Hidden Markov Models (II): Log-Likelihood (Forward Algorithm)

CMSC 473/673
UMBC
October 11th, 2017
Course Announcement: Assignment 2

Due next Saturday, 10/21 at 11:59 PM (~10 days)

Any questions?
Recap from last time...
Expectation Maximization (EM)

0. Assume *some* value for your parameters

Two step, iterative algorithm

1. E-step: count under uncertainty, assuming these parameters

\[ p(z_i) \quad \rightarrow \quad \text{count}(z_i, w_i) \]

2. M-step: maximize log-likelihood, assuming these uncertain counts

\[ p^{(t)}(z) \quad \leftrightarrow \quad \text{estimated counts} \quad \rightarrow \quad p^{(t+1)}(z) \]
Counting Requires Marginalizing

E-step: count under uncertainty, assuming these parameters break into 4 disjoint pieces

\[ p(z_i) \xrightarrow{\text{break into 4 disjoint pieces}} \text{count}(z_i, w_i) \]

\[ p(w) = p(z_1, w) + p(z_2, w) + p(z_3, w) + p(z_4, w) = \sum_{z=1}^{4} p(z_i, w) \]
Agenda

HMM Motivation (Part of Speech) and Brief Definition

What is Part of Speech?

HMM Detailed Definition

HMM Tasks
## Penn Treebank Part of Speech

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
<th>Example</th>
<th>Tag</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>CC</td>
<td>coordin. conjunction</td>
<td><em>and, but, or</em></td>
<td>SYM</td>
<td>symbol</td>
<td>+, %, &amp;</td>
</tr>
<tr>
<td>CD</td>
<td>cardinal number</td>
<td><em>one, two</em></td>
<td>TO</td>
<td>“to”</td>
<td><em>to</em></td>
</tr>
<tr>
<td>DT</td>
<td>determiner</td>
<td><em>a, the</em></td>
<td>UH</td>
<td>interjection</td>
<td><em>ah, oops</em></td>
</tr>
<tr>
<td>EX</td>
<td>existential ‘there’</td>
<td><em>there</em></td>
<td>VB</td>
<td>verb base form</td>
<td><em>eat</em></td>
</tr>
<tr>
<td>FW</td>
<td>foreign word</td>
<td><em>mea culpa</em></td>
<td>VBD</td>
<td>verb past tense</td>
<td><em>ate</em></td>
</tr>
<tr>
<td>IN</td>
<td>preposition/sub-conj</td>
<td><em>of, in, by</em></td>
<td>VBG</td>
<td>verb gerund</td>
<td><em>eating</em></td>
</tr>
<tr>
<td>JJ</td>
<td>adjective</td>
<td><em>yellow</em></td>
<td>VBN</td>
<td>verb past participle</td>
<td><em>eaten</em></td>
</tr>
<tr>
<td>JJR</td>
<td>adj., comparative</td>
<td><em>bigger</em></td>
<td>VBP</td>
<td>verb non-3sg pres</td>
<td><em>eat</em></td>
</tr>
<tr>
<td>JJS</td>
<td>adj., superlative</td>
<td><em>wildest</em></td>
<td>VBZ</td>
<td>verb 3sg pres</td>
<td><em>eats</em></td>
</tr>
<tr>
<td>LS</td>
<td>list item marker</td>
<td><em>1, 2, One</em></td>
<td>WDT</td>
<td>wh-determiner</td>
<td><em>which, that</em></td>
</tr>
<tr>
<td>MD</td>
<td>modal</td>
<td><em>can, should</em></td>
<td>WP</td>
<td>wh-pronoun</td>
<td><em>what, who</em></td>
</tr>
<tr>
<td>NN</td>
<td>noun, sing. or mass</td>
<td><em>llama</em></td>
<td>WPS</td>
<td>possessive wh-</td>
<td><em>whose</em></td>
</tr>
<tr>
<td>NNS</td>
<td>noun, plural</td>
<td><em>llamas</em></td>
<td>WRB</td>
<td>wh-adverb</td>
<td><em>how, where</em></td>
</tr>
<tr>
<td>NNP</td>
<td>proper noun, sing.</td>
<td><em>IBM</em></td>
<td>$</td>
<td>dollar sign</td>
<td>$</td>
</tr>
<tr>
<td>NNPS</td>
<td>proper noun, plural</td>
<td><em>Carolinatas</em></td>
<td>#</td>
<td>pound sign</td>
<td>#</td>
</tr>
<tr>
<td>PDT</td>
<td>predeterminer</td>
<td><em>all, both</em></td>
<td>“</td>
<td>left quote</td>
<td>‘ or “</td>
</tr>
<tr>
<td>POS</td>
<td>possessive ending</td>
<td>’s</td>
<td>”</td>
<td>right quote</td>
<td>’ or ”</td>
</tr>
<tr>
<td>PRP</td>
<td>personal pronoun</td>
<td><em>I, you, he</em></td>
<td>(</td>
<td>left parenthesis</td>
<td>[ , ( { , &lt;</td>
</tr>
<tr>
<td>PRPS</td>
<td>possessive pronoun</td>
<td><em>your, one’s</em></td>
<td>)</td>
<td>right parenthesis</td>
<td>] , ) , } , &gt;</td>
</tr>
<tr>
<td>RB</td>
<td>adverb</td>
<td><em>quickly, never</em></td>
<td>,</td>
<td>comma</td>
<td>,</td>
</tr>
<tr>
<td>RBR</td>
<td>adverb, comparative</td>
<td><em>faster</em></td>
<td>.</td>
<td>sentence-final punctuation</td>
<td>. ! ?</td>
</tr>
<tr>
<td>RBS</td>
<td>adverb, superlative</td>
<td><em>fastest</em></td>
<td>:</td>
<td>mid-sentence punctuation</td>
<td>: ; ... --</td>
</tr>
<tr>
<td>RP</td>
<td>particle</td>
<td><em>up, off</em></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Figure 10.1* Penn Treebank part-of-speech tags (including punctuation).
<table>
<thead>
<tr>
<th>Language</th>
<th>UD size</th>
<th>UD data</th>
<th>Treebank</th>
</tr>
</thead>
<tbody>
<tr>
<td>African languages</td>
<td>48K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ancient Greek</td>
<td>202K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arabic</td>
<td>242K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arabic-huAD</td>
<td>632K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arabic-PUD</td>
<td>20K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Basque</td>
<td>121K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Belarusian</td>
<td>8K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bulgarian</td>
<td>158K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Byurut</td>
<td>16K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Catalan</td>
<td>532K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chinese</td>
<td>123K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chinese-CFL</td>
<td>7K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chinese-PUD</td>
<td>21K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Coptic</td>
<td>4K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Croatian</td>
<td>107K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Czech</td>
<td>1,530K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Czech-CAC</td>
<td>453K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Czech-CLTT</td>
<td>37K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Czech-PUD</td>
<td>18K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Danish</td>
<td>100K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dutch</td>
<td>209K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dutch-LassySmall</td>
