PCFGs: Parameter Estimation, Lexicalization and Parsing
Informatics 2A: Lecture 19

John Longley (slides by Bonnie Webber)

School of Informatics
University of Edinburgh

11 November 2010
1. Standard PCFGs
   - Parameter Estimation
   - Problem 1: Ignoring Lexical Information
   - Problem 2: Assuming Independence

2. Lexicalized PCFGs
   - Lexicalization
   - Head Lexicalization
   - Parameter Estimation

3. Parsing PCFGs

   Reading:

   NLTK Book, Chapter 8, final section on Weighted Grammar
Where do we get the parameters (i.e., rule probabilities) for a PCFG?

The easiest way is from a large parsed corpus such as the Penn Treebank.

Given a large parsed corpus, we can obtain:

- the grammar rules by reading them off the trees in the corpus;
- the rule probabilities by comparing the frequency with which a given rule occurs in the corpus, compared with the frequency of its LHS. That is,

\[
P(\alpha \rightarrow \beta | \alpha) = \frac{\text{Count}(\alpha \rightarrow \beta)}{\sum_{\gamma} \text{Count}(\alpha \rightarrow \gamma)} = \frac{\text{Count}(\alpha \rightarrow \beta)}{\text{Count}(\alpha)}
\]
Here’s a parsed corpus of sentences and parse trees.

S1: [S [NP grass] [VP grows]]
S2: [S [NP grass] [VP grows] [AP slowly]]
S3: [S [NP grass] [VP grows] [AP fast]]
S4: [S [NP bananas] [VP grow]]

We can compute PCFG probabilities as follows:

| $r$ | Rule               | $\alpha$ | $P(r|\alpha)$ |
|-----|--------------------|----------|---------------|
| $r_1$ | $S \rightarrow NP$ $VP$ | $S$      |               |
| $r_2$ | $S \rightarrow NP$ $VP$ $AP$ | $S$      |               |
Here's a parsed corpus of sentences and parse trees.

S1: [S [NP grass] [VP grows]]
S2: [S [NP grass] [VP grows] [AP slowly]]
S3: [S [NP grass] [VP grows] [AP fast]]
S4: [S [NP bananas] [VP grow]]

We can continue computing PCFG probabilities:

| $r$  | Rule          | $\alpha$ | $P(r|\alpha)$ |
|------|---------------|----------|----------------|
| $r1$ | $S \rightarrow$ NP VP | S        | 2/4            |
| $r2$ | $S \rightarrow$ NP VP AP  | S        | 2/4            |
| $r3$ | NP $\rightarrow$ grass   | NP       |                |
| $r4$ | NP $\rightarrow$ bananas | NP       |                |
Here’s a parsed corpus of sentences and parse trees.

S1: [S [NP grass] [VP grows]]
S2: [S [NP grass] [VP grows] [AP slowly]]
S3: [S [NP grass] [VP grows] [AP fast]]
S4: [S [NP bananas] [VP grow]]

We can continue computing PCFG probabilities:

| $r$  | Rule                  | $\alpha$ | $P(r|\alpha)$ |
|------|-----------------------|----------|---------------|
| $r_1$ | $S \rightarrow$ NP VP | S        | 2/4           |
| $r_2$ | $S \rightarrow$ NP VP AP | S        | 2/4           |
| $r_3$ | NP $\rightarrow$ grass | NP       | 3/4           |
| $r_4$ | NP $\rightarrow$ bananas | NP       | 1/4           |
| $r_5$ | VP $\rightarrow$ grows | VP       |               |
| $r_6$ | VP $\rightarrow$ grow  | VP       |               |
Parameter Estimation

Here’s a parsed corpus of sentences and parse trees.

