POS Tagging
Penn Treebank English POS tagging (see homework): 44 tags

Baseline: assign each word its most frequent tag: ~90% accuracy

Trigram HMM (model pairs of tags): ~95% accuracy / 55% on words not seen in train

TnT tagger (Brants 1998, tuned HMM): 96.2% acc / 86.0% on unks

CRF tagger (Toutanova + Manning 2000): 96.9% / 87.0%

State-of-the-art (BiLSTM-CRFs, BERT): 97.5% / 89%+

Slide credit: Dan Klein
## Errors

<table>
<thead>
<tr>
<th>JJ</th>
<th>NN</th>
<th>NNP</th>
<th>NNPS</th>
<th>RB</th>
<th>RP</th>
<th>IN</th>
<th>VB</th>
<th>VBD</th>
<th>VBN</th>
<th>VBP</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>JJ</td>
<td>0</td>
<td>177</td>
<td>56</td>
<td>0</td>
<td>61</td>
<td>2</td>
<td>5</td>
<td>10</td>
<td>15</td>
<td>108</td>
<td>0</td>
</tr>
<tr>
<td>NN</td>
<td>244</td>
<td>0</td>
<td>103</td>
<td>0</td>
<td>12</td>
<td>1</td>
<td>1</td>
<td>29</td>
<td>5</td>
<td>6</td>
<td>19</td>
</tr>
<tr>
<td>NNP</td>
<td>107</td>
<td>106</td>
<td>0</td>
<td>132</td>
<td>5</td>
<td>0</td>
<td>7</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>NNPS</td>
<td>1</td>
<td>0</td>
<td>110</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>RB</td>
<td>72</td>
<td>21</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>16</td>
<td>138</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>RP</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>39</td>
<td>0</td>
<td>65</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>IN</td>
<td>11</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>169</td>
<td>103</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>VB</td>
<td>17</td>
<td>64</td>
<td>9</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>4</td>
<td>7</td>
<td>85</td>
</tr>
<tr>
<td>VBD</td>
<td>10</td>
<td>5</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>143</td>
<td>2</td>
</tr>
<tr>
<td>VBN</td>
<td>101</td>
<td>3</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>108</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>VBP</td>
<td>5</td>
<td>34</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>49</td>
<td>6</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>Total</td>
<td>626</td>
<td>536</td>
<td>348</td>
<td>144</td>
<td>317</td>
<td>122</td>
<td>279</td>
<td>102</td>
<td>140</td>
<td>269</td>
<td>108</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>JJ/NN</th>
<th>NN</th>
</tr>
</thead>
<tbody>
<tr>
<td>official knowledge</td>
<td>VBD RP/IN DT NN made up the story recently sold shares</td>
</tr>
</tbody>
</table>

(NN NN: tax cut, art gallery, ...)

Slide credit: Dan Klein / Toutanova + Manning (2000)
Remaining Errors

- Lexicon gap (word not seen with that tag in training): 4.5% of errors
- Unknown word: 4.5%
- Could get right: 16% (many of these involve parsing!)
- Difficult linguistics: 20%

VBD / VBP? (past or present?)

They set up absurd situations, detached from reality

Underspecified / unclear, gold standard inconsistent / wrong: 58%

adjective or verbal participle? JJ / VBN?

a $10 million fourth-quarter charge against discontinued operations

Manning 2011 “Part-of-Speech Tagging from 97% to 100: Is It Time for Some Linguistics?”
CRFs and NER
Barack Obama will travel to Hangzhou today for the G20 meeting.

- Frame as a sequence problem with a BIO tagset: begin, inside, outside
- Why might an HMM not do so well here?
  - Lots of O’s, so tags aren’t as informative about context
  - Want to use context features (to Hangzhou => Hangzhou is a LOC)
- Conditional random fields (CRFs) can help solve these problems
HMMs

- Big advantage: transitions, scoring pairs of adjacent y’s

- Big downside: not able to incorporate useful word context information

- Solution: switch from generative to discriminative model (conditional random fields) so we can condition on the entire input.

