ITCS 4111/5111: Introduction to NLP

Sequence Labeling for POS Tagging and NE Recognition

Razvan C. Bunescu
Department of Computer Science @ CCI
rbunescu@uncc.edu
Part of Speech (POS) Tagging

- Annotate each word in a sentence with its POS:
  - noun, verb, adjective, adverb, pronoun, preposition, interjection, ...

  She promised to back the bill
Part of Speech (POS) Tagging

• Annotate each word in a sentence with its POS:
  – noun, verb, adjective, adverb, pronoun, preposition, interjection, …

PRP  VBD  TO   VB   TO   DT   NN   IN   NN   VBD   VBG
OBJECT

They used to object to the use of object oriented programming

• Useful for many other NLP tasks:
  – speech recognition and synthesis
  – syntactic parsing
  – word sense disambiguation
  – information retrieval, …
Parts of Speech

• Lexical categories that are defined based on:
  – **Syntactic function**:
    • nouns can occur with determiners: a *goat*.
    • nouns can take possessives: IBM’s annual revenue.
    • most nouns can occur in the plural: *goats*.
  – **Morphological function**:
    • many verbs can be composed with the prefix “un”.

• There are tendencies toward **semantic coherence**:
  – nouns often refer to “people, places, or things”.
  – adjectives often refer to properties.
• **Closed Class:**
  – relatively fixed membership.
  – usually **function words**:
    • short common words which have a structuring role in grammar.
    – **Prepositions**: of, in, by, on, under, over, …
    – **Auxiliaries**: may, can, will, had, been, should, …
    – **Pronouns**: I, you, she, mine, his, them, …
    – **Determiners**: a, an, the, which, that, …
    – **Conjunctions**: and, but, or (coord.), as, if, when, (subord.), …
    – **Particles**: up, down, on, off, …
    – **Numerals**: one, two, three, third, …
POS: Open Class vs. Closed Class

• **Open Class:**
  – new members are continually added.
    • *to fax, to google, futon,* …
  – English has 4: **Nouns, Verbs, Adjectives, Adverbs.**
    • Many languages have these 4, but not all (e.g. Korean).
  – **Nouns:** people, places, or things
  – **Verbs:** actions and processes
  – **Adjectives:** properties or qualities
  – **Adverbs:** a hodge-podge
    • *Unfortunately, John walked home extremely slowly yesterday.*
    • directional, locative, temporal, degree, manner, …
Parts of Speech: Granularity

• Grammatical sketch of Greek [Dionysius Thrax, c. 100 B.C.]:
  – 8 tags: noun, verb, pronoun, preposition, adjective, conjunction, participle, and article.
• Brown corpus [Francis, 1979]:
  – 87 tags.
• Penn Treebank [Marcus et al., 1993]:
  – 45 tags.
• British National Corpus (BNC) [Garside et al., 1997]:
  – C5 tagset: 61 tags.
  – C7 tagset: 146 tags.

We will focus on the Penn Treebank POS tags.
## Penn Treebank POS Tagset

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

• Selected from the original 87 tags of the Brown corpus:
  ⇒ lost finer distinctions between lexical categories.

1) Prepositions and subordinating conjunctions:
   – after/CS spending/VBG a/AT day/NN at/IN the/AT palace/NN
   – after/IN a/AT wedding/NN trip/NN to/IN Hawaii/NNP ./.

2) Infinitive to and prepositional to:
   – to/TO give/VB priority/NN to/IN teachers/NNS

3) Adverbial nouns:
   – Brown: Monday/NR, home/NR, west/NR, tomorrow/NR
   – PTB: Monday/NNP, (home, tomorrow, west)/(NN, RB)
POS Tagging ≡ POS Disambiguation

• Words often have more than one POS tag, e.g. back:
  – the back/JJ door
  – on my back/NN
  – win the voters back/RB
  – promised to back/VB the bill

• Brown corpus statistics [DeRose, 1998]:
  – 11.5% ambiguous English word types.
  – 40% of all word occurrences are ambiguous.
    • most are easy to disambiguate
      – the tags are not equally likely, i.e. low tag entropy: table
POS Tag Ambiguity

