Introduction to Artificial Intelligence

V22.0472-001 Fall 2009

Lecture 19: Speech Recognition & Viterbi Decoding

Rob Fergus – Dept of Computer Science, Courant Institute, NYU

Slides from John DeNero, Dan Klein, Zheng Chen

---

**Today**

- HMMs: Most likely explanation queries
- Speech recognition
  - A massive HMM!
  - Details of this section not required

---

**Speech and Language**

- Speech technologies
  - Automatic speech recognition (ASR)
  - Text-to-speech synthesis (TTS)
  - Dialog systems
- Language processing technologies
  - Machine translation
  - Information extraction
  - Web search, question answering
  - Text classification, spam filtering, etc…

---

**HMMs: MLE Queries**

- HMMs defined by
  - States \( X \)
  - Observations \( E \)
  - Initial distr: \( P(X_1) \)
  - Transitions: \( P(X_t|X_{t-1}) \)
  - Emissions: \( P(E|X) \)

- Query: most likely explanation:

\[
\arg\max_{x_{1:T}} P(x_{1:T}|e_{1:T}) \quad \text{Viterbi algorithm}
\]

---

**State Path Trellis**

- State trellis: graph of states and transitions over time

\[
X_1 \rightarrow X_2 \rightarrow \cdots \rightarrow X_N
\]

- Each arc represents some transition \( x_{t-1} \rightarrow x_t \)
- Each arc has weight \( P(x_t|x_{t-1})P(e_t|x_t) \)
- 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
Viterbi Algorithm

\[ x_{1:T} = \arg \max_{x_{1:T}} P(x_{1:T} \mid \mathbf{e}_{1:T}) = \arg \max_{x_{1:T}} P(x_{1:T}, e_{1:T}) \]

\[ m_t[x_t] = \max_{e_{1:t-1}} P(x_{1:t-1}, e_{1:t-1}) \]

\[ = \max_{e_{1:t-1}} P(x_{1:t-1}, e_{1:t-1}) P(x_t \mid x_{t-1}) P(e_t \mid x_t) \]

\[ = P(x_t \mid x_{t-1}, e_{1:t-1}) \max_{e_{1:t-2}} P(x_{1:t-1}, e_{1:t-2}) \]

\[ = P(x_t \mid x_{t-1}) m_{t-1}[x_{t-1}] \]

Example

Andrew Viterbi

Digitizing Speech

Speech in an Hour

- Speech input is an acoustic wave form

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

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 0.036 seconds)
- Small wave roughly 4 times this, or roughly 1000 Hz
- Two little tiny waves on top of peak of 1000 Hz waves

Resonances of the vocal tract

- The human vocal tract as an open tube

  Length 17.5 cm

  - 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.

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.
**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**

- $P(E\mid X)$ encodes which acoustic vectors are appropriate for each phoneme (each kind of sound)
- $P(X\mid X')$ 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**

**Schematic Architecture for a (simplified) Speech Recognizer**

The most common model used for speech is constrained, allowing a state to transition only to itself or to a single succeeding state.

**Fine-grained HMM model to represent a phone**
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 $e_{1:T}$:

$$x_{1:T} = \underset{x_{1:T}}{\arg \max} P(x_{1:T} | e_{1:T})$$

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

Also use Language Model

For the given acoustic observation $O = o_1, o_2, \ldots, o_n$, the goal of speech recognition is to find out the corresponding word sequence $W = w_1, w_2, \ldots, w_n$ that has the maximum posterior probability $P(W|O)$:

$$\hat{W} = \underset{W \in \mathcal{L}}{\arg \max} P(W | O)$$

Using the Bayes' rule:

$$\hat{W} = \underset{W \in \mathcal{L}}{\arg \max} \frac{P(O|W)P(W)}{P(O)} = \underset{W \in \mathcal{L}}{\arg \max} \frac{P(O|W)}{P(O)}$$

where $P(O|W)$ is the likelihood and $P(W)$ is the prior.

![Diagram of HMM and Language Model](image)