<td>101K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>English</td>
<td>254K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>English-ESL</td>
<td>88K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>English-LINDS</td>
<td>82K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>English-PUD</td>
<td>21K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>English-ParTUT</td>
<td>49K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Estonian</td>
<td>47K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Finnish</td>
<td>202K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Finnish-PUD</td>
<td>15K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Finnish-FTB</td>
<td>15K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>French</td>
<td>351K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>French-FTB</td>
<td>556K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>French-PUD</td>
<td>24K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>French-ParTUT</td>
<td>27K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>French-SeqVola</td>
<td>68K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Galician</td>
<td>126K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Galician-TreGal</td>
<td>23K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>German</td>
<td>287K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>German-PUD</td>
<td>20K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gothic</td>
<td>55K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Greek</td>
<td>61K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hebrew</td>
<td>115K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hindi</td>
<td>351K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hindi-PUD</td>
<td>23K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hungarian</td>
<td>42K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Indonesian</td>
<td>121K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Indonesian-PUD</td>
<td>25K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Irish</td>
<td>23K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Italian</td>
<td>273K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Italian-PUD</td>
<td>22K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Japanese</td>
<td>186K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Japanese-KTC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Japanese-PUD</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kazakh</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Korean</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Korean-PUD</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kumań</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latin</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latin-PROIEL</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latin-ProTIB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latinian</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lithuanian</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maltese</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>North Sami</td>
<td>59K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Norwegian-Bokmaen</td>
<td>310K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Norwegian-Nynorsk</td>
<td>301K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Old Church Slavonic</td>
<td>57K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Persian</td>
<td>151K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Persian</td>
<td>151K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Portuguese</td>
<td>92K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Portuguese-Br</td>
<td>297K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Portuguese-PUD</td>
<td>21K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Romanian</td>
<td>218K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Russian</td>
<td>95K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Russian-PUD</td>
<td>19K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Russian-SynTagRus</td>
<td>1,107K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sanskrit</td>
<td>1K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Serbian</td>
<td>80K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Slovene</td>
<td>104K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Slovenian</td>
<td>146K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Slovenian-SST</td>
<td>29K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spanish</td>
<td>423K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spanish-AnCorA</td>
<td>647K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spanish-PUD</td>
<td>22K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Swedish</td>
<td>96K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Swedish-LinEES</td>
<td>79K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Swedish-PUD</td>
<td>19K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Swedish-Sign Language</td>
<td>&lt;1K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tamil</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Thai-PUD</td>
<td>8K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Turkish</td>
<td>56K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Turkish-PUD</td>
<td>16K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ukrainian</td>
<td>25K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Upper Sorbian</td>
<td>11K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Urdu</td>
<td>138K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Uyghur</td>
<td>11K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vietnamese</td>
<td>43K</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Upcoming UD Treebanks**

- Amharic
- Armenian
- Bangla
- Bengali-DDS
- Cantonese
- Chinese-HK
- Czech-FicTree
- Dargwa
- Faroese
- French-Spoken
- Italian-PostTWITA
Hidden Markov Models: Part of Speech

(i): Adjective → Noun → Verb → Prep → Noun → Noun
(ii): Noun → Verb → Noun → Prep → Noun → Noun