S1: [S [NP grass] [VP grows]]
S2: [S [NP grass] [VP grows] [AP slowly]]
S3: [S [NP grass] [VP grows] [AP fast]]
S4: [S [NP bananas] [VP grow]]

We can continue computing PCFG probabilities:

| $r$   | Rule             | $\alpha$ | $P(r|\alpha)$ |
|-------|------------------|----------|---------------|
| $r1$  | S $\rightarrow$ NP VP | S        | 2/4           |
| $r2$  | S $\rightarrow$ NP VP AP | S        | 2/4           |
| $r3$  | NP $\rightarrow$ grass | NP      | 3/4           |
| $r4$  | NP $\rightarrow$ bananas | NP      | 1/4           |
| $r5$  | VP $\rightarrow$ grows | VP      | 3/4           |
| $r6$  | VP $\rightarrow$ grow | VP      | 1/4           |
| $r7$  | AP $\rightarrow$ fast | AP      |               |
| $r8$  | AP $\rightarrow$ slowly | AP      |               |
Here’s a parsed corpus of sentences and parse trees.

S1: [S [NP grass] [VP grows]]
S2: [S [NP grass] [VP grows] [AP slowly]]
S3: [S [NP grass] [VP grows] [AP fast]]
S4: [S [NP bananas] [VP grow]]

We now have all possible PCFG probabilities:

| $r$ | Rule | $\alpha$ | $P(r|\alpha)$ |
|-----|------|----------|----------------|
| $r_1$ | $S \rightarrow NP \ VP$ | $S$ | 2/4 |
| $r_2$ | $S \rightarrow NP \ VP \ AP$ | $S$ | 2/4 |
| $r_3$ | $NP \rightarrow grass$ | $NP$ | 3/4 |
| $r_4$ | $NP \rightarrow bananas$ | $NP$ | 1/4 |
| $r_5$ | $VP \rightarrow grows$ | $VP$ | 3/4 |
| $r_6$ | $VP \rightarrow grow$ | $VP$ | 1/4 |
| $r_7$ | $AP \rightarrow fast$ | $AP$ | 1/2 |
| $r_8$ | $AP \rightarrow slowly$ | $AP$ | 1/2 |
With these parameters (rule probabilities), we can now compute the probabilities of the four sentences S1–S4:

\[
P(S1) = P(r_1|S)P(r_3|NP)P(r_5|VP) \\
= \frac{2}{4} \cdot \frac{3}{4} \cdot \frac{3}{4} = 0.28125
\]

\[
P(S2) = P(r_2|S)P(r_3|NP)P(r_5|VP)P(r_7|AP) \\
= \frac{2}{4} \cdot \frac{3}{4} \cdot \frac{3}{4} \cdot \frac{1}{2} = 0.140625
\]

\[
P(S3) = P(r_2|S)P(r_3|NP)P(r_5|VP)P(r_7|AP) \\
= \frac{2}{4} \cdot \frac{3}{4} \cdot \frac{3}{4} \cdot \frac{1}{2} = 0.140625
\]

\[
P(S4) = P(r_1|S)P(r_4|NP)P(r_6|VP) \\
= \frac{2}{4} \cdot \frac{1}{4} \cdot \frac{1}{4} = 0.03125
\]
Problems with Standard PCFGs

While standard PCFGs are useful for a number of applications, they can produce a wrong result when used to choose the correct parse for an ambiguous sentence.

How can that be?

1. They ignore lexical information until the very end of the analysis, when word classes are rewritten to word tokens.
2. They make unwarranted independence assumptions.

How can this lead to the wrong choice among possible parses?
Problem 1: Ignoring Lexical Information

Consider the sentences:

(1)  
  a. They admired the painting of the queen.  
  b. They put the painting on the wall.

