Lexical Semantics
Wrap-up and Midterm Review
How do we know when a word has more than one sense?

• ATIS examples
  • Which flights serve breakfast?
  • Does America West serve Philadelphia?

• The “zeugma” test:
  • ?Does United serve breakfast and San Jose?
Synonyms

- Word that have the same meaning in some or all contexts.
  - filbert / hazelnut
  - couch / sofa
  - big / large
  - automobile / car
  - vomit / throw up
  - Water / H₂O

- Two lexemes are synonyms if they can be successfully substituted for each other in all situations
  - If so they have the same **propositional meaning**
Synonyms

• But there are few (or no) examples of perfect synonymy.
  • Why should that be?
  • Even if many aspects of meaning are identical
  • Still may not preserve the acceptability based on notions of politeness, slang, register, genre, etc.

• Example:
  • Water and $\text{H}_2\text{O}$
Some more terminology

- **Lemmas and wordforms**
  - A **lexeme** is an abstract pairing of meaning and form
  - A **lemma** or **citation form** is the grammatical form that is used to represent a **lexeme**.
    - *Carpet* is the lemma for *carpets*
    - *Dormir* is the lemma for *duermes*.
  - Specific surface forms *carpets*, *sung*, *duermes* are called **wordforms**

- **The lemma** *bank* **has two senses:**
  - Instead, a **bank** can hold the investments in a custodial account in the client’s name
  - But as agriculture burgeons on the east **bank**, the river will shrink even more.

- **A sense** is a discrete representation of one aspect of the meaning of a word
Synonymy is a relation between senses rather than words

• Consider the words *big* and *large*

• Are they synonyms?
  • How *big* is that plane?
  • Would I be flying on a *large* or small plane?

• How about here:
  • Miss Nelson, for instance, became a kind of *big* sister to Benjamin.
  • Miss Nelson, for instance, became a kind of *large* sister to Benjamin.

• Why?
  • *big* has a sense that means being older, or grown up
  • *large* lacks this sense
Antonyms

• Senses that are opposites with respect to one feature of their meaning

• Otherwise, they are very similar!
  • dark / light
  • short / long
  • hot / cold
  • up / down
  • in / out

• More formally: antonyms can
  • define a binary opposition or at opposite ends of a scale (*long*/*short*, *fast*/*slow*)
  • Be *reversives*: *rise*/*fall*, *up*/*down*
Hyponymy

- One sense is a **hyponym** of another if the first sense is more specific, denoting a subclass of the other
  - *car* is a hyponym of *vehicle*
  - *dog* is a hyponym of *animal*
  - *mango* is a hyponym of *fruit*

- Conversely
  - *vehicle* is a hypernym/superordinate of *car*
  - *animal* is a hypernym of *dog*
  - *fruit* is a hypernym of *mango*

<table>
<thead>
<tr>
<th>superordinate</th>
<th>vehicle</th>
<th>fruit</th>
<th>furniture</th>
<th>mammal</th>
</tr>
</thead>
<tbody>
<tr>
<td>hyponym</td>
<td>car</td>
<td>mango</td>
<td>chair</td>
<td>dog</td>
</tr>
</tbody>
</table>
Hypernymy more formally

• Extensional:
  • The class denoted by the superordinate
  • extensionally includes the class denoted by the hyponym

• Entailment:
  • A sense A is a hyponym of sense B if being an A entails being a B

• Hyponymy is usually transitive
  • (A hypo B and B hypo C entails A hypo C)
Why would hypernyms/hyponyms be important to constructing a meaning representation?
Why would hypernyms/hyponyms be important for meaning representation?
II. WordNet

- A hierarchically organized lexical database
- On-line thesaurus + aspects of a dictionary
  - Versions for other languages are under development

<table>
<thead>
<tr>
<th>Category</th>
<th>Unique Forms</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noun</td>
<td>117,097</td>
</tr>
<tr>
<td>Verb</td>
<td>11,488</td>
</tr>
<tr>
<td>Adjective</td>
<td>22,141</td>
</tr>
<tr>
<td>Adverb</td>
<td>4,601</td>
</tr>
</tbody>
</table>
WordNet