<table>
<thead>
<tr>
<th></th>
<th>87-tag Original Brown</th>
<th>45-tag Treebank Brown</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unambiguous (1 tag)</td>
<td>44,019</td>
<td>38,857</td>
</tr>
<tr>
<td>Ambiguous (2–7 tags)</td>
<td>5,490</td>
<td>8844</td>
</tr>
<tr>
<td>Details:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 tags</td>
<td>4,967</td>
<td>6,731</td>
</tr>
<tr>
<td>3 tags</td>
<td>411</td>
<td>1621</td>
</tr>
<tr>
<td>4 tags</td>
<td>91</td>
<td>357</td>
</tr>
<tr>
<td>5 tags</td>
<td>17</td>
<td>90</td>
</tr>
<tr>
<td>6 tags</td>
<td>2 (well, beat)</td>
<td>32</td>
</tr>
<tr>
<td>7 tags</td>
<td>2 (still, down)</td>
<td>6 (well, set, round, open, fit, down)</td>
</tr>
<tr>
<td>8 tags</td>
<td></td>
<td>4 (’s, half, back, a)</td>
</tr>
<tr>
<td>9 tags</td>
<td></td>
<td>3 (that, more, in)</td>
</tr>
</tbody>
</table>
POS Disambiguation: Form & Context

• **Open Class**: new members are continually added.

1. Annie: Do you love me?
   Alvy: Love is too weak a word for what I feel... I **lurve** you. Y'know, I **loove** you, I, I **luff** you. There are two f's. I have to invent... Of course I love you. *(Annie Hall)*

2. 'Twas brillig, and the slithy toves
   Did gyre and gimble in the wabe;
   All mimsy were the borogoves,
   And the mome raths outgrabe.

   "Beware the Jabberwock, my son!
   The jaws that bite, the claws that catch!
   Beware the Jubjub bird, and shun
   The frumious Bandersnatch!"

   *(Jabberwocky, Lewis Caroll)*
POS Disambiguation:
Context overriding Form

“Here's a movie where you forgive the preposterous because it takes you to the perplexing.”

[Source Code, by Roger Ebert, March 31, 2011]

“The good, the bad, and the ugly”

“The young and the restless”

“The bold and the beautiful”
POS Tagging ≡ POS Disambiguation

• Some distinctions are difficult even for humans:
  – Mrs. Shaefer never got around to joining
    NNP  NNP   RB  VBD  RP  TO  VBG
  – All we gotta do is go around the corner
    DT  PRP  VBN  VB  VBZ  VB  IN  DT  NN
  – Chateau Petrus costs around 250
    NNP  NNP  VBZ   RB  CD

• Use heuristics [Santorini, 1990]:
  – She told off/RP her friends
  – She told her friends off/RP
    She stepped off/IN the train
    *She stepped the train off/IN
How Difficult is POS Tagging?

- Most current tagging algorithms: ~ 96% – 97% accuracy for Penn Treebank tagsets.
  - 2015 SofA 97.55% tagging accuracy. How good is this?
    - Bidirectional LSTM-CRF Models for Sequence Tagging [Huang, Xu, Yu, 2015].
  - **Human Ceiling**: how well humans do?
    - human annotators: about 96% – 97% [Marcus et al., 1993].
    - when allowed to discuss tags, consensus is 100% [Voutilainen, 95]
  - **Most Frequent Class Baseline**:
    - 90% – 91% on the 87-tag Brown tagset [Charniak et al., 1993].
    - 93.69% on the 45-tag Penn Treebank, with unknown word model [Toutanova et al., 2003].
POS Tagging Methods

• **Rule Based:**
  – Rules are designed by human experts based on linguistic knowledge.

• **Machine Learning:**
  – Trained on data that has been manually labeled by humans.
  – Rule learning:
    • **Transformation Based Learning (TBL).**
    – **Sequence tagging:**
      • **Maximum Entropy (Logistic Regression).**
      • **Hidden Markov Models (HMMs).**
      • **(Sequential) Conditional Random Fields (CRFs).**
      • **Recurrent Neural Networks (RNNs):**
        – bidirectional, with a CRF layer (BI-LSTM-CRF).
POS Tagging: Rule Based

1) Start with a dictionary.