$p(\text{British Left Waffles on Falkland Islands})$

Class-based model

$p(w_i | z_i)$

Bigram model of the classes

$p(z_i | z_{i-1})$

Model all class sequences

$\sum_{z_1, \ldots, z_N} p(z_1, w_1, z_2, w_2, \ldots, z_N, w_N)$
Hidden Markov Model

\[ p(z_1, w_1, z_2, w_2, ..., z_N, w_N) = p(z_1 | z_0) p(w_1 | z_1) \cdots p(z_N | z_{N-1}) p(w_N | z_N) = \prod_i p(w_i | z_i) p(z_i | z_{i-1}) \]

Goal: maximize (log-)likelihood

In practice: we don’t actually observe these \( z \) values; we just see the words \( w \)

if we did observe \( z \), estimating the probability parameters would be easy... but we don’t! :(  

if we knew the probability parameters then we could estimate \( z \) and evaluate likelihood... but we don’t! :( 
Each $z_i$ can take the value of one of $K$ latent states

Transition and emission distributions do not change

**Q:** How many different probability values are there with $K$ states and $V$ vocab items?

**A:** $VK$ emission values and $K^2$ transition values
Hidden Markov Model Representation

\[ p(z_1, w_1, z_2, w_2, \ldots, z_N, w_N) = p(z_1 | z_0)p(w_1 | z_1) \cdots p(z_N | z_{N-1})p(w_N | z_N) = \prod_{i} p(w_i | z_i)p(z_i | z_{i-1}) \]

Initial starting distribution ("BOS")
\[ p(z_1 | z_0) \]

Graphical Models (see 478/678)

Each \( z_i \) can take the value of one of \( K \) latent states

Transition and emission distributions do not change
Example: 2-state Hidden Markov Model as a Lattice
Comparison of Joint Probabilities

Unigram Language Model

\[ p(w_1, w_2, \ldots, w_N) = p(w_1)p(w_2) \cdots p(w_N) = \prod_i p(w_i) \]

Unigram Class-based Language Model ("K" coins)

\[ p(z_1, w_1, z_2, w_2, \ldots, z_N, w_N) = p(z_1)p(w_1|z_1) \cdots p(z_N)p(w_N|z_N) \]
\[ = \prod_i p(w_i|z_i) p(z_i) \]

Hidden Markov Model

\[ p(z_1, w_1, z_2, w_2, \ldots, z_N, w_N) = p(z_1|z_0)p(w_1|z_1) \cdots p(z_N|z_{N-1})p(w_N|z_N) \]
\[ = \prod_i p(w_i|z_i) p(z_i|z_{i-1}) \]
Agenda

HMM Motivation (Part of Speech) and Brief Definition

What is Part of Speech?

HMM Detailed Definition

HMM Tasks
Hidden Markov Model Tasks

\[ p(z_1, w_1, z_2, w_2, \ldots, z_N, w_N) = p(z_1 | z_0)p(w_1 | z_1) \cdots p(z_N | z_{N-1})p(w_N | z_N) \]

= \prod_i p(w_i | z_i)p(z_i | z_{i-1})

Calculate the (log) likelihood of an observed sequence \( w_1, \ldots, w_N \)

Calculate the most likely sequence of states (for an observed sequence)

Learn the emission and transition parameters
Hidden Markov Model Tasks

\[ p(z_1, w_1, z_2, w_2, \ldots, z_N, w_N) = p(z_1 | z_0) p(w_1 | z_1) \cdots p(z_N | z_{N-1}) p(w_N | z_N) \]

Calculate the (log) likelihood of an observed sequence \( w_1, \ldots, w_N \)

Calculate the most likely sequence of states (for an observed sequence)

Learn the emission and transition parameters
HMM Likelihood Task

Marginalize over all latent sequence joint likelihoods

\[ p(w_1, w_2, ..., w_N) = \sum_{z_1, ..., z_N} p(z_1, w_1, z_2, w_2, ..., z_N, w_N) \]

Q: In a K-state HMM for a length N observation sequence, how many summands (different latent sequences) are there?
HMM Likelihood Task

Marginalize over all latent sequence joint likelihoods

\[ p(w_1, w_2, ..., w_N) = \sum_{z_1, ..., z_N} p(z_1, w_1, z_2, w_2, ..., z_N, w_N) \]

Q: In a K-state HMM for a length N observation sequence, how many summands (different latent sequences) are there?

A: \( K^N \)
HMM Likelihood Task

Marginalize over all latent sequence joint likelihoods

\[ p(w_1, w_2, ..., w_N) = \sum_{z_1, ..., z_N} p(z_1, w_1, z_2, w_2, ..., z_N, w_N) \]

Q: In a K-state HMM for a length N observation sequence, how many summands (different latent sequences) are there?

A: \( K^N \)

Goal:
Find a way to compute this exponential sum efficiently (in polynomial time)
HMM Likelihood Task

Marginalize over all likelyhoods

\[ p(w_1, w_2, \ldots, w_N) = \sum_{z_1, \ldots, z_N} p(z_1, w_1, z_2, w_2, \ldots, z_N, w_N) \]

Q: In a K-state HMM for a length N observation sequence, how many summands (different latent sequences) are there?