Because rules for rewriting non-terminals ignore word tokens until the very end, let’s consider these simply as strings of POS tags:

(2)  
  a. PRO TV DET N PREP DET N  
  b. PRO TV DET N PREP DET N

using the lexical rules:

\[
\begin{align*}
N & \rightarrow \text{painting} \mid \text{queen} \mid \text{wall} \\
TV & \rightarrow \text{admired} \mid \text{put} \\
PREP & \rightarrow \text{of} \mid \text{on} \\
\end{align*}
\]

\[
\begin{align*}
\text{PRO} & \rightarrow \text{they} \\
\text{DET} & \rightarrow \text{the} \\
\end{align*}
\]
Problem 1: Ignoring Lexical Information

The grammar

\[ S \rightarrow NP \ VP \]
\[ NP \rightarrow PRO \mid DET \ N \mid NP \ PP \]
\[ VP \rightarrow TV \ NP \mid TV \ NP \ PP \]
\[ PP \rightarrow PREP \ NP \]

provides two possible analyses for the string of POS tags

\[ PRO \ TV \ DET \ N \ PREP \ DET \ N \]
Problem 1: Ignoring Lexical Information

Informatics 2A: Lecture 19

PCFGs: Parameter Estimation, Lexicalization and Parsing
Problem 1: Ignoring Lexical Information

Which do we want for "They admired the painting of the queen"? Which for "They put the painting on the wall"?

The most appropriate analysis depends, in part, on the actual words in the sentence, and not just their POS tags.
Problem 2: Assuming Independence

By definition, a CFG assumes that the expansion of non-terminals is completely independent: It doesn’t matter

- where a non-terminal is in the analysis;
- what else is (or isn’t) in the analysis.

The same assumption holds for standard PCFGs: The probability of a rule is the same, no matter

- where it is applied in the analysis;
- what else is (or isn’t) in the analysis.

But this assumption is too simple.
Consider the rules:

- $S \rightarrow NP \ VP \ (p1)$
- $VP \rightarrow TV \ NP \ (p2)$
- $NP \rightarrow PRO \ (p3)$
- $NP \rightarrow DET \ NOM \ (p4)$

They assign the same probability to both these trees, because they use the same 5 re-write rules, and probability calculations don’t depend on where rules are used.
Independence Assumption

But in one large speech corpus, 91% of 31021 subject NPs are pronouns:

(3)  
  a. She’s able to take her baby to work with her.  
  b. My wife worked until we had a family.

while only 34% of 7489 object NPs are pronouns:

(4)  
  a. Some laws absolutely prohibit it.  
  b. It wasn’t clear how NL and Mr. Simmons would respond if Georgia Gulf spurns them again.

So the probability of NP → PRO should depend on where in the analysis it applies.
In Lecture 13, we saw that each non-terminal in a natural language has a head which determines the syntactic properties of the phrase (e.g., which other phrases it can combine with).

**Example**

- Noun Phrase (NP): Noun
- Adjective Phrase (AP): Adjective
- Verb Phrase (VP): Verb
- Prepositional Phrase (PP): Preposition

**Key idea:** Have each grammar rule specify its lexical head and use it to avoid unwarranted independence assumptions.

**How?**
We can lexicalize a PCFG by annotating each non-terminal with its head word, starting with the terminals – replacing

\[
\begin{align*}
VP & \rightarrow V \ NP \ PP \quad (p1) \\
VP & \rightarrow V \ NP \quad (p2) \\
NP & \rightarrow \text{DET} \ \text{NOM} \quad (p3) \\
\text{NOM} & \rightarrow N \ PP \quad (p4)
\end{align*}
\]

with rules of the form

\[
\begin{align*}
\text{VP(put)} & \rightarrow V(\text{put}) \ \text{NP(} \text{painting} \text{)} \ \text{PP(on)} \quad (p1) \\
\text{VP(admired)} & \rightarrow V(\text{admired}) \ \text{NP(} \text{painting} \text{)} \ \text{PP(on)} \quad (p2) \\
\text{VP(find)} & \rightarrow V(\text{find}) \ \text{NP(} \text{painting} \text{)} \ \text{PP(on)} \quad (p3) \\
\text{VP(put)} & \rightarrow V(\text{put}) \ \text{NP(} \text{painting} \text{)} \quad (p4) \\
\text{VP(admired)} & \rightarrow V(\text{admired}) \ \text{NP(} \text{painting} \text{)} \quad (p5) \\
\text{NP(painting)} & \rightarrow \text{DET(} \text{the} \text{)} \ \text{NOM(} \text{painting} \text{)} \quad (p6)
\end{align*}
\]
Standard PCFGs
Lexicalized PCFGs
Parsing PCFGs