2) Assign all possible tags to words from the dictionary.

3) Write rules by hand to selectively remove tags, leaving the correct tag for each word.
POS Tagging: Rule Based

1) Start with a dictionary:

- **she:** PRP
- **promised:** VBN, VBD
- **to:** TO
- **back:** VB, JJ, RB, NN
- **the:** DT
- **bill:** NN, VB

... for the ~100,000 words of English.
2) Assign every possible tag:

NN
RB
VBN  JJ  VB
PRP  VBD  TO  VB  DT  NN
She  promised  to  back  the  bill
3) Write rules to eliminate incorrect tags.
   
   - Eliminate VBN if VBD is an option when VBN|VBD follows “<S> PRP”

```
She promised to back the bill
```

<table>
<thead>
<tr>
<th>NN</th>
<th>RB</th>
<th>VBN</th>
<th>JJ</th>
<th>VB</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRP</td>
<td>VBD</td>
<td>TO</td>
<td>VB</td>
<td>DT</td>
</tr>
</tbody>
</table>
POS Tagging as Sequence Labeling

- **Sequence Labeling:**
  - Tokenization and Sentence Segmentation.
  - Part of Speech Tagging.
  - Information Extraction
    - Named Entity Recognition
  - Shallow Parsing.
  - Semantic Role Labeling.
  - DNA Analysis.
  - Music Segmentation.

- Solved using **ML models** for classification:
  - Token-level vs. Sequence-level.
(Character-Level) Sequence Labeling

- **Sentence Segmentation:**
  
  Mr. Burns is Homer Simpson’s boss. He is very rich.

- **Tokenization:**
  
  Mr. Burns is Homer Simpson’s boss. He is very rich.
Drug giant **Pfizer Inc.** has reached an agreement to buy the private biotechnology firm **Rinat Neuroscience Corp.**
Sequence Labeling

- **Information Extraction:**
  - **Text Segmentation** into topical sections.

Vine covered cottage, near Contra Costa Hills. 2 bedroom house,

modern kitchen and dishwasher. No pets allowed. $1050 / month

[Haghighi & Klein, NAACL ‘06]
Sequence Labeling

- **Information Extraction:**
  - segmenting classifieds into topical sections.

  Vine covered cottage, near Contra Costa Hills. 2 bedroom house,

  modern kitchen and dishwasher. No pets allowed. $1050/month

  [Haghighi & Klein, NAACL ‘06]

  - Features: B-F, I-F
  - Neighborhood: B-N, I-N
  - Size: B-S, I-S
  - Restrictions: B-Res, I-Res
  - Rent: B-Ren, I-Ren
  - O
Sequence Labeling

• **Semantic Role Labeling:**
  – For each clause, determine the semantic role played by each noun phrase that is an argument to the verb:

  John drove Mary from Athens to Columbus in his Toyota Prius. The hammer broke the window.

  • agent
  • patient
  • source
  • destination
  • instrument
Sequence Labeling

- **DNA Analysis:**
  - transcription factor binding sites.
  - promoters.
  - introns, exons, …

AATGC\textcolor{red}{GCTA}ACGT\textcolor{red}{TTCG}A\textcolor{red}{TAGATAGGCCTAAGAGGTCA}
Sequence Labeling

- **Music Analysis:**
  - segmentation into “musical phrases”

*Example musicscore*

[Romeo & Juliet, Nino Rota]
1) **Classify** each token **individually** into one of a number of classes:

- Token represented as a vector of features extracted from context.
- To build classification model, use general ML algorithms:
  - **Maximum Entropy** (i.e. **Logistic Regression**)
  - Support Vector Machines (SVMs)
  - Perceptrons.
  - Winnow.
  - Naïve Bayes, Bayesian Networks.
  - Decision Trees.
  - k-Nearest Neighbor.
  - Neural Networks (RNNs, Transformer, …)
A Maximum Entropy Model for POS Tagging

- Represent each position \( i \) in text as \( \varphi(t, h_i) = \{ \varphi_k(t, h_i) \} \):
  - \( t \) is the *potential* POS tag at position \( i \).
  - \( h_i \) is the history/context of position \( i \).