A: \( K^N \)

Goal:
Find a way to compute this exponential sum efficiently (in polynomial time)

Like in language modeling, you need to model when to stop generating.

This ending state is generally not included in “K.”
Q: What are the latent sequences here (EOS excluded)?
Q: What are the latent sequences here (EOS excluded)?

A:

(N, w₁), (N, w₂), (N, w₃), (N, w₄)  (N, w₁), (V, w₂), (N, w₃), (N, w₄)  (V, w₁), (N, w₂), (N, w₃), (N, w₄)  (V, w₁), (N, w₂), (N, w₃), (V, w₄)  (N, w₁), (N, w₂), (V, w₃), (N, w₄)  (N, w₁), (V, w₂), (V, w₃), (N, w₄)  (N, w₁), (V, w₂), (V, w₃), (V, w₄)  (N, w₁), (V, w₂), (V, w₃), (V, w₄)  ...

... (six more)
Q: What are the latent sequences here (EOS excluded)?

A:

\[(N, w_1), (N, w_2), (N, w_3), (N, w_4), (N, w_1), (V, w_2), (N, w_3), (N, w_4)\]
\[(V, w_1), (N, w_2), (N, w_3), (N, w_4)\]
\[(N, w_1), (N, w_2), (N, w_3), (V, w_4)\]
\[(N, w_1), (V, w_2), (N, w_3), (V, w_4)\]
\[(N, w_1), (V, w_2), (V, w_3), (N, w_4)\]
\[(V, w_1), (N, w_2), (N, w_3), (V, w_4)\]
\[(V, w_1), (V, w_2), (V, w_3), (V, w_4)\]

... (six more)
2 (3)-State HMM Likelihood

\[ p(V|\text{start}) \]

\[ p(N|\text{start}) \]

\[ z_1 = V \]
\[ p(V|V) \]
\[ z_2 = V \]
\[ p(V|V) \]
\[ z_3 = V \]
\[ p(V|V) \]
\[ z_4 = V \]
\[ p(V|V) \]

\[ p(w_1|V) \]
\[ p(w_2|V) \]
\[ p(w_3|V) \]
\[ p(w_4|V) \]

\[ z_1 = N \]
\[ p(N|N) \]
\[ z_2 = N \]
\[ p(N|N) \]
\[ z_3 = N \]
\[ p(N|N) \]
\[ z_4 = N \]
\[ p(N|N) \]

\[ p(w_1|N) \]
\[ p(w_2|N) \]
\[ p(w_3|N) \]
\[ p(w_4|N) \]

\[ p(V|N) \]
\[ p(N|V) \]
\[ p(N|V) \]
\[ p(N|V) \]

\[ p(V|w_1) \]
\[ p(V|w_2) \]
\[ p(V|w_3) \]
\[ p(V|w_4) \]

\[ p(N|w_1) \]
\[ p(N|w_2) \]
\[ p(N|w_3) \]
\[ p(N|w_4) \]

\[ \begin{array}{c|c|c|c|c} \hline V & N & \text{end} \\ \hline \text{start} & .7 & .2 & .1 \\ N & .15 & .8 & .05 \\ V & .6 & .35 & .05 \end{array} \]

\[ \begin{array}{c|c|c|c|c} \hline w_1 & w_2 & w_3 & w_4 \\ \hline N & .7 & .2 & .05 & .05 \\ V & .2 & .6 & .1 & .1 \end{array} \]
2 (3)-State HMM Likelihood

Q: What's the probability of (N, w₁), (V, w₂), (V, w₃), (N, w₄)?

<table>
<thead>
<tr>
<th></th>
<th>N</th>
<th>V</th>
<th>end</th>
</tr>
</thead>
<tbody>
<tr>
<td>start</td>
<td>.7</td>
<td>.2</td>
<td>.1</td>
</tr>
<tr>
<td>N</td>
<td>.15</td>
<td>.8</td>
<td>.05</td>
</tr>
<tr>
<td>V</td>
<td>.6</td>
<td>.35</td>
<td>.05</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>w₁</th>
<th>w₂</th>
<th>w₃</th>
<th>w₄</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>.7</td>
<td>.2</td>
<td>.05</td>
<td>.05</td>
</tr>
<tr>
<td>V</td>
<td>.2</td>
<td>.6</td>
<td>.1</td>
<td>.1</td>
</tr>
</tbody>
</table>
2 (3)-State HMM Likelihood

Q: What’s the probability of 
(N, w₁), (V, w₂), (V, w₃), (N, w₄)?

A: (.7*.7) * (.8*.6) * (.35*.1) * (.6*.05) = 0.0002822
Q: What’s the probability of (N, w₁), (V, w₂), (V, w₃), (N, w₄) with ending included (unique ending symbol “#”)?

A: (.7*.7) * (.8*.6) * (.35*.1) * (.6*.05) * (.05 * 1) = 0.00001235
2 (3)-State HMM Likelihood

Q: What’s the probability of 
(N, w₁), (V, w₂), (N, w₃), (N, w₄)?
2 (3)-State HMM Likelihood

Q: What’s the probability of \((N, w_1), (V, w_2), (N, w_3), (N, w_4)\)?