Lexicalization
Head Lexicalization
Parameter Estimation

Example

S(admired)
  NP(they)
  PRO(they)
    They
  VP(admired)
    TV(admired)
      admired
    NP(painting)
      NP(painting)
        DET(the)
        N(painting)
        the
        painting
      PP(of)
        PREP(of)
        NP(queen)
        DET(the)
        N(queen)
        of
        the
        queen

Informatics 2A: Lecture 19
PCFGs: Parameter Estimation, Lexicalization and Parsing
But this would mean an enormous expansion in grammar rules, with no parsed corpus big enough to estimate their probabilities accurately.

Instead we just lexicalize the head of phrase:

<table>
<thead>
<tr>
<th>Grammar Rule</th>
<th>Parse Rule</th>
</tr>
</thead>
<tbody>
<tr>
<td>VP(put)</td>
<td>V(put) NP PP</td>
</tr>
<tr>
<td>VP(admired)</td>
<td>V(admired) NP PP</td>
</tr>
<tr>
<td>VP(find)</td>
<td>V(find) NP PP</td>
</tr>
<tr>
<td>VP(put)</td>
<td>V(put) NP</td>
</tr>
<tr>
<td>VP(admired)</td>
<td>V(admired) NP</td>
</tr>
<tr>
<td>VP(find)</td>
<td>V(find) NP</td>
</tr>
<tr>
<td>NP(painting)</td>
<td>DET NOM(painting)</td>
</tr>
<tr>
<td>NOM(painting)</td>
<td>N(painting) PP</td>
</tr>
</tbody>
</table>

Such grammars are called **lexicalized PCFGs** or, alternatively, probabilistic lexicalized CFGs.
The probabilities of lexicalized PCFGs can be estimated from a syntactically annotated corpus like the Penn TreeBank.

We want to estimate both:

- **rule probabilities** $P(\alpha \rightarrow \beta|\alpha, h(\alpha))$ that condition a rule on its LHS and head, $h(\alpha)$;
- **head probabilities** $P(h(\alpha)|\alpha, h(m(\alpha)))$ that condition the head of a rule on the head of the rule’s mother, $h(m(\alpha))$.

These probabilities can be estimated as follows:

\[
P(\alpha \rightarrow \beta|\alpha, h(\alpha)) = \frac{\text{Count}(\alpha(h(\alpha)) \rightarrow \beta)}{\text{Count}(\alpha(h(\alpha)))}
\]

\[
P(h(\alpha)|\alpha, h(m(\alpha))) = \frac{\text{Count}(X(h(m(\alpha))) \rightarrow \ldots \alpha(h(\alpha)) \ldots)}{\text{Count}(X(h(m(\alpha))) \rightarrow \ldots \alpha \ldots)}
\]
We can estimate rule probabilities from the parsed corpus:

S1: [S [NP grass] [VP grows]]
S2: [S [NP grass] [VP grows] [AP slowly]]
S3: [S [NP grass] [VP grows] [AP fast]]
S4: [S [NP bananas] [VP grow]]

| $r$  | Rule                | $\alpha$ | $h(\alpha)$ | $P(r|\alpha, h(\alpha))$ |
|------|---------------------|-----------|-------------|--------------------------|
| $r_1$ | S $\rightarrow$ NP VP | S         | grows       |                          |
| $r_2$ | S $\rightarrow$ NP VP AP | S         | grows       |                          |
| $r_3$ | S $\rightarrow$ NP VP | S         | grow        |                          |

where $P(\alpha \rightarrow \beta|\alpha, h(\alpha)) = \frac{\text{Count}(\alpha(h(\alpha)) \rightarrow \beta)}{\text{Count}(\alpha(h(\alpha)))}$
We can estimate rule probabilities from the parsed corpus:

S1: \([S \ [NP \ grass] \ [VP \ grows]]\)
S2: \([S \ [NP \ grass] \ [VP \ grows] \ [AP \ slowly]]\)
S3: \([S \ [NP \ grass] \ [VP \ grows] \ [AP \ fast]]\)
S4: \([S \ [NP \ bananas] \ [VP \ grow]]\)

\[
\begin{array}{|c|c|c|c|}
\hline
\mathbf{r} & \text{Rule} & \alpha & h(\alpha) & P(\mathbf{r}|\alpha, h(\alpha)) \\
\hline
r_1 & S \rightarrow NP \ VP & S & \text{grows} & 1/3 \\
r_2 & S \rightarrow NP \ VP \ AP & S & \text{grows} & 2/3 \\
r_3 & S \rightarrow NP \ VP & S & \text{grow} & 1/1 \\
r_4 & NP \rightarrow \text{grass} & NP & \text{grass} & \\
r_5 & NP \rightarrow \text{bananas} & NP & \text{bananas} & \\
\hline
\end{array}
\]

where \(P(\alpha \rightarrow \beta|\alpha, h(\alpha)) = \frac{\text{Count}(\alpha(h(\alpha)) \rightarrow \beta)}{\text{Count}(\alpha(h(\alpha)))}\)
We can estimate rule probabilities from the parsed corpus:

S1: [S [NP grass] [VP grows]]
S2: [S [NP grass] [VP grows] [AP slowly]]
S3: [S [NP grass] [VP grows] [AP fast]]
S4: [S [NP bananas] [VP grow]]

|   | Rule          | \( \alpha \) | \( h(\alpha) \) | \( P(r|\alpha, h(\alpha)) \) |
|---|---------------|---------------|------------------|-------------------------------|
| r1| S → NP VP     | S             | grows           | 1/3                           |
| r2| S → NP VP AP  | S             | grows           | 2/3                           |
| r3| S → NP VP     | S             | grow            | 1/1                           |
| r4| NP → grass    | NP            | grass           | 3/3                           |
| r5| NP → bananas  | NP            | bananas         | 1/1                           |
| r6| VP → grows    | VP            | grows           | 3/3                           |
| r7| VP → grow     | VP            | grow            | 1/1                           |
| r8| AP → fast     | AP            | fast            | 1/1                           |
| r9| AP → slowly   | AP            | slowly          | 1/1                           |
And the head probabilities from this corpus as well:

|   | Rule        | α  | $h(\alpha)$ | $h(m(\alpha))$ | $P(h(\alpha)|h(m(\alpha)))$ |
|---|-------------|----|--------------|-----------------|-------------------------------|
| $r_1$ | $S \rightarrow NP\ VP$ | $S$ | grows | - | 1 |
| $r_2$ | $S \rightarrow NP\ VP\ AP$ | $S$ | grows | - | 1 |
| $r_3$ | $S \rightarrow NP\ VP$ | $S$ | grow | - | 1 |
| $r_4$ | $NP \rightarrow grass$ | NP | grass | grows | $3/3$ |
| $r_5$ | $NP \rightarrow bananas$ | NP | bananas | grow | $1/1$ |
| $r_6$ | $VP \rightarrow grows$ | VP | grows | grows | $3/3$ |
| $r_7$ | $VP \rightarrow grow$ | VP | grow | grow | $1/1$ |
| $r_8$ | $AP \rightarrow fast$ | AP | fast | grows | $1/2$ |
| $r_9$ | $AP \rightarrow slowly$ | AP | slowly | grows | $1/2$ |

where $P(h(\alpha)|\alpha, h(m(\alpha))) = \frac{\text{Count}(X(h(m(\alpha))) \rightarrow \ldots \alpha(h(\alpha)) \ldots)}{\text{Count}(X(h(m(\alpha))) \rightarrow \ldots \alpha \ldots)}$
The lexicalized PCFG probability of sentence $S$ with parse tree $T$ is:

$$P(T, S) = \prod_{\alpha \in T} P(\alpha \rightarrow \beta | \alpha, h(\alpha)) P(h(\alpha) | \alpha, h(m(\alpha)))$$

$\Rightarrow$ the product of the rule probability and head probability at each non-terminal node.