\[
h_i = \{ w_i, w_{i+1}, w_{i+2}, w_{i-1}, w_{i-2}, t_{i-1}, t_{i-2} \}
\]

- \( \varphi(t, h_i) \) is a vector of features \( \varphi_k(t, h_i) \), for \( k = 1..K \).

\[
\phi_k(t,h_i) = \begin{cases} 
1 & \text{if suffix}(w_i) = "ing" \& \ t = \text{VBG} \\
0 & \text{otherwise}
\end{cases}
\]

- Represent the “unnormalized” score of a tag \( t \) as:

\[
\text{score}(t, h_i) = w^T \phi(t, h_i) = \sum_{k=1}^{K} w_k \phi_k(t, h_i)
\]

want \( w_k \) to be large here
A Maximum Entropy Model for POS Tagging

[Ratnaparkhi, EMNLP’96]

<table>
<thead>
<tr>
<th>Condition</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>( w_i ) is not rare</td>
<td>( w_i = X ) &amp; ( t_i = T )</td>
</tr>
<tr>
<td>( w_i ) is rare</td>
<td>( X ) is prefix of ( w_i ), (</td>
</tr>
<tr>
<td></td>
<td>( X ) is suffix of ( w_i ), (</td>
</tr>
<tr>
<td></td>
<td>( w_i ) contains number &amp; ( t_i = T )</td>
</tr>
<tr>
<td></td>
<td>( w_i ) contains uppercase character &amp; ( t_i = T )</td>
</tr>
<tr>
<td></td>
<td>( w_i ) contains hyphen &amp; ( t_i = T )</td>
</tr>
<tr>
<td>( \forall \ w_i )</td>
<td>( t_{i-1} = X ) &amp; ( t_i = T )</td>
</tr>
<tr>
<td></td>
<td>( t_{i-2}t_{i-1} = XY ) &amp; ( t_i = T )</td>
</tr>
<tr>
<td></td>
<td>( w_{i-1} = X ) &amp; ( t_i = T )</td>
</tr>
<tr>
<td></td>
<td>( w_{i-2} = X ) &amp; ( t_i = T )</td>
</tr>
<tr>
<td></td>
<td>( w_{i+1} = X ) &amp; ( t_i = T )</td>
</tr>
<tr>
<td></td>
<td>( w_{i+2} = X ) &amp; ( t_i = T )</td>
</tr>
</tbody>
</table>

Table 1: Features on the current history \( h_i \)

<table>
<thead>
<tr>
<th>Word:</th>
<th>the</th>
<th>stories</th>
<th>about</th>
<th>well-heeled</th>
<th>communities</th>
<th>and</th>
<th>developers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tag:</td>
<td>DT</td>
<td>NNS</td>
<td>IN</td>
<td>JJ</td>
<td>NNS</td>
<td>CC</td>
<td>NNS</td>
</tr>
<tr>
<td>Position:</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
</tbody>
</table>

Table 2: Sample Data
A Maximum Entropy Model for POS Tagging

[ Ratnaparkhi, EMNLP’96 ]

<table>
<thead>
<tr>
<th>Word:</th>
<th>the stories about well-heeled communities and developers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tag:</td>
<td>DT   NNS   IN   JJ   NNS   CC   NNS</td>
</tr>
<tr>
<td>Position:</td>
<td>1 2 3 4 5 6 7</td>
</tr>
</tbody>
</table>

Table 2: Sample Data

<table>
<thead>
<tr>
<th>Condition</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>$w_i$ is not rare</td>
<td>$w_i = X$ &amp; $t_i = T$</td>
</tr>
<tr>
<td>$w_i$ is rare</td>
<td>$X$ is prefix of $w_i$, $</td>
</tr>
<tr>
<td></td>
<td>$X$ is suffix of $w_i$, $</td>
</tr>
<tr>
<td></td>
<td>$w_i$ contains number &amp; $t_i = T$</td>
</tr>
<tr>
<td></td>
<td>$w_i$ contains uppercase character &amp; $t_i = T$</td>
</tr>
<tr>
<td></td>
<td>$w_i$ contains hyphen &amp; $t_i = T$</td>
</tr>
<tr>
<td>$\forall w_i$</td>
<td>$t_{i-1} = X$ &amp; $t_i = T$</td>
</tr>
<tr>
<td></td>
<td>$t_{i-2}t_{i-1} = XY$ &amp; $t_i = T$</td>
</tr>
<tr>
<td></td>
<td>$w_{i-1} = X$ &amp; $t_i = T$</td>
</tr>
<tr>
<td></td>
<td>$w_{i-2} = X$ &amp; $t_i = T$</td>
</tr>
<tr>
<td></td>
<td>$w_{i+1} = X$ &amp; $t_i = T$</td>
</tr>
<tr>
<td></td>
<td>$w_{i+2} = X$ &amp; $t_i = T$</td>
</tr>
</tbody>
</table>

