

CS 343: Artificial Intelligence Natural Language Processing

Raymond J. Mooney
University of Texas at Austin

1

Communication

- The goal in the production and comprehension of natural language is communication.
- Communication for the speaker:
 - Intention:** Decide when and what information should be transmitted. May require planning and reasoning about agents' goals and beliefs.
 - Generation:** Translate the information to be communicated (in internal logical representation or "language of thought") into string of words in desired natural language.
 - Synthesis:** Output the string in desired modality, text or speech.

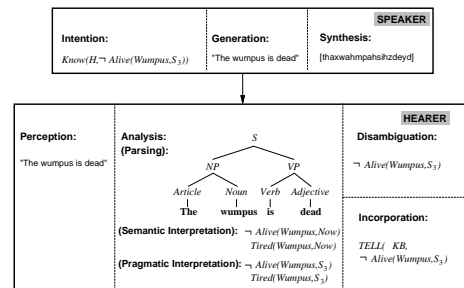
2

Communication (cont)

- Communication for the hearer:
 - Perception:** Map input modality to a string of words, e.g. *optical character recognition (OCR)* or *speech recognition*.
 - Analysis:** Determine the information content of the string.
 - Syntactic interpretation (parsing):** Find the correct parse tree showing the phrase structure of the string.
 - Semantic Interpretation:** Extract the (literal) meaning of the string (*logical form*).
 - Pragmatic Interpretation:** Consider effect of the overall context on altering the literal meaning of a sentence.
 - Incorporation:** Decide whether or not to believe the content of the string and add it to the KB.

3

Communication (cont)



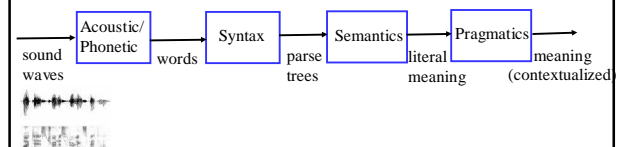
4

Syntax, Semantic, Pragmatics

- Syntax concerns the proper ordering of words and its affect on meaning.
 - The dog bit the boy.
 - The boy bit the dog.
 - * Bit boy dog the the.
 - Colorless green ideas sleep furiously.
- Semantics concerns the (literal) meaning of words, phrases, and sentences.
 - "plant" as a photosynthetic organism
 - "plant" as a manufacturing facility
 - "plant" as the act of sowing
- Pragmatics concerns the overall communicative and social context and its effect on interpretation.
 - The ham sandwich wants another beer. (co-reference, anaphora)
 - John thinks vanilla. (ellipsis)

5

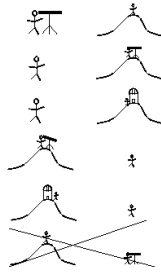
Modular Processing



6

Ambiguity

- Natural language is highly ambiguous and must be *disambiguated*.
 - I saw the man on the hill with a telescope.
 - I saw the Grand Canyon flying to LA.
 - Time flies like an arrow.
 - Horse flies like a sugar cube.
 - Time runners like a coach.
 - Time cars like a Porsche.



7

Ambiguity is Ubiquitous

- Speech Recognition
 - “recognize speech” vs. “wreck a nice beach”
 - “youth in Asia” vs. “euthanasia”
- Syntactic Analysis
 - “I ate spaghetti with a fork” vs. “I ate spaghetti with meat balls.”
- Semantic Analysis
 - “The dog is in the pen.” vs. “The ink is in the pen.”
 - “I put the plant in the window” vs. “Ford put the plant in Mexico”
- Pragmatic Analysis
 - Pedestrian: “Does your dog bite?,” Clouseau: “No.”
 - Pedestrian pets dog and is bitten.
 - Pedestrian: “I thought you said your dog does not bite?”
 - Clouseau: “That, sir, is not my dog.”

