

CS474  
Introduction to Natural Language Processing

**Final Exam**  
**December 15, 2005**

**Name:**

**Netid:**

**Instructions:** You have 2 hours and 30 minutes to complete this exam. The exam is a closed-book exam.

#	description	score	max score
1	Parsing with PCFGs	_____	/ 25
2	Bottom-up Chart Parsing	_____	/ 10
3	Partial Parsing/Question Answering	_____	/ 30
4	Inference	_____	/ 15
5	The Grab Bag	_____	/ 20
Total score:		_____	/ 100

# 1 Parsing with PCFGs (25 pts)

- (a) (3 pts) A sentence can easily have more than one parse tree that is consistent with a given CFG. How do PCFGs and non-probability-based CFGs differ in terms of handling parsing ambiguity?

## Answer

PCFG parsers resolve ambiguity by preferring constituents (and parse trees) with the highest probability.

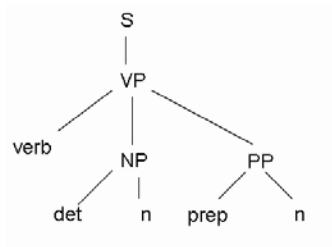
Consider the following PCFG for problems (b)-(e).

production rule	probability
$S \rightarrow VP$	1.0
$VP \rightarrow \text{Verb NP}$	0.7
$VP \rightarrow \text{Verb NP PP}$	0.3
$NP \rightarrow \text{NP PP}$	0.3
$NP \rightarrow \text{Det Noun}$	0.7
$PP \rightarrow \text{Prep Noun}$	1.0
$\text{Det} \rightarrow \text{the}$	0.1
$\text{Verb} \rightarrow \text{Cut} \mid \text{Ask} \mid \text{Find} \mid \dots$	0.1
$\text{Prep} \rightarrow \text{with} \mid \text{in} \mid \dots$	0.1
$\text{Noun} \rightarrow \text{envelope} \mid \text{grandma} \mid \text{scissors} \mid \text{men} \mid \text{suits} \mid \text{summer} \mid \dots$	0.1

- (b) (3 pts) Draw the top-ranked parse tree for the sentence below by applying the given PCFG. Does the result seem reasonable to you? Why or why not?

Cut the envelope with scissors.

## Answer



The top-ranked sentence structure is shown in figure 1. (The leaf nodes representing words are omitted.) The probability of the resulting parse tree is  $1.0 * 0.3 * 0.7 * 1.0 *$

$(0.1)^5$ , which is larger than  $1.0 * 0.7 * 0.3 * 0.7 * 1.0 * (0.1)^5$ , the probability of the alternative parse tree (with the  $[VP \rightarrow \text{Verb NP}]$  rule expansion). Semantically, "with scissors" should attach to the verb, hence the resulting parse tree is a reasonable one.

- (c) (3 pts) Draw the top-ranked parse tree for the sentence below by applying the given PCFG. Does the result seem reasonable to you? Why or why not?

Ask the grandma with scissors.

## Answer

The top-ranked sentence structure is the same as for part (c). Semantically, "with scissors" should attach to the noun phrase, hence the resulting parse tree is not a reasonable one.

- (d) (5 pts) Describe how you would lexicalize the given PCFG in order to address the problem you hopefully noticed in (b) and/or (c). Then show specifically how the production rules below should be modified according to your lexicalization scheme.

production rule	probability
$VP \rightarrow \text{Verb NP}$	0.7
$VP \rightarrow \text{Verb NP PP}$	0.3

## Answer

Lexicalization of production rules can capture lexical specific preference of certain rule expansions. In order to mitigate the sparse data problem, we will lexicalize with respect to the head word of the left hand side of each production rule, instead of all nonterminals in each production rule. In particular, the rules expanding from VP should be modified as

production rule	probability
$VP(x) \rightarrow \text{Verb NP}$	$(p_x)$
$VP(x) \rightarrow \text{Verb NP PP}$	$(q_x)$

where

$x \in \{ \text{Cut, Ask, Find, ...} \},$

$p_x \stackrel{\text{def}}{=} P( VP(x) \rightarrow \text{Verb NP} \mid VP, x),$

$q_x \stackrel{\text{def}}{=} P( VP(x) \rightarrow \text{Verb NP PP} \mid VP, x),$

and  $p_x + q_x = 1.$

## Comment

Because we didn't restrict lexicalization to head words of the right hand side of rules, it is okay to propose lexicalized PCFGs in many different ways; in particular, you don't have to condition on the head word, you can condition on the entire combination of words for all nonterminals, as long as you made it clear what you are conditioning on, although it would be much less practical.

