NLP Basics: Representations, Tools and Resources
Levels of processing

speech analysis
morphological and lexical analysis
morphological realization
syntactic realization
parsing
lexicon and grammar
discourse context
contextual reasoning
utterance planning
application reasoning and execution
domain knowledge

Phonology
Morphology
Syntax
Semantics
Reasoning
I would like to purchase a flight ...
Levels of Processing II

![Diagram showing different levels of processing in natural language understanding and generation. The diagram includes nodes like Interlingua, Semantic Structure, Syntactic Structure, Word Structure, Semantic Composition, Semantic Decomposition, Semantic Analysis, Syntactic Analysis, Direct, and Morphological Generation. Arrows indicate the flow of processing from source text to target text.]
What kinds of things does this course look at?
Levels of processing

**Phonology**
- Speech analysis
- Speech synthesis
- Pronunciation model

**Morphology**
- Morphological and lexical analysis
- Morphological realization
- Morphological rules

**Syntax**
- Parsing
- Syntactic realization
- Lexicon and grammar

**Semantics**
- Contextual reasoning
- Utterance planning
- Discourse context

**Reasoning**
- Application reasoning and execution
- Domain knowledge

**Discourse, Dialogue, Pragmatics**

Language Meaning requiring CONTEXT or KNOWLEDGE
Computational Models of Discourse/Dialogue

- Language that requires representations of context above the sentence or phrase to understand and extract the meaning
- Natural Language Processing using different representations of context => Discourse Model
- Language that results from interaction of two or more people (dialogue)
- Language that requires inference using world knowledge to understand and extract meaning
- Language that depends on social norms to understand what is being said
What kinds of problems?

- Anaphora resolution,
  - what is being talked about when a speaker uses pronouns such as *he, she, it, that*

- Inference of discourse relations within a speaker’s turn or across speakers or turns
  - In forums, do speaker A and speaker B *disagree*
  - Is speaker B’s response to what A said *sarcastic*

- Predicting missing events or likely next events after hearing or reading about a sequence of events
  - Narrative Schemas: Built out of event chains
    - if I see the word “*arrested*” what is a likely next word I will see, e.g. “*tried, sentenced, jailed*”
Why do we care about or need to know about the other levels of processing?
Levels of processing

We believe that representations from these levels will help us computationally model or recognize meaning at the discourse level.
What do we need to know about the other levels of processing?
Features, Labels: By hand, computational, ML

Features: things we count

- Words
- Cue words: particular words in particular positions in the utterance
- Patterns of words
- Predications (a verb and one of its arguments)
- Representations of the context (utterances before this one)

Labels: what we try to characterize with the things we count

- Discourse Relations, within and across speakers
- (Dis) Agreement
- Nastiness/Nice (Insults)
- Factual/Emotional
- Sarcasm
- Which discourse entity this pronoun refers to
Tools that people use

- Lots of people distribute code for many NLP tasks
  - You can email a paper’s authors to ask for their code
- Some lists of software, but no central site
- Some end-to-end pipelines for text analysis
  - Cleanup/tokenization + morphology + tagging + parsing + …
  - NLTK is easy for beginners and has a free book
- What we have used
  - Stanford Parser
  - Open NLP Coref
  - Regular expressions to find particular patterns
  - NLTK modules (python)
  - Mechanical Turk to get labels
Tools (an incomplete list)

- Tokenization/Sentence Splitting
- Part Of Speech
- Verbnet, Framenet
- Chunking
- Named Entity Recognition
- Stanford parser and dependencies
- Charniak Parser
- Coreference
- Semantic Role Labeling
- AddDiscourse
More on Tools

- To find good or popular tools:
  - Search current papers, ask around, use the web
- Still, often hard to identify the best tool for your job:
  - Produces appropriate, sufficiently detailed output?
  - Accurate? (on the measure you care about)
  - Robust? (accurate on your data, not just theirs)
  - Fast? Easy and flexible to use? Nice file formats, command line options, visualization?
  - Trainable for new data and languages? How slow is training?
  - Open-source and easy to extend?

- One point of homework: stanford parser and coref should work pretty well on news stories but perhaps not so well on the kinds of data we are interested in
Data: we have some and there’s lots out there

- **Raw text or speech corpora**
  - Or just their **n-gram counts**, for super-big corpora
  - Various languages and genres
  - Usually there’s some metadata (document’s date, author, etc.)

- **Text or speech with manual or automatic annotations**
  - What kind of annotations?
How to get interesting data

- Read papers to find out what datasets others are using
  - Linguistic Data Consortium (searchable) hosts many large datasets
  - Many projects and competitions post data on their websites
- In this class, NLDS lab already has data targeted to the papers we are reading, but you may want to make or get your own
- CORPORA mailing list is also a good place to ask around
- LREC Conference publishes papers about new datasets & metrics
- Amazon Mechanical Turk – pay humans (very cheaply) to annotate your data or to correct automatic annotations
  - We have some expertise here at UCSC. Semantics Lab, NLDS
  - Old task, new domain: Annotate parses etc. on your kind of data
  - New task: Annotate something new that you want your system to find
  - Auxiliary task: Annotate something new that your system may benefit from finding (e.g., annotate sarcasm or disagreement or event reference chains)
Levels of processing

We believe that representations from these levels will help us computationally model or recognize meaning at the discourse level.
Basic Text Processing

Tokens (Words), Sentences
Tokenization and Sentence Segmentation

- Given a document, find the sentence and token boundaries

  The police chased Mr. Smith of Pink Forest, Fla. all the way to Bethesda, where he lived. Smith had escaped after a shoot-out at his workplace, Machinery Inc.