8

Humor and Ambiguity

- Many jokes rely on the ambiguity of language:
 - Groucho Marx: One morning I shot an elephant in my pajamas. How he got into my pajamas, I’ll never know.
 - She criticized my apartment, so I knocked her flat.
 - Noah took all of the animals on the ark in pairs. Except the worms, they came in apples.
 - Policeman to little boy: “We are looking for a thief with a bicycle.” Little boy: “Wouldn’t you be better using your eyes.”
 - Why is the teacher wearing sun-glasses. Because the class is so bright.

9

Syntactic Analysis

- Syntactic analysis is probably the most well-studied and well-understood aspects of language processing.
- Use formalisms and algorithms from formal language theory.
- Programming languages are designed to be *unambiguous*.
- Parsing *natural* language requires syntactic disambiguation.

10

Formal Grammars

- A *grammar* is a set of *production rules* that generates a set of strings (a *language*) by repeatedly rewriting symbols, starting with a top symbol *S* (for Sentence).
 - $S \rightarrow NP VP$
 - $NP \rightarrow Det N$
 - $VP \rightarrow V NP$
- Nonterminal* symbols are intermediate results that are not contained in the strings of the language.
- Terminal* symbols are the ultimate symbols (words) that compose the strings of the language.
- Productions that generate terminal symbols from *part of speech* (POS) categories constitute the *lexicon*.
 - $N \rightarrow boy$
 - $V \rightarrow eat$
- A *context free grammar* (CFG) only has productions with a single nonterminal on the left-hand-side.
 - Not CFG:
 - $AB \rightarrow C$
 - $BC \rightarrow FG$

11

Simplified English Grammar

- $S \rightarrow NP VP$
- $S \rightarrow VP$
- $NP \rightarrow Det Adj^* N PP^*$
- $NP \rightarrow ProN$
- $NP \rightarrow PName$
- $VP \rightarrow V PP^*$
- $VP \rightarrow V NP PP^*$
- $PP^* \rightarrow \epsilon$
- $PP^* \rightarrow PP PP^*$
- $PP \rightarrow Prep NP$
- $Adj^* \rightarrow \epsilon$
- $Adj^* \rightarrow Adj Adj^*$

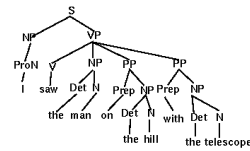
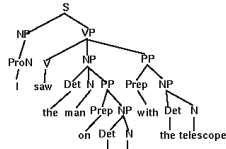
LEXICON:

- $ProN \rightarrow I$; $ProN \rightarrow you$; $ProN \rightarrow he$; $ProN \rightarrow she$
- $PName \rightarrow John$; $PName \rightarrow Mary$
- $Adj \rightarrow big$; $Adj \rightarrow little$; $Adj \rightarrow blue$
- $Det \rightarrow a$; $Det \rightarrow an$; $Det \rightarrow the$
- $N \rightarrow man$; $N \rightarrow telescope$; $N \rightarrow hill$; $N \rightarrow saw$
- $Prep \rightarrow with$; $Prep \rightarrow for$; $Prep \rightarrow of$; $Prep \rightarrow on$
- $V \rightarrow hit$; $V \rightarrow took$; $V \rightarrow saw$; $V \rightarrow likes$

12

Parse Trees and Syntactic Ambiguity

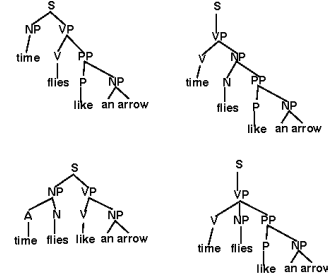
- A *parse tree* shows the derivation of a sentence in the language from the start symbol to its sequence of terminal symbols.
- If a sentence has more than one possible derivation (parse tree) it is said to be *syntactically ambiguous*.



13

Spurious Ambiguity

- Most parse trees of most NL sentences make no sense.



14

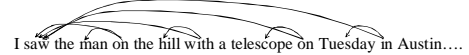
Syntactic Parsing

- Given a sequence of words, determine if can be derived from a given grammar and is therefore a sentence in the language.
- In NLP, want to determine all *possible* parse trees and use semantics and pragmatics to eliminate spurious parses and then use the best one to produce a semantic representation.
- Problem: Most sentences have *many* parses.