- (e) (5 pts) The following two sentences exhibit parsing ambiguities. How would your lexicalized PCFG from (d) handle these ambiguities?

Find the men in suits.

Find the men in summer.

## Answer

Notice that the head word for any other node in a parse tree except the node for the last word is identical in both sentences. Therefore, the conditional probability of each node for a particular rule expansion is identical in both sentences except the node for the last word. However, the last word in both given sentences do not control which rule expansion to be used on the ancestor's nodes. Hence the exact same parse tree will be chosen by PCFGs, even though the prepositional phrase in the first sentence should attach to the noun phrase, and the prepositional phrase in the second sentence should attach to the verb phrase. (Although it is not impossible to do it the other way, but it would sound less sensible.) Which attachment to be chosen will depend on the actual value of  $P(\text{VP}(\text{Find}) \rightarrow \text{Verb NP} \mid \text{VP}, \text{Find})$  and  $P(\text{VP}(\text{Find}) \rightarrow \text{Verb NP PP} \mid \text{VP}, \text{Find})$ . In summary, the head word lexicalization does not solve all ambiguities, as shown in the given sentences.

## Comment

If your proposal in (d) didn't condition on head words of the right hand side of rules, you might have a chance to have a different conclusion here, depending on how exactly you chose the set of words to condition on. However, unless you somehow invented a clever way to condition on the entire words for one nonterminal PP, or unless you have changed the definition of head word, you probably end up encountering the same problem as above.

**Read for problems (f)-(g): One problem with a lexicalized PCFG is that some (perfectly reasonable) words might never show up in the training data for certain production rules. This results in rules with a probability of 0.**

- (f) (3 pts) Describe why production rules with zero probability are problematic.

## **Answer**

If a production rule has a zero probability, then the parse tree derived from that production rule will have to have a zero probability also. However, a production rule may have a zero probability not because it is invalid, but because the particular production rule has not been observed in the training data. PCFGs in this case will not be able to return the correct parse tree involving an unseen rule.

- (g) (3 pts) Describe one method to avoid zero probabilities for lexicalized PCFGs.

## **Answer**

Smoothing techniques from the language models can be similarly applied here. One simple method would be assigning a minimum count 1 for all possible lexicalized rules. (In order to make it a proper probability, we will need to augment the probability values collected from the training data when we see the new test data by renormalizing them.)

## 2 Bottom-up Chart Parsing (10 pts)

Given the grammar and lexicon below, show the **final chart** for the following sentence after applying the bottom-up chart parser. Remember that the final chart contains all edges added during the parsing process. You may use either the notation from class (i.e. nodes/links) or the notation from the book to depict the chart.

---

$S \rightarrow VP$

$VP \rightarrow \text{Verb NP}$

$NP \rightarrow NP PP$

$NP \rightarrow \text{Det Noun}$

$PP \rightarrow \text{Prep Noun}$

Find the men in suits.