- Why?
  - Word counts may be important features
  - Words may themselves be the object you want to classify
  - “lived.” and “lived” should give the same information
  - Different labels need to align – have to decide what the basic units are that labels go on if you want to leverage multiple annotators from different sources/tasks
Text Normalization

- Every NLP task needs to do text normalization:
  1. Segmenting/tokenizing words in running text
  2. Normalizing word formats
  3. Segmenting sentences in running text
Issues in Tokenization

- Finland’s capital → Finland Finlands Finland’s ?
- what’re, I’m, isn’t → What are, I am, is not
- Hewlett-Packard → Hewlett Packard ?
- state-of-the-art → state of the art ?
- Lowercase → lower-case lowercase lower case ?
- San Francisco → one token or two?
- m.p.h., PhD. → ??
I do uh mainly business data processing. Fragments, filled pauses.

Seuss’s *cat* in the hat is different from other *cats*!

- **Lemma**: same stem, part of speech, rough word sense
  - *cat* and *cats* = same lemma

- **Wordform**: the full inflected surface form
  - *cat* and *cats* = different wordforms
How many words?

$N = \text{number of tokens}$

$V = \text{vocabulary = set of types}$

$|V|$ is the size of the vocabulary

Church and Gale (1990): $|V| > O(N^{\frac{1}{2}})$

|                          | Tokens = $N$   | Types = $|V|$         |
|--------------------------|----------------|----------------------|
| Switchboard phone        | 2.4 million    | 20 thousand          |
| conversations            |                |                      |
| Shakespeare              | 884,000        | 31 thousand          |
| Google N-grams           | 1 trillion     | 13 million           |
How many words?

they lay back on the San Francisco grass and looked at the stars and their

- **Type**: an element of the vocabulary.
- **Token**: an instance of that type in running text.
- **How many?**
  - 15 tokens (or 14)
  - 13 types (or 12) (or 11?)
Morphology

- **Morphemes:**
  - The small meaningful units that make up words
  - **Stems:** The core meaning-bearing units
  - **Affixes:** Bits and pieces that adhere to stems
    - Often with grammatical functions
Stemming

- Reduce terms to their stems in information retrieval
- *Stemming* is crude chopping of affixes
  - language dependent
  - e.g., *automate(s), automatic, automation* all reduced to *automat*.

*for example compressed and compression are both accepted as equivalent to compress.*
Porter’s algorithm
The most common English stemmer

Step 1a
- sses $\rightarrow$ ss caresses $\rightarrow$ caress
- ies $\rightarrow$ i ponies $\rightarrow$ poni
- ss $\rightarrow$ ss caress $\rightarrow$ caress
- s $\rightarrow$ ø cats $\rightarrow$ cat

Step 1b
- (*v*)ing $\rightarrow$ ø walking $\rightarrow$ walk
- sing $\rightarrow$ sing
- (*v*)ed $\rightarrow$ ø plastered $\rightarrow$ plaster

Step 2 (for long stems)
- ational $\rightarrow$ ate relational $\rightarrow$ relate
- izer $\rightarrow$ ize digitizer $\rightarrow$ digitize
- ator $\rightarrow$ ate operator $\rightarrow$ operate

Step 3 (for longer stems)
- al $\rightarrow$ ø revival $\rightarrow$ reviv
- able $\rightarrow$ ø adjustable $\rightarrow$ adjust
- ate $\rightarrow$ ø activate $\rightarrow$ activ
Viewing morphology in a corpus
Why only strip –ing if there is a vowel?

(*v*)ing → ø   walking  → walk
sing       → sing
Viewing morphology in a corpus
Why only strip –ing if there is a vowel?

(*v*)ing → ø
walking → walk

sing → sing

```bash
tr -sc 'A-Za-z' '
' < shakes.txt | grep 'ing$' | sort | uniq -c | sort -nr
```

```
1312 King
548 being
548 being
541 nothing
388 king
375 bring
358 thing
307 ring
152 something
145 coming
130 morning
```

```bash
tr -sc 'A-Za-z' '
' < shakes.txt | grep '[aeiou].*ing$' | sort | uniq -c | sort -nr
```

30
Dealing with complex morphology is sometimes necessary

- Some languages require complex morpheme segmentation
  - Turkish
  - Uygarlastiramadiklarimizdanmissinizcasina
  - `(behaving) as if you are among those whom we could not civilize’
  - **Uygar** `civilized’ + **las** `become’
    + **tir** `cause’ + **ama** `not able’
    + **dik** `past’ + **lar** `plural’
    + **imiz** ‘p1pl’ + **dan** ‘abl’
    + **mis** ‘past’ + **siniz** ‘2pl’ + **casina** ‘as if’
Part of Speech (POS)

- Allows simple abstraction for pattern detection

<table>
<thead>
<tr>
<th>POS</th>
<th>DT</th>
<th>NN</th>
<th>VBD</th>
<th>PP</th>
<th>DT</th>
<th>JJ</th>
<th>NN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Word</td>
<td>The</td>
<td>boy</td>
<td>stood</td>
<td>on</td>
<td>the</td>
<td>burning</td>
<td>deck</td>
</tr>
</tbody>
</table>

- Disambiguate a target, e.g.
  
  “make (a cake)” vs. “make (of car)”

- Specify more abstract patterns,
  
  e.g. Noun Phrase: ( DT JJ* NN )

- Specify context in abstract way

  - e.g. “DT boy VBX” for “actions boys do”
  - This expression will catch “a boy cried”, “some boy ran”, …
Structures above the word:

First I will go through the big picture then go back to more detail
Chunking

- Identifies phrase-level constituents in sentences
  - [NP Boris] [ADVP regretfully] [VP told] [NP his wife] [SBAR that] [NP their child] [VP could not attend] [NP night school] [PP without] [NP permission].

- Useful for filtering: identify e.g. only noun phrases, or only verb phrases
  - Groups modifiers with heads
  - Useful for e.g. Mention Detection

- Used as source of features, e.g. distance (abstracts away determiners, adjectives, for example), sequence
  - More efficient to compute than full syntactic parse
  - Applications in e.g. Information Extraction – getting (simple) information about concepts of interest from text documents
Named Entity Recognition

- Identifies and classifies strings of characters representing proper nouns
  - [PER Neil A. Armstrong], the 38-year-old civilian commander, radioed to earth and the mission control room here: "[LOC Houston], [ORG Tranquility] Base here; the Eagle has landed."