A: \((.7*.7) * (.8*.6) * (.6*.05) * (.15*.05) = 0.00007056\)
Q: What’s the probability of 
(N, w₁), (V, w₂), (N, w₃), (N, w₄) with 
ending (unique symbol “#”)?

A: \((.7 \times .7) \times (.8 \times .6) \times (.6 \times .05) \times (.15 \times .05) \times (.05 \times 1) = 0.000002646\)
2 (3)-State HMM Likelihood

\[ p(N|\text{start}) \]
\[ p(V|N) \]
\[ p(w_1|N) \]
\[ z_1 = N \]
\[ w_1 \]

\[ p(V|N) \]
\[ p(w_2|V) \]
\[ z_2 = V \]
\[ w_2 \]

\[ p(w_1|N) \]
\[ p(V|N) \]
\[ p(V|V) \]
\[ z_1 = V \]
\[ w_1 \]

\[ p(N|V) \]
\[ p(w_3|V) \]
\[ z_3 = N \]
\[ w_3 \]

\[ p(N|V) \]
\[ p(w_4|N) \]
\[ z_4 = N \]
\[ w_4 \]
2 (3)-State HMM Likelihood

Up until here, all the computation was the same
2 (3)-State HMM Likelihood

Up until here, all the computation was the same

Let’s reuse what computations we can
Reusing Computation: Attempt #1

\[
\begin{align*}
\text{start} & \quad | \quad .7 & .2 & .1 \\
N & \quad | \quad .15 & .8 & .05 \\
V & \quad | \quad .6 & .35 & .05 \\
\end{align*}
\]

\[
\begin{array}{cccc}
w_1 & w_2 & w_3 & w_4 \\
.7 & .2 & .05 & .05 \\
.2 & .6 & .1 & .1 \\
\end{array}
\]

\[
\begin{align*}
z_1 & = N \\
z_2 & = V \\
z_3 & = N \\
z_4 & = N \\
\end{align*}
\]
Remember: these are only two of the 16 paths through the trellis.
Reusing Computation: Attempt #1

Remember: these are only two of the 16 paths through the trellis.

<table>
<thead>
<tr>
<th></th>
<th>N</th>
<th>V</th>
<th>end</th>
</tr>
</thead>
<tbody>
<tr>
<td>start</td>
<td>.7</td>
<td>.2</td>
<td>.1</td>
</tr>
<tr>
<td>N</td>
<td>.15</td>
<td>.8</td>
<td>.05</td>
</tr>
<tr>
<td>V</td>
<td>.6</td>
<td>.35</td>
<td>.05</td>
</tr>
</tbody>
</table>

\[
a_{N}[2, V] = (.7 \times .7) \times (.8 \times .6)
\]

\[
a_{N}[2, V] = (.4 \times .1)
\]

\[
a_{N}[2, V] = (.6 \times .05)
\]
Remember: these are only two of the 16 paths through the trellis.

<table>
<thead>
<tr>
<th>N</th>
<th>V</th>
<th>end</th>
</tr>
</thead>
<tbody>
<tr>
<td>start</td>
<td>.7</td>
<td>.2</td>
</tr>
<tr>
<td>N</td>
<td>.15</td>
<td>.8</td>
</tr>
<tr>
<td>V</td>
<td>.6</td>
<td>.35</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>w_1</th>
<th>w_2</th>
<th>w_3</th>
<th>w_4</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>.7</td>
<td>.2</td>
<td>.05</td>
</tr>
<tr>
<td>V</td>
<td>.2</td>
<td>.6</td>
<td>.1</td>
</tr>
</tbody>
</table>

These \( a_p(t, s) \) values represent the probability of a state \( s \) at time \( t \) in a particular shared path \( P \).
Reusing Computation: Attempt #1

Remember: these are only two of the 16 paths through the trellis

\[
\begin{align*}
   a_{N}[2, V] &= a[1, V] * (.8* .6) \\
   a_{N}[2, V] &= a_{N}[2, V] * (.4* .1)
\end{align*}
\]

These \(a_p(t, s)\) values represent the probability of a state \(s\) at time \(t\) in a particular shared path \(P\)
Reusing Computation: Attempt #1

Remember: these are only two of the 16 paths through the trellis.

How do we handle the other 14 paths?

Marginalize at each time step.
Reusing Computation: Attempt #1

so far, we’ve only considered “particular shared path (AB) → *”

\[ a_{AB}(i, \*) = a_A(i-1, B) \times p(*) \mid B\times p(\text{obs at } i \mid \*) \]

what about any of the other paths (AAB, ACB, etc.)?
Reusing Computation: Attempt #2

let’s first consider “any shared path ending with B (AB, BB, or CB) → B”
Reusing Computation: Attempt #2

let’s first consider “any shared path ending with B (AB, BB, or CB) → B”
marginalize across the previous hidden state values

\[ \alpha(i, B) = \sum_s \alpha(i - 1, s) \times p(B | s) \times p(\text{obs at } i | B) \]
Reusing Computation: Attempt #2

Let's first consider "any shared path ending with B (AB, BB, or CB) → B" marginalize across the previous hidden state values.