Table 1: Features on the current history $h_i$

$w_i = \text{about} \quad \& \quad t_i = \text{IN}$

$w_{i-1} = \text{stories} \quad \& \quad t_i = \text{IN}$

$w_{i-2} = \text{the} \quad \& \quad t_i = \text{IN}$

$w_{i+1} = \text{well-heeled} \quad \& \quad t_i = \text{IN}$

$w_{i+2} = \text{communities} \quad \& \quad t_i = \text{IN}$

$t_{i-1} = \text{NNS} \quad \& \quad t_i = \text{IN}$

$t_{i-2}t_{i-1} = \text{DT NNS} \quad \& \quad t_i = \text{IN}$

the non-zero features for position 3

feature templates
A Maximum Entropy Model for POS Tagging

Table 2: Sample Data

<table>
<thead>
<tr>
<th>Condition</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>$w_i$ is not rare</td>
<td>$w_i = X$ &amp; $t_i = T$</td>
</tr>
<tr>
<td>$w_i$ is rare</td>
<td>$X$ is prefix of $w_i$, $</td>
</tr>
<tr>
<td></td>
<td>$X$ is suffix of $w_i$, $</td>
</tr>
<tr>
<td></td>
<td>$w_i$ contains number &amp; $t_i = T$</td>
</tr>
<tr>
<td></td>
<td>$w_i$ contains uppercase character &amp; $t_i = T$</td>
</tr>
<tr>
<td></td>
<td>$w_i$ contains hyphen &amp; $t_i = T$</td>
</tr>
<tr>
<td>$\forall w_i$</td>
<td>$t_{i-1} = X$ &amp; $t_i = T$</td>
</tr>
<tr>
<td></td>
<td>$t_{i-2}t_{i-1} = XY$ &amp; $t_i = T$</td>
</tr>
<tr>
<td></td>
<td>$w_{i-1} = X$ &amp; $t_i = T$</td>
</tr>
<tr>
<td></td>
<td>$w_{i-2} = X$ &amp; $t_i = T$</td>
</tr>
<tr>
<td></td>
<td>$w_{i+1} = X$ &amp; $t_i = T$</td>
</tr>
<tr>
<td></td>
<td>$w_{i+2} = X$ &amp; $t_i = T$</td>
</tr>
</tbody>
</table>

Table 1: Features on the current history $h_i$

$w_{i-1} = \text{about} \quad \& \quad t_i = JJ$

$w_{i-2} = \text{stories} \quad \& \quad t_i = JJ$

$w_{i+1} = \text{communities} \quad \& \quad t_i = JJ$

$w_{i+2} = \text{and} \quad \& \quad t_i = JJ$

$t_{i-1} = \text{IN} \quad \& \quad t_i = JJ$

$t_{i-2}t_{i-1} = \text{NNS IN} \quad \& \quad t_i = JJ$

prefix($w_i$) = w \quad \& \quad t_i = JJ

prefix($w_i$) = we \quad \& \quad t_i = JJ

prefix($w_i$) = wel \quad \& \quad t_i = JJ

prefix($w_i$) = well \quad \& \quad t_i = JJ

suffix($w_i$) = d \quad \& \quad t_i = JJ

suffix($w_i$) = ed \quad \& \quad t_i = JJ

suffix($w_i$) = led \quad \& \quad t_i = JJ

suffix($w_i$) = eled \quad \& \quad t_i = JJ

$w_i$ contains hyphen \quad \& \quad t_i = JJ

the non-zero features for position 4
A Maximum Entropy Model for POS Tagging

• How do we learn the weights $w$?
  – Train on manually annotated data (supervised learning).