- Useful for filtering documents
  - “I need to find news articles about organizations in which Bill Gates might be involved…”

- Disambiguate tokens: “Chicago” (team) vs. “Chicago” (city)

- Source of abstract features
  - E.g. “Verbs that appear with entities that are Organizations”
  - E.g. “Documents that have a high proportion of Organizations”
Coreference

- Identify all phrases that refer to each entity of interest – i.e., group mentions of concepts

  [Neil A. Armstrong], [the 38-year-old civilian commander], radioed to [earth]. [He] said the famous words, “[the Eagle] has landed.”

- The Named Entity recognizer only gets us part-way…
- …if we ask, “what actions did Neil Armstrong perform?”, we will miss many instances (e.g. “He said…”)
- Coreference resolver abstracts over different ways of referring to the same person
  - Useful in feature extraction, information extraction
Parsers

- Identify the grammatical structure of a sentence

Full parse

Dependency parse

Parsers reveal the grammatical relationships between words and phrases
Stanford dependencies tools


Dependencies for *Bills on ports and immigration were submitted by Senator Brownback, Republican of Kansas*

- nsubjpass(submitted, Bills)
- auxpass(submitted, were)
- agent(submitted, Brownback)
- nn(Brownback, Senator)
- appos(Brownback, Republican)
- prep_of(Republican, Kansas)
- prep_on(Bills, ports)
- conj_and(port, immigration)
- prep_on(Bills, immigration)

---

Figure 1. Standard Stanford dependencies (collapsed and propagated)
Labeled Dependency Parsing

Raw sentence
He reckons the current account deficit will narrow to only 1.8 billion in September.

POS-tagged sentence
He reckons the current account deficit will narrow to only 1.8 billion in September.
PRP    VBZ    DT    JJ    NN    NN    MD    VB    TO    RB    CD    CD    IN    NNP    .

Word dependency parsed sentence
He reckons the current account deficit will narrow to only 1.8 billion in September .

Slide from Jason Eisner and Yuji Matsumoto
Semantic Role Labeler

Semantic Role Labeling Output

Input Text:
A car bomb that exploded outside the U.S. military base in Beniji killed 11 Iraqi citizens.

Result: Complete!

- General Explanation of Argument Labels

<table>
<thead>
<tr>
<th>A</th>
<th>bomb [A1]</th>
<th>killer [A0]</th>
</tr>
</thead>
<tbody>
<tr>
<td>car</td>
<td></td>
<td></td>
</tr>
<tr>
<td>bomb</td>
<td></td>
<td></td>
</tr>
<tr>
<td>that</td>
<td></td>
<td></td>
</tr>
<tr>
<td>exploded</td>
<td>V: explode</td>
<td></td>
</tr>
<tr>
<td>outside</td>
<td>location [AM-LOC]</td>
<td></td>
</tr>
<tr>
<td>the</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U.S.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>military</td>
<td>temporal [AM-TMP]</td>
<td></td>
</tr>
<tr>
<td>base</td>
<td></td>
<td></td>
</tr>
<tr>
<td>in</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Beniji</td>
<td></td>
<td></td>
</tr>
<tr>
<td>killed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Iraqi</td>
<td></td>
<td></td>
</tr>
<tr>
<td>citizens</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- SRL reveals relations and arguments in the sentence (where relations are expressed as verbs)
- Cannot abstract over variability of expressing the relations – e.g. kill vs. murder vs. slay…
Text categorization

- Lots of possible tasks and types of text
  - turns in dialogue, user review, whole discussion, news article
- Is it spam? (see features)
- Sentiment and Subjectivity:
  - Is it positive?
  - Is it negative?
  - Is it objective?
  - Is it insulting?
- Dialogue Structure:
  - Is it a disagreement?
  - Is it sarcastic?
Sentiment and Subjectivity analysis: general area of much of what we are reading

- Also known as opinion mining
- A kind of text classification that attempts to identify the opinion/sentiment that a person may hold towards an object
- Sentiment a bit finer grained analysis compared to subjectivity analysis

<table>
<thead>
<tr>
<th>Sentiment Analysis</th>
<th>Subjectivity analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive</td>
<td>Subjective</td>
</tr>
<tr>
<td>Negative</td>
<td></td>
</tr>
<tr>
<td>Neutral</td>
<td>Objective</td>
</tr>
</tbody>
</table>
Sentiment classification

What features of the text could help predict # of stars?
How to identify more?
Are the features hard to compute? (syntax? sarcasm?)

An extremely versatile machine!, November 22, 2006

By Dr. Nickolas E. Jorgensen "njorgens3"

This review is from: Cuisinart DGB-600BC Grind & Brew, Brushed Chrome (Kitchen)
This coffee-maker does so much! It makes weak, watery coffee! It grinds beans if you want it to! It inexplicably floods the entire counter with half-brewed coffee when you aren't looking! Perhaps it could be used to irrigate crops... It is time-consuming to clean, but in fairness I should also point out that the stainless-steel thermal carafe is a durable item that has withstood being hurled onto the floor in rage several times. And if all these features weren't enough, it's pretty expensive too. If faced with the choice between having a car door repeatedly slamming into my genitalia and buying this coffee-maker, I'd unhesitatingly choose the Cuisinart! The coffee would be lousy, but at least I could still have children...
Components of an opinion

- Basic components of an opinion:
  - Opinion holder: The person or organization that holds a specific opinion on a particular object.
  - Object: on which an opinion is expressed
  - Opinion: a view, attitude, or appraisal on an object from an opinion holder.
Opinion Mining Tasks (cont.)

- **At the feature level:**
  - **Task 1:** Identify and extract object features that have been commented on by an opinion holder (e.g., a reviewer).
  - **Task 2:** Determine whether the opinions on the features are positive, negative or neutral.
  - **Task 3:** Group feature synonyms.
    - Produce a feature-based opinion summary of multiple reviews.

