Natural Language Processing

Acoustic Models
Dan Klein – UC Berkeley

The Noisy Channel Model

Acoustic model: HMMs over word positions with mixtures of Gaussians as emissions
Language model: Distributions over sequences of words (sentences)

Speech Recognition Architecture

Feature Extraction

Digitizing Speech

Frame Extraction

- A frame (25 ms wide) extracted every 10 ms

Figure: Bryan Pellom

Figure: Simon Arnfield
Mel Freq. Cepstral Coefficients

- Do FFT to get spectral information
  - Like the spectrogram we saw earlier
- Apply Mel scaling
  - Models human ear; more sensitivity in lower freqs
  - Approx linear below 1kHz, log above, equal samples above and below 1kHz
- Plus discrete cosine transform

Final Feature Vector

- 39 (real) features per 10 ms frame:
  - 12 MFCC features
  - 12 delta MFCC features
  - 12 delta-delta MFCC features
  - 1 (log) frame energy
  - 1 delta (log) frame energy
  - 1 delta-delta (log frame energy)

  So each frame is represented by a 39D vector

Emission Model

HMMs for Continuous Observations

- Before: discrete set of observations
- Now: feature vectors are real-valued
- Solution 1: discretization
- Solution 2: continuous emissions
  - Gaussians
  - Multivariate Gaussians
  - Mixtures of multivariate Gaussians

- A state is progressively
  - Context independent subphone (~3 per phone)
  - Context dependent phone (triphones)
  - State tying of CD phone

Vector Quantization

- Idea: discretization
  - Map MFCC vectors onto discrete symbols
  - Compute probabilities just by counting

  This is called vector quantization or VQ

- Not used for ASR any more
- But: useful to consider as a starting point

Gaussian Emissions

- VQ is insufficient for top-quality ASR
  - Hard to cover high-dimensional space with codebook
  - Moves ambiguity from the model to the preprocessing

- Instead: assume the possible values of the observation vectors are normally distributed.
  - Represent the observation likelihood function as a Gaussian?
Gaussians for Acoustic Modeling

A Gaussian is parameterized by a mean and a variance:

\[ P(x|\mu, \sigma) = \frac{1}{\sigma \sqrt{2\pi}} \exp \left( -\frac{(x-\mu)^2}{2\sigma^2} \right) \]

- **P(x)**:
  - \( P(x) \) is highest here at mean
  - \( P(x) \) is low here, far from mean

Multivariate Gaussians

- Instead of a single mean \( \mu \) and variance \( \sigma^2 \):
  \[ P(x|\mu, \Sigma) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp \left( -\frac{1}{2}(x-\mu)^\top \Sigma^{-1}(x-\mu) \right) \]
- Vector of means \( \mu \) and covariance matrix \( \Sigma \): **P(x)**
- Usually assume diagonal covariance (!)
  - This isn't very true for FFT features, but is less bad for MFCC features

Gaussians: Size of \( \Sigma \)

- \( \mu = [0 \ 0] \)
- \( \Sigma = I \)
- As \( \Sigma \) becomes larger, Gaussian becomes more spread out; as \( \Sigma \) becomes smaller, Gaussian more compressed

Gaussians: Shape of \( \Sigma \)

- \( \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \)
- \( \Sigma = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix} \)
- \( \Sigma = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix} \)
- As we increase the off diagonal entries, more correlation between value of \( x \) and value of \( y \)

But we’re not there yet

- Single Gaussians may do a bad job of modeling a complex distribution in any dimension
- Even worse for diagonal covariances
- Solution: mixtures of Gaussians

Mixtures of Gaussians

- Mixtures of Gaussians:
  \[ P(x|\mu_i, \Sigma_i) = \frac{1}{(2\pi)^{d/2}|\Sigma_i|^{1/2}} \exp \left( -\frac{1}{2}(x-\mu_i)^\top \Sigma_i^{-1}(x-\mu_i) \right) \]
  \[ P(x|\mu, \Sigma, c) = \sum_i c_i P(x|\mu_i, \Sigma_i) \]
GMMs

- Summary: each state has an emission distribution $P(x|s)$ (likelihood function) parameterized by:
  - $M$ mixture weights
  - $M$ mean vectors of dimensionality $D$
  - Either $M$ covariance matrices of $D \times D$ or $M$ $D \times 1$ diagonal variance vectors

- Like soft vector quantization after all:
  - Think of the mixture means as being learned codebook entries
  - Think of the Gaussian densities as a learned codebook distance function
  - Think of the mixture of Gaussians like a multinomial over codes
  - (Even more true given shared Gaussian inventories, cf next week)

State Transition Diagrams

- Bayes Net: HMM as a Graphical Model

- State Transition Diagram: Markov Model as a Weighted FSA

Lexical State Structure

ASR Lexicon

Adding an LM

Figure: J & M
**State Space**

- State space must include
  - Current word ($|V|$ on order of 20K+)
  - Index within current word ($|L|$ on order of 5)

- Acoustic probabilities only depend on phone type
  - E.g. $P(x|\text{lecture}) = P(x|t)$

- From a state sequence, can read a word sequence

---

**State Refinement**

---

**Phones Aren’t Homogeneous**

---

**Need to Use Subphones**

---

**A Word with Subphones**

---

**Modeling phonetic context**

---
“Need” with triphone models

Lots of Triphones

- Possible triphones: 50x50x50=125,000
- How many triphone types actually occur?
- 20K word WSJ Task (from Bryan Pellom)
  - Word internal models: need 14,300 triphones
  - Cross word models: need 54,400 triphones
- Need to generalize models, tie triphones

State Tying / Clustering

- [Young, Odell, Woodland 1994]
- How do we decide which triphones to cluster together?
- Use phonetic features (or “broad phonetic classes”)
  - Stop
  - Nasal
  - Fricative
  - Sibilant
  - Vowel
  - lateral

State Space

- State space now includes
  - Current word: |W| is order 20K
  - Index in current word: |L| is order 5
  - Subphone position: 3
- Acoustic model depends on clustered phone context
  - But this doesn’t grow the state space

Decoding

Inference Tasks

Most likely word sequence:

Most likely state sequence:

\[ d_1, d_2, d_3, d_4, d_5, d_6, d_7, d_8, d_9, d_{10}, d_{11}, d_{12}, d_{13} \]
### Emission Caching
- **Problem:** scoring all the P(x|s) values is too slow
- **Idea:** many states share tied emission models, so cache them

### Prefix Trie Encodings
- **Problem:** many partial-word states are indistinguishable
- **Solution:** encode word production as a prefix trie (with pushed weights)

### Beam Search
- **Problem:** trellis is too big to compute v(s) vectors
- **Idea:** most states are terrible, keep v(s) only for top states at each time
- **Important:** still dynamic programming; collapse equiv states

### LM Factoring
- **Problem:** Higher-order n-grams explode the state space
- **(One) Solution:**
  - Factor state space into (word index, lm history)
  - Score unigram prefix costs while inside a word
  - Subtract unigram cost and add trigram cost once word is complete
LM Reweighting

- Noisy channel suggests
  \[ P(x|w)P(w) \]
- In practice, want to boost LM
  \[ P(x|w)P(w)^\alpha \]
- Also, good to have a “word bonus” to offset LM costs
  \[ P(x|w)P(w)^\alpha P(w)^\beta \]
- These are both consequences of broken independence assumptions in the model