- Where it is:
  - https://wordnet.princeton.edu/
Format of Wordnet Entries

The noun “bass” has 8 senses in WordNet.
1. bass¹ - (the lowest part of the musical range)
2. bass², bass part¹ - (the lowest part in polyphonic music)
3. bass³, basso¹ - (an adult male singer with the lowest voice)
4. sea bass¹, bass⁴ - (the lean flesh of a saltwater fish of the family Serranidae)
5. freshwater bass¹, bass⁵ - (any of various North American freshwater fish with lean flesh (especially of the genus Micropterus))
6. bass⁶, bass voice¹, basso² - (the lowest adult male singing voice)
7. bass⁷ - (the member with the lowest range of a family of musical instruments)
8. bass⁸ - (nontechnical name for any of numerous edible marine and freshwater spiny-finned fishes)

The adjective “bass” has 1 sense in WordNet.
1. bass¹, deep⁶ - (having or denoting a low vocal or instrumental range)
   "a deep voice"; "a bass voice is lower than a baritone voice"; "a bass clarinet"
# WordNet Noun Relations

<table>
<thead>
<tr>
<th>Relation</th>
<th>Also called</th>
<th>Definition</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hypernym</td>
<td>Superordinate</td>
<td>From concepts to superordinates</td>
<td>breakfast(^1) → meal(^1)</td>
</tr>
<tr>
<td>Hyponym</td>
<td>Subordinate</td>
<td>From concepts to subtypes</td>
<td>meal(^1) → lunch(^1)</td>
</tr>
<tr>
<td>Member Meronym</td>
<td>Has-Member</td>
<td>From groups to their members</td>
<td>faculty(^2) → professor(^1)</td>
</tr>
<tr>
<td>Instance</td>
<td>Has-Instance</td>
<td>From concepts to instances of the concept</td>
<td>composer(^1) → Bach(^1)</td>
</tr>
<tr>
<td>Member Holonym</td>
<td>Member-Of</td>
<td>From instances to their concepts</td>
<td>Austen(^1) → author(^1)</td>
</tr>
<tr>
<td>Part Meronym</td>
<td>Has-Part</td>
<td>From members to their groups</td>
<td>copilot(^1) → crew(^1)</td>
</tr>
<tr>
<td>Part Holonym</td>
<td>Part-Of</td>
<td>From wholes to parts</td>
<td>table(^2) → leg(^3)</td>
</tr>
<tr>
<td>Antonym</td>
<td></td>
<td>From parts to wholes</td>
<td>course(^7) → meal(^1)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Opposites</td>
<td>leader(^1) → follower(^1)</td>
</tr>
</tbody>
</table>
# WordNet Verb Relations

<table>
<thead>
<tr>
<th>Relation</th>
<th>Definition</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hypernym</td>
<td>From events to superordinate events</td>
<td>$fly^9 \rightarrow travel^5$</td>
</tr>
<tr>
<td>Troponym</td>
<td>From a verb (event) to a specific manner elaboration of that verb</td>
<td>$walk^1 \rightarrow stroll^1$</td>
</tr>
<tr>
<td>Entails</td>
<td>From verbs (events) to the verbs (events) they entail</td>
<td>$snore^1 \rightarrow sleep^1$</td>
</tr>
<tr>
<td>Antonym</td>
<td>Opposites</td>
<td>$increase^1 \leftrightarrow decrease^1$</td>
</tr>
</tbody>
</table>
WordNet Hierarchies

Sense 3
bass, basso --
(an adult male singer with the lowest voice)
=> singer, vocalist, vocalizer, vocaliser
  => musician, instrumentalist, player
  => performer, performing artist
  => entertainer
  => person, individual, someone...
  => organism, being
    => living thing, animate thing,
      => whole, unit
        => object, physical object
          => physical entity
            => entity
  => causal agent, cause, causal agency
  => physical entity
  => entity

Sense 7
bass --
(the member with the lowest range of a family of musical instruments)
=> musical instrument, instrument
  => device
    => instrumentality, instrumentation
      => artifact, artefact
        => whole, unit
          => object, physical object
            => physical entity
How is “sense” defined in WordNet?