- **Opinion holders:** identify holders is also useful, e.g., in news articles, etc, but they are usually known in the user generated content, i.e., authors of the posts.
Facts and Opinions

- Two main types of textual information on the Web.
  - Facts and Opinions

- Current search engines search for facts (assume they are true)
  - Facts can be expressed with topic keywords.

- Search engines do not search for opinions
  - Opinions are hard to express with a few keywords
    - What do people think of Motorola Cell phones?
  - Current search ranking strategy is not appropriate for opinion retrieval/search.
## Applications

- **Businesses and organizations:**
  - product and service benchmarking.
  - market intelligence.
  - Business spends a huge amount of money to find consumer sentiments and opinions.
    - Consultants, surveys and focused groups, etc
- **Individuals:** interested in other’s opinions when
  - purchasing a product or using a service,
  - finding opinions on political topics
- **Ads placements:** Placing ads in the user-generated content
  - Place an ad when one praises a product.
  - Place an ad from a competitor if one criticizes a product.
- **Opinion retrieval/search:** providing general search for opinions.
Adjectives

- **positive:** honest important mature large patient

- Ron Paul is the only *honest* man in Washington.
- Kitchell’s writing is unbelievably *mature* and is only likely to get better.
- To humour me my *patient* father agrees yet again to my choice of film
Words

- **Adjectives**
  - negative: harmful hypocritical inefficient insecure
    - It was a macabre and hypocritical circus.
    - Why are they being so inefficient?
Words

- **Adjectives**
  - Subjective (but not positive or negative sentiment): curious, peculiar, odd, likely, probable
    - He spoke of Sue as his probable successor.
    - The two species are likely to flower at different times.
Words

  
  - **Verbs**
    - positive: *praise*, *love*
    - negative: *blame*, *criticize*
    - subjective: *predict*
  
  - **Nouns**
    - positive: *pleasure*, *enjoyment*
    - negative: *pain*, *criticism*
    - subjective: *prediction*, *feeling*
Phrases

- Phrases containing adjectives and adverbs: Turney 2002, Takamura, Inui & Okumura 2007
  - positive: high intelligence, low cost
  - negative: little variation, many troubles
How? Patterns

- **Lexico-syntactic patterns** Riloff & Wiebe 2003
- **way with <np>:** … to ever let China use force to have its way with …
- **expense of <np>:** at the expense of the world’s security and stability
- **underlined <dobj>:** Jiang’s subdued tone … underlined his desire to avoid disputes …
How?

- How do we identify subjective items?
- Assume that contexts are coherent
- Also remember the plot units paper? Evil agents do evil things....
Conjunction

The Homestay Experience - Cultural Kaleidoscope 2006
My host's home was very nice and comfortable. I got to try all types of food; Malaysian, Chinese, Indonesian and I loved it all. My host's parents were very ...
www.gardenschool.edu.my/studentportal/aec/Kaleidoscope06/experience.asp - 10k - Cached - Similar pages - Note this

PriceGrabber User Rating for Watch Your Budget - PriceGrabber.com
Reviews, Camera I purchased was very nice and a bargain. There was a problem with shipping, but was resolved quickly. Buy with confidence from this vendor. ...
www.pricegrabber.com/rating_getreview.php?retid=5821 - Similar pages - Note this

Testimonials
"Everybody was very nice and service was as fast as they possibly could. ... "Staff member who helped me was very nice and easy to talk to." ...
www.sa.psu.edu/uhs/news/testimonials.cfm - 22k - Cached - Similar pages - Note this

Naxos Villages - Naxos Town or Chora Reviews: Very nice and very ...
-Did you enjoy the trip to Naxos Town: Yes it was very nice and very scenic. -In order to get to the village were there enough signs in order to find it: It ...
Statistical association

- If words of the same orientation likely to co-occur together, then the presence of one makes the other more probable (co-occur within a window, in a particular context, etc.)

- Use statistical measures of association to capture this interdependence
  - E.g., Mutual Information (Church & Hanks 1989)
How?

- How do we identify subjective items?

- Assume that contexts are coherent

- Assume that alternatives are similarly subjective ("plug into" subjective contexts)
How? Summary

