Dialogue & Narrative Structures: Advanced Research Seminar in NLP and Narrative

Prof. Marilyn Walker
Baskin School of Engineering
University of California, Santa Cruz
Announcements.

- Reid is in charge this week (I am away giving a talk).
- Today: class session on Stanford Toolkit. Will help with everything that follows that uses it (e.g. Reids thesis work, Riloff plot structures, Narrative Schema etc).
- Homework 3: homework to familiarize yourself with Stanford toolkit. Due next Tuesday.
- May be useful to look ahead and Riloff paper to see how she uses parse structures when you are playing around with the Stanford parser.
- Will link to Aesop Fables corpus so we can play with it.
Syllabus & Course Structure

- [http://courses.soe.ucsc.edu/courses/cmsps245/Spring13/01/pages/computational-models](http://courses.soe.ucsc.edu/courses/cmsps245/Spring13/01/pages/computational-models)

- Is everyone signed up on Piazza now?
Coding stories in conversations (adapted from Labov & Waletzky, 1967)

I. A clause is the unit of analysis for parsing a story
   A clause is basically a simple sentence containing one thought unit.

   Ex: “I went to the store / and got some sushi” = 2 clauses.

II. Elements of a story

   Action [action clauses build the plot of the story]
   An action clause has an active verb in the past tense (“So I hit the brakes”) or
   historical present tense (“so I’m hitting the brakes”) [versus “he was sitting
   there” – no change of state]

   Action clauses build on each other around some sort of problem (Bruner, 1990)

   Action clauses are not necessarily contiguous (next to each other); they can be interspersed
   with other information.

   Action clauses can be posed as statements (“I dashed to the window”), or questions (“Did
Sample story utterance from Thorne

Understanding L&W Narrative Analysis.

Homework 2. Please submit via Ecommons.

Here is the same story repeated 5 times. For each story, underline the appropriate clauses (e.g. Action, Temporal Juncture, Orientation, Evaluation and Coda) and separate underlined clauses with a slash (/)

1. Action.

It was a bright and sunny day, and Frank the cat was sitting under Chuck’s open birdcage. Suddenly Frank grabbed Chuck, dragged Chuck out of the cage, and ate him. I really miss Chuck, and I can’t stand to even look at Frank anymore. And that’s the end of the story.

2. Temporal juncture (underline the action clauses that require a fixed time sequence, and connect the clauses with an arrow.)

It was a bright and sunny day, and Frank the cat was sitting under Chuck’s open birdcage.
HW2: Who wants to talk about L&W on story blog?

- **Someone who hasn’t talked in front of class yet?**

  - http://theszmurlos.blogspot.com/2013/02/hawaii-2012-day-10-surfing-lessons.html
  - http://surfinggrandma.blogspot.com/2013/01/feeling-pretty-awmesome.html
  - http://theequestrianvagabond.blogspot.com/2013/02/owyheee-deputized.html
Stanford Toolkit: CoreNLP
A Pipeline of useful tools.

nlp.stanford.edu/software/corenlp.shtml

Stanford CoreNLP

A Suite of Core NLP Tools

About | Download | Usage | SUTime | Adding Annotators | Caseless Models | Extensions | Questions | Mailing lists | Online demo | FAQ | Release history

About

Stanford CoreNLP provides a set of natural language analysis tools which can take raw English language text input and give the base forms of words, their parts of speech, whether they are names of companies, people, etc., normalize dates, times, and numeric quantities, and mark up the structure of sentences in terms of phrases and word dependencies, and indicate which noun phrases refer to the same entities. Stanford CoreNLP is an integrated framework, which make it very easy to apply a bunch of language analysis tools to a piece of text. Starting from plain text, you can run all the tools on it with just two lines of code. Its analyses provide the foundational building blocks for higher-level and domain-specific text understanding applications.

Stanford CoreNLP integrates all our NLP tools, including the part-of-speech (POS) tagger, the named entity recognizer (NER), the parser, and the coreference resolution system, and provides model files for analysis of English. The goal of this project is to enable people to quickly and painlessly get complete linguistic annotations of natural language texts. It is designed to be highly flexible and extensible. With a single option you can change which tools should be enabled and which should be disabled.

The Stanford CoreNLP code is written in Java and licensed under the GNU General Public License (v2 or later). Source is included. Note that this is the full GPL, which allows many free uses, but not its use in distributed proprietary software. The download is 259 MB and requires Java 1.6+.

Download

Download Stanford CoreNLP version 1.3.5.

Or you can find it on Maven Central.
<table>
<thead>
<tr>
<th>Property name</th>
<th>Annotator class name</th>
<th>Generated Annotation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tokenize</td>
<td>PTBTokenizerAnnotator</td>
<td>TokensAnnotation (list of tokens), and CharacterOffsetBeginAnnotation, CharacterOffsetEndAnnotation, TextAnnotation (for each token)</td>
<td>Tokenizes the text. This component started as a PTB-style tokenizer, but was extended since then to handle noisy and web text. The tokenizer saves the character offsets of each token in the input text, as CharacterOffsetBeginAnnotation and CharacterOffsetEndAnnotation.</td>
</tr>
<tr>
<td>cleanxml</td>
<td>CleanXmlAnnotator</td>
<td>XmiContextAnnotation</td>
<td>Remove xml tokens from the document</td>
</tr>
<tr>
<td>ssplit</td>
<td>WordToSentenceAnnotator</td>
<td>SentencesAnnotation</td>
<td>Splits a sequence of tokens into sentences.</td>
</tr>
<tr>
<td>pos</td>
<td>POSTaggerAnnotator</td>
<td>PartOfSpeechAnnotation</td>
<td>Labels tokens with their POS tag. For more details see this page.</td>
</tr>
<tr>
<td>lemma</td>
<td>MorphaAnnotator</td>
<td>LemmaAnnotation</td>
<td>Generates the word lemmas for all tokens in the corpus.</td>
</tr>
<tr>
<td>ner</td>
<td>NERClassifierCombiner</td>
<td>NamedEntityTagAnnotation and NormalizedNamedEntityTagAnnotation</td>
<td>Recognizes named (PERSON, LOCATION, ORGANIZATION, MISC) and numerical entities (DATE, TIME, MONEY, NUMBER). Named entities are recognized using a combination of three CRF sequence taggers trained on various corpora, such as ACE and MUC. Numerical entities are recognized using a rule-based system. Numerical entities that require normalization, e.g., dates, are normalized to NormalizedNamedEntityTagAnnotation. For more details on the CRF tagger see this page.</td>
</tr>
<tr>
<td>regexner</td>
<td>RegexNERAnnotator</td>
<td>NamedEntityTagAnnotation</td>
<td>Implements a rule-based NER using Java regular expressions. The goal of this Annotator is to provide a simple framework to incorporate NE labels that are not annotated in traditional NL corpora. For example, the default list of regular expressions that we distribute recognizes ideologies (IDEOLOGY), nationalities (NATIONALITY), religions (RELIGION), and titles (TITLE).</td>
</tr>
<tr>
<td>ner</td>
<td>NERClassifierCombiner</td>
<td>NamedEntityTagAnnotation and NormalizedNamedEntityTagAnnotation</td>
<td></td>
</tr>
<tr>
<td>------------</td>
<td>------------------------</td>
<td>---------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>regexner</td>
<td>RegexNERAnnotator</td>
<td>NamedEntityTagAnnotation</td>
<td></td>
</tr>
<tr>
<td>truecase</td>
<td>TrueCaseAnnotator</td>
<td>TrueCaseAnnotation and TrueCaseTextAnnotation</td>
<td></td>
</tr>
<tr>
<td>parse</td>
<td>ParserAnnotator</td>
<td>TreeAnnotation, BasicDependenciesAnnotation, CollapsedDependenciesAnnotation, CollapsedCCProcessedDependenciesAnnotation</td>
<td></td>
</tr>
<tr>
<td>dcoref</td>
<td>DeterministicCoreAnnotator</td>
<td>CorefChainAnnotation</td>
<td></td>
</tr>
</tbody>
</table>

- Named entities, Parsing, Coreference
  
  Named entities are recognized using a rule-based system. Numerical entities that require normalization, e.g., dates, are normalized to NormalizedNamedEntityTagAnnotation. For more details on the CRF tagger see this page.

- Implements a rule-based NER using Java regular expressions. The goal of this Annotator is to provide a simple framework to incorporate NE labels that are not annotated in traditional NL corpora. For example, the default list of regular expressions that we distribute recognizes ideologies (IDEOLOGY), nationalities (NATIONALITY), religions (RELIGION), and titles (TITLE).

- Recognizes the true case of tokens in text where this information was lost, e.g., all upper case text. This is implemented with a discriminative model implemented using a CRF sequence tagger. The true case label, e.g., INIT_UPPER is saved in TrueCaseAnnotation. The token text adjusted to match its true case is saved as TrueCaseTextAnnotation.

- Provides full syntactic analysis, using both the constituent and the dependency representations. The constituent-based output is saved in TreeAnnotation. We generate three dependency-based outputs, as follows: basic, uncollapsed dependencies, saved in BasicDependenciesAnnotation; collapsed dependencies saved in CollapsedDependenciesAnnotation; and collapsed dependencies with processed coordinations, in CollapsedCCProcessedDependenciesAnnotation. Most users of our parser will prefer the latter representation. For more details on the parser, please see this page. For more details about the dependencies, please refer to this page.

- Implements both pronominal and nominal coreference resolution. The entire coreference graph (with head words of mentions as nodes) is saved in CorefChainAnnotation. For more details on the underlying coreference resolution algorithm, see this page.
Part–Of–Speech (POS) Tagger
Stanford POS Tagger

- A POS Tagger is a piece of software that reads text in some language and assigns parts of speech to each word.
  - such as noun, verb, adjective, etc.

- This software is a Java implementation of the log-linear part-of-speech tagger. The system requires Java 1.6+ to be installed.

- The following examples are base on 64 bit Windows OS.
Input: Plain Text

- Input:

But then, slowly all nine planets of our Solar System move into frame and align. The last of them is the giant, burning sphere of the sun. Just as the sun enters frame, a solar storm of gigantic proportion unfolds. The eruptions shoot thousands of miles into the blackness of space.
Input: Plain Text

- Command:

```
java -mx300m
-classpath stanford-postagger.jar
edu.stanford.nlp.tagger.maxent.MaxentTagger
-model models/bidirectional-wsj-0-18.tagger
-textFile input.txt > output.txt
```

But _CC then _RB , _, slowly _RB all _DT nine _CD planets _NNS of _IN our _PRP$ Solar _NNP System _NNP move _NN into _IN frame _NN and _CC align _NN __. The _DT last _JJ of _IN them _PRP is _VBZ the _DT giant _NN , _, burning _NN sphere _NN of _IN the _DT sun _NN __. Just _RB as _IN the _DT sun _NN enters _VBZ frame _NN , _, a _DT solar _JJ storm _NN of _IN gigantic _JJ proportion _NN unfolds _VBZ __. The _DT eruptions _NNS shoot _VBP thousands _NNS of _IN miles _NNS into _IN the _DT blackness _NN of _IN space _NN __. 
Last week, I rediscovered my yoga asana practice. After nearly five months with only a few attempts at practice I signed up for a month of classes at a studio near my apartment that offers Ashtanga, Iyengar and Hatha classes. Even a light Hatha class was a challenge after so long without practice. Coming out of our relaxation the end of class left me lingering on my mat for a few minutes, consumed by joy, sadness, and frustration.
Input: XML

- Command:


XML tags whose content we want the POS tagger to tag
Last week, I rediscovered my yoga asana practice. After nearly five months with only a few attempts at practice, I signed up for a month of classes at a studio near my apartment that offers Ashtanga, Iyengar and Hatha classes. Even a light Hatha class was a challenge after so long without practice. Coming out of our relaxation the end of class left me lingering on my mat for a few minutes, consumed by joy, sadness, and frustration.
The English taggers use the Penn Treebank tag set:

### The Penn Treebank POS tagset.

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CC</td>
<td>Coordinating conjunction</td>
<td>TO</td>
<td>to</td>
</tr>
<tr>
<td>CD</td>
<td>Cardinal number</td>
<td>UH</td>
<td>Interjection</td>
</tr>
<tr>
<td>DT</td>
<td>Determiner</td>
<td>VB</td>
<td>Verb, base form</td>
</tr>
<tr>
<td>EX</td>
<td>Existential <em>there</em></td>
<td>VBD</td>
<td>Verb, past tense</td>
</tr>
<tr>
<td>FW</td>
<td>Foreign word</td>
<td>VBG</td>
<td>Verb, gerund/present participle</td>
</tr>
<tr>
<td>IN</td>
<td>Preposition/subordinating conjunction</td>
<td>VBN</td>
<td>Verb, past participle</td>
</tr>
<tr>
<td>JJ</td>
<td>Adjective</td>
<td>VBP</td>
<td>Verb, non-3rd ps. sing. present</td>
</tr>
<tr>
<td>JJR</td>
<td>Adjective, comparative</td>
<td>VBPZ</td>
<td>Verb, 3rd ps. sing. present</td>
</tr>
<tr>
<td>JJS</td>
<td>Adjective, superlative</td>
<td>WDT</td>
<td><em>wh</em>-determiner</td>
</tr>
<tr>
<td>LS</td>
<td>List item marker</td>
<td>WP</td>
<td><em>wh</em>-pronoun</td>
</tr>
<tr>
<td>MD</td>
<td>Modal</td>
<td>WP$</td>
<td>Possessive <em>wh</em>-pronoun</td>
</tr>
<tr>
<td>NN</td>
<td>Noun, singular or mass</td>
<td>WRB</td>
<td><em>wh</em>-adverb</td>
</tr>
<tr>
<td>NNS</td>
<td>Noun, plural</td>
<td>#</td>
<td>Pound sign</td>
</tr>
<tr>
<td>NNP</td>
<td>Proper noun, singular</td>
<td>$</td>
<td>Dollar sign</td>
</tr>
<tr>
<td>NNPS</td>
<td>Proper noun, plural</td>
<td>.</td>
<td>Sentence-final punctuation</td>
</tr>
<tr>
<td>PDT</td>
<td>Predeterminer</td>
<td>,</td>
<td>Comma</td>
</tr>
<tr>
<td>POS</td>
<td>Possessive ending</td>
<td>:</td>
<td>Colon, semi-colon</td>
</tr>
<tr>
<td>PRP</td>
<td>Personal pronoun</td>
<td>(</td>
<td>Left bracket character</td>
</tr>
<tr>
<td>PP$</td>
<td>Possessive pronoun</td>
<td>)</td>
<td>Right bracket character</td>
</tr>
<tr>
<td>RB</td>
<td>Adverb</td>
<td>&quot;</td>
<td>Straight double quote</td>
</tr>
<tr>
<td>RBR</td>
<td>Adverb, comparative</td>
<td>'</td>
<td>Left open single quote</td>
</tr>
<tr>
<td>RBS</td>
<td>Adverb, superlative</td>
<td>&quot;</td>
<td>Left open double quote</td>
</tr>
<tr>
<td>RP</td>
<td>Particle</td>
<td>&quot;</td>
<td>Right close single quote</td>
</tr>
<tr>
<td>SYM</td>
<td>Symbol (mathematical or scientific)</td>
<td>&quot;</td>
<td>Right close double quote</td>
</tr>
</tbody>
</table>

Penn Treebank Tag set: [http://www.comp.leeds.ac.uk/amalgam/tagsets/upenn.html](http://www.comp.leeds.ac.uk/amalgam/tagsets/upenn.html)
Named Entity Tagger
Stanford Named Entity Recognizer (NER)

• Named Entity Recognition (NER) labels sequences of words in a text which are the names of things, such as person and company names.

• Recognizes named (PERSON, LOCATION, ORGANIZATION, MISC) and numerical entities (DATE, TIME, MONEY, NUMBER).
*Germany’s representative to the European Union’s veterinary committee Werner Zwingman said on Wednesday consumers should take action.*
Jim's income is 30000 dollars a year, which is 80 percent of his family income.
Stanford Named Entity Recognizer (NER)

Why NER?
- Question Answering
- Textual Entailment
- Coreference Resolution
- Computational Semantics
- ...
Stanford Dependencies

- The Stanford dependencies provide a representation of grammatical relations between words in a sentence.
- Simple descriptions, easily understood.
- Triples of a relation between pairs of words, such as “the subject of hit is Jim.”

- Reference: Stanford typed dependencies manual
Bell, based in Los Angeles, makes and distributes electronic, computer and building products.
Input: Bell, based in Los Angeles, makes and distributes electronic, computer and building products.

<table>
<thead>
<tr>
<th>Dependency</th>
<th>Definition</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nsubj</td>
<td>nominal subject</td>
<td>A nominal subject is a noun phrase which is the syntactic subject of a clause.</td>
</tr>
<tr>
<td>partmod</td>
<td>participial modifier</td>
<td>A participial modifier of an NP or VP or sentence is a participial verb form that serves to modify the meaning of a noun phrase or sentence.</td>
</tr>
<tr>
<td>nn</td>
<td>noun compound modifier</td>
<td>A noun compound modifier of an NP is any noun that serves to modify the head noun.</td>
</tr>
<tr>
<td>prep_in</td>
<td>prepositional modifier</td>
<td>A prepositional modifier of a verb, adjective, or noun is any prepositional phrase that serves to modify the meaning of the verb, adjective, noun, or even another preposition.</td>
</tr>
<tr>
<td>root</td>
<td>root</td>
<td>The root grammatical relation points to the root of the sentence.</td>
</tr>
<tr>
<td>conj_and</td>
<td>conjunct</td>
<td>A conjunct is the relation between two elements connected by a coordinating conjunction, such as “and”, “or”, etc.</td>
</tr>
<tr>
<td>amod</td>
<td>adjectival modifier</td>
<td>An adjectival modifier of an NP is any adjectival phrase that serves to modify the meaning of the NP.</td>
</tr>
<tr>
<td>dobj</td>
<td>indirect object</td>
<td>The indirect object of a VP is the noun phrase which is the (dative) object of the verb.</td>
</tr>
</tbody>
</table>
Example of how I used NER and Dependencies:

1. Annotate film scene files (tokenize, pos, lemma, ner, parse, dcoref)
2. Pick all the verbs (POS tags beginning with “VB”)
3. Figure out the subject and object of a verb (typed dependencies “nsubj”, “agent”, “dobj”, “iobj”, “nsubjpass”)
4. Generalize the subject and object (use NER results if they have, otherwise use lemma)
5. Integrate all the subjects and objects across all film scenes

**In one film scene file:** “Jack unlocks the door.”

<table>
<thead>
<tr>
<th>id</th>
<th>Word</th>
<th>Lemma</th>
<th>Char begin</th>
<th>Char end</th>
<th>POS</th>
<th>NER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Jack</td>
<td>Jack</td>
<td>0</td>
<td>4</td>
<td>NNP</td>
<td>PERSON</td>
</tr>
<tr>
<td>2</td>
<td>unlocks</td>
<td>unlock</td>
<td>5</td>
<td>12</td>
<td>VBZ</td>
<td>O</td>
</tr>
<tr>
<td>3</td>
<td>the</td>
<td>the</td>
<td>13</td>
<td>16</td>
<td>DT</td>
<td>O</td>
</tr>
<tr>
<td>4</td>
<td>door</td>
<td>door</td>
<td>17</td>
<td>21</td>
<td>NN</td>
<td>O</td>
</tr>
<tr>
<td>5</td>
<td>.</td>
<td>.</td>
<td>21</td>
<td>22</td>
<td>-</td>
<td>O</td>
</tr>
</tbody>
</table>

*person UNLOCK door*

**In other film scene files:**
He unlocked the door. (person UNLOCK door)
The door was unlocked by a strange force. (force UNLOCK door)

**Integrated:**
{person, force} UNLOCK (door)
Parsing the Stanford Parser
Text input

Stanford Parser

Constituency based parse tree

Apply syntactic templates

Extracted patterns
Plain text:
At the time of the Constitution there weren't exactly vast suburbs that could be prowled by thieves looking for an open window.
Constituency-based parse tree

- Difference from dependency based tree

- Parser output:

```
(ROOT (S (PP (IN At) (NP (NP (DT the) (NN time)) (PP (IN of) (NP (DT the) (NNP Constitution))))))
(NP (EX there)) (VP (VBD were) (RB n't) (ADVP (RB exactly)) (NP (NP (JJ vast) (NNS suburbs)))
(SBAR (WHNP (WDT that)) (S (VP (MD could) (VP (VB be) (VP (VBN prowled) (PP (IN by) (NP (NP (NNS thieves)) (VP (VBG looking) (PP (IN for) (NP (DT an) (JJ open) (NN window))))))))))))
```
Stanford Parser Code

```java
// initialize the pipeline
Properties props = new Properties();
props.put("annotators", "tokenize, ssplit, pos, lemma, parse");
StanfordCoreNLP pipeline = new StanfordCoreNLP(props);

// annotate document with given annotations
Annotation document = new Annotation(text);
pipeline.annotate(document);

// these are all the sentences in this document
List<CoreMap> sentences =
    document.get(SentencesAnnotation.class);

String parseTree = "";
for(CoreMap sentence: sentences) {
    // this is the Stanford dependency graph of the current sentence
    // e.g. (ROOT (S (PP (IN At) ...)
    parseTree += sentence.get(TreeAnnotation.class).toString();
}
```
At the time of the Constitution, there were n't exactly vast suburbs that could be prowled by thieves looking for an open window.

(ROOT
  (S
    (PP (IN At)
      (NP
        (NP (DT the) (NN time))
        (PP (IN of)
          (NP (DT the) (NNP Constitution))))
      (NP (EX there))
      (VP (VBD were) (RB n't)
        (ADVP (RB exactly))
        (NP
          (NP (JJ vast) (NNS suburbs))
          (SBAR
            (WHNP (WDT that))
            (S
              (VP (MD could)
                (VP (VB be)
                  (VP (VBN prowled)
                    (PP (IN by)
                      (NP
                        (NP (NNS thieves))
                        (VP (VBG looking)
                          (PP (IN for)
                            (NP
                              (NP (DT an) (JJ open) (NN window))))))))))))))
  )
)
there were n't exactly vast suburbs

(ROOT
  (S
    (PP (IN At)
      (NP
        (NP (DT the) (NN time))
        (PP (IN of)
          (NP (DT the) (NNP Constitution))))))
    (NP (EX there))
    (VP (VBD were) (RB n't)
      (ADVP (RB exactly))
      (NP
        (NP (JJ vast) (NNS suburbs))
        (SBAR
          (WHNP (WDT that))
          (S
            (VP (MD could)
              (VP (VB be)
                (VP (VBN prowled)
                  (PP (IN by)
                    (NP
                      (NP (NNS thieves))
                      (VP (VBG looking)
                        (PP (IN for)
                          (NP
                            (NP (DT an) (JJ open) (NN window))))))))))))))
that would be prowled by thieves looking for an open window.
<table>
<thead>
<tr>
<th>Synactic Form</th>
<th>Example Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;subj&gt;</code> passive-verb</td>
<td><code>&lt;subj&gt;</code> was explained</td>
</tr>
<tr>
<td><code>&lt;subj&gt;</code> active-verb</td>
<td><code>&lt;subj&gt;</code> appears</td>
</tr>
<tr>
<td><code>&lt;subj&gt;</code> active-verb dobj</td>
<td><code>&lt;subj&gt;</code> have problem</td>
</tr>
<tr>
<td><code>&lt;subj&gt;</code> verb infinitive</td>
<td><code>&lt;subj&gt;</code> have to do</td>
</tr>
<tr>
<td><code>&lt;subj&gt;</code> aux noun</td>
<td><code>&lt;subj&gt;</code> is nothing</td>
</tr>
<tr>
<td>active-verb <code>&lt;dobj&gt;</code></td>
<td>gives <code>&lt;dobj&gt;</code></td>
</tr>
<tr>
<td>infinitive <code>&lt;dobj&gt;</code></td>
<td>to force <code>&lt;dobj&gt;</code></td>
</tr>
<tr>
<td>verb infinitive <code>&lt;dobj&gt;</code></td>
<td>want to take <code>&lt;dobj&gt;</code></td>
</tr>
<tr>
<td>noun aux <code>&lt;dobj&gt;</code></td>
<td>fact is <code>&lt;dobj&gt;</code></td>
</tr>
<tr>
<td>noun prep <code>&lt;np&gt;</code></td>
<td>argument against <code>&lt;np&gt;</code></td>
</tr>
<tr>
<td>active-verb prep <code>&lt;np&gt;</code></td>
<td>looking for <code>&lt;np&gt;</code></td>
</tr>
<tr>
<td>passive-verb prep <code>&lt;np&gt;</code></td>
<td>was put in <code>&lt;np&gt;</code></td>
</tr>
<tr>
<td>infinitive prep <code>&lt;np&gt;</code></td>
<td>to go to <code>&lt;np&gt;</code></td>
</tr>
</tbody>
</table>
active-verb prep <np>
Rules:
1. VP with VB_ and PP children
2. PP with IN and NP children
active-verb prep <np>

Rules:
1. VP with VB_ and PP children
2. PP with IN and NP children
active-verb prep <np>

Rules:
1. VP with VB_ and PP children
2. PP with IN and NP children
active-verb prep <np>

Rules:
1. VP with VB_ and PP children
2. PP with IN and NP children
active-verb prep <np>

Rules:
1. VP with VB_ and PP children
2. PP with IN and NP children
active-verb prep <np>

Rules:
1. VP with VB_ and PP children
2. PP with IN and NP children

→ prowled by <np>
<subj> active-verb dobj

Rules:
1. Node with NP and VP children
2. VP with VB and NP children
   \[ \Rightarrow \text{exert their utmost skill} \]

(Root
  (S
    (NP (PRP$ their) (JJ utmost) (NN skill)))
  (VP (VB exert)
    (NP (PRP their) (JJ utmost) (NN skill)))))
<table>
<thead>
<tr>
<th>Pattern Instance</th>
<th>FREQ</th>
<th>%SARC</th>
<th>Example Utterance</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;subj&gt; was explained</td>
<td>2</td>
<td>100%</td>
<td>Well, I incorrectly assumed that anyone attempting to enter the discussion would at least have a grasp of the most fundamental principles. It is quite strange to encounter someone in this day and age who lacks any knowledge whatsoever of the mechanism of adaptation since it was explained 150 years ago.</td>
</tr>
<tr>
<td>&lt;subj&gt; appears</td>
<td>1</td>
<td>94%</td>
<td>It appears this thread has been attacked by the “line item” poster.</td>
</tr>
<tr>
<td>&lt;subj&gt; have problem</td>
<td>4</td>
<td>50%</td>
<td>I see your point, langb but I’m not about to be leaving before you’ve had a chance to respond. I won’t be “leaving” at all. You challenged me to produce an argument, so I’m going to produce my argument. I will then summarize the argument, and you can respond to it and we can then discuss / debate those specifics that you have a problem with.</td>
</tr>
<tr>
<td>&lt;subj&gt; have to do</td>
<td>15</td>
<td>86%</td>
<td>How does purchasing a house have to do with abortion? Ok, so what if the kid wants to have the baby and the adults want to get rid of it? What if the adults want her to have the baby and the kid wants to get rid of it? You would force the kid to have a child (that doesn’t seem responsible at all), or you would force the kid to abort her child (thereby taking away her son or daughter). Both of those decisions don’t sound very consistent or responsible. The decision is best left up to the person that is pregnant, regardless of their age.</td>
</tr>
<tr>
<td>&lt;subj&gt; is nothing</td>
<td>10</td>
<td>90%</td>
<td>Even though there is nothing but ad hoc answers to the questions, creationists touted the book as “proof” that Noah’s ark was possible. They never seem to notice that no one has ever tried to build and float an ark. They prefer to put the money into creation museums and amusement parks.</td>
</tr>
<tr>
<td>gives &lt;dobj&gt;</td>
<td>25</td>
<td>88%</td>
<td>Just knowing that there are many Senators and Congressmen who would like to abolish gun rights gives credence to the fact that government could actually try to limit or ban the 2nd Amendment in the future.</td>
</tr>
<tr>
<td>to force &lt;dobj&gt;</td>
<td>9</td>
<td>89%</td>
<td>And I just say that it would be unjust and unfair of you to force metaphysical belief systems of your own which constitute religious belief upon your followers who may believe otherwise than you. Get pregnant and treat your fetus as a full person if you wish, nobody will force you to abort it. Let others follow their own beliefs differing or the same. Otherwise you attempt to obtain justice by doing injustice.</td>
</tr>
<tr>
<td>want to take &lt;dobj&gt;</td>
<td>5</td>
<td>80%</td>
<td>How far do you want to take the preemptive strike thing? Should we make it illegal for people to gather in public in groups of two or larger because anything else might be considered a violent mob assembly for the basis of creating terror and chaos?</td>
</tr>
<tr>
<td>fact is &lt;dobj&gt;</td>
<td>6</td>
<td>83%</td>
<td>No, the fact is PP was founded by an avowed racist and staunch supporter of Eugenics.</td>
</tr>
<tr>
<td>argument against &lt;np&gt;</td>
<td>4</td>
<td>75%</td>
<td>Perhaps I am too attached to this particular debate that you are having but if you actually have a sensible argument against gay marriage then please give it your best shot here. I look forward to reading your comments.</td>
</tr>
<tr>
<td>looking for &lt;np&gt;</td>
<td>14</td>
<td>92%</td>
<td>At the time of the Constitution there weren’t exactly vast suburbs that could be prowled by thieves looking for an open window.</td>
</tr>
</tbody>
</table>
Coreference Resolution

Larissa Munishkina
UCSC 2013
What is reference?

**Words** are used to represent things and experiences in the real or imagined world. Different words can be used to describe the same thing or experience. **Referent** is the concrete object or concept that is designated by a word or expression. A referent is an object, action, state, relationship, or attribute in the referential realm.

**Reference:**

1. is the symbolic relationship that a linguistic expression has with the concrete object or abstraction it represents.

2. is the relationship of one linguistic expression to another, in which one provides the information necessary to interpret the other.

What is coreference resolution?

**Coreference** is the reference in one expression to the same **referent** in another expression.

**Example:** in the following sentence, both *you's* have the same referent:

*You* said *you* would come.

**Coreference resolution** is a task of finding all references to the same entity.

**Coreference resolution includes:**

- Pronominal anaphora resolution
- Nominal coreference resolution
Example of Coreferences

- [Douglas Quail]₁ and [his wife Kirsten]₂, are asleep in bed.
- Gradually the room lights brighten.
- The clock chimes and begins speaking in a soft, feminine voice.
- [They]₁,₂ don't budge.
- Shortly, the clock chimes again.
- [Quail's wife]₂ stirs.
- Maddeningly, the clock chimes a third time.
- [Quail]₁ reaches out and shuts the clock off.
- Then [he]₁ sits up in bed.
Coreference and Event Chains

Coreference Chain:
(Douglas Quail; his; Quail's; he; He; his; He; his; He; a good-looking but conventional man in his early thirties; his; He; his; him; ),

Event Chain:
[(be_asleep,nsubj),(sit,nsubj),(swing,nsubj),(sit,nsubj),
 (put,nsubj),(sit,nsubj),(lose,nsubj),(be_man,nsubj),
 (seem,nsubj)]
StanfordCoreNLP Tool


**Excerpt from the code:**

```java
StanfordCoreNLP pipeline = new StanfordCoreNLP(props);
Annotation document = newAnnotation(text);
pipeline.annotate(document);

Map<Integer, CorefChain> graph = document.get(CorefChainAnnotation.class);
System.out.println(graph);
```
StanfordCoreNLP Example

Input Raw Text:
Jack and Jill went up the hill to fetch a pail of water. Jack fell down and broke his crown, and Jill came tumbling after. Up he got, and home did trot, as fast as he could caper; to old Dame Dob, who patched his nob with vinegar and brown paper.

Output Coreference Chains:
1=CHAIN1-"Jack" in sentence 1, "Jack" in sentence 2, "his" in sentence 2, "he" in sentence 3, "he" in sentence 3, "his" in sentence 3],
2=CHAIN2-"Jill" in sentence 1, "Jill" in sentence 2],
12=CHAIN12-"old Dame Dob , who patched his nob with vinegar and brown paper" in sentence 3, "old Dame Dob" in sentence 3, "Dame Dob" in sentence 3],
15=CHAIN15-"trot , as fast as he could caper ;" in sentence 3]
Questions?

Thank you!
Lots and Lots of other tools
Other tools out there

- FREEBASE ontology. Currently being used heavily for NLU
  - http://www.freebase.com/
- Semantic dictionaries of various kinds:
  - LIWC: Have a version in the lab tags at word level
  - MPQA, Sentiwordnet: polarity, sentiment
- Wordnet
- Verbnet
- FrameNet
- Other more detailed named entity frameworks.
LIWC. We have used a lot.

<table>
<thead>
<tr>
<th>Category</th>
<th>Sample 1</th>
<th>Sample 2</th>
<th>Sample 3</th>
<th>Sample 4</th>
<th>Sample 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Swear words</td>
<td>0.31</td>
<td>0.64</td>
<td>0.30</td>
<td>0.63</td>
<td>0.99</td>
</tr>
<tr>
<td>Social words</td>
<td>8.63</td>
<td>3.97</td>
<td>7.92</td>
<td>3.82</td>
<td>0.98</td>
</tr>
<tr>
<td>Family</td>
<td>0.53</td>
<td>0.85</td>
<td>0.51</td>
<td>0.84</td>
<td>0.99</td>
</tr>
<tr>
<td>Friends</td>
<td>0.33</td>
<td>0.46</td>
<td>0.32</td>
<td>0.46</td>
<td>0.99</td>
</tr>
<tr>
<td>Humans</td>
<td>0.73</td>
<td>0.66</td>
<td>0.67</td>
<td>0.61</td>
<td>0.95</td>
</tr>
<tr>
<td>Affect</td>
<td>5.12</td>
<td>2.25</td>
<td>4.04</td>
<td>1.91</td>
<td>0.93</td>
</tr>
<tr>
<td>Positive emotions</td>
<td>3.02</td>
<td>1.62</td>
<td>2.26</td>
<td>1.33</td>
<td>0.89</td>
</tr>
<tr>
<td>Negative emotions</td>
<td>2.04</td>
<td>1.43</td>
<td>1.76</td>
<td>1.31</td>
<td>0.97</td>
</tr>
<tr>
<td>Anxiety</td>
<td>0.39</td>
<td>0.46</td>
<td>0.28</td>
<td>0.39</td>
<td>0.91</td>
</tr>
<tr>
<td>Anger</td>
<td>0.69</td>
<td>0.86</td>
<td>0.59</td>
<td>0.79</td>
<td>0.97</td>
</tr>
<tr>
<td>Sadness</td>
<td>0.41</td>
<td>0.50</td>
<td>0.37</td>
<td>0.47</td>
<td>0.97</td>
</tr>
<tr>
<td>Cognitive mechanisms</td>
<td>16.34</td>
<td>4.02</td>
<td>6.41</td>
<td>2.50</td>
<td>0.75</td>
</tr>
<tr>
<td>Insight</td>
<td>2.20</td>
<td>1.26</td>
<td>1.86</td>
<td>1.05</td>
<td>0.86</td>
</tr>
<tr>
<td>Causal</td>
<td>1.44</td>
<td>0.80</td>
<td>0.90</td>
<td>0.61</td>
<td>0.83</td>
</tr>
<tr>
<td>Discrepancy</td>
<td>1.63</td>
<td>0.98</td>
<td>2.14</td>
<td>1.13</td>
<td>0.87</td>
</tr>
<tr>
<td>Tentative</td>
<td>2.60</td>
<td>1.30</td>
<td>2.45</td>
<td>1.27</td>
<td>0.84</td>
</tr>
<tr>
<td>Certainty</td>
<td>1.31</td>
<td>0.80</td>
<td>1.08</td>
<td>0.71</td>
<td>0.81</td>
</tr>
</tbody>
</table>
Have a version in NLDS, tags at word level

http://www.liwc.net/

**What is LIWC?**

Linguistic Inquiry and Word Count (LIWC) is a text analysis software program designed by James W. Pennebaker, Roger J. Booth, and Martha E. Francis. LIWC calculates the degree to which people use different categories of words across a wide array of texts, including emails, speeches, poems, or transcribed daily speech. With a click of a button, you can determine the degree any text uses positive or negative emotions, self-references, causal words, and 70 other language dimensions.

The LIWC program can analyze hundreds of standard ASCII text files or Microsoft Word documents in seconds. The LIWC2007 program also allows you to build your own dictionaries to analyze dimensions of language specifically relevant to your interests. The Macintosh version of LIWC2007 has a feature that will highlight in color all the words found in a particular file when it is analyzed. Users can also create dictionaries that include literal phrases (e.g., 'you know') as well as individual words and word stems. These features will soon be available for the Windows LIWC2007 version as well.

The student version of LIWC, LIWClite7, only analyzes plain text files using the LIWC2007 and earlier LIWC2001 dictionaries. LIWClite7 is the student version that is ideal for people with limited text analysis needs.

**LIWC license**

A single-user license for LIWC2007 or LIWClite7 entitles you to install the software on no more than two computers, however discounts available for multi-user versions (see [End User License Agreement here](http://www.liwc.net/)).
Using WordNet: Online Thesaurus.

- [http://wordnetweb.princeton.edu/](http://wordnetweb.princeton.edu/)
- What is the service ceiling of a U-2?
- Can access it FROM a program (not just this interface).
Using WordNet: Online Thesaurus.

- What is the service ceiling of a U-2
- Can access it as a program.

http://wordnetweb.princeton.edu/
Wikipedia: What knowledge can we get from Wikipedia?


Lockheed U-2
From Wikipedia, the free encyclopedia

The Lockheed U-2, nicknamed "Dragon Lady", is a single-engine, very high-altitude reconnaissance aircraft operated by the United States Air Force (USAF) and previously flown by the Central Intelligence Agency (CIA). It provides day and night, very high-altitude (70,000 feet / 21,000 meters), all-weather intelligence gathering.[1] The aircraft is also used for electronic sensor research and development, satellite calibration, and satellite data validation.

A Lockheed U-2 in flight

Reconnaissance aircraft
From Wikipedia, the free encyclopedia

A reconnaissance aircraft is a manned military aircraft designed, or adapted, to carry out aerial reconnaissance.
Extracting Named Entities

Person: Mr. Hubert J. Smith, Adm. McInnes, Grace Chan
Title: Chairman, Vice President of Technology, Secretary of State
Country: USSR, France, Haiti, Haitian Republic
City: New York, Rome, Paris, Birmingham, Seneca Falls
Province: Kansas, Yorkshire, Uttar Pradesh
Business: GTE Corporation, FreeMarkets Inc., Acme
University: Bryn Mawr College, University of Iowa
Organization: Red Cross, Boys and Girls Club
This is where technology was when IBM started their project
An Example

Who won the Nobel Peace Prize in 1991?

But many foreign investors remain sceptical, and western governments are withholding aid because of the Slorc's dismal human rights record and the continued detention of Ms Aung San Suu Kyi, the opposition leader who won the Nobel Peace Prize in 1991.

The military junta took power in 1988 as pro-democracy demonstrations were sweeping the country. It held elections in 1990, but has ignored their result. It has kept the 1991 Nobel peace prize winner, Aung San Suu Kyi - leader of the opposition party which won a landslide victory in the poll - under house arrest since July 1989.

The regime, which is also engaged in a battle with insurgents near its eastern border with Thailand, ignored a 1990 election victory by an opposition party and is detaining its leader, Ms Aung San Suu Kyi, who was awarded the 1991 Nobel Peace Prize. According to the British Red Cross, 5,000 or more refugees, mainly the elderly and women and children, are crossing into Bangladesh each day.