- The set of near-synonyms for a WordNet sense is called a **synset (synonym set)**; it’s their version of a sense or a concept.
- Example: **chump** as a noun to mean
  - ‘a person who is gullible and easy to take advantage of’

\{chump\(^1\), fool\(^2\), gull\(^1\), mark\(^9\), patsy\(^1\), fall guy\(^1\), sucker\(^1\), soft touch\(^1\), mug\(^2\)\}

- Each of these senses share this same gloss.
- Thus for WordNet, the meaning of this sense of **chump** is this list.
Wordnet example
Questions?
Midterm

• Format
  • Multiple Choice questions
  • Short answer questions
  • Problem solving

• What will it cover?
  • Anything covered in class
  • From reading that supports material in class
  • Math as needed for neural nets, machine learning, smoothing
Midterm

• Closed book, no notes, no electronics

• Will avoid asking you to recall formulas
  • That said, you should know how to compute the probability of ngrams, of POS tags, basics for smoothing, language modeling, how to do computation for neural nets.

• Will cover anything from beginning through today

• Sample midterm questions posted
Top topics

- Viterbi algorithm
- Dependency parsing
- RNNs
Questions?
Viterbi and POS
Two kinds of probabilities (1)

- Tag transition probabilities $p(t_i | t_{i-1})$
  - Determiners likely to precede adjs and nouns
    - That/DT flight/NN
    - The/DT yellow/JJ hat/NN
    - So we expect $P(\text{NN} | \text{DT})$ and $P(\text{JJ} | \text{DT})$ to be high
    - But $P(\text{DT} | \text{JJ})$ to be low
  - Compute $P(\text{NN} | \text{DT})$ by counting in a labeled corpus:

\[
P(t_i | t_{i-1}) = \frac{C(t_{i-1}, t_i)}{C(t_{i-1})}
\]

\[
P(\text{NN} | \text{DT}) = \frac{C(\text{DT}, \text{NN})}{C(\text{DT})} = \frac{56,509}{116,454} = .49
\]
Two kinds of probabilities (2)

- Word likelihood probabilities $p(w_i | t_i)$
  - VBZ (3sg Pres verb) likely to be “is”
  - Compute $P(is | VBZ)$ by counting in a labeled corpus:

\[
P(w_i | t_i) = \frac{C(t_i, w_i)}{C(t_i)}
\]

\[
P(is | VBZ) = \frac{C(VBZ, is)}{C(VBZ)} = \frac{10,073}{21,627} = .47
\]
An Example: the verb “race”

- Secretariat/NNP is/VBZ expected/VBN to/TO \textbf{race}/VB tomorrow/NR
- People/NNS continue/VB to/TO inquire/VB the/DT reason/NN for/IN the/DT \textbf{race}/NN for/IN outer/JJ space/NN

- How do we pick the right tag?
Disambiguating “race”

(a) Secretariat is expected to race tomorrow

(b) Secretariat is expected to race tomorrow
Disambiguating “race”
Disambiguating “race”

(a) Secretariat is expected to race tomorrow

(b) Secretariat is expected to race tomorrow
Disambiguating “race”

(a) Secretariat is expected to race tomorrow

(b) Secretariat is expected to race tomorrow
- \( P(\text{NN} | \text{TO}) = 0.00047 \)
- \( P(\text{VB} | \text{TO}) = 0.83 \)
- \( P(\text{race} | \text{NN}) = 0.00057 \)
- \( P(\text{race} | \text{VB}) = 0.00012 \)
- \( P(\text{NR} | \text{VB}) = 0.0027 \)
- \( P(\text{NR} | \text{NN}) = 0.0012 \)
- \( P(\text{VB} | \text{TO})P(\text{NR} | \text{VB})P(\text{race} | \text{VB}) = 0.00000027 \)
- \( P(\text{NN} | \text{TO})P(\text{NR} | \text{NN})P(\text{race} | \text{NN}) = 0.00000000032 \)
- So we (correctly) choose the verb reading,
HMMS
Hidden Markov Models

- We don’t observe POS tags
  - We infer them from the words we see

- Observed events

- Hidden events
Hidden Markov Model

• For Markov chains, the output symbols are the same as the states.
  • See **hot** weather: we’re in state **hot**

• But in part-of-speech tagging (and other things)
  • The output symbols are **words**
  • The hidden states are **part-of-speech tags**

• So we need an extension!

• A **Hidden Markov Model** is an extension of a Markov chain in which the input symbols are not the same as the states.