$\text{Det} \rightarrow \text{the}$

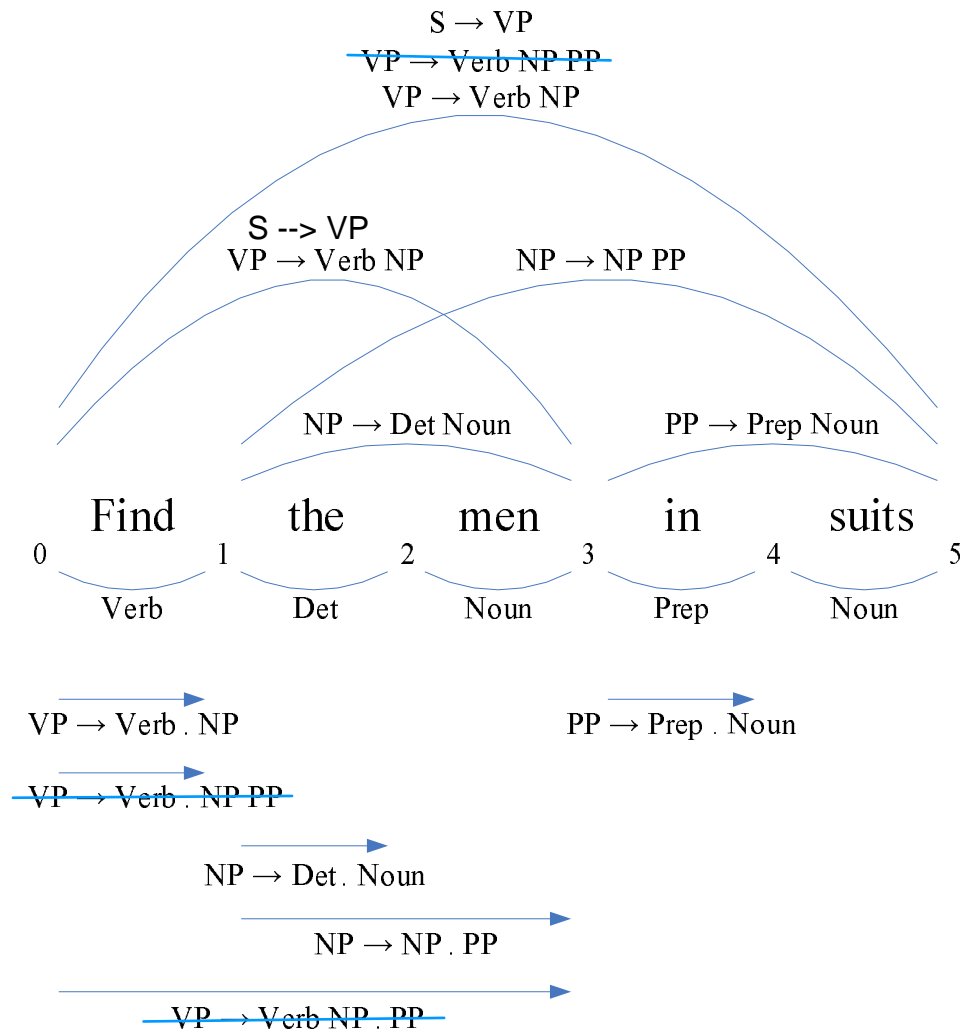
$\text{Verb} \rightarrow \text{Find}$

$\text{Prep} \rightarrow \text{in}$

$\text{Noun} \rightarrow \text{men} \mid \text{suits}$

---

# Answer



### 3 Partial Parsing / Question Answering (30 pts)

Consider the following article for problems (a) - (e).

---

[From product reviews for various computer peripherals.]

I bought my wireless keyboard/mouse set several months ago, and, like a lot of new products, it has some unanticipated issues. On the plus side, obviously, is the styling. The design is fresh, clean, and interesting. The keyboard can tilt at different angles, which was important because I had some difficulty typing with it flat. The bluetooth receiver in the charger was functional, and I appreciated having a bluetooth hub for my cellphone. The mouse and the keyboard have both proved durable and reliable despite a number of mishaps.

In regards to the software, there are some real issues. When the mouse powers down to save battery life there is a second or two of lag before it reconnects with the receiver. I found this really annoying to deal with every time I stepped away from my desk for ten or fifteen minutes. Also, during system startup when the bluetooth software has yet to initialize, both the keyboard and the mouse are useless. This made it impossible to do any kind of pre-windows-startup tasks such as F8 for windows configuration. I suspect this is a result of how bluetooth interacts with the OS and bios, but whatever the cause, it was, for me, a deal-breaker.

---

- (a) (5pts) Mark or draw the output of a **partial parser** for the following sentence, stating any necessary assumptions.

The bluetooth receiver in the charger was functional, and I appreciated having a bluetooth hub for my cellphone.

#### Answer

[The bluetooth receiver]<sub>np</sub> in [the charger]<sub>np</sub> was functional, and I appreciated having [a bluetooth hub]<sub>np</sub> for [my cellphone]<sub>np</sub>.

#### Comment

There can be different correct answers depending on the definition of constituents.

- (b) (5 pts) State two advantages of partial parsers over parsers that provide in-depth syntactic information.



## Answer

First, partial parsers can be more robust than the regular parsers, because partial parsers work on easier tasks. Second, for some NLP applications such as information extraction, information derived from partial parsers can be more relevant than that from regular parsers.