- How do we identify subjective items?
- Assume that contexts are coherent
- Assume that alternatives are similarly subjective
- Take advantage of specific words
<table>
<thead>
<tr>
<th>object</th>
<th>15651</th>
<th>5.8</th>
<th>subject</th>
<th>9100</th>
<th>6.2</th>
<th>modifier</th>
<th>1971</th>
<th>1.4</th>
<th>and/or</th>
<th>130</th>
<th>0.0</th>
<th>pp_by-p</th>
<th>3374</th>
<th>16.4</th>
</tr>
</thead>
<tbody>
<tr>
<td>damage</td>
<td>938</td>
<td>10.09</td>
<td>negligence</td>
<td>55</td>
<td>7.34</td>
<td>reasonable</td>
<td>26</td>
<td>8.72</td>
<td>permit</td>
<td>18</td>
<td>6.09</td>
<td>negligence</td>
<td>46</td>
<td>8.14</td>
</tr>
<tr>
<td>harm</td>
<td>276</td>
<td>8.91</td>
<td>virus</td>
<td>53</td>
<td>7.14</td>
<td>indirectly</td>
<td>16</td>
<td>7.77</td>
<td>contribute</td>
<td>5</td>
<td>4.21</td>
<td>defect</td>
<td>21</td>
<td>6.81</td>
</tr>
<tr>
<td>injury</td>
<td>295</td>
<td>8.38</td>
<td>smoking</td>
<td>27</td>
<td>6.36</td>
<td>possibly</td>
<td>27</td>
<td>6.77</td>
<td>use</td>
<td>6</td>
<td>0.22</td>
<td>bacterium</td>
<td>17</td>
<td>6.62</td>
</tr>
<tr>
<td>problem</td>
<td>1014</td>
<td>8.37</td>
<td>defect</td>
<td>29</td>
<td>6.32</td>
<td>thereby</td>
<td>26</td>
<td>7.66</td>
<td>virus</td>
<td>17</td>
<td>6.4</td>
<td>smoking</td>
<td>13</td>
<td>6.4</td>
</tr>
<tr>
<td>trouble</td>
<td>249</td>
<td>8.32</td>
<td>bacterium</td>
<td>26</td>
<td>6.23</td>
<td>mainly</td>
<td>32</td>
<td>7.51</td>
<td>fault</td>
<td>19</td>
<td>6.15</td>
<td>fault</td>
<td>19</td>
<td>6.15</td>
</tr>
<tr>
<td>death</td>
<td>383</td>
<td>7.96</td>
<td>infection</td>
<td>32</td>
<td>6.09</td>
<td>inevitably</td>
<td>20</td>
<td>7.48</td>
<td>lack</td>
<td>37</td>
<td>5.98</td>
<td>lack</td>
<td>37</td>
<td>5.98</td>
</tr>
<tr>
<td>delay</td>
<td>146</td>
<td>7.87</td>
<td>factor</td>
<td>76</td>
<td>6.07</td>
<td>partly</td>
<td>22</td>
<td>7.47</td>
<td>deficiency</td>
<td>10</td>
<td>5.95</td>
<td>deficiency</td>
<td>10</td>
<td>5.95</td>
</tr>
<tr>
<td>confusion</td>
<td>137</td>
<td>7.8</td>
<td>assault</td>
<td>28</td>
<td>6.05</td>
<td>probably</td>
<td>51</td>
<td>7.1</td>
<td>shortage</td>
<td>12</td>
<td>5.75</td>
<td>shortage</td>
<td>12</td>
<td>5.75</td>
</tr>
<tr>
<td>difficulty</td>
<td>223</td>
<td>7.74</td>
<td>pollution</td>
<td>31</td>
<td>6.04</td>
<td>thus</td>
<td>36</td>
<td>7.01</td>
<td>blockage</td>
<td>6</td>
<td>5.71</td>
<td>blockage</td>
<td>6</td>
<td>5.71</td>
</tr>
<tr>
<td>disruption</td>
<td>111</td>
<td>7.71</td>
<td>recession</td>
<td>28</td>
<td>5.99</td>
<td>recklessly</td>
<td>8</td>
<td>6.97</td>
<td>breach</td>
<td>14</td>
<td>5.66</td>
<td>breach</td>
<td>14</td>
<td>5.66</td>
</tr>
<tr>
<td>distress</td>
<td>101</td>
<td>7.52</td>
<td>stress</td>
<td>28</td>
<td>5.88</td>
<td>in part</td>
<td>10</td>
<td>6.94</td>
<td>parasite</td>
<td>7</td>
<td>5.65</td>
<td>parasite</td>
<td>7</td>
<td>5.65</td>
</tr>
<tr>
<td>concern</td>
<td>190</td>
<td>7.35</td>
<td>accident</td>
<td>36</td>
<td>5.8</td>
<td>undoubtedly</td>
<td>11</td>
<td>6.91</td>
<td>default</td>
<td>7</td>
<td>5.59</td>
<td>default</td>
<td>7</td>
<td>5.59</td>
</tr>
<tr>
<td>pain</td>
<td>126</td>
<td>7.24</td>
<td>bomb</td>
<td>26</td>
<td>5.78</td>
<td>deliberately</td>
<td>14</td>
<td>6.81</td>
<td>error</td>
<td>18</td>
<td>5.58</td>
<td>error</td>
<td>18</td>
<td>5.58</td>
</tr>
<tr>
<td>chaos</td>
<td>82</td>
<td>7.22</td>
<td>disease</td>
<td>50</td>
<td>5.74</td>
<td>intentionally</td>
<td>7</td>
<td>6.77</td>
<td>abnormality</td>
<td>7</td>
<td>5.38</td>
<td>abnormality</td>
<td>7</td>
<td>5.38</td>
</tr>
<tr>
<td>accident</td>
<td>126</td>
<td>7.22</td>
<td>fire</td>
<td>45</td>
<td>5.63</td>
<td>intentionall</td>
<td>7</td>
<td>6.77</td>
<td>theft</td>
<td>9</td>
<td>5.58</td>
<td>theft</td>
<td>9</td>
<td>5.58</td>
</tr>
<tr>
<td>loss</td>
<td>190</td>
<td>7.14</td>
<td>lack</td>
<td>37</td>
<td>5.6</td>
<td>sometimes</td>
<td>31</td>
<td>6.69</td>
<td>pollution</td>
<td>14</td>
<td>5.57</td>
<td>pollution</td>
<td>14</td>
<td>5.57</td>
</tr>
<tr>
<td>controversy</td>
<td>81</td>
<td>7.12</td>
<td>organism</td>
<td>18</td>
<td>5.58</td>
<td>often</td>
<td>72</td>
<td>6.67</td>
<td>enteritis</td>
<td>5</td>
<td>5.53</td>
<td>enteritis</td>
<td>5</td>
<td>5.53</td>
</tr>
<tr>
<td>pollution</td>
<td>88</td>
<td>7.04</td>
<td>deficiency</td>
<td>16</td>
<td>5.57</td>
<td>directly</td>
<td>24</td>
<td>6.66</td>
<td>build-up</td>
<td>6</td>
<td>5.51</td>
<td>build-up</td>
<td>6</td>
<td>5.51</td>
</tr>
<tr>
<td>havoc</td>
<td>64</td>
<td>7.01</td>
<td>fault</td>
<td>21</td>
<td>5.57</td>
<td>reportedly</td>
<td>9</td>
<td>6.64</td>
<td>fall</td>
<td>17</td>
<td>5.5</td>
<td>fall</td>
<td>17</td>
<td>5.5</td>
</tr>
<tr>
<td>cancer</td>
<td>93</td>
<td>7.01</td>
<td>delay</td>
<td>20</td>
<td>5.55</td>
<td>usually</td>
<td>37</td>
<td>6.61</td>
<td>exposure</td>
<td>11</td>
<td>5.5</td>
<td>exposure</td>
<td>11</td>
<td>5.5</td>
</tr>
<tr>
<td>stir</td>
<td>62</td>
<td>7.0</td>
<td>damage</td>
<td>30</td>
<td>5.51</td>
<td>either</td>
<td>28</td>
<td>6.49</td>
<td>warming</td>
<td>6</td>
<td>5.46</td>
<td>warming</td>
<td>6</td>
<td>5.46</td>
</tr>
<tr>
<td>suffering</td>
<td>70</td>
<td>6.99</td>
<td>weather</td>
<td>22</td>
<td>5.39</td>
<td>primarily</td>
<td>10</td>
<td>6.45</td>
<td>drought</td>
<td>6</td>
<td>5.45</td>
<td>drought</td>
<td>6</td>
<td>5.45</td>
</tr>
<tr>
<td>disease</td>
<td>141</td>
<td>6.95</td>
<td>explosion</td>
<td>15</td>
<td>5.27</td>
<td>largely</td>
<td>18</td>
<td>6.39</td>
<td>failure</td>
<td>24</td>
<td>5.38</td>
<td>failure</td>
<td>24</td>
<td>5.38</td>
</tr>
<tr>
<td>explosion</td>
<td>72</td>
<td>6.93</td>
<td>drought</td>
<td>12</td>
<td>5.27</td>
<td>allegedly</td>
<td>7</td>
<td>6.39</td>
<td>recession</td>
<td>11</td>
<td>5.36</td>
<td>recession</td>
<td>11</td>
<td>5.36</td>
</tr>
<tr>
<td>embarrassment</td>
<td>63</td>
<td>6.87</td>
<td>parasite</td>
<td>12</td>
<td>5.26</td>
<td>also</td>
<td>196</td>
<td>6.39</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Performance Measures (again)
Measuring Performance