15

Prepositional Phrase Attachment Explosion

- A transitive English sentence ending in m prepositional phrases has *at least* 2^m parses.



- The exact number of parses is given by the *Catalan numbers* (where $n=m+1$)

$$\binom{2n}{n} - \binom{2n}{n-1} \approx \frac{4^n}{n^{3/2} \sqrt{\pi}}$$

1, 2, 5, 14, 132, 429, 1430, 4862, 16796,

16

Parsing Algorithms

- By exploiting *dynamic programming* to avoid repeated work, it can be determined if a sentence with n words has at least one parse tree (i.e. is grammatical according to a given CFG) in $O(n^3)$ time.
- Typical parsing algorithms are either:
 - **Top down**: Try to derive the sentence starting from the start symbol.
 - **Bottom up**: Try to derive the sentence starting from the terminal symbols.

17

Top Down Parsing

- Search the space of possible derivations of S (e.g. depth-first) for one that matches the input sentence.
 - I saw the man
 - S → NP VP
 - NP → Det Adj* N
 - Det → the
 - Det → a
 - Det → an
 - NP → ProN
 - ProN → I
 - VP → V NP
 - V → hit
 - V → saw
 - NP → Det Adj* N
 - Det → the
 - Adj* → ε
 - N → man

18

Bottom-Up Parsing

- Search upward from words finding larger and larger phrases until a full sentence is found.

| | |
|---------------------|--------------------|
| - I saw the man. | ProN → I |
| - ProN saw the man | NP → ProN |
| - NP saw the man | N → saw (dead end) |
| - NP N the man | V → saw |
| - NP V the man | Det → the |
| - NP V Det man | Adj* → ε |
| - NP V Det Adj* man | N → man |
| - NP V Det Adj* N | NP → Det Adj* N |
| - NP V NP | VP → V NP |
| - NP VP | S → NP VP |
| - S | |

19

Bottom Up Parsing Algorithm

```

function BOTTOM-UP-PARSE(words, grammar) returns a parse tree
    forest ← words
    loop do
        if LENGTH(forest) = 1 and CATEGORY(forest[1]) = START(grammar) then
            return forest[1]
        else
            i ← choose from {1...LENGTH(forest)}
            rule ← choose from RULES(grammar)
            n ← LENGTH(RULE-RHS(rule))
            subsequence ← SUBSEQUENCE(forest, i, i+n-1)
            if MATCH(subsequence, RULE-RHS(rule)) then
                forest[i...i+n-1] ← [MAKE-NODE(RULE-LHS(rule), subsequence)]
            else fail
        end
    end
  
```

20

Statistical Learning and Disambiguation

- Standard parsing does not help resolve ambiguities.
- Grammars must be developed manually, a laborious and difficult process.
 - “All grammars leak.” (linguist Edward Sapir)
- Statistical learning methods (*empirical, corpus-based*) have become very popular in NLP.
 - Automatically learn grammars and other linguistic and world knowledge from annotated or unannotated corpora.
 - Compute most probable parse or interpretation to resolve ambiguities.

21

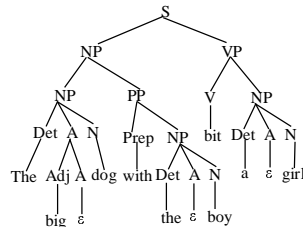
Probabilistic Context Free Grammar (PCFG)

- A PCFG is a probabilistic version of a CFG where each production has a probability.
- Probabilities of all productions rewriting a given non-terminal must add to 1, defining a distribution for each non-terminal.
- String generation is now probabilistic where production probabilities are used to non-deterministically select a production for rewriting a given non-terminal.