- Conditional random fields: logistic regression + features on pairs of y’s
Tagging with Logistic Regression

- Logistic regression over each tag individually:
  \[ P(y_i = y | x, i) = \frac{\exp(w^T f(y, i, x))}{\sum_{y' \in \mathcal{Y}} \exp(w^T f(y', i, x))} \]

- Over all tags:
  \[ P(y = \tilde{y} | x) = \prod_{i=1}^{n} P(y_i = \tilde{y}_i | x, i) = \frac{1}{Z} \exp \left( \sum_{i=1}^{n} w^T f(\tilde{y}_i, i, x) \right) \]

- Score of a prediction: sum of weights dot features over each individual predicted tag (this is a simple CRF but not the general form)

- Set Z equal to the product of denominators; we’ll discuss this in a few slides

“different features” approach to features for a single tag
Example

B-PER  I-PER  O  O
Barack Obama will travel
feats = f_e(B-PER, i=1, x) + f_e(I-PER, i=2, x) + f_e(O, i=3, x) + f_e(O, i=4, x)

B-PER  B-PER  O  O
Barack Obama will travel
feats = f_e(B-PER, i=1, x) + f_e(B-PER, i=2, x) + f_e(O, i=3, x) + f_e(O, i=4, x)
Adding Structure

\[ P(y = \tilde{y}|x) = \frac{1}{Z} \exp \left( \sum_{i=1}^{n} w^\top f(\tilde{y}_i, i, x) \right) \]

- We want to be able to learn that some tags don’t follow other tags — want to have features on tag pairs

\[ P(y = \tilde{y}|x) = \frac{1}{Z} \exp \left( \sum_{i=1}^{n} w^\top f_e(\tilde{y}_i, i, x) + \sum_{i=1}^{n} w^\top f_t(\tilde{y}_i, \tilde{y}_{i+1}, i, x) \right) \]

- Score: sum of weights dot \( f_e \) features over each predicted tag (“emissions”) plus sum of weights dot \( f_t \) features over tag pairs (“transitions”)

- This is a sequential CRF
Example

\[ \text{feats} = f_e(\text{B-PER}, i=1, x) + f_e(\text{I-PER}, i=2, x) + f_e(\text{O}, i=3, x) + f_e(\text{O}, i=4, x) \\
+ f_t(\text{B-PER}, \text{I-PER}, i=1, x) + f_t(\text{I-PER}, \text{O}, i=2, x) + f_t(\text{O}, \text{O}, i=3, x) \]

Obama can start a new named entity (emission feats look okay), but we’re not likely to have two PER entities in a row (transition feats)
Features for NER

\[
P(y = \tilde{y} | x) = \frac{1}{Z} \exp \left( \sum_{i=1}^{n} w^T f_e(\tilde{y}_i, i, x) + \sum_{i=1}^{n} w^T f_t(\tilde{y}_i, \tilde{y}_{i+1}, i, x) \right)
\]

Barack Obama will travel to **Hangzhou** today for the G20 meeting.

Transitions: \( f_t(O, \text{B-LOC}, i = 5, x) = \text{Indicator}[O \rightarrow \text{B-LOC}] \)

Emissions: \( f_e(\text{B-LOC}, i = 6, x) = \text{Indicator}[\text{B-LOC \& Curr word = Hangzhou}] \)

- Indicator[\text{B-LOC \& Prev word = to}]

- We couldn’t use a “previous word” feature in the HMM at all!
Conditional Random Fields

\[ P(y = \tilde{y} | x) = \frac{1}{Z} \exp \left( \sum_{i=1}^{n} w^\top f_e(\tilde{y}_i, i, x) + \sum_{i=1}^{n} w^\top f_t(\tilde{y}_i, \tilde{y}_{i+1}, i, x) \right) \]

normalizer \( Z \): must make this a probability distribution over all possible seqs

\[ Z = \sum_{y' \in \mathcal{Y}^n} \exp \left( \sum_{i=1}^{n} w^\top f_e(y'_i, i, x) + \sum_{i=1}^{n} w^\top f_t(y'_i, y'_{i+1}, i, x) \right) \]
Inference and Learning

\[ P(y = \tilde{y} \mid x) = \frac{1}{Z} \exp \left( \sum_{i=1}^{n} w^\top f_e(\tilde{y}_i, i, x) + \sum_{i=1}^{n} w^\top f_t(\tilde{y}_i, \tilde{y}_{i+1}, i, x) \right) \]

- **Inference**: Can use the Viterbi algorithm to find the highest scoring path. Replace HMM log probs with “scores” from weights dot features

  \[
  \log P(x_i \mid y_i) \rightarrow w^\top f_e(y_i, i, x) \quad \text{(initial distribution is removed)}
  \]

  \[
  \log P(y_i \mid y_{i-1}) \rightarrow w^\top f_t(y_{i-1}, y_i, i, x)
  \]

- **Learning**: requires running *forward-backward* (like Viterbi but with summing instead of maxing over y’s) to compute Z, then doing some tricky math to compute gradients [outside scope of the course/not on midterm]
**Takeaways**

- CRFs provide a way to build structured feature-based models: logistic regression over structured objects like sequences.

- Inference and learning can still be done efficiently but require dynamic programming.

- CRFs don’t have to be linear models; can use scores derived from neural networks (“neural CRFs”).