Syntax and Parsing I

Constituency Parsing

Slav Petrov – Google

Thanks to:

Dan Klein, Ryan McDonald, Alexander Rush, Joakim Nivre, Greg Durrett, David Weiss, Luheng He, Timothy Dozat

Lisbon Machine Learning School 2018
On January 13, 2018, a false ballistic missile alert was issued via the Emergency Alert System and Commercial Mobile Alert System over television, radio, and cellphones in the U.S. state of Hawaii. The alert stated that there was an incoming ballistic missile threat to Hawaii, advised residents to seek shelter, and concluded "This is not a drill". The message was sent at 8:07 a.m. local time.
Analyzing Natural Language

```
S
  /   
/     
NP    VP
  /   /   
P   V   NP
|    /   PP
|   |   |
They solved the problem with statistics
```
They solved the problem with statistics
They solved the problem with statistics.
Constituency and Dependency

ROOT

PRON

VERB

DET

NOUN

ADP

NOUN

They solved the problem with statistics
Influential members of the House Ways and Means Committee introduced legislation that would restrict how the new savings-and-loan bailout agency can raise capital, creating another potential obstacle to the government's sale of sick thrifts.
Dependency Parsing

- Directed edges between pairs of word (head, dependent)
- Can handle free word-order languages
- Very efficient decoding algorithms exist
- Second part of today’s lecture
Attachments

• I cleaned the dishes from dinner
• I cleaned the dishes with detergent
• I cleaned the dishes in my pajamas
• I cleaned the dishes in the sink
Classical NLP: Parsing

- Write symbolic or logical rules:
  - VBD VB
  - VBN VBZ VBP VBZ
  - NNP NNS NN NNS CD NN
  - Fed raises interest rates 0.5 percent

<table>
<thead>
<tr>
<th>Grammar (CFG)</th>
<th>Lexicon</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT → S</td>
<td>NP → NP PP</td>
</tr>
<tr>
<td>S → NP VP</td>
<td>VP → VBP NP</td>
</tr>
<tr>
<td>NP → DT NN</td>
<td>VP → VBP NP PP</td>
</tr>
<tr>
<td>NP → NN NNS</td>
<td>PP → IN NP</td>
</tr>
<tr>
<td>NNP NNS CD NN</td>
<td></td>
</tr>
</tbody>
</table>

- Use deduction systems to prove parses from words
  - Minimal grammar on “Fed raises” sentence: 36 parses
  - Real-size grammar: many millions of parses

- This scaled very badly, didn’t yield broad-coverage tools
A context-free grammar is a tuple \(<N, T, S, R>\)

- \(N\) : the set of non-terminals
  - Phrasal categories: \(S, NP, VP, ADJP, \text{ etc.}\)
  - 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 \(X \rightarrow Y_1 Y_2 \ldots Y_k\), with \(X, Y_i \in N\)
  - Examples: \(S \rightarrow NP \ VP, \ VP \rightarrow \ VP \ CC \ VP\)
  - Also called rewrites, productions, or local trees

A PCFG adds:

- A top-down production probability per rule \(P(Y_1 Y_2 \ldots Y_k | X)\)
Treebank Grammars

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

  \[
  S \rightarrow NP \ VP . \\
  NP \rightarrow PRP \ 0.5 \ \\
  NP \rightarrow DT \ NN \ 0.5 \ \\
  VP \rightarrow VBD \ NP \ 1.0 \ \\
  PRP \rightarrow She \ 1.0 \\
  \ldots
  \]

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

- Treebank grammars can be enormous
  - As FSAs, the raw grammar has ~10K states, excluding the lexicon
  - Better parsers usually make the grammars larger, not smaller
Chomsky Normal Form

- **Chomsky normal form:**
  - All rules of the form $X \rightarrow Y Z$ or $X \rightarrow w$
  - In principle, this is no limitation on the space of (P)CFGs
    - N-ary rules introduce new non-terminals
  - Unaries / empties are "promoted"

- In practice it’s kind of a pain:
  - Reconstructing n-aries is easy
  - Reconstructing unaries is trickier
  - The straightforward transformations don’t preserve tree scores

- Makes parsing algorithms simpler!
A Recursive Parser

bestScore(X,i,j,s)
   if (j = i+1)
      return tagScore(X,s[i])
   else
      return max score(X->YZ) *
         bestScore(Y,i,k) *
         bestScore(Z,k,j)

