k+1} | w_{k-1})} + \lambda_3 \frac{F(w_k)}{\sum_{w_{k+1}} F(w_{k+1})}
  \]
  - the \( \lambda \)s typically estimated by maximum likelihood estimation on held out data \( \hat{P}(w_{k+1} | w_k) \) are relative frequencies
  - many other interpolations exist (another standard is a non-linear backoff)