- **Classification accuracy**: What % of messages were classified correctly?
- **Is this what we care about?**

<table>
<thead>
<tr>
<th></th>
<th>Overall accuracy</th>
<th>Accuracy on spam</th>
<th>Accuracy on gen</th>
</tr>
</thead>
<tbody>
<tr>
<td>System 1</td>
<td>95%</td>
<td>99.99%</td>
<td>90%</td>
</tr>
<tr>
<td>System 2</td>
<td>95%</td>
<td>90%</td>
<td>99.99%</td>
</tr>
</tbody>
</table>

- Which system do you prefer?
Measuring Performance

- **Precision** = \( \frac{\text{good messages kept}}{\text{all messages kept}} \)
- **Recall** = \( \frac{\text{good messages kept}}{\text{all good messages}} \)

Move from high precision to high recall by deleting fewer messages (delete only if spamminess > high threshold)
Measuring Performance

- **Low threshold:**
  - Keep all the good stuff,
  - But a lot of the bad too

- **High threshold:**
  - All we keep is good,
  - But we don’t keep much

- OK for spam filtering and legal search
- OK for search engines (users only want top 10)

- Point where precision=recall (occasionally reported)
- Would prefer to be here!
Measuring Performance

- **Precision** = \( \frac{\text{good messages kept}}{\text{all messages kept}} \)
- **Recall** = \( \frac{\text{good messages kept}}{\text{all good messages}} \)
- **F-measure** = \( \frac{1}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}}} \)

Move from high precision to high recall by deleting fewer messages (raise threshold)

Conventional to tune system and threshold to optimize F-measure on dev data

But it’s more informative to report the whole curve

Since in real life, the user should be able to pick a tradeoff point they like
How do we find things we want to look at?

Regular Expressions: who has never seen these?
Regular expressions

- A formal language for specifying text strings
- How can we search for any of these?
  - woodchuck
  - woodchucks
  - Woodchuck
  - Woodchucks
Regular Expressions: Disjunctions

- Letters inside square brackets []

    | Pattern          | Matches                  |
    |------------------|--------------------------|
    | [wW]oodchuck     | Woodchuck, woodchuck     |
    | [1234567890]     | Any digit                |

- Ranges [A–Z]

    | Pattern | Matches                      |
    |---------|------------------------------|
    | [A–Z]   | An upper case letter         |
    |         | Drenched Blossoms            |
    | [a–z]   | A lower case letter          |
    |         | my beans were impatient      |
    | [0–9]   | A single digit               |
    |         | Chapter 1: Down the Rabbit Hole |
Regular Expressions: Negation in Disjunction

- **Negations** \[^{\text{Ss}}\]
  - Carat means negation only when first in []

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Matches</th>
</tr>
</thead>
<tbody>
<tr>
<td>[^A-Z]</td>
<td>Not an upper case letter</td>
</tr>
<tr>
<td>[^\text{Ss}]</td>
<td>Neither ‘S’ nor ‘s’</td>
</tr>
<tr>
<td>[^e]</td>
<td>Neither e nor ^</td>
</tr>
<tr>
<td>a^b</td>
<td>The pattern a carat b</td>
</tr>
</tbody>
</table>

- Oyfn pripetchik
- I have no exquisite reason”
- Look here
- Look up a^b now
Regular Expressions: More Disjunction

- Woodchucks is another name for groundhog!
- The pipe | for disjunction

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Matches</th>
</tr>
</thead>
<tbody>
<tr>
<td>groundhog</td>
<td>woodchuck</td>
</tr>
<tr>
<td>yours</td>
<td>mine</td>
</tr>
<tr>
<td>mine</td>
<td></td>
</tr>
<tr>
<td>a</td>
<td>b</td>
</tr>
<tr>
<td>[gG]roundhog</td>
<td>[Ww]oodchuck</td>
</tr>
</tbody>
</table>

Photo D. Fletcher

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Matches</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>colour?r</td>
<td>Optional previous char</td>
<td>colour, colour</td>
</tr>
<tr>
<td>oo*h!</td>
<td>0 or more of previous char</td>
<td>oh!, ooh!, oooh!, ooooh!</td>
</tr>
<tr>
<td>o+h!</td>
<td>1 or more of previous char</td>
<td>oh!, ooh!, oooh!, ooooh!</td>
</tr>
<tr>
<td>baa+</td>
<td></td>
<td>baa, baaa, baaaa, baaaaa</td>
</tr>
<tr>
<td>beg.n</td>
<td></td>
<td>begin, begun, begun, begun, begun, begun, begun, begun</td>
</tr>
</tbody>
</table>

Stephen C Kleene

Kleene *, Kleene +
**Example**

- Find me all instances of the word “the” in a text.