• What does it mean “train $w$ on annotated corpus”?
  – Probabilistic Discriminative Models:
    • Maximum Entropy (Logistic Regression). [Ratnaparkhi, EMNLP’96]
    • Conditional Random Fields (CRFs).
    • Neural Networks.
  – Discriminant-based Methods:
    • (Average) Perceptrons. [Collins, ACL 2002]
    • Support Vector Machines (SVMs).
A Maximum Entropy Model for POS Tagging

[Ratnaparkhi, EMNLP’96]

• Probabilistic Discriminative Model:
  \[ p(t | h_i) = \frac{\exp(w^T \phi(t, h_i))}{\sum_i \exp(w^T \phi(t', h_i))} \]

• Training using:
  – Maximum Likelihood (ML).
  – Maximum A Posteriori (MAP) with a Gaussian prior on \( w \).

• Inference (i.e. Testing):
  \[ \hat{t}_i = \arg \max_{t_i \in T} p(t_i | h_i) = \arg \max_{t_i \in T} \exp(w^T \phi(t_i, h_i)) = \arg \max_{t_i \in T} w^T \phi(t_i, h_i) \]
A Maximum Entropy Model for POS Tagging

• Inference, need to do Forward traversal of input sequence:

  [Animation by Ray Mooney, UT Austin]

John saw the saw and decided to take it to the table.

classifier

NNP
A Maximum Entropy Model for POS Tagging

• Inference, need to do Forward traversal of input sequence:

  [Animation by Ray Mooney, UT Austin]

NNP
John saw the saw and decided to take it to the table.

classifier

VBD
A Maximum Entropy Model for POS Tagging [Ratnaparkhi, EMNLP’96]

- Inference, need to do Forward traversal of input sequence:

NNP VBD
John saw the saw and decided to take it to the table.

Animation by Ray Mooney, UT Austin
A Maximum Entropy Model for POS Tagging

- Inference, need to do Forward traversal of input sequence:

  John saw the saw and decided to take it to the table.

NNP VBD DT
classifier
NN
A Maximum Entropy Model for POS Tagging

• Inference, need to do Forward traversal of input sequence:

NNP VBD DT NN
John saw the saw and decided to take it to the table.

[Animation by Ray Mooney, UT Austin]
A Maximum Entropy Model for POS Tagging

- Inference, need to do Forward traversal of input sequence:

  John saw the saw and decided to take it to the table.

  **NNP VBD DT NN CC**

  ![Diagram showing POS tagging process]

  **classifier**

  **VBD**
A Maximum Entropy Model for POS Tagging

- Inference, need to do Forward traversal of input sequence:

NNP VBD DT NN CC VBD
John saw the saw and decided to take it to the table.
A Maximum Entropy Model for POS Tagging

• Inference, need to do Forward traversal of input sequence:

NNP VBD DT NN CC VBD TO
John saw the saw and decided to take it to the table.
A Maximum Entropy Model for POS Tagging

[Animation by Ray Mooney, UT Austin]

NNP VBD DT NN CC VBD TO VB
John saw the saw and decided to take it to the table.
A Maximum Entropy Model for POS Tagging

• Inference, need to do Forward traversal of input sequence:


NNP VBD DT NN CC VBD TO VB PRP
John saw the saw and decided to take it to the table.
A Maximum Entropy Model for POS Tagging

• Inference, need to do Forward traversal of input sequence:

NNP VBD DT NN CC VBD TO VB PRP IN
John saw the saw and decided to take it to the table.

classifier

DT
A Maximum Entropy Model for POS Tagging

• Inference, need to do Forward traversal of input sequence:

NNP VBD DT NN CC VBD TO VB PRP IN DT
John saw the saw and decided to take it to the table.

[Animation by Ray Mooney, UT Austin]

[Animation by Ray Mooney, UT Austin]
A Maximum Entropy Model for POS Tagging

- Inference, need to do Forward traversal of input sequence:

  [Animation by Ray Mooney, UT Austin]

John saw the saw and decided to take it to the table.