• Will this parser work?
• Why or why not?
• Memory requirements?
A Memoized Parser

• One small change:

```java
bestScore(X,i,j,s) {
    if (scores[X][i][j] == null)
    if (j = i+1)
        score = tagScore(X,s[i])
    else
        score = max score(X->YZ) * 
                bestScore(Y,i,k) * 
                bestScore(Z,k,j)

    scores[X][i][j] = score
    return scores[X][i][j]
}
```
A Bottom-Up Parser (CKY)

- Can also organize things bottom-up

```plaintext
bestScore(s)
    for (i : [0, n-1])
        for (X : tags[s[i]])
            score[X][i][i+1] =
                tagScore(X, s[i])
    for (diff : [2, n])
        for (i : [0, n-diff])
            j = i + diff
            for (X->YZ : rule)
                for (k : [i+1, j-1])
                    score[X][i][j] = max score[X][i][j],
                        score(X->YZ) * score[Y][i][k] * score[Z][k][j]
```
Time: Theory

• How much time will it take to parse?

• For each diff (<= n)
  • For each i (<= n)
    • For each rule X → Y Z
      • For each split point k
        Do constant work

• Total time: |rules|*n^3

• Something like 5 sec for an unoptimized parse of a 20-word sentences, or 0.2sec for an optimized parser
Unary Rules

Unary rules?

\[
\text{bestScore}(X,i,j,s) \\
\quad \text{if } (j = i+1) \\
\quad \quad \text{return } \text{tagScore}(X,s[i]) \\
\quad \text{else} \\
\quad \quad \text{return } \max \ \max \ \text{score}(X\rightarrow YZ) \ \times \\
\quad \quad \quad \text{bestScore}(Y,i,k) \ \times \\
\quad \quad \quad \text{bestScore}(Z,k,j) \\
\quad \quad \quad \quad \max \ \text{score}(X\rightarrow Y) \ \times \\
\quad \quad \quad \quad \quad \text{bestScore}(Y,i,j)
\]
CNF + Unary Closure

• We need unaries to be non-cyclic
  • Can address by pre-calculating the unary closure
  • Rather than having zero or more unaries, always have exactly one

• Alternate unary and binary layers
• Reconstruct unary chains afterwards
Alternating Layers

\[
\text{bestScoreB}(X,i,j,s) \\
\quad \text{return } \max \max \text{ score}(X-YZ) * \\
\quad \text{bestScoreU}(Y,i,k) * \\
\quad \text{bestScoreU}(Z,k,j)
\]

\[
\text{bestScoreU}(X,i,j,s) \\
\quad \text{if } (j = i+1) \\
\quad \quad \text{return } \text{tagScore}(X,s[i]) \\
\quad \text{else} \\
\quad \quad \text{return } \max \max \text{ score}(X-Y) * \\
\quad \quad \quad \text{bestScoreB}(Y,i,j)
\]
Treebank Grammars

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

```
S \rightarrow NP VP .

NP \rightarrow PRP 0.5
NP \rightarrow DT NN 0.5
VP \rightarrow VBD NP 1.0
PRP \rightarrow She 1.0
```

- Better results by enriching the grammar (e.g., lexicalization).
- Can also get reasonable parsers

<table>
<thead>
<tr>
<th>Model</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Charniak ’96</td>
<td>72.0</td>
</tr>
</tbody>
</table>
Conditional Independence?

- Not every NP expansion can fill every NP slot

- A grammar with symbols like “NP” won’t be context-free

- Statistically, conditional independence too strong
Non-Independence

- Independence assumptions are often too strong.

- Example: the expansion of an NP is highly dependent on the parent of the NP (i.e., subjects vs. objects).
- Also: the subject and object expansions are correlated!
The Game of Designing a Grammar

- Structure Annotation [Johnson ’98, Klein & Manning ’03]
- Lexicalization [Collins ’99, Charniak ’00]
- Latent Variables [Matsuzaki et al. 05, Petrov et al. ’06]
- (Neural) CRF Parsing [Hall et al. ’14, Durrett & Klein ’15]
A Fully Annotated (Unlexicalized) Tree

[Klein & Manning ‘03]