• This means **we don’t know which state we are in**.
Hidden Markov Models

- States $Q = q_1, q_2 \ldots q_N$;
- Observations $O = o_1, o_2 \ldots o_N$;
  - Each observation is a symbol from a vocabulary $V = \{v_1, v_2, \ldots v_V\}$
- Transition probabilities
  - Transition probability matrix $A = \{a_{ij}\}$
    \[ a_{ij} = P(q_t = j \mid q_{t-1} = i) \quad 1 \leq i, j \leq N \]
- Observation likelihoods
  - Output probability matrix $B = \{b_i(k)\}$
    \[ b_i(k) = P(X_t = o_k \mid q_t = i) \]
- Special initial probability vector $\pi$
  \[ \pi_i = P(q_1 = i) \quad 1 \leq i \leq N \]
Hidden Markov Models

- Some constraints

\[ \sum_{j=1}^{N} a_{ij} = 1; \quad 1 \leq i \leq N \]

\[ \pi_i = P(q_1 = i) \quad 1 \leq i \leq N \]

\[ \sum_{k=1}^{M} b_i(k) = 1 \]

\[ \sum_{j=1}^{N} \pi_j = 1 \]
Assumptions

• **Markov assumption:**

\[ P(q_i | q_1...q_{i-1}) = P(q_i | q_{i-1}) \]

• **Output-independence assumption**

\[ P(o_t | O_1^{t-1}, q_t^t) = P(o_t | q_t) \]
Three fundamental Problems for HMMs

• **Likelihood:** Given an HMM $\lambda = (A, B)$ and an observation sequence $O$, determine the likelihood $P(O, \lambda)$.

• **Decoding:** Given an observation sequence $O$ and an HMM $\lambda = (A, B)$, discover the best hidden state sequence $Q$.

• **Learning:** Given an observation sequence $O$ and the set of states in the HMM, learn the HMM parameters $A$ and $B$.

*What kind of data would we need to learn the HMM parameters?*
Decoding

• The best hidden sequence
  • Weather sequence in the ice cream task
  • POS sequence given an input sentence

• We could use argmax over the probability of each possible hidden state sequence
  • Why not?

• Viterbi algorithm
  • Dynamic programming algorithm
  • Uses a dynamic programming trellis
    • Each trellis cell represents, $v_t(j)$, represents the probability that the HMM is in state $j$ after seeing the first $t$ observations and passing through the most likely state sequence
Viterbi intuition: we are looking for the best ‘path’

promised to back the bill

Slide from Dekang Lin
Intuition

• The value in each cell is computed by taking the MAX over all paths that lead to this cell.

\[ v_t(j) = \max_{1 \leq i \leq N-1} v_{t-1}(i) a_{ij} b_j(o_t) \]

• An extension of a path from state i at time t-1 is computed by multiplying:

- the previous Viterbi path probability from the previous time step
- the transition probability from previous state \( q_i \) to current state \( q_j \)
- the state observation likelihood of the observation symbol \( o_t \) given the current state \( j \)
The Viterbi Algorithm

\textbf{function} VITERBI(observations of len T, state-graph) \textbf{returns} best-path

\texttt{num-states} \leftarrow \text{NUM-OF-STATES(state-graph)}

Create a path probability matrix \texttt{viterbi[num-states+2,T+2]}

\texttt{viterbi[0,0]} \leftarrow 1.0

\textbf{for} each time step \texttt{t} \textbf{from 1 to} \texttt{T} \textbf{do}

\textbf{for} each state \texttt{s} \textbf{from 1 to} \texttt{num-states} \textbf{do}

\texttt{viterbi[s,t]} \leftarrow \max_{1 \leq s' \leq \text{num-states}} \texttt{viterbi[s',t-1]} \ast a_{s',s} \ast b_{s}(o_{t})

\texttt{backpointer[s,t]} \leftarrow \arg\max_{1 \leq s' \leq \text{num-states}} \texttt{viterbi[s',t-1]} \ast a_{s',s}

Backtrace from highest probability state in final column of \texttt{viterbi[]} and return path
### The A matrix for the POS HMM

<table>
<thead>
<tr>
<th></th>
<th>VB</th>
<th>TO</th>
<th>NN</th>
<th>PPSS</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;s&gt;</td>
<td>.019</td>
<td>.0043</td>
<td>.041</td>
<td>.067</td>
</tr>
<tr>
<td>VB</td>
<td>.0038</td>
<td>.035</td>
<td>.047</td>
<td>.0070</td>
</tr>
<tr>
<td>TO</td>
<td>.83</td>
<td>0</td>
<td>.00047</td>
<td>0</td>
</tr>
<tr>
<td>NN</td>
<td>.0040</td>
<td>.016</td>
<td>.087</td>
<td>.0045</td>
</tr>
<tr>
<td>PPSS</td>
<td>.23</td>
<td>.00079</td>
<td>.0012</td>
<td>.00014</td>
</tr>
</tbody>
</table>

**Figure 4.15** Tag transition probabilities (the $a$ array, $p(t_i|t_{i-1})$) computed from the 87-tag Brown corpus without smoothing. The rows are labeled with the conditioning event; thus $P(PPSS|VB)$ is .0070. The symbol <s> is the start-of-sentence symbol.