  `the`

  Misses capitalized examples

  `[tT]he`

  Incorrectly returns other or theology

  `[^a-zA-Z][tT]he[^a-zA-Z]`
Errors

- The process we just went through was based on fixing two kinds of errors
  - Matching strings that we should not have matched (there, then, other)
    - False positives (Type I)
  - Not matching things that we should have matched (The)
    - False negatives (Type II)
Errors cont.

- In NLP we are always dealing with these kinds of errors.
- Reducing the error rate for an application often involves two antagonistic efforts:
  - Increasing accuracy or precision (minimizing false positives)
  - Increasing coverage or recall (minimizing false negatives).
Summary

- Regular expressions play a surprisingly large role
  - Sophisticated sequences of regular expressions are often the first model for any text processing text
- For many hard tasks, we use machine learning classifiers
  - But regular expressions are used as features in the classifiers
  - Can be very useful in capturing generalizations
Parsing
Dependency Trees

- Shows which words modify (“depend on”) another word
- Each subtree of the dependency tree is still a constituent
  - But not all of the original constituents are subtrees (e.g., VP)

- Easy to spot semantic relations (“who did what to whom?”)
  - Good source of syntactic features for other tasks
- Easy to annotate (high agreement)
- Easy to evaluate (what % of words have correct parent?)
Stanford dependencies tools

- http://nlp.stanford.edu/software/stanford-

Dependencies for Bills on ports and immigration were submitted by Senator Brownback, Republican of Kansas

Figure 1. Standard Stanford dependencies (collapsed and propagated)
He reckons the current account deficit will narrow to only 1.8 billion in September.

Mailing lists
To ask questions about the dependencies, you can use the same lists as for the parser, each @lists.stanford.edu:

1. parser-user  This is the best list to post to in order to ask questions, make announcements, or for discussion among parser users. Join the list via this webpage or by emailing parser-user-join@lists.stanford.edu. (Leave the subject and message body empty.) You can also look at the list archives.
2. parser-announce  This list will be used only to announce new parser versions. So it will be very low volume (expect 1-3 messages a year). Join the list via this webpage or by emailing parser-announce-join@lists.stanford.edu. (Leave the subject and message body empty.)
3. parser-support  This list goes only to the parser maintainers. It's a good address for licensing questions, etc. For general use and support questions, you're better off joining and using parser-user. You cannot join parser-support, but you can mail questions to parser-support@lists.stanford.edu.

• The dependencies are produced using hand-writtent regex patterns over phrase-structure trees as described in: Marie-Catherine de Marneffe, Bill MacCartney and Christopher D. Manning. 2006.


Dependency Trees vs. Phrase Structure Trees
Det The plan

VP

V has

NP

Wanda

VP

V been

NP

Otto

VP

V swallow

NP

Wanda

1. Assign heads
2. Each word is the head of a whole connected subgraph.
The plan to swallow Wanda has been thrilling Otto.
Dependency Trees

3. Just look at which words are related
Dependency Trees

- Shows which words modify ("depend on") another word
- Each subtree of the dependency tree is still a constituent
  - But not all of the original constituents are subtrees (e.g., VP)

- Easy to spot semantic relations ("who did what to whom?")
  - Good source of syntactic features for other tasks
- Easy to annotate (high agreement)
- Easy to evaluate (what % of words have correct parent?)
Semantic role labeling
Semantic Role Labeling (SRL)

- For each **predicate** (e.g., verb)
  1. find its arguments (e.g., NPs)
  2. determine their **semantic roles**

John **drove** Mary from Austin to Dallas in his Toyota Prius.

The hammer **broke** the window.

- **agent**: Actor of an action
- **patient**: Entity affected by the action
- **source**: Origin of the affected entity
- **destination**: Destination of the affected entity
- **instrument**: Tool used in performing action.
- **beneficiary**: Entity for whom action is performed
Uses of Semantic Roles

- Find the answer to a user’s question
  - “Who” questions usually want Agents
  - “What” question usually want Patients
  - “How” and “with what” questions usually want Instruments
  - “Where” questions frequently want Sources/Destinations.
  - “For whom” questions usually want Beneficiaries
  - “To whom” questions usually want Destinations

- Generate text
  - Many languages have specific syntactic constructions that must or should be used for specific semantic roles.

- Word sense disambiguation, using selectional restrictions
  - The **bat ate** the **bug**.  (what kind of bat? what kind of bug?)
    - Agents (particularly of “eat”) should be animate – animal bat, not baseball bat
    - Patients of “eat” should be edible – animal bug, not software bug
  - John **fired** the secretary.
    - John **fired** the rifle.
      - Patients of fire$_1$ are different than patients of fire$_2$
As usual, can solve as classification …

- Consider one verb at a time: “bit”
- Classify the role (if any) of each of the 3 NPs

Color Code:
- not-a-role
- agent
- patient
- source
- destination
- instrument
- beneficiary

Slide thanks to Ray Mooney (modified)
Parse tree paths as classification features

Path feature is

\[ V \uparrow VP \uparrow S \downarrow NP \]

which tends to be associated with agent role
Parse tree paths as classification features

Path feature is

V ↑ VP ↑ S ↓ NP ↓ PP ↓ NP

which tends to be associated with no role

Slide thanks to Ray Mooney (modified)
Head words as features

- Some roles prefer to be filled by certain kinds of NPs.
- This can give us useful features for classifying accurately:
  - “John ate the spaghetti with chopsticks.” (instrument)
  - “John ate the spaghetti with meatballs.” (patient)
  - "John ate the spaghetti with Mary."
    - Instruments should be tools
    - Patient of “eat” should be edible
  - “John bought the car for $21K.” (instrument)
  - “John bought the car for Mary.” (beneficiary)
    - Instrument of “buy” should be Money
    - Beneficiaries should be animate (things with desires)
  - “John drove Mary to school in the van”
  - “John drove the van to work with Mary.”
    - What do you think?

Slide thanks to Ray Mooney (modified)
Other Current Semantic Annotation Tasks (similar to SRL)

- PropBank – coarse-grained roles of verbs
- NomBank – similar, but for nouns
- FrameNet – fine-grained roles of any word
- TimeBank – temporal expressions
FrameNet Example

We avenged the insult by setting fire to his village.

- **Avenger**
- **Offender** (unexpressed in this sentence)
- **Injury**
- **Injured Party** (unexpressed in this sentence)
- **Punishment**

A word/phrase that triggers the REVENGE frame.
FrameNet Example

REVENGE FRAME

triggering words and phrases
(not limited to verbs)

avenge, revenge, retaliate, get back at, pay back, get even, …
revenge, vengeance, retaliation, retribution, reprisal, …
vengeful, retaliatory, retributive; in revenge, in retaliation, …
take revenge, wreak vengeance, exact retribution, …

Slide thanks to CJ Fillmore (modified)
## Framenet for hiring: APIs

<table>
<thead>
<tr>
<th>Hiring</th>
<th>Definition</th>
<th>Examples</th>
<th>Some core elements</th>
<th>Some non-core elements</th>
<th>Frame relations</th>
<th>Lexical units that activate this frame</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>An <strong>Employer</strong> hires an <strong>Employee</strong>, promising the <strong>Employee</strong> a certain <strong>Compensation</strong> in exchange for the performance of a job. The job may be described either in terms of a <strong>Task</strong> or a <strong>Position</strong>.</td>
<td>John was HIRED to clean up the file system. IBM HIRED Gates as chief janitor.</td>
<td><strong>Employee</strong>: the person whom the <strong>Employer</strong> takes on as an <strong>Employee</strong>, obligating them to perform some <strong>Task</strong> in order to receive <strong>Compensation</strong>. <strong>Employer</strong>: the person (or institution) that takes on an <strong>Employee</strong>, giving them <strong>Compensation</strong> in return for the performance of an assigned <strong>Task</strong>. <strong>Task</strong>: the action that the <strong>Employee</strong> is taken on by the <strong>Employer</strong> to do.</td>
<td><strong>Place</strong>: this FE identifies the <strong>Place</strong> where the <strong>Employer</strong> hires the <strong>Employee</strong>. <strong>Time</strong>: this FE identifies the <strong>Time</strong> when an <strong>Employer</strong> hires the <strong>Employee</strong>.</td>
<td><strong>Inherits From</strong>: Intentionally_affectSubframe of: <strong>Employer</strong>’s_scenario <strong>Precedes</strong>: <strong>Employing</strong> <strong>Perspective on</strong>: <strong>Employment</strong> <strong>start</strong></td>
<td><strong>commission.n</strong>, <strong>commission.v</strong>, <strong>contract.v</strong>, <strong>give job.v</strong>, <strong>hire.n</strong>, <strong>hire.v</strong>, <strong>retain.v</strong>, <strong>sign on.v</strong>, <strong>sign up.v</strong>, <strong>sign.v</strong>, <strong>subcontract.v</strong>, <strong>take on.v</strong></td>
</tr>
</tbody>
</table>

Table 5. Reduced version of *hiring* frame.
Back to Discourse and Dialogue
Dialogue Structure

- What makes a text coherent?
- What are discourse structures?
- Theories of discourse structures
- Approaches to build discourse structures
Event anaphora resolution, Discourse Deixis

- Hardly any work has been done in the last 25 years
  - Webber 1988, Byron 2005
  - Recent interest with CoNLL competitions
  - Tons of it in the forums data

<table>
<thead>
<tr>
<th>Stance</th>
<th>Post</th>
</tr>
</thead>
<tbody>
<tr>
<td>FOR</td>
<td>Studies have shown that using the death penalty saves 4 to 13 lives per execution. That alone makes killing murderers worthwhile.</td>
</tr>
<tr>
<td>AGAINST</td>
<td>What studies? I have never seen ANY evidence that capital punishment acts as a deterrent to crime. I have not seen any evidence that it is “just” either.</td>
</tr>
<tr>
<td>FOR</td>
<td>When Texas and Florida were executing people one after the other in the late 90’s, the murder rates in both states plunged, like Rosie O’donnel off a diet.</td>
</tr>
<tr>
<td>AGAINST</td>
<td>That’s your evidence? What happened to those studies? In the late 90s a LOT of things were different than the periods preceding and following the one you mention. We have no way to determine what of those contributed to a lower murder rate, if indeed there was one. You have to prove a cause and effect relationship and you have failed.</td>
</tr>
</tbody>
</table>
Inference of discourse relations

- within a speaker’s turn or across speakers or turns
  - Intentional or rhetorical, narrative structure etc
Discourse structure

S1: John took a train to Bill’s car dealership.
S2: He needed to buy a car.
S3: The company he works for now isn’t near any public transportation.
S4: He also wanted to talk to Bill about their softball leagues.

Explanation
Discourse structure

S1: John took a train to Bill’s car dealership.
S2: He needed to buy a car.
S3: The company he works for now isn’t near any public transportation.
S4: He also wanted to talk to Bill about their softball leagues.
S1: John took a train to Bill’s car dealership.
S2: He needed to buy a car.
S3: The company he works for now isn’t near any public transportation.
S4: He also wanted to talk to Bill about their softball leagues.
Discourse Relations

- Penn Discourse Tree Bank
- Mechanical Turk? Hand Label?
- Look at correlations (section of our WASSA paper)
- Could do like Marcu and Echihabi?
  - Take them out and see if you can put them back.
Questions?