- Some POS tags would be easier to disambiguate backward, what can we do?
  - Use backward traversal, with backward features … but lose forward info.
Inference in Systems

Sequence Level

Sequence Data

Local Level

Local Data

Feature Extraction

Label

Features

Classifier Type

Optimization

Smoothing

Maximum Entropy Models

Conjugate Gradient

Quadratic Penalties

Feature Extraction

Label

Features

Sequence Model

Inference

Sequence Model
Greedy Inference

- **Greedy inference:**
  - We just start at the left, and use our classifier at each position to assign a label
  - The classifier can depend on previous labeling decisions as well as observed data

- **Advantages:**
  - Fast, no extra memory requirements
  - Very easy to implement
  - With rich features including observations to the right, it may perform quite well

- **Disadvantage:**
  - Greedy. We make commit errors we cannot recover from
Beam Inference

- **Beam inference:**
  - At each position keep the top $k$ complete sequences.
  - Extend each sequence in each local way.
  - The extensions compete for the $k$ slots at the next position.

- **Advantages:**
  - Fast; beam sizes of $k = 3$ to 5 are almost as good as exact inference in many cases.
  - Easy to implement (no dynamic programming required).

- **Disadvantage:**
  - Inexact: the globally best sequence can fall off the beam.
Viterbi Inference

• **Viterbi inference:**
  – Dynamic programming or memoization.
  – Requires small window of state influence (e.g., past two states are relevant).
• Advantage:
  – Exact: the **global best sequence** is returned.
• Disadvantage:
  – Harder to implement long-distance state-state interactions (but beam inference tends not to allow long-distance resurrection of sequences anyway).
Sequence Labeling as Classification

1) **Classify** each token **individually** into one of a number of classes.

2) **Classify** all tokens **jointly** into one of a number of classes:

   \[
   \hat{t}_1 \ldots \hat{t}_n = \arg \max_{t_1, \ldots, t_n} \lambda^T \varphi(t_1, \ldots, t_n, w_1, \ldots, w_n)
   \]

   - Hidden Markov Models.
   - Conditional Random Fields.
   - Structural SVMs.
   - Discriminatively Trained HMMs [Collins, EMNLP’02].
   - Bi-directional RNNs / LSTM-CRFs.
Hidden Markov Models

- Probabilistic Generative Models:

\[
\hat{t}_1...\hat{t}_n = \arg \max_{t_1,...,t_n} p(t_1,...,t_n \mid w_1,...,w_n)
\]

\[
= \arg \max_{t_1,...,t_n} p(w_1,...,w_n \mid t_1,...,t_n) p(t_1,...,t_n)
\]

Likelihood
- Use state emission probs

Prior
- Use state transition probs
Hidden Markov Models: Assumptions

1) A word event depends only on its POS tag:

\[ p(w_1, ..., w_n \mid t_1, ..., t_n) = \prod_{i=1}^{n} p(w_i \mid t_i) \]

2) A tag event depends only on the previous tag:

\[ p(t_1, ..., t_n) = \prod_{i=1}^{n} p(t_i \mid t_{i-1}) \]

\( \Rightarrow \) POS tagging is \( \hat{t}_1, ..., \hat{t}_n = \arg \max_{t_1, ..., t_n} \prod_{i=1}^{n} p(w_i \mid t_i) p(t_i \mid t_{i-1}) \)
Linear-Chain Conditional Random Fields

- A linear-chain CRF is a distribution $p(y|x)$ over sequences of labels $y$ and conditioned on observations $x$ that takes the form:

$$
p(y|x) = \frac{1}{Z(x)} \exp \left( \sum_{t=1}^{T} w^T f_s(y_{t-1}, y_t) + \sum_{t=1}^{T} u^T f_o(y_t, t, x) \right)
$$

where

$$
Z(x) = \sum_{y' \in Y} \exp \left( \sum_{t=1}^{T} w^T f_s(y'_{t-1}, y'_t) + \sum_{t=1}^{T} u^T f_o(y'_t, t, x) \right)
$$

- The state transition features $f_s$ and observation emission features $f_o$ can be any real-valued functions.
Supplemental Reading

• Chapter 8 in Jurafsky & Martin:

• Appendix A in Jurafsky & Martin: