

CS 188: Artificial Intelligence Spring 2006

Lecture 27: NLP 4/27/2006

Dan Klein – UC Berkeley

What is NLP?



- Fundamental goal: *deep* understand of *broad* language
 - Not just string processing or keyword matching!
- End systems that we want to build:
 - Ambitious: speech recognition, machine translation, information extraction, dialog interfaces, question answering...
 - Modest: spelling correction, text categorization...

Why is Language Hard?

- Ambiguity
 - EYE DROPS OFF SHELF
 - MINERS REFUSE TO WORK AFTER DEATH
 - KILLER SENTENCED TO DIE FOR SECOND TIME IN 10 YEARS
 - LACK OF BRAINS HINDERS RESEARCH

The Big Open Problems

- Machine translation
- Information extraction
- Solid speech recognition
- Deep content understanding

Machine Translation

Atlanta, preso il killer del palazzo di Giustizia

ATLANTA - La grande paura che per 26 ore ha attanagliato Atlanta è finita: Brian Nichols, l'uomo che aveva ucciso tre persone a palazzo di Giustizia e che ha poi ucciso un agente di dogana, s'è consegnato alla polizia, dopo avere cercato rifugio nell'alloggio di una donna in un complesso d'appartamenti alla periferia della città. Per tutto il giorno, il centro della città, sede della Coca Cola e dei Giochi 1996, cuore di una popolosa area metropolitana, era rimasto paralizzato.

- Translation systems encode:
 - Something about fluent language
 - Something about how two languages correspond
- SOTA: for easy language pairs, better than nothing, but more an understanding aid than a replacement for human translators

Atlanta, taken the killer of the palace of Justice

ATLANTA - The great fear that for 26 hours has gripped Atlanta is ended: Brian Nichols, the man who had killed three persons to palace of Justice and that a customs agent has then killed, s' is delivered to the police, after to have tried shelter in the lodging of one woman in a complex of apartments to the periphery of the city. For all the day, the center of the city, center of the Coke Strains and of Giochi 1996, heart of one popolosa metropolitan area, was remained paralyzed.

Information Extraction

- Information Extraction (IE)
 - Unstructured text to database entries

New York Times Co. named **Russell T. Lewis**, 45, **president and general manager** of its flagship New York Times newspaper, responsible for all business-side activities. He was **executive vice president and deputy general manager**. He succeeds **Lance R. Primis**, who in September was named **president and chief operating officer** of the parent.

Person	Company	Post	State
Russell T. Lewis	New York Times newspaper	president and general manager	start
Russell T. Lewis	New York Times newspaper	executive vice president	end
Lance R. Primis	New York Times Co.	president and CEO	start

- SOTA: perhaps 70% accuracy for multi-sentence templates, 90%+ for single easy fields

Question Answering

- Question Answering:
 - More than search
 - Ask general comprehension questions of a document collection
 - Can be really easy: "What's the capital of Wyoming?"
 - Can be harder: "How many US states' capitals are also their largest cities?"
 - Can be open ended: "What are the main issues in the global warming debate?"
- SOTA: Can do factoids, even when text isn't a perfect match



Models of Language

- Two main ways of modeling language
 - Language modeling:** putting a distribution $P(s)$ over sentences s
 - Useful for modeling fluency in a noisy channel setting, like machine translation or ASR
 - Typically simple models, trained on lots of data
 - Language analysis:** determining the structure and/or meaning behind a sentence
 - Useful for deeper processing like information extraction or question answering
 - Starting to be used for MT

The Speech Recognition Problem

- We want to predict a sentence given an acoustic sequence:

$$s^* = \arg \max_s P(s | A)$$

- The noisy channel approach:

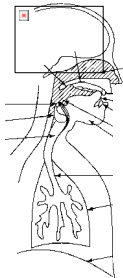
- Build a generative model of production (encoding)

$$P(A, s) = P(s) P(A | s)$$

- To decode, we use Bayes' rule to write

$$\begin{aligned} s^* &= \arg \max_s P(s | A) \\ &= \arg \max_s P(s) P(A | s) / P(A) \\ &= \arg \max_s P(s) P(A | s) \end{aligned}$$

- Now, we have to find a sentence maximizing this product



N-Gram Language Models

- No loss of generality to break sentence probability down with the chain rule

$$P(w_1 w_2 \dots w_n) = \prod_i P(w_i | w_1 w_2 \dots w_{i-1})$$

- Too many histories!

- N-gram solution: assume each word depends only on a short linear history

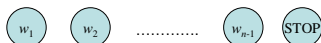
$$P(w_1 w_2 \dots w_n) = \prod_i P(w_i | w_{i-k} \dots w_{i-1})$$

Unigram Models

- Simplest case: unigrams

$$P(w_1 w_2 \dots w_n) = \prod_i P(w_i)$$

- Generative process: pick a word, pick another word, ...
- As a graphical model:



- To make this a proper distribution over sentences, we have to generate a special STOP symbol last. (Why?)

- Examples:

- [fifth, an, of, futures, the, an, incorporated, a, a, the, inflation, most, dollars, quarter, in, is, mass.]
- [thrift, did, eighty, said, hard, 'm, july, bullish]
- [that, or, limited, the]
- []
- [after, any, on, consistently, hospital, lake, of, of, other, and, factors, raised, analyst, too, allowed, mexico, never, consider, fall, bungled, davisson, that, obtain, price, lines, the, to, sass, the, the, further, board, a, details, machinists, the, companies, which, rivals, an, because, longer, cakes, percent, a, they, three, edward, it, currier, an, within, in, three, wrote, is, you, s, longer, institute, dentistry, pay, however, said, possible, to, rooms, hiding, eggs, approximate, financial, canada, the, so, workers, advancers, half, between, nasdaq]

Bigram Models

- Big problem with unigrams: $P(\text{the the the the}) \gg P(\text{I like ice cream})$
- Condition on last word:

$$P(w_1 w_2 \dots w_n) = \prod_i P(w_i | w_{i-1})$$



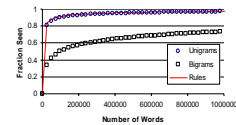
- Any better?

- [texaco, rose, one, in, this, issue, is, pursuing, growth, in, a, boiler, house, said, mr., gurria, mexico, 's, motion, control, proposal, without, permission, from, five, hundred, fifty, five, yen]
- [outside, new, car, parking, lot, of, the, agreement, reached]
- [although, common, shares, rose, forty, six, point, four, hundred, dollars, from, thirty, seconds, at, the, greatest, play, disingenuous, to, be, reset, annually, the, buy, out, of, american, brands, vying, for, mr., womack, currently, sharedata, incorporated, believe, chemical, prices, undoubtedly, will, be, as, much, is, scheduled, to, conscientious, teaching]
- [this, would, be, a, record, november]

Sparsity

Problems with n-gram models:

- New words appear all the time:
 - Synaptitude
 - 132,701.03
 - fuzzificational
- New bigrams: even more often
- Trigrams or more – still worse!



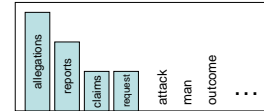
Zipf's Law

- Types (words) vs. tokens (word occurrences)
- Broadly: most word types are rare
- Specifically:
 - Rank word types by token frequency
 - Frequency inversely proportional to rank
- Not special to language: randomly generated character strings have this property

Smoothing

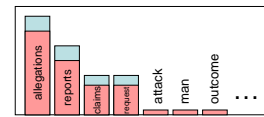
- We often want to make estimates from sparse statistics:

$P(w \mid \text{denied the})$
 3 allegations
 2 reports
 1 claims
 1 request
 7 total



- Smoothing flattens spiky distributions so they generalize better

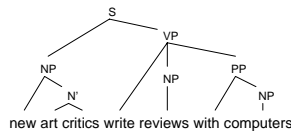
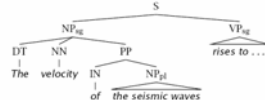
$P(w \mid \text{denied the})$
 2.5 allegations
 1.5 reports
 0.5 claims
 0.5 request
 2 other
 7 total



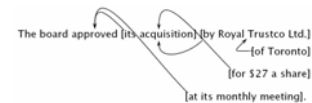
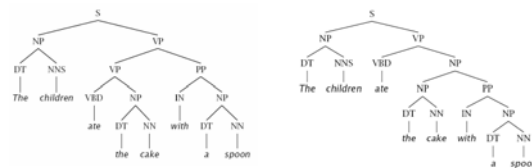
- Very important all over NLP, but easy to do badly!

Phrase Structure Parsing

- Phrase structure parsing organizes syntax into *constituents or brackets*
- In general, this involves nested trees
- Linguists can, and do, argue about details
- Lots of ambiguity
- Not the only kind of syntax...



PP Attachment



Attachment is a Simplification

- I cleaned the dishes from dinner
- I cleaned the dishes with detergent
- I cleaned the dishes in the sink

Syntactic Ambiguities I

- Prepositional phrases:**
They cooked the beans in the pot on the stove with handles.
- Particle vs. preposition:**
A good pharmacist dispenses with accuracy.
The puppy tore up the staircase.
- Complement structures**
The tourists objected to the guide that they couldn't hear.
She knows you like the back of her hand.
- Gerund vs. participial adjective**
Visiting relatives can be boring.
Changing schedules frequently confused passengers.

Syntactic Ambiguities II

- **Modifier scope within NPs**
impractical design requirements
plastic cup holder
- **Multiple gap constructions**
The chicken is ready to eat.
The contractors are rich enough to sue.
- **Coordination scope:**
Small rats and mice can squeeze into holes or cracks in the wall.

Human Processing

Garden pathing:

the man who hunts ducks out on weekends
the cotton shirts are made from grows in Mississippi
the old train the young
the daughter of the king's son loves himself

Ambiguity maintenance

Have the police ... eaten their supper?
come in and look around.
taken out and shot.

Context-Free Grammars

- A context free grammar is a tuple $\langle N, T, S, R \rangle$
 - N : the set of non-terminals
 - Phrasal categories: S, NP, VP, ADJP, etc.
 - Parts-of-speech (pre-terminals): NN, JJ, DT, VB
 - T : the set of terminals (the words)
 - S : the start symbol
 - Often written as ROOT or TOP
 - Not usually the sentence non-terminal S
 - R : the set of rules
 - Of the form $X \rightarrow Y_1 Y_2 \dots Y_k$, with $X, Y_i \in N$
 - Examples: $S \rightarrow NP VP$, $VP \rightarrow VP CC VP$
 - Also called rewrites, productions, or local trees

Example CFG

- Can just write the grammar (rules with non-terminal LHSs) and lexicon (rules with pre-terminal LHSs)

Grammar		Lexicon
ROOT \rightarrow S	NP \rightarrow NNS	JJ \rightarrow new
S \rightarrow NP VP	NP \rightarrow NN	NN \rightarrow art
VP \rightarrow VBP	NP \rightarrow JJ NP	NNS \rightarrow critics
VP \rightarrow VBP NP	NP \rightarrow NP NNS	NNS \rightarrow reviews
VP \rightarrow VP PP	NP \rightarrow NP PP	NNS \rightarrow computers
PP \rightarrow IN NP		VBP \rightarrow write
		IN \rightarrow with

Top-Down Generation from CFGs

- A CFG generates a language
 - Fix an order: apply rules to leftmost non-terminal
- ROOT
S
NP VP
NNS VP
critics VP
critics VBP NP
critics write NP
critics write NNS
critics write reviews

```

graph TD
    ROOT[ROOT] --> S[S]
    S --> NP1[NP]
    S --> VP1[VP]
    NP1 --> NNS1[NNS]
    NNS1 --> critics[critics]
    VP1 --> VBP[VBP]
    VP1 --> NP2[NP]
    VBP --> write[write]
    NP2 --> NNS2[NNS]
    NNS2 --> reviews[reviews]
    
```
- Gives a *derivation* of a tree using rules of the grammar

Corpora



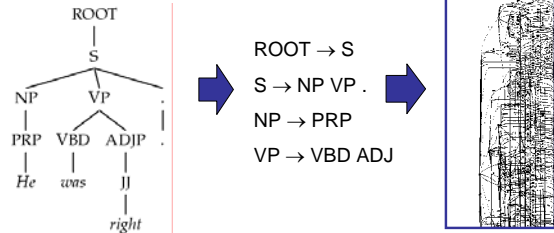
- A corpus is a collection of text
 - Often annotated in some way
 - Sometimes just lots of text
 - Balanced vs. uniform corpora
- Examples
 - Newswire collections: 500M+ words
 - Brown corpus: 1M words of tagged "balanced" text
 - Penn Treebank: 1M words of parsed WSJ
 - Canadian Hansards: 10M+ words of aligned French / English sentences
 - The Web: billions of words of who knows what

Treebank Sentences

```
( (S (NP-SBJ The move)
  (VP followed
    (NP (NP a round)
      (PP of
        (NP (NP similar increases)
          (PP by
            (NP other lenders))
          (PP against
            (NP Arizona real estate loans))))))
  (S-ADV (NP-SBJ *)
    (VP reflecting
      (NP (NP a continuing decline)
        (PP-LOC in
          (NP that market))))))
.))
```

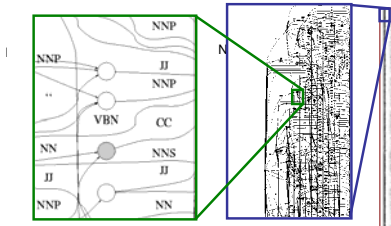
Corpus-Based Methods

- A corpus like a treebank gives us three important tools:
 - It gives us broad coverage



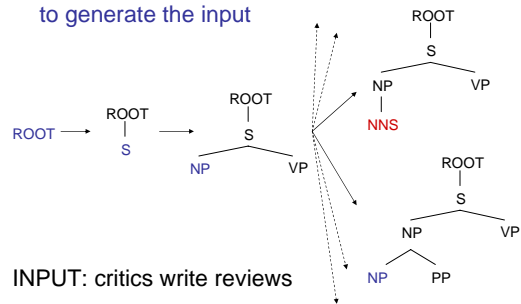
Why is Language Hard?

- Scale



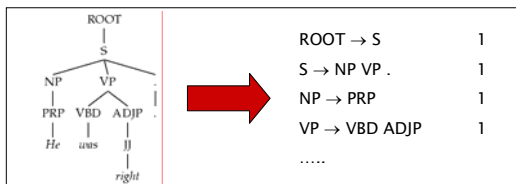
Parsing as Search: Top-Down

- Top down parsing: starts with the root and tries to generate the input



Treebank Parsing in 20 sec

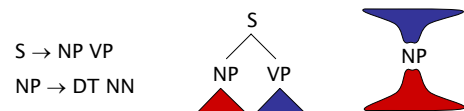
- Need a PCFG for broad coverage parsing.
- Can take a grammar right off the trees (doesn't work well):



- Better results by enriching the grammar (e.g., lexicalization).
- Can also get reasonable parsers without lexicalization.

PCFGs and Independence

- Symbols in a PCFG define independence assumptions:



- At any node, the material inside that node is independent of the material outside that node, given the label of that node.
- Any information that statistically connects behavior inside and outside a node must flow through that node.

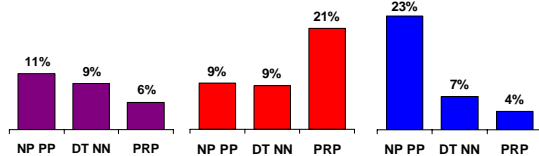
Corpus-Based Methods

- It gives us statistical information

All NPs

NPs under S

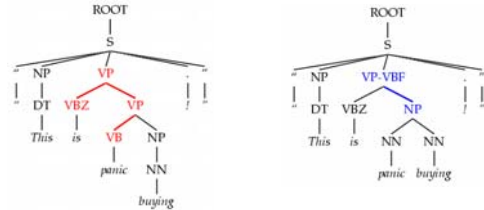
NPs under VP



This is a very different kind of subject/object asymmetry than what many linguists are interested in.