\[ a_{AB}(i, B) = a_A(i-1, B) \times p(B \mid B) \times p(\text{obs at } i \mid B) \]

\[ \alpha(i, B) = \sum_s \alpha(i - 1, s) \times p(B \mid s) \times p(\text{obs at } i \mid B) \]
let’s first consider “*any* shared path ending with B (AB, BB, or CB) → B” 
marginalize across the previous hidden state values

\[
a_{AB}(i, B) = a_A(i-1, B) \cdot p(B \mid B) \cdot p(\text{obs at } i \mid B)
\]

\[
\alpha(i, B) = \sum_s \alpha(i - 1, s) \cdot p(B \mid s) \cdot p(\text{obs at } i \mid B)
\]

computing \( \alpha \) at time \( i-1 \) will correctly incorporate paths through time \( i-2 \): we correctly obey the Markov property.
Forward Probability

Let's first consider “any shared path ending with B (AB, BB, or CB) → B” marginalize across the previous hidden state values

\[ a_{AB}(i, B) = a_A(i-1, B) \times p(B | B) \times p(\text{obs at } i | B) \]

\[ \alpha(i, B) = \sum_{s'} \alpha(i - 1, s') \times p(B | s') \times p(\text{obs at } i | B) \]

\( \alpha(i, B) \) is the total probability of all paths to that state B from the beginning

Computing \( \alpha \) at time \( i-1 \) will correctly incorporate paths through time \( i-2 \): we correctly obey theMarkov property
Forward Probability

\[
\alpha(i, s) = \sum_{s'} \alpha(i - 1, s') \times p(s | s') \times p(\text{obs at } i \mid s)
\]

\(\alpha(i, s)\) is the total probability of all paths:
1. that start from the beginning
2. that end (currently) in \(s\) at step \(i\)
3. that emit the observation \(\text{obs at } i\)
Forward Probability

\[ \alpha(i, s) = \sum_{s'} \alpha(i - 1, s') \times p(s | s') \times p(\text{obs at } i | s) \]

**What are the immediate ways to get into state \( s \)?**

**What’s the total probability up until now?**

**How likely is it to get into state \( s \) this way?**

\( \alpha(i, s) \) is the total probability of all paths:

1. that start from the beginning
2. that end (currently) in \( s \) at step \( i \)
3. that emit the observation \( \text{obs at } i \)
2 (3) - State HMM Likelihood with Forward Probabilities

\[ \alpha[3, V] = \alpha[2, V] \times (0.35 \times 0.1) + \alpha[2, N] \times (0.8 \times 0.1) \]

\[ \alpha[2, V] = \alpha[1, N] \times (0.8 \times 0.6) + \alpha[1, V] \times (0.35 \times 0.6) \]

\[ \alpha[1, N] = (0.7 \times 0.7) \]

\[ p(V | \text{start}) = \alpha[1, V] \]

\[ p(N | \text{start}) = \alpha[1, N] \]

\[ p(w_1 | N) = \alpha[1, N] \times (0.7 \times 0.7) \]

\[ p(w_2 | V) = \alpha[2, V] \]

\[ p(w_3 | V) = \alpha[3, V] \]

\[ p(w_4 | N) = \alpha[2, N] \times (0.8 \times 0.1) + \alpha[2, V] \times (0.6 \times 0.05) \]

\[ \alpha[3, N] = \alpha[2, V] \times (0.6 \times 0.05) + \alpha[2, N] \times (0.15 \times 0.05) \]

\[ p(N | N) = \alpha[3, N] \]

\[ p(N | V) = \alpha[3, V] \]

\[ p(w_3 | N) = \alpha[3, N] \]

\[ p(w_4 | N) = \alpha[2, N] \times (0.8 \times 0.1) + \alpha[2, V] \times (0.6 \times 0.05) \]
2 (3) - State HMM Likelihood with Forward Probabilities

\[
\begin{align*}
\alpha[1, V] &= (.2 \times .2) \\
\alpha[1, N] &= (.7 \times .7) \\
\alpha[2, V] &= \alpha[1, N] \times (.8 \times .6) + \alpha[1, V] \times (.35 \times .6) \\
\alpha[2, N] &= \alpha[1, V] \times (.8 \times .1) + \alpha[2, N] \times (.15 \times .05) \\
\alpha[3, V] &= \alpha[2, V] \times (.35 \times .1) + \alpha[2, N] \times (.8 \times .1) \\
\alpha[3, N] &= \alpha[2, V] \times (.6 \times .05) + \alpha[2, N] \times (.15 \times .05)
\end{align*}
\]
2 (3) - State HMM Likelihood with Forward Probabilities

\[ \alpha[1, V] = (0.2 \times 0.2) = 0.04 \]
\[ \alpha[1, N] = (0.7 \times 0.7) = 0.49 \]
\[ \alpha[2, V] = \alpha[1, N] \times (0.8 \times 0.6) + \alpha[1, V] \times (0.35 \times 0.6) = 0.2436 \]
\[ \alpha[2, N] = \alpha[1, N] \times (0.15 \times 0.2) + \alpha[1, V] \times (0.6 \times 0.2) = 0.0195 \]

### Transition Probabilities

<table>
<thead>
<tr>
<th></th>
<th>N</th>
<th>V</th>
<th>end</th>
</tr>
</thead>
<tbody>
<tr>
<td>start</td>
<td>0.7</td>
<td>0.2</td>
<td>0.1</td>
</tr>
<tr>
<td>N</td>
<td>0.15</td>
<td>0.8</td>
<td>0.05</td>
</tr>
<tr>
<td>V</td>
<td>0.6</td>
<td>0.35</td>
<td>0.05</td>
</tr>
</tbody>
</table>

### Emission Probabilities

<table>
<thead>
<tr>
<th></th>
<th>w₁</th>
<th>w₂</th>
<th>w₃</th>
<th>w₄</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>0.7</td>
<td>0.2</td>
<td>0.05</td>
<td>0.05</td>
</tr>
<tr>
<td>V</td>
<td>0.2</td>
<td>0.6</td>
<td>0.1</td>
<td>0.1</td>
</tr>
</tbody>
</table>
Use dynamic programming to build the $\alpha$ left-to-right.
Forward Algorithm

\( \alpha \): a 2D table, \( N+2 \times K^* \)

\( N+2 \): number of observations (+2 for the BOS & EOS symbols)

\( K^* \): number of states

Use dynamic programming to build the \( \alpha \) left-to-right
Forward Algorithm

\[ \alpha = \text{double}[N+2][K^*] \]
\[ \alpha[0][*] = 1.0 \]

for(i = 1; i \leq N+1; ++i) {

}
Forward Algorithm

\( \alpha = \text{double} [N+2][K^*] \)
\( \alpha[0][*] = 1.0 \)

for( \( i = 1; \ i \leq N+1; \ ++i \) ) {
    for( \( \text{state} = 0; \ \text{state} < K^*; \ ++\text{state} \) ) {
        
    }
}

}
Forward Algorithm

\( \alpha = \text{double}[N+2][K^*] \)
\( \alpha[0][*] = 1.0 \)

for(\( i = 1; \ i \leq N+1; \ ++i \) ) {
    for(\( \text{state} = 0; \ \text{state} < K^*; \ ++\text{state} \) ) {
        \( p_{\text{obs}} = p_{\text{emission}}(\text{obs}_i \mid \text{state}) \)
    }
}
Forward Algorithm

\[ \alpha = \text{double}[N+2][K^*] \]
\[ \alpha[0][*] = 1.0 \]

for \( i = 1; i \leq N+1; ++i \) {
    for (state = 0; state < K*; ++state) {
        \[ p_{\text{obs}} = p_{\text{emission}}(o_{\text{obs}_i} \mid \text{state}) \]
        for (old = 0; old < K*; ++old) {
            \[ p_{\text{move}} = p_{\text{transition}}(\text{state} \mid \text{old}) \]
            \[ \alpha[i][\text{state}] += \alpha[i-1][\text{old}] \times p_{\text{obs}} \times p_{\text{move}} \]
        }
    }
}
Forward Algorithm

\( \alpha = \text{double}[N+2][K^*] \)

\( \alpha[0][*] = 1.0 \)

\[
\begin{align*}
\text{for}(i = 1; i \leq N+1; ++i) \{ \\
\quad \text{for}(\text{state} = 0; \text{state} < K^*; ++\text{state}) \{ \\
\quad \quad p_{\text{obs}} = p_{\text{emission}}(\text{obs}_i \mid \text{state}) \\
\quad \quad \text{for}(\text{old} = 0; \text{old} < K^*; ++\text{old}) \{ \\
\quad \quad \quad p_{\text{move}} = p_{\text{transition}}(\text{state} \mid \text{old}) \\
\quad \quad \quad \alpha[i][\text{state}] += \alpha[i-1][\text{old}] \times p_{\text{obs}} \times p_{\text{move}} \\
\quad \quad \} \\
\quad \} \\
\} 
\end{align*}
\]

we still need to learn these (EM)
Forward Algorithm

\[ \alpha = \text{double}[N+2][K^*] \]
\[ \alpha[0][*] = 1.0 \]

for \( i = 1; i \leq N+1, ++i \) {
    for (state = 0; state < K*; ++state) {
        \[ p_{\text{obs}} = p_{\text{emission}}(\text{obs}_i \mid \text{state}) \]
        for (old = 0; old < K*; ++old) {
            \[ p_{\text{move}} = p_{\text{transition}}(\text{state} \mid \text{old}) \]
            \[ \alpha[i][\text{state}] += \alpha[i-1][\text{old}] \times p_{\text{obs}} \times p_{\text{move}} \]
        }
    }
}
Forward Algorithm

\[ \alpha = \text{double}[N+2][K^*] \]
\[ \alpha[0][*] = 1.0 \]

\[
\text{for}(i = 1; i \leq N+1; ++i) \{
    \text{for}(\text{state} = 0; \text{state} < K^*; ++\text{state}) \{
        p_{\text{obs}} = p_{\text{emission}}(\text{obs}_i | \text{state})
        \text{for}(\text{old} = 0; \text{old} < K^*; ++\text{old}) \{
            p_{\text{move}} = p_{\text{transition}}(\text{state} | \text{old})
            \alpha[i][\text{state}] += \alpha[i-1][\text{old}] \times p_{\text{obs}} \times p_{\text{move}}
        } 
    } 
\}
\]