For example, the probability of $S2$ is:

$$P(S2) = P(S \rightarrow NP \ VP \ AP | S, grows) P(grows | S, -)$$

$$P(NP \rightarrow grass | NP, grass) P(grass | NP, grows)$$

$$P(VP \rightarrow grows | VP, grows) P(grows | VP, grows)$$

$$P(AP \rightarrow slowly | AP, slowly) P(slowly | AP, grows)$$

$$= 1/3 \cdot 1 \cdot 3/3 \cdot 3/3 \cdot 3/3 \cdot 3/3 \cdot 1/1 \cdot 1/2$$

$$= 0.1667$$
Different chart parsing algorithms can be used with PCFGs. They can use the probabilities

- to add only the most likely analysis to the chart
- to decide in what order to add edges to the chart (i.e., in order of likelihood
- to limit which edges get added to the chart (i.e., only the more likely ones)
In Lecture 18, we saw two analyses of

\[0 \text{ Can } 1 \text{ you } 2 \text{ book } 3 \text{ TWA } 4 \text{ flights? } 5\]

from the PCFG

<table>
<thead>
<tr>
<th>Rule</th>
<th>Production</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td>S → NP VP</td>
<td>0.85</td>
</tr>
<tr>
<td>R2</td>
<td>S → Aux NP VP</td>
<td>0.15</td>
</tr>
<tr>
<td>R3</td>
<td>NP → PRO</td>
<td>0.4</td>
</tr>
<tr>
<td>R4</td>
<td>NP → NOM</td>
<td>0.05</td>
</tr>
<tr>
<td>R5</td>
<td>NP → NPR</td>
<td>0.35</td>
</tr>
<tr>
<td>R6</td>
<td>NP → NPR NOM</td>
<td>0.2</td>
</tr>
<tr>
<td>R7</td>
<td>NOM → N</td>
<td>0.75</td>
</tr>
<tr>
<td>R8</td>
<td>NOM → N PP</td>
<td>0.25</td>
</tr>
<tr>
<td>R9</td>
<td>VP → TV NP NP</td>
<td>0.05</td>
</tr>
<tr>
<td>R10</td>
<td>VP → TV NP</td>
<td>0.4</td>
</tr>
<tr>
<td>R11</td>
<td>VP → IV</td>
<td>0.55</td>
</tr>
<tr>
<td>R12</td>
<td>Aux → can</td>
<td>0.4</td>
</tr>
<tr>
<td>R13</td>
<td>N → flights</td>
<td>0.5</td>
</tr>
<tr>
<td>R14</td>
<td>PRO → you</td>
<td>0.4</td>
</tr>
<tr>
<td>R15</td>
<td>TV → book</td>
<td>0.3</td>
</tr>
<tr>
<td>R16</td>
<td>NPR → TWA</td>
<td>0.4</td>
</tr>
</tbody>
</table>
This grammar allows two ways of adding VP to the chart from 2 to 5 (ie, 2 book 3 TWA 4 flights? 5).

\[
\text{VP: } P(R9) \times P(R5) \times P(R4) = 0.05 \times 0.35 \times 0.05 = 0.000875 \\
\text{VP: } P(R10) \times P(R6) = 0.4 \times 0.2 = 0.08
\]

So only an edge for the latter would be added to the chart.
The rule probabilities of a PCFG can be estimated by counting how often the rules occur in a corpus.

The usefulness of PCFGs is limited by the lack of lexical information and by strong independence assumptions.

These limitations can be overcome by lexicalizing the grammars, i.e., by conditioning the rule probabilities on the head word of the rule.

Several parameter estimation methods are available for lexicalized PCFGs.

A chart parser can be adapted to use the probabilities in a PCFG.