Corpus-Based Methods

- It lets us check our answers!



Semantic Interpretation

- Back to meaning!
 - A very basic approach to computational semantics
 - Truth-theoretic notion of semantics (Tarskian)
 - Assign a "meaning" to each word
 - Word meanings combine according to the parse structure
 - People can and do spend entire courses on this topic
 - We'll spend about an hour!
- What's NLP and what isn't?
 - Designing meaning representations?
 - Computing those representations?
 - Reasoning with them?
- Supplemental reading will be on the web page.

Meaning

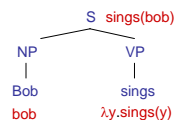
- "Meaning"
 - What is meaning?
 - "The computer in the corner."
 - "Bob likes Alice."
 - "I think I am a gummi bear."
 - Knowing whether a statement is true?
 - Knowing the conditions under which it's true?
 - Being able to react appropriately to it?
 - "Who does Bob like?"
 - "Close the door."
- A distinction:
 - Linguistic (semantic) meaning
 - "The door is open."
 - Speaker (pragmatic) meaning
- Today: assembling the semantic meaning of sentence from its parts

Entailment and Presupposition

- Some notions worth knowing:
 - Entailment:
 - A entails B if A being true necessarily implies B is true
 - ? "Twitchy is a big mouse" → "Twitchy is a mouse"
 - ? "Twitchy is a big mouse" → "Twitchy is big"
 - ? "Twitchy is a big mouse" → "Twitchy is furry"
 - Presupposition:
 - A presupposes B if A is only well-defined if B is true
 - "The computer in the corner is broken" presupposes that there is a (salient) computer in the corner

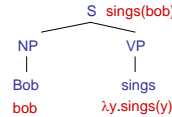
Truth-Conditional Semantics

- Linguistic expressions:
 - "Bob sings"
- Logical translations:
 - sings(bob)
 - Could be `p_1218(e_397)`
- Denotation:
 - [[bob]] = some specific person (in some context)
 - [[sings(bob)]] = ???
- Types on translations:
 - bob : e (for entity)
 - sings(bob) : t (for truth-value)



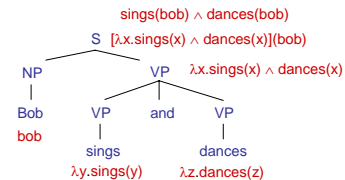
Truth-Conditional Semantics

- Proper names:
 - Refer directly to some entity in the world
 - Bob : **bob** $[[\text{bob}]]^w \rightarrow ???$
- Sentences:
 - Are either true or false (given how the world actually is)
 - Bob sings : **sings(bob)**
- So what about verbs (and verb phrases)?
 - sings** must combine with **bob** to produce **sings(bob)**
 - The λ -calculus is a notation for functions whose arguments are not yet filled.
 - sings** : $\lambda x.\text{sings}(x)$
 - This is *predicate* – a function which takes an entity (type e) and produces a truth value (type t). We can write its type as $e \rightarrow t$.
 - Adjectives?



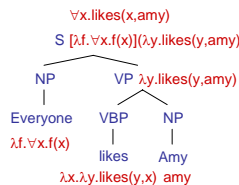
Compositional Semantics

- So now we have meanings for the words
- How do we know how to combine words?
- Associate a combination rule with each grammar rule:
 - $S : \beta(\alpha) \rightarrow NP : \alpha \quad VP : \beta$ (function application)
 - $VP : \lambda x. \alpha(x) \wedge \beta(x) \rightarrow VP : \alpha \quad \text{and} : \emptyset \quad VP : \beta$ (intersection)
- Example:



Other Cases

- Transitive verbs:
 - likes** : $\lambda x.\lambda y.\text{likes}(y,x)$
 - Two-place predicates of type $e \rightarrow (e \rightarrow t)$.
 - likes Amy** : $\lambda y.\text{likes}(y,\text{Amy})$ is just like a one-place predicate.
- Quantifiers:
 - What does "Everyone" mean here?
 - Everyone** : $\lambda f.\forall x.f(x)$
 - Mostly works, but some problems
 - Have to change our NP/VP rule.
 - Won't work for "Amy likes everyone."
 - "Everyone like someone."
 - This gets tricky quickly!