What is $P(VB|TO)$? What is $P(NN|TO)$? Why does this make sense?

What is $P(TO|VB)$? What is $P(TO|NN)$? Why does this make sense?
The B matrix for the POS HMM

<table>
<thead>
<tr>
<th></th>
<th>I</th>
<th>want</th>
<th>to</th>
<th>race</th>
</tr>
</thead>
<tbody>
<tr>
<td>VB</td>
<td>0</td>
<td>.0093</td>
<td>0</td>
<td>.00012</td>
</tr>
<tr>
<td>TO</td>
<td>0</td>
<td>0</td>
<td>.99</td>
<td>0</td>
</tr>
<tr>
<td>NN</td>
<td>0</td>
<td>.000054</td>
<td>0</td>
<td>.00057</td>
</tr>
<tr>
<td>PPSS</td>
<td>.37</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Figure 4.16** Observation likelihoods (the b array) computed from the 87-tag Brown corpus without smoothing.

Look at $P(\text{want}|\text{VB})$ and $P(\text{want}|\text{NN})$. Give an explanation for the difference in the probabilities.
<table>
<thead>
<tr>
<th></th>
<th>VB</th>
<th>TO</th>
<th>NN</th>
<th>PPSS</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;s&gt;</td>
<td>.019</td>
<td>.0043</td>
<td>.041</td>
<td>.067</td>
</tr>
<tr>
<td>VB</td>
<td>.0038</td>
<td>.035</td>
<td>.047</td>
<td>.0070</td>
</tr>
<tr>
<td>TO</td>
<td>.83</td>
<td>0</td>
<td>.00047</td>
<td>0</td>
</tr>
<tr>
<td>NN</td>
<td>.0040</td>
<td>.016</td>
<td>.087</td>
<td>.0045</td>
</tr>
<tr>
<td>PPSS</td>
<td>.23</td>
<td>.00079</td>
<td>.0012</td>
<td>.00014</td>
</tr>
</tbody>
</table>

**Figure 4.15**  Tag transition probabilities (the $a$ array, $p(t_i|t_{i-1})$) computed from the 87-tag Brown corpus without smoothing. The rows are labeled with the conditioning event; thus $P(PPSS|VB)$ is .0070. The symbol $<s>$ is the start-of-sentence symbol.

<table>
<thead>
<tr>
<th></th>
<th>I</th>
<th>want</th>
<th>to</th>
<th>race</th>
</tr>
</thead>
<tbody>
<tr>
<td>VB</td>
<td>0</td>
<td>.0093</td>
<td>0</td>
<td>.00012</td>
</tr>
<tr>
<td>TO</td>
<td>0</td>
<td>0</td>
<td>.99</td>
<td>0</td>
</tr>
<tr>
<td>NN</td>
<td>0</td>
<td>.000054</td>
<td>0</td>
<td>.00057</td>
</tr>
<tr>
<td>PPSS</td>
<td>.37</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Figure 4.16**  Observation likelihoods (the $b$ array) computed from the 87-tag Brown corpus without smoothing.
Problem

• I want to race (possible states: PPS VB TO NN)
\[ v_t(j) = \max_{1 \leq i \leq N-1} v_{t-1}(i) a_{ij} b_j(o_t) \]

**Example**

\[ v_1(4) = 0.041 \times 0 = 0 \]

\[ v_1(3) = 0.0033 \times 0 = 0 \]

\[ v_1(2) = 0.019 \times 0 = 0 \]

\[ v_1(1) = 0.067 \times 0.37 = 0.025 \]

**Backtrace**

- \[ v_1(4) \rightarrow v_1(3) \rightarrow v_1(2) \rightarrow v_1(1) \]
- **t=1**

**Words**

- \( o_1 \): i
- \( o_2 \): want
- \( o_3 \): to
- \( o_4 \): race
\[ v_t(j) = \max_{1 \leq i \leq N-1} v_{1}(i) a_{ij} b_j(o_t) \]
\[ v_t(j) = \max_{1 \leq i \leq N-1} v_1(i) a_{ij} b_j(o_t) \]

\[ v_0(0) = 1.0 \]

\[ v_1(1) = 0.067 \times 0.37 = 0.025 \]

\[ v_1(2) = 0.019 \times 0 = 0 \]

\[ v_1(3) = 0.0043 \times 0 = 0 \]

\[ v_1(4) = 0.041 \times 0 = 0 \]

\[ v_1(5) = 0.025 \times 0.23 = 0.0055 \]

\[ J = \text{NN} \]

\[ I = \text{S} \]

\[ t = 1 \]

\[ o_1 \text{ i } \]

\[ o_2 \text{ want } \]

\[ o_3 \text{ to } \]

\[ o_4 \text{ race } \]
The A matrix for the POS HMM

<table>
<thead>
<tr>
<th></th>
<th>VB</th>
<th>TO</th>
<th>NN</th>
<th>PPSS</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;s&gt;</td>
<td>0.019</td>
<td>0.0043</td>
<td>0.041</td>
<td>0.067</td>
</tr>
<tr>
<td>VB</td>
<td>0.0038</td>
<td>0.035</td>
<td>0.047</td>
<td>0.0070</td>
</tr>
<tr>
<td>TO</td>
<td>0.83</td>
<td>0</td>
<td>0.00047</td>
<td>0</td>
</tr>
<tr>
<td>NN</td>
<td>0.0040</td>
<td>0.016</td>
<td>0.087</td>
<td>0.0045</td>
</tr>
<tr>
<td>PPSS</td>
<td>0.23</td>
<td>0.00079</td>
<td>0.0012</td>
<td>0.00014</td>
</tr>
</tbody>
</table>

Figure 4.15 Tag transition probabilities (the $a$ array, $p(t_i|t_{i-1})$) computed from the 87-tag Brown corpus without smoothing. The rows are labeled with the conditioning event; thus $P(PPSS|VB)$ is 0.0070. The symbol $<s>$ is the start-of-sentence symbol.
\[ v_t(j) = \max_{1 \leq i \leq N-1} v_{i+1}(i) a_{ij} b_j(o_t) \]

\[ v_{1}(4) = 0.041 \times 0 = 0 \]

\[ v_{1}(3) = 0.0043 \times 0 = 0 \]

\[ v_{1}(2) = 0.019 \times 0 = 0 \]

\[ v_{1}(1) = 0.067 \times 0.37 = 0.025 \]

I=S

J=NN

T=1

i

want

to

race
The B matrix for the POS HMM

<table>
<thead>
<tr>
<th></th>
<th>I</th>
<th>want</th>
<th>to</th>
<th>race</th>
</tr>
</thead>
<tbody>
<tr>
<td>VB</td>
<td>0</td>
<td>0.0093</td>
<td>0</td>
<td>0.00012</td>
</tr>
<tr>
<td>TO</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>NN</td>
<td>0</td>
<td>0.000054</td>
<td>0</td>
<td>0.00057</td>
</tr>
<tr>
<td>PPSS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 4.16 Observation likelihoods (the b array) computed from the 87-tag Brown corpus without smoothing.

Look at $P(\text{want}|\text{VB})$ and $P(\text{want}|\text{NN})$. Give an explanation for the difference in the probabilities.
\[
v_t(j) = \max_{1 \leq i \leq N-1} v_{i}(i) a_{ij} b_j(o_t)
\]

**Viterbi example**

<table>
<thead>
<tr>
<th>t=1</th>
<th>i = S</th>
<th>\text{start}</th>
</tr>
</thead>
<tbody>
<tr>
<td>\text{start}</td>
<td>\text{PPS}</td>
<td>\text{PPS}</td>
</tr>
<tr>
<td>\text{PPS}</td>
<td>\text{VB}</td>
<td>\text{VB}</td>
</tr>
<tr>
<td>\text{VB}</td>
<td>TO</td>
<td>TO</td>
</tr>
<tr>
<td>TO</td>
<td>NN</td>
<td>NN</td>
</tr>
<tr>
<td>NN</td>
<td>end</td>
<td>end</td>
</tr>
</tbody>
</table>

- \( v_{i}(1) = 0.067 \times 0.37 = 0.025 \)
- \( v_{1}(1) = 0.041 \times 0 = 0 \)
- \( v_{1}(2) = 0.019 \times 0 = 0 \)
- \( v_{1}(3) = 0.0043 \times 0 = 0 \)
- \( v_{1}(4) = 0.041 \times 0 = 0 \)
- \( v_{1}(4) = 0.0043 \times 0.69 = 0 \)
- \( v_{1}(3) = 0.0043 \times 0 = 0 \)
- \( v_{1}(2) = 0.0043 \times 0 = 0 \)

The word sequence is **i want to race**.