- (c) (5 pts) Consider a closed domain QA system for the domain of the above text, i.e. product reviews of computer peripherals. Assume that the QA system uses a simple TFIDF-based information retrieval method to identify documents and sentences that contain the answer to the input question. Assume also that the QA system only has access to the above document, i.e. the above document is the only document in the collection. (Yes, we know that this is not a reasonable assumption.)

Devise one reasonable wh-question (i.e. who, what, where, when, why) that has an answer in the document but that the QA system would not be able to answer sensibly. Explain why the question is difficult for the system.

## Answer

**Fall 2006 students: we did not cover TFIDF-based IR methods. They represent each document and query as a vector indicating the presence or absence of each word in the language (minus stopwords), and then compute similarity between a document and a query by computing the cosine of the angle between the two vectors. In addition, words that appear frequently across the entire corpus receive small weights; words that appear frequently in a document receive high weights. This isn't the whole story, but is enough to let you think about answering the question. No answer yet...**

- (d) (7 pts) Now suppose a closed domain QA system that has access to a large number of product reviews for various computer peripherals. Assume the possible questions for the QA system are limited to the following two types of questions.
- What features of product X are buyers satisfied with?
  - What features of product X are buyers dissatisfied with?

Since the types of questions are restricted, we can design **predictive annotations** to assist the question answering system. Describe a set of useful predictive annotation types for this restricted question answering task. Then annotate one sentence from the article according to your annotation scheme.

## Answer

**Fall 2006 students: we did not cover predictive annotation. No answer yet.**

- (e) (8 pts) Suppose that you have convinced your friends to annotate 500 documents per your definition of predictive annotations given in (d). Once the 500 documents are annotated, one can use them to train a supervised machine learning algorithm to automatically annotate many more documents (and thereby avoid losing one's friends who have become increasingly unwilling to help with the manual annotations).

Select **one** of your predictive annotation types from (d). Explain step-by-step how you would go about the task of training a learning algorithm to automate this type of annotation. Be sure to define your learning task and to describe a reasonable set of features.

## **Answer**

**Fall 2006 students: we did not cover predictive annotation.** No answer yet.

## 4 Inference (10 pts)

Consider the following article for this problem.

---

[This is just the first paragraph from the previous question's text.]

I bought my wireless keyboard/mouse set several months ago, and, like a lot of new products, it has some unanticipated issues. On the plus side, obviously, is the styling. The design is fresh, clean, and interesting. The keyboard can tilt at different angles, which was important because I had some difficulty typing with it flat. The bluetooth receiver in the charger was functional, and I appreciated having a bluetooth hub for my cellphone. The mouse and the keyboard have both proved durable and reliable despite a number of mishaps.

---

For each of inferences (a) through (d) below,

1. state whether the inference depends on the **discourse context, knowledge about actions**, and/or **general world knowledge**; and
  2. describe what natural language processing techniques, if any, might enable a system to make the inference automatically.
- (a) The reviewer owns the keyboard.  
(b) The charger is part of the keyboard.  
(c) The reviewer had difficulty typing with the keyboard.  
(d) The reviewer likes the keyboard.

## 5 Grab Bag (20 pts)

- (a) (4 pts) (True or False. Explain your answer.) Information extraction is harder than text categorization.

### Answer

**Fall 2006 students: we did not cover information extraction.**

- (b) (6 pts) Briefly describe the key differences between **Autoslog-TS** and **Autoslog**.

### Answer

**Fall 2006 students: we did not cover this.** Autoslog-TS is largely unsupervised. It does not require annotations, but instead, requires two sets of documents: relevant and not relevant. After extracting every NP from the texts, it selects patterns by *relevance rate* and frequency.

- (c) (4 pts) (True or False. Explain your answer.) 4-grams are better than trigrams for part-of-speech tagging.

### Answer

False. There is not generally enough data for 4-grams to outperform trigrams.

- (d) (6 pts) Noun phrase coreference resolution includes **pronoun** resolution, **proper noun** resolution, and **common noun** resolution. Which of the three would you expect to be the most difficult to handle computationally? Explain why.

### Answer

Common noun is the hardest, because there can be drastically broad way of corefering the same entity. The variety of proper noun and pronoun coreference patterns are relatively much narrower.