Denotation

- What do we do with logical translations?
 - Translation language (logical form) has fewer ambiguities
 - Can check truth value against a database
 - Denotation ("evaluation") calculated using the database
 - More usefully: assert truth and modify a database
 - Questions: check whether a statement in a corpus entails the (question, answer) pair:
 - "Bob sings and dances" \rightarrow "Who sings?" + "Bob"
 - Chain together facts and use them for comprehension

Grounding

- Grounding
 - So why does the translation **likes** : $\lambda x.\lambda y.\text{likes}(y,x)$ have anything to do with actual liking?
 - It doesn't (unless the denotation model says so)
 - Sometimes that's enough: wire up **bought** to the appropriate entry in a database
- Meaning postulates
 - Insist, e.g. $\forall x,y.\text{likes}(y,x) \rightarrow \text{knows}(y,x)$
 - This gets into lexical semantics issues
- Statistical version?

Tense and Events

- In general, you don't get far with verbs as predicates
- Better to have event variables e
 - "Alice danced" : **danced(alice)**
 - $\exists e : \text{dance}(e) \wedge \text{agent}(e,\text{alice}) \wedge (\text{time}(e) < \text{now})$
- Event variables let you talk about non-trivial tense / aspect structures
 - "Alice had been dancing when Bob sneezed"
 - $\exists e, e' : \text{dance}(e) \wedge \text{agent}(e,\text{alice}) \wedge \text{sneeze}(e') \wedge \text{agent}(e',\text{bob}) \wedge (\text{start}(e) < \text{start}(e') \wedge \text{end}(e) = \text{end}(e')) \wedge (\text{time}(e') < \text{now})$

Propositional Attitudes

- “Bob thinks that I am a gummi bear”
 - $\text{thinks}(\text{bob}, \text{gummi}(\text{me}))$?
 - $\text{Thinks}(\text{bob}, \text{“I am a gummi bear”})$?
 - $\text{thinks}(\text{bob}, \wedge \text{gummi}(\text{me}))$?
- Usual solution involves intensions ($\wedge X$) which are, roughly, the set of possible worlds (or conditions) in which X is true
- Hard to deal with computationally
 - Modeling other agents models, etc
 - Can come up in simple dialog scenarios, e.g., if you want to talk about what your bill claims you bought vs. what you actually bought

Trickier Stuff

- Non-Intersective Adjectives
 - $\text{green ball} : \lambda x. [\text{green}(x) \wedge \text{ball}(x)]$
 - $\text{fake diamond} : \lambda x. [\text{fake}(x) \wedge \text{diamond}(x)]$? $\longrightarrow \lambda x. [\text{fake}(\text{diamond}(x))]$
- Generalized Quantifiers
 - $\text{the} : \lambda f. [\text{unique-member}(f)]$
 - $\text{all} : \lambda f. \lambda g. [\forall x. f(x) \rightarrow g(x)]$
 - most?
 - Could do with more general second order predicates, too (why worse?)
 - $\text{the}(\text{cat}, \text{meows}), \text{all}(\text{cat}, \text{meows})$
- Generics
 - “Cats like naps”
 - “The players scored a goal”
- Pronouns (and bound anaphora)
 - “If you have a dime, put it in the meter.”
- ... the list goes on and on!

Multiple Quantifiers

- Quantifier scope
 - Groucho Marx celebrates quantifier order ambiguity:
“In this country a woman gives birth every 15 min.
Our job is to find that woman and stop her.”
- Deciding between readings
 - “Bob bought a pumpkin every Halloween”
 - “Bob put a pumpkin in every window”