CS 378 Lecture 10
HMMs, Viterbi

Announcements
- A2 due today
- A3 out today
- Midterm: in 2.5 weeks
- Final proj: if independent project, start thinking

Recap POS tagging

Fed raises interest rates

How to do with a classifier?

\[
P(y_i = \hat{y} | l_x) = \frac{\sum e^{w_{y}^T f(x, i)}}{\sum e^{w_{y}^T f(x, i)}}
\]
Problem: all $y_i$ are predicted independently
How do we model the sequence all at once?

Today Hidden Markov Models
- Definition
- Training
- Inference (Viterbi)

HMM: generative model $P(\bar{y}, \bar{x})$

There are discriminative seq. models, Conditional Random Fields $P(\bar{y} | \bar{x})$
**HMMs** Tags $y_i \in T$, words $x_i \in V$

$$P(\gamma, x) = P(y_1) P(x_1 | y_1) P(y_2 | y_1) P(x_2 | y_2) \cdots P(\text{STOP} | y_n)$$

- $\gamma = (y_1, y_2, y_3, \ldots, y_n)$
- $x = (x_1, x_2, x_3, \ldots, x_n)$

**Generative story:** draw $y_1$ to start tag

- Fed sequence

- draw $x_1 | y_1$ as the first word
- draw $y_2 | y_1$ as the next tag
Parameters

$P(y_i)$
initial distribution

call this $S$

$P(y_i | y_{i-1})$
transitions

$E$
emissions

$P(x_i | y_i)$

$P(N)$
$P(V)$

$|T| 	imes |V|$ matrix

$|T| - 1$ en vector that sums to 1
why Markov model: $P(y_i | y_{i-1})$ $y_i$ doesn't depend on $y_{i-2}$

Goals with HMMs

HMM $P(\overline{y,x})$

What we want: $P(\overline{y,x}) = \frac{P(\overline{y,x})}{P(x)}$

Ex $T = \{N, V, STOP\}$ $V = \{they, can, fish\}$

$S$ $P(y_i)$
$N$ 1.0
$V$ 0.0
STOP 0.0

$T$ $N$ $V$ $STOP$
$N$ 1/5 3/5 1/5
$V$ 1/5 1/5 3/5

$E$ $N$ $V$
$N$ 1 0 0
$V$ 0 1/2 1/2

"they, can, fish"
Poll

1) What is the probability of 
$$\binom{N}{V} \binom{V}{V}$$ (they can fish)

$$P(y|x) = P(N)P(\text{they IN})$$

$$1.0 \cdot 1.0$$

$$1.0 \cdot \frac{3}{5} \cdot \frac{1}{5} = \frac{3}{5}$$

$$1.0 \cdot \frac{1}{2} \cdot \frac{1}{2}$$

2) N V V
they can fish

$$P(y|x) = 0.0$$

V N V

3) Other seagulls for
they can fish?

No
Training: Given labeled sequences

\[
\begin{pmatrix}
X^{(i)} \\
Y^{(i)}
\end{pmatrix}_{i=1}^D
\]

Data: NV NV

they can they fish

What parameters maximize the likelihood of the data?

Can find this exactly by counting + normalizing

Biased coin w/probability p of heads

See HHHT

\[p = \frac{3}{4}\] is the MLE
\[ P(x_i \mid v) = \begin{cases} \text{can: } 1 & \text{they: } 0 \\ \text{fish: } 1 & \Rightarrow \text{can} \ 1/2 \\ \text{they: } 0 & \text{fish } 1/2 \end{cases} \]

This is what "training" involves

\[ P(y_i \mid y_{i-1} = v) = \begin{cases} N: 0 & N: 0 \\ V: 0 & \Rightarrow V: 0 \\ \text{stop: } 2 & \text{stop: } 1 \end{cases} \]

**Smoothing**: pretend we saw everything with some nonzero count

Add 0.1 to every count

\[ \begin{array}{c|c|c} N & 0 & 0.1 \\ \hline V & 0 & \Rightarrow V & 0.1 & \Rightarrow \text{probs} \\ \hline \text{stop } 2 & \text{stop } 2.1 \end{array} \]
Inference in HMMs

We defined HMMs as $P(\bar{y}, \bar{x})$

What we really want is a tagger

$P(\bar{y} | \bar{x})$ "you give me a sent, I tell you tags"

$\arg \max_{\bar{y}} \ P(\bar{y} | \bar{x})$ Two problems:

1. Space is exponentially large $(|Y|^n)$

2. We don't have $P(\bar{y} | \bar{x})$ yet

Solving #2:

$\arg \max_{\bar{y}} \ P(\bar{y} | \bar{x}) = \arg \max_{\bar{y}} \ \frac{P(\bar{y} | \bar{x})}{P(\bar{x})} \ P(\bar{y}, \bar{x})$

$= \arg \max_{\bar{y}} \ \frac{P(\bar{y}, \bar{x})}{P(\bar{x})}$
\[ P(\bar{x}) = \sum_{\bar{y}} P(\bar{y}, \bar{x}) \]

We don't need this

\[
\arg\max_{\bar{y}} \frac{P(\bar{y}, \bar{x})}{P(\bar{x})} = \arg\max_{\bar{y}} \frac{P(\bar{y}, \bar{x})}{P(\bar{x})}
\]

Conclusion: \( \arg\max_{\bar{y}} P(\bar{y}, \bar{x}) \)

to build our tagger #2

Solving #1: the Viterbi algorithm

First: move to log space

\[
\arg\max_{\bar{y}} \log P(\bar{y}, \bar{x}) = \arg\max_{\bar{y}} \log P(\bar{y}, \bar{x})
\]
\[
\arg\max_{\bar{Y}} \log P(\bar{Y}, x)
\]

\[
= \arg\max_{\tilde{Y}_1, \ldots, \tilde{Y}_n} \log P(\tilde{Y}_1) + \log P(x_i | \tilde{Y}_i) + \log P(\tilde{Y}_2 | \tilde{Y}_1) + \ldots
\]

Basic concept: dynamic programming

Best sequence of \(\tilde{Y}_1, \ldots, \tilde{Y}_n\) can be found from best sequence from \(\tilde{Y}_1, \ldots, \tilde{Y}_{n-1}\) + HMM probs

optimal seq for each possible \(\tilde{Y}_{n-1}\)