**Words:**
- \( o_1 = i \)
- \( o_2 = \text{want} \)
- \( o_3 = \text{to} \)
- \( o_4 = \text{race} \)
\[ v_t(j) = \max_{1 \leq i \leq N-1} v_{i+1}(i) \cdot a_{ij} \cdot b_j(o_t) \]
Show the 4 formulas you would use to compute the value at this node and the max.
Dependency Parsing
Dependency parsing

• An example from the NY Times today:

Last week, on the third floor of a small building in San Francisco’s Mission District, a woman scrambled the tiles of a Rubik’s Cube
Dependency parsing

• An example from the NY Times today:

_Last week, on the third floor of a small building in San Francisco’s Mission District, a woman scrambled the tiles of a Rubik’s Cube._
Dependency parsing

- An example from the NY Times today:

*Last week, on the third floor, a woman scrambled the tiles of a Rubik’s Cube*
RNNs and LSTMs
Recurrent Neural Networks

\[ h_{t} = \sigma(W_{h} h_{t-1} + W_{x} x_{t}) \]
\[ h_{t} = \sigma(W_{h} h_{t-1} + W_{x} x_{t}) \]
\[ y_{t} = \text{softmax}(W_{y} h_{t}) \]
RNN

The cat sat

Slide from Radev
Updating Parameters of an RNN

Backpropagation through time
RNN – I had in mind your facts, buddy, not hers.

In this overview, \( w \) refers to the weights. But there are different kinds of weights. Let’s be more specific.
RNN – I had in mind your facts, buddy, not hers.
RNN – I had in mind your facts, buddy, not hers.

$W$ are the weights: the word embedding matrix multiplication with $x_t$ yields the embedding for $x$.

$U$ is another weight matrix.

$H_0$ is often not specified. $H$ is the hidden layer.
RNN – I had in mind your facts, buddy, not hers.

\[ h_t = \sigma \left( U \begin{bmatrix} w_{xt} \\ h_{t-1} \end{bmatrix} \right) \]
RNN – I had in mind your facts, buddy, not hers.

Final embedding run through the sigmoid function -> [0,1]

1 = positive
0 = negative

Often final h is used as word embedding for the sentence
Updating Parameters of an RNN

Backpropagation through time
Gold label = 0 (negative)
Adjust weights using gradient
Repeat many times with all examples
Transforming RNN to LSTM

\[ u_{lt} = \sigma(W_{hl} h_{lt-1} + W_{lx} x_{lt}) \]

[slides from Catherine Finegan-Dollak]
Transforming RNN to LSTM

[slides from Catherine Finegan-Dollak]
Transforming RNN to LSTM

\[ c_t = f_t \odot c_{t-1} + i_t \odot u_t \]

[slides from Catherine Finegan-Dollak]
Transforming RNN to LSTM

\[ c_t = f_t \odot c_{t-1} + i_t \odot u_t \]

[slides from Catherine Finegan-Dollak]
Transforming RNN to LSTM

\[ c_t = f_t \odot c_{t-1} + i_t \odot u_t \]

[slides from Catherine Finegan-Dollak]
Transforming RNN to LSTM

\[ f_{\downarrow t} = \sigma(W_{hf} h_{\downarrow t-1} + W_{xf} x_{\downarrow t}) \]

[slides from Catherine Finegan-Dollak]
Transforming RNN to LSTM

\[ \dot{c}_t = \sigma(W_{hi} h_{t-1} + W_{xi} x_t) \]
Transforming RNN to LSTM

\[ h_{\downarrow t} = o_{\downarrow t} \odot \tanh c_{\downarrow t} \]

[slides from Catherine Finegan-Dollak]
The cat sat

[slides from Catherine Finegan-Dollak]
Problem 10 from sample midterm questions