22

Sample PCFG

| | | |
|--------------------------|-----|-------|
| $S \rightarrow NP VP$ | 0.9 | } = 1 |
| $S \rightarrow VP$ | 0.1 | |
| $NP \rightarrow Det A N$ | 0.5 | } = 1 |
| $NP \rightarrow NP PP$ | 0.3 | |
| $NP \rightarrow PropN$ | 0.2 | |
| $A \rightarrow \epsilon$ | 0.6 | } = 1 |
| $A \rightarrow Adj A$ | 0.4 | |
| $PP \rightarrow Prep NP$ | 1.0 | } = 1 |
| $VP \rightarrow V NP$ | 0.7 | |
| $VP \rightarrow VP PP$ | 0.3 | } = 1 |

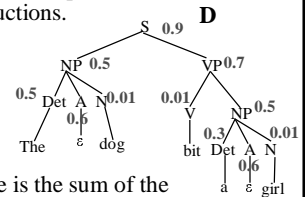


23

Sentence Probability

- Assume productions for each node are chosen independently.
- Probability of derivation is the product of the probabilities of its productions.

$$\begin{aligned}
 P(D) &= 0.9 \times 0.5 \times 0.7 \times 0.5 \times \\
 &\quad 0.6 \times 0.01 \times 0.01 \times 0.5 \times \\
 &\quad 0.3 \times 0.6 \times 0.01 \\
 &= 8.505 \times 10^{-9}
 \end{aligned}$$



- Probability of a sentence is the sum of the probability of all of its derivations.

Since it is unambiguous, $P(\text{“The dog bit a girl”}) = 8.505 \times 10^{-9}$

24

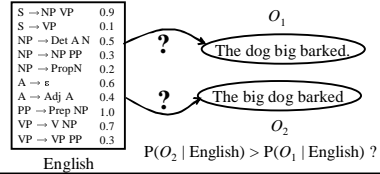
Three Useful PCFG Tasks

- Observation likelihood: To classify and produce a grammatical preference order on sentences.
- Most likely derivation: To determine the most likely parse tree for a sentence.
- Maximum likelihood training: To train a PCFG to fit empirical linguistic data.

25

PCFG: Observation Likelihood

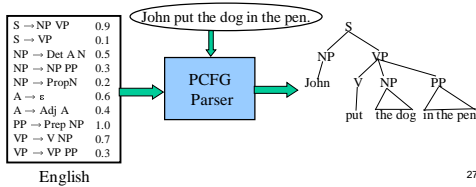
- The Inside/Outside algorithm efficiently determines how likely a string is to be produced by a PCFG.
- Can use a PCFG as a language model to choose between alternative sentences for speech recognition or machine translation.



26

PCFG: Most Likely Derivation

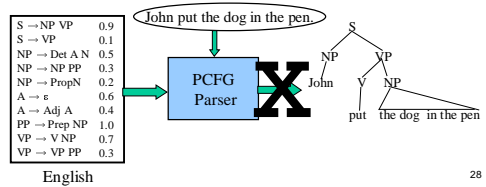
- There is an analog to the Viterbi algorithm to efficiently determine the most probable derivation (parse tree) for a sentence.
- Time complexity is $O(N^3T^3)$ where N is the number of non-terminals in the grammar and T is the length of the sentence.



27

PCFG: Most Likely Derivation

- There is an analog to the Viterbi algorithm to efficiently determine the most probable derivation (parse tree) for a sentence.
- Time complexity is $O(N^3T^3)$ where N is the number of non-terminals in the grammar and T is the length of the sentence.



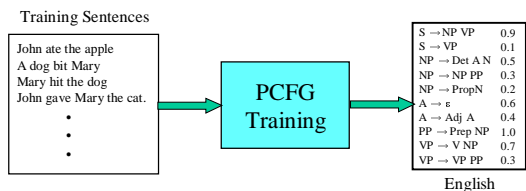
28

PCFG: Maximum Likelihood Training

- Given a set of sentences, induce a grammar that maximizes the probability that this data was generated from this grammar.
- Assume the number of non-terminals in the grammar is specified.
- Only need to have an unannotated set of sequences generated from the model. Does not need correct parse trees for these sentences. In this sense, it is unsupervised.