\textbf{Q: What do we return? (How do we return the likelihood of the sequence?)}

\textbf{A: } \alpha[N+1][\text{end}]
Interactive HMM Example


Original: http://www.cs.jhu.edu/~jason/465/PowerPoint/lect24-hmm.xls
Forward Algorithm in Log-Space

\[ \alpha = \text{double}[N+2][K^*] \]
\[ \alpha[0][*] = 0.0 \]

\[
\text{for}(i = 1; i \leq N+1; ++i) \{ \\
\quad \text{for}(\text{state} = 0; \text{state} < K^*; ++\text{state}) \{ \\
\quad \quad p_{\text{obs}} = \log p_{\text{emission}}(\text{obs}_i \mid \text{state}) \\
\quad \quad \text{for}(\text{old} = 0; \text{old} < K^*; ++\text{old}) \{ \\
\quad \quad \quad p_{\text{move}} = \log p_{\text{transition}}(\text{state} \mid \text{old}) \\
\quad \quad \quad \alpha[i][\text{state}] = \logadd(\alpha[i][\text{state}], \\
\quad \quad \quad \quad \alpha[i-1][\text{old}] + p_{\text{obs}} + p_{\text{move}}) \\
\quad \quad \} \\
\quad \} \\
\} 
\]
Forward Algorithm in Log-Space

\[
\alpha = \text{double}[N+2][K^*]
\]
\[
\alpha[0][*] = 0.0
\]

for (i = 1; i ≤ N+1; ++i) {
    for (state = 0; state < K*; ++state) {
        \[ p_{\text{obs}} = \log p_{\text{emission}}(\text{obs}_i \mid \text{state}) \]
        for (old = 0; old < K*; ++old) {
            \[ p_{\text{move}} = \log p_{\text{transition}}(\text{state} \mid \text{old}) \]
            \[ \alpha[i][state] = \logadd(\alpha[i][state], \alpha[i-1][old] + p_{\text{obs}} + p_{\text{move}}) \]
        }
    }
}

\[
\logadd(lp, lq) = \log(\exp(lp) + \exp(lq))
\]
Forward Algorithm in Log-Space

\[ \alpha = \text{double}[N+2][K^*] \]
\[ \alpha[0][*] = 0.0 \]

for \( i = 1; i \leq N+1; ++i \) {
    for (state = 0; state < K*; ++state) {
        \[ p_{\text{obs}} = \log p_{\text{emission}}(\text{obs}_i \mid \text{state}) \]
        for (old = 0; old < K*; ++old) {
            \[ p_{\text{move}} = \log p_{\text{transition}}(\text{state} \mid \text{old}) \]
            \[ \alpha[i][\text{state}] = \logadd(\alpha[i][\text{state}], \alpha[i-1][\text{old}] + p_{\text{obs}} + p_{\text{move}}) \]
        }
    }
    \[ \logadd(lp, lq) = \log(\exp(lp) + \exp(lq)) \]
Forward Algorithm in Log-Space

\[ \alpha = \text{double}[N+2][K^*] \]
\[ \alpha[0][*] = 0.0 \]

\[
\text{for}(i = 1; i \leq N+1; ++i) \{ \\
\quad \text{for}(\text{state} = 0; \text{state} < K^*; ++\text{state}) \{ \\
\quad\quad p_{\text{obs}} = \log p_{\text{emission}}(\text{obs}_i \mid \text{state}) \\
\quad\quad \text{for}(\text{old} = 0; \text{old} < K^*; ++\text{old}) \{ \\
\quad\quad\quad p_{\text{move}} = \log p_{\text{transition}}(\text{state} \mid \text{old}) \\
\quad\quad\quad \alpha[i][\text{state}] = \text{logadd}(\alpha[i][\text{state}], \alpha[i-1][\text{old}] + p_{\text{obs}} + p_{\text{move}}) \\
\quad\quad \} \\
\quad \} \\
\}
\]

\[
\text{logadd}(lp, lq) = \begin{cases} 
lp + \log(1 + \exp(lq)), & \quad lp \geq lq \\
lq + \log(1 + \exp(lp)), & \quad lq > lp
\end{cases}
\]
Forward Algorithm in Log-Space

\[ \alpha = \text{double}[N+2][K^*] \]
\[ \alpha[0][*] = 0.0 \]

for \( i = 1; i \leq N+1; ++i \) {
    for(state = 0; state < K*; ++state) {
        \[ p_{\text{obs}} = \log p_{\text{emission}}(\text{obs}_i \mid \text{state}) \]
        for(old = 0; old < K*; ++old) {
            \[ p_{\text{move}} = \log p_{\text{transition}}(\text{state} \mid \text{old}) \]
            \[ \alpha[i][state] = \text{logadd}(\alpha[i][state], \alpha[i-1][old] + p_{\text{obs}} + p_{\text{move}}) \]
        }
    }
}

\[ \text{logadd}(lp, lq) = \begin{cases} 
lp + \log(1 + \exp(lq)), & \text{if } lp \geq lq \\
lp + \log(1 + \exp(lp)), & \text{if } lq > lp
\end{cases} \]