Model | F1
---|---
Charniak ’96 | 72.0
Klein&Manning ’03 | 86.3

ROOT

S^ROOT-v

"^S NP^S-B

" DT-U^NP

This

VP^S-VBF-v

VBZ^BE^VP

is

NP^VP-B

NP^NP

NN^NP

.^[S "^S
The Game of Designing a Grammar

- Annotation refines base treebank symbols to improve statistical fit of the grammar
  - Head lexicalization [Collins ’99, Charniak ’00]
Problems with PCFGs

- If we do no annotation, these trees differ only in one rule:
  - \( VP \rightarrow VP \, PP \)
  - \( NP \rightarrow NP \, PP \)
- Parse will go one way or the other, regardless of words
- We addressed this in one way with unlexicalized grammars (how?)
- Lexicalization allows us to be sensitive to specific words
Problems with PCFGs

- What’s different between basic PCFG scores here?
- What (lexical) correlations need to be scored?
Lexicalized Trees

- Add “headwords” to each phrasal node
  - Syntactic vs. semantic heads
  - Headship not in (most) treebanks
  - Usually use head rules, e.g.:  
    - **NP:**
      - Take leftmost NP
      - Take rightmost N*
      - Take rightmost JJ
      - Take right child
    - **VP:**
      - Take leftmost VB*
      - Take leftmost VP
      - Take left child

[Charniak ’97, Collins ’97]
Lexicalized Grammars

- **Challenges:**
  - Many parameters to estimate: requires sophisticated smoothing techniques
  - Exact inference is too slow: requires pruning heuristics
  - Difficult to adapt to new languages: At least head rules need to be specified, typically more changes needed

<table>
<thead>
<tr>
<th>Model</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Klein&amp;Manning ’03</td>
<td>86.3</td>
</tr>
<tr>
<td>Charniak ’00</td>
<td>89.7</td>
</tr>
</tbody>
</table>
The Game of Designing a Grammar

- Annotation refines base treebank symbols to improve statistical fit of the grammar
  - Automatic clustering
Latent Variable Grammars

[Matsuzaki et al. '05, Petrov et al. '06]
Learning Latent Annotations

EM algorithm:
- Brackets are known
- Base categories are known
- Only induce subcategories

Just like Forward-Backward for HMMs.

<table>
<thead>
<tr>
<th>Model</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Charniak ’00</td>
<td>89.7</td>
</tr>
<tr>
<td>Petrov et al. ‘06</td>
<td>90.1</td>
</tr>
</tbody>
</table>
The Game of Designing a Grammar

- Annotation refines base treebank symbols to improve statistical fit of the grammar
  - CRF Parsing (+Neural Network Representations)
Generative vs. Discriminative

**Generative**

Maximize joint likelihood of gold tree and sentence

EM-algorithm

EASY: expectations over observed trees

[Matsuzaki et al. ‘05, Petrov et al. ‘06]

**Discriminative**

Maximize conditional likelihood of gold tree given sentence

Gradient-based algorithm

HARD: expectations over all trees

[Petrov & Klein ‘07, ‘08]
Objective Functions

Generative Objective Function:

$$\max_{\theta} \mathcal{L}_\theta (y, w_1...w_n)$$

[ Petrov, Barrett, Thibaux & Klein ’06]

Discriminative Objective Function:

$$\max_{\theta} \mathcal{L}_\theta (y| w_1...w_n)$$

[ Petrov & Klein ’08, Finkel et. al ’08]

Bayesian Objective Function:

$$\max_{\theta} \mathcal{P}(\theta| y) \mathcal{L}_\theta (y, w_1...w_n)$$

[ Liang, Petrov, Jordan & Klein ’07]
Be a tree

He gave a speech

Score of VP over this span

\( w \cdot f_s \)

dense neural network

sparse log-linear model
CRF Parsing Sparse Features

\[
P(T|x) \propto \prod_{r \in T} \exp(\text{score}(r))
\]

\[
\text{score}(2\text{NP}_7 \rightarrow 2\text{NP}_{44}\text{PP}_7) = w^\top f(2\text{NP}_7 \rightarrow 2\text{NP}_{44}\text{PP}_7)
\]

FirstWord = a & NP \rightarrow NP PP
PrevWord = gave & NP \rightarrow NP PP
AfterSplit = on & NP \rightarrow NP PP
FirstWord = a & NP
...