29

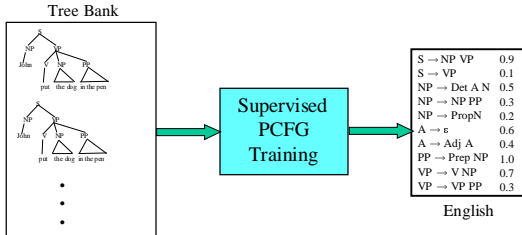
PCFG: Maximum Likelihood Training



30

PCFG: Supervised Training

- If parse trees are provided for training sentences, a grammar and its parameters can be estimated directly from counts accumulated from the tree-bank (with appropriate smoothing).



31

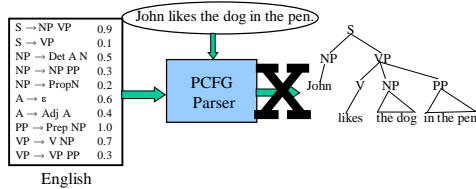
PCFG Comments

- Unsupervised training (of PCFGs or HMMs) do not to work very well. They tend to capture alternative structure in the data that does not directly reflect general syntax.
- Since probabilities of productions do not rely on specific words or concepts, only general structural disambiguation is possible.
- Consequently, vanilla PCFGs cannot resolve syntactic ambiguities that require semantics to resolve, e.g. ate with fork vs. meatballs.
- In order to work well, PCFGs must be lexicalized, i.e. productions must be specialized to specific words by including their head-word in their LHS non-terminals (e.g. VP-ate).

32

Example of Importance of Lexicalization

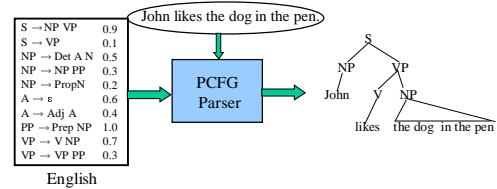
- A general preference for attaching PPs to verbs rather than NPs in certain structural situations could be learned by a vanilla PCFG.
- But the desired preference can depend on specific words.



33

Example of Importance of Lexicalization

- A general preference for attaching PPs to verbs rather than NPs in certain structural situations could be learned by a vanilla PCFG.
- But the desired preference can depend on specific words.



34

Trebanks

- English Penn Treebank: Standard corpus for testing syntactic parsing consists of 1.2 M words of text from the Wall Street Journal (WSJ).
- Typical to train on about 40,000 parsed sentences and test on an additional standard disjoint test set of 2,416 sentences.
- Chinese Penn Treebank: 100K words from the Xinhua news service.
- Other corpora existing in many languages, see the Wikipedia article "Treebank"

35

Trebank Results

- Standard accuracy measurements judge the fraction of the constituents that match between the computed and human parse trees. If P is the system's parse tree and T is the human parse tree (the "gold standard"):
 - Recall = (# correct constituents in P) / (# constituents in T)
 - Precision = (# correct constituents in P) / (# constituents in P)
- Labeled Precision and labeled recall require getting the non-terminal label on the constituent node correct to count as correct.
- Results of current state-of-the-art systems on the English Penn WSJ treebank are about 90% labeled precision and recall.

36

Word Sense Disambiguation (WSD) as Text Categorization

- Each sense of an ambiguous word is treated as a category.
 - “play” (verb)
 - play-game
 - play-instrument
 - play-role
 - “pen” (noun)
 - writing-instrument
 - enclosure
- Treat current sentence (or preceding and current sentence) as a document to be classified.
 - “play”:
 - play-game: “John played soccer in the stadium on Friday.”
 - play-instrument: “John played guitar in the band on Friday.”
 - play-role: “John played Hamlet in the theater on Friday.”
 - “pen”:
 - writing-instrument: “John wrote the letter with a pen in New York.”
 - enclosure: “John put the dog in the pen in New York.”

37

Learning for WSD

- Assume part-of-speech (POS), e.g. noun, verb, adjective, for the target word is determined.
- Treat as a classification problem with the appropriate potential senses for the target word given its POS as the categories.
- Encode context using a set of features to be used for disambiguation.
- Train a classifier on labeled data encoded using these features.
- Use the trained classifier to disambiguate future instances of the target word given their contextual features.

38

WSD “line” Corpus

- 4,149 examples from newspaper articles containing the word “line.”
- Each instance of “line” labeled with one of 6 senses from WordNet.
- Each example includes a sentence containing “line” and the previous sentence for context.

39

Senses of “line”

- Product: “While he wouldn’t estimate the sale price, analysts have estimated that it would exceed \$1 billion. Kraft also told analysts it plans to develop and test a line of refrigerated entrees and desserts, under the Chillery brand name.”
- Formation: “C-LD-R L-V-S V-NNA reads a sign in Caldor’s book department. The 1,000 or so people fighting for a place in line have no trouble filling in the blanks.”
- Text: “Newspaper editor Francis P. Church became famous for a 1897 editorial, addressed to a child, that included the line “Yes, Virginia, there is a Santa Clause.”
- Cord: “It is known as an aggressive, tenacious litigator. Richard D. Parsons, a partner at Patterson, Belknap, Webb and Tyler, likes the experience of opposing Sullivan & Cromwell to “having a thousand-pound tuna on the line.”
- Division: “Today, it is more vital than ever. In 1983, the act was entrenched in a new constitution, which established a tricameral parliament along racial lines, whith separate chambers for whites, coloreds and Asians but none for blacks.”
- Phone: “On the tape recording of Mrs. Guba’s call to the 911 emergency line, played at the trial, the baby sitter is heard begging for an ambulance.” 40

Experimental Data for WSD of “line”

- Sample equal number of examples of each sense to construct a corpus of 2,094.
- Represent as simple binary vectors of word occurrences in 2 sentence context.
 - Stop words eliminated
 - Stemmed to eliminate morphological variation
- Final examples represented with 2,859 binary word features.

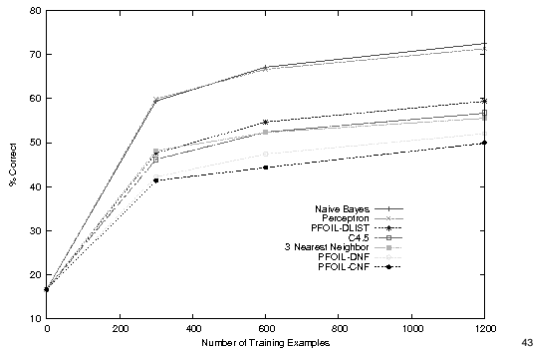
41

Learning Algorithms

- Naïve Bayes
 - Binary features
- K Nearest Neighbor
 - Simple instance-based algorithm with k=3 and Hamming distance
- Perceptron
 - Simple neural-network algorithm.
- C4.5
 - State of the art decision-tree induction algorithm
- PFOIL-DNF
 - Simple logical rule learner for Disjunctive Normal Form
- PFOIL-CNF
 - Simple logical rule learner for Conjunctive Normal Form
- PFOIL-DLIST
 - Simple logical rule learner for decision-list of conjunctive rules

42

Learning Curves for WSD of “line”



43

Discussion of Learning Curves for WSD of “line”

- Naïve Bayes and Perceptron give the best results.
- Both use a weighted linear combination of evidence from many features.
- Symbolic systems that try to find a small set of relevant features tend to overfit the training data and are not as accurate.
- Nearest neighbor method that weights all features equally is also not as accurate.
- Of symbolic systems, decision lists work the best.

44

NLP Conclusions

- The need for disambiguation makes language understanding difficult.
- Levels of linguistic processing:
 - Syntax
 - Semantics
 - Pragmatics
- CFGs can be used to parse natural language but produce many spurious parses.
- Statistical learning methods can be used to:
 - Automatically learn grammars from (annotated) corpora.
 - Compute the most likely interpretation based on a learned statistical model.

45