He gave a speech on foreign policy.
Neural CRF Model

score(2NP₇ → 2NP₄₄PP₇) =

\[ W \odot \left( f_s(2X_7 \rightarrow 2X_44X_7) f_o^T(NP \rightarrow NP PP) \right) \]

\[ f_s = g(Hv) \]

(-arbitrary neural network)

He gave a speech on foreign policy.

[Durrett et al. ’15]

<table>
<thead>
<tr>
<th>Model</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Petrov et al. ‘06</td>
<td>90,1</td>
</tr>
<tr>
<td>Durrett et. al. ‘15</td>
<td>91,1</td>
</tr>
</tbody>
</table>
LSTM Parsing [Vinyals et al. ’15]

- Treat parsing as a sequence-to-sequence prediction problem
- Completely ignores tree structure, uses LSTMs as black boxes
- No global normalization, only local normalization

<table>
<thead>
<tr>
<th>Model</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Durrett et. al. ‘15</td>
<td>91.1</td>
</tr>
<tr>
<td>Vinyals et al. ‘15</td>
<td>88.6*</td>
</tr>
</tbody>
</table>
Parsing with Self-Attention

Output

Decoder

Encoder

Input

...(VP(VBD fled))(NP(DT the)(NN market))...

[Kitaev & Kleint ’18]
Detailed English Results (Old)

<table>
<thead>
<tr>
<th>Method</th>
<th>Single Parser</th>
<th>Self-Trained</th>
<th>Reranker</th>
<th>Product</th>
<th>Combination</th>
</tr>
</thead>
<tbody>
<tr>
<td>Charniak '00</td>
<td>91.0</td>
<td>91.4</td>
<td>92.3</td>
<td>92.4</td>
<td>92.8</td>
</tr>
<tr>
<td>Petrov et al. '06</td>
<td>91.1</td>
<td>91.5</td>
<td>92.4</td>
<td>92.6</td>
<td>92.8</td>
</tr>
<tr>
<td>Carreras et al. '08</td>
<td>91.1</td>
<td>91.7</td>
<td>92.3</td>
<td>92.4</td>
<td>92.8</td>
</tr>
<tr>
<td>Durrett et al. '15</td>
<td>91.3</td>
<td>91.7</td>
<td>92.4</td>
<td>92.6</td>
<td>92.8</td>
</tr>
<tr>
<td>Zhu et al. '13</td>
<td>91.3</td>
<td>91.7</td>
<td>92.4</td>
<td>92.6</td>
<td>92.8</td>
</tr>
<tr>
<td>Dyer et al. '16</td>
<td>93.6</td>
<td>92</td>
<td>91.8</td>
<td>92</td>
<td>92.8</td>
</tr>
<tr>
<td>[Charniak '00]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[Petrov et al. '06]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[Carreras et al. '08]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[Durrett et al. '15]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[Zhu et al. '13]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[Dyer et al. '16]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[Huang &amp; Harper '08]</td>
<td>91.0</td>
<td>91.4</td>
<td>92.3</td>
<td>92.4</td>
<td>92.8</td>
</tr>
<tr>
<td>[Huang &amp; Harper, Petrov '10]</td>
<td>91.1</td>
<td>91.5</td>
<td>92.4</td>
<td>92.6</td>
<td>92.8</td>
</tr>
<tr>
<td>[Charniak &amp; Johnson '05]</td>
<td>91.1</td>
<td>91.5</td>
<td>92.4</td>
<td>92.6</td>
<td>92.8</td>
</tr>
<tr>
<td>[McClosky et al. '06]</td>
<td>91.3</td>
<td>91.7</td>
<td>92.4</td>
<td>92.6</td>
<td>92.8</td>
</tr>
<tr>
<td>[Huang '08]</td>
<td>91.4</td>
<td>91.6</td>
<td>92.3</td>
<td>92.4</td>
<td>92.8</td>
</tr>
<tr>
<td>[Huang '08]</td>
<td>91.5</td>
<td>91.7</td>
<td>92.4</td>
<td>92.6</td>
<td>92.8</td>
</tr>
<tr>
<td>[McClosky et al. '06]</td>
<td>91.6</td>
<td>91.8</td>
<td>92.5</td>
<td>92.7</td>
<td>92.9</td>
</tr>
<tr>
<td>[Petrov '10]</td>
<td>92.3</td>
<td>92.4</td>
<td>92.7</td>
<td>92.9</td>
<td>93.0</td>
</tr>
<tr>
<td>[Huang &amp; Harper, Petrov '10]</td>
<td>92.4</td>
<td>92.5</td>
<td>92.8</td>
<td>93.0</td>
<td>93.1</td>
</tr>
<tr>
<td>[Sagae &amp; Lavie '06]</td>
<td>92.5</td>
<td>92.6</td>
<td>93.0</td>
<td>93.2</td>
<td>93.3</td>
</tr>
<tr>
<td>[Fossum &amp; Knight '09]</td>
<td>92.6</td>
<td>92.7</td>
<td>93.1</td>
<td>93.3</td>
<td>93.4</td>
</tr>
<tr>
<td>[Zhang et al. '09]</td>
<td>92.7</td>
<td>92.8</td>
<td>93.2</td>
<td>93.4</td>
<td>93.5</td>
</tr>
<tr>
<td>[Vinyals et al. '16]</td>
<td>92.8</td>
<td>92.9</td>
<td>93.3</td>
<td>93.5</td>
<td>93.6</td>
</tr>
</tbody>
</table>
Detailed English Results (New)

<table>
<thead>
<tr>
<th>Single Parser</th>
<th>Multi-Modal / Additional Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Charniak '00]</td>
<td>[Petrov et al. '06]</td>
</tr>
<tr>
<td>[Petrov et al. '06]</td>
<td>[Carreras et al. '08]</td>
</tr>
<tr>
<td>[Vinyals et al. '16]</td>
<td>[Cross &amp; Huang '16]</td>
</tr>
<tr>
<td>[Cross &amp; Huang '16]</td>
<td>[Gaddy et al. '18]</td>
</tr>
<tr>
<td>[Gaddy et al. '18]</td>
<td>[Stern et al. '17]</td>
</tr>
<tr>
<td>[Stern et al. '17]</td>
<td>[Kitaev &amp; Klein '18]</td>
</tr>
<tr>
<td>[Kitaev &amp; Klein '18]</td>
<td>[Petrov '10]</td>
</tr>
<tr>
<td>[Petrov '10]</td>
<td>[Huang et al. '10]</td>
</tr>
<tr>
<td>[Huang et al. '10]</td>
<td>[Durrett &amp; Klein '15]</td>
</tr>
<tr>
<td>[Durrett &amp; Klein '15]</td>
<td>[Vinyals et al. '15]</td>
</tr>
<tr>
<td>[Vinyals et al. '15]</td>
<td>[Dyer et al. '16]</td>
</tr>
<tr>
<td>[Dyer et al. '16]</td>
<td>[Choe &amp; Charniak '16]</td>
</tr>
<tr>
<td>[Choe &amp; Charniak '16]</td>
<td>[Liu &amp; Chang '17]</td>
</tr>
<tr>
<td>[Liu &amp; Chang '17]</td>
<td>[Fried et al. '17]</td>
</tr>
<tr>
<td>[Fried et al. '17]</td>
<td>[Kitaev &amp; Klein '18]</td>
</tr>
</tbody>
</table>

89.7 90.1 91.0 88.3 91.3 92.1 92.6 93.6 91.8 92.4 92.8 93.3 93.8 94.2 94.7 95.1
Multi-Lingual Results

<table>
<thead>
<tr>
<th>Language</th>
<th>Test set F$_1$ all lengths</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arabic</td>
<td>78.8, 81.3, 88.4, 93.0</td>
</tr>
<tr>
<td>Basque</td>
<td>74.7, 81.3, 85.4, 90.7</td>
</tr>
<tr>
<td>French</td>
<td>79.7, 81.0, 88.6, 90.7</td>
</tr>
<tr>
<td>German</td>
<td>78.4, 81.0, 85.4, 90.7</td>
</tr>
<tr>
<td>Hebrew</td>
<td>85.2, 88.3, 88.6, 90.7</td>
</tr>
<tr>
<td>Hungarian</td>
<td>78.6, 80.2, 82.2, 86.8</td>
</tr>
<tr>
<td>Korean</td>
<td>80.2, 82.0, 82.2, 90.7</td>
</tr>
<tr>
<td>Polish</td>
<td>80.6, 83.5, 82.0, 93.0</td>
</tr>
<tr>
<td>Swedish</td>
<td>80.9, 83.2, 83.5, 85.1</td>
</tr>
<tr>
<td>Average</td>
<td>85.1</td>
</tr>
</tbody>
</table>