The SUBTLE NL Parsing Pipeline: 
A Complete Parser for English

Mitch Marcus
University of Pennsylvania
PICTURE OF ANALYSIS PIPELINE

- Tokenize
- Maximum Entropy POS tagger MXPOST Ratnaparkhi
- Core Parser Collins Generative Prob. CFG
- Functional Tag Replacement – Kulick
- Null Element Placement – Ryan Gabbard PhD Dec 2010 (just submitted)
Junior, defuse any bombs immediately when going into a room.

Junior_NNP ,_, defuse_VBP any_DT bombs_NNS immediately_RB when_WRB going_VBG into_IN a_DT room_NN ._.

(S (NP (NNP Junior))
 , ,)

(VP (VB defuse)
 (NP (DT any)
  (NNS bombs))
 (ADVP (RB immediately))
 (SBAR (WHADVP (WRB when))
  (S
   (VP (VBG going)
    (PP (IN into)
     (NP (DT a)
      (NN room)))
   )))))

( . .))
NLP Analysis Pipeline

Typed text in

Tokenize

Maximum Entropy POS Tagger
*(Ratnaparkhi ‘97)*

Parse *(Collins –Bikel ‘04)*
Restore Function Tags
*(Kulick ’06)*

Junior, defuse any bombs immediately when going into a room.

*Junior_NNP ,_, defuse_VBP any_DT bombs_NNS immediately_RB when_WRB going_VBG into_IN a_DT room_NN . _.*

( (S (NP (NNP Junior))
  (, ,)
  (VP (VB defuse)
    (NP (DT any)
      (NNS bombs))
    (ADVP (RB immediately))
    (SBAR (WHADVP (WRB when)))
  (S
    (VP (VBG going)
      (PP (IN into)
        (NP (DT a)
          (NN room)))))
  )))
)

11/4/2010

SUBTLE Annual Review
NLP Analysis Pipeline

Typed text in

Tokenize

Maximum Entropy
POS Tagger
(Ratnaparkhi ‘97)

Parse (Collins –Bikel ‘04)

Restore Function Tags
(Kulick ‘06)

Restore Null
Elements
(Gabbard ‘10)

Junior, defuse any bombs immediately when going into a room.

Junior_NNP ,_, defuse_VBP any_DT bombs_NNS immediately_RB when_WRB going_VBG into_IN a_DT room_NN . _.

( (S (NP-VOC (NNP Junior))
  (, ,)
  (VP (VB defuse)
   (NP (DT any)
    (NNS bombs))
   (ADVP-TMP (RB immediately))
   (SBAR-TMP (WHADVP (WRB when)))
   (S
    (VP (VBG going)
     (PP-DIR (IN into)
      (NP (DT a)
       (NN room))
     )))
   )))

( . .)))
Typed text in

Tokenize

Maximum Entropy POS Tagger
(Ratnaparkhi ‘97)

Parse (Collins –Bikel ‘04)

Restore Function Tags
(Kulick ‘06)

Restore Null Elements
(Gabbard ‘10)

Semantic Analysis
(Perera)

Junior, defuse any bombs immediately when going into a room.

Junior_NNP _, defuse_VBP any_DT bombs_NNS immediately_RB when_WRB going_VBG into_IN a_DT room_NN _.

( (S (NP-VOC (NNP Junior))
  (, ,)
  (NP-SBJ-0 (-NONE- *))
  (VP (VB defuse)
    (NP (DT any)
      (NNS bombs))
    (ADVP-TMP (RB immediately))
    (SBAR-TMP (WHADVP-1 (WRB when)))
    (S (NP-SBJ-0 (-NONE- *))
      (VP (VBG going)
        (PP-DIR (IN into)
          (NP (DT a)
            (NN room)))
        (ADVP-1 (-NONE- */T*)))))
  (. .)))
1995: A breakthrough in parsing

$10^6$ words of Penn Treebank Annotation + Machine Learning = Robust Parsers

(Magerman ’95)

The founder of Pakistan's nuclear program, Abdul Qadeer Khan, has admitted he transferred nuclear technology to Iran, Libya, and North Korea

- 1990 Best hand-built parsers: ~40% accuracy
- 1995+ Statistical parsers: >90% accuracy

(both on short sentences)
Generative Parsing Results (<40 w)

<table>
<thead>
<tr>
<th>Parser</th>
<th>F₁</th>
</tr>
</thead>
<tbody>
<tr>
<td>Magerman 95</td>
<td>84.7</td>
</tr>
<tr>
<td>Collins 96</td>
<td>86.0</td>
</tr>
<tr>
<td>Charniak 97</td>
<td>87.4</td>
</tr>
<tr>
<td>Collins 99 (Bikel 04)</td>
<td>88.6</td>
</tr>
<tr>
<td>Liang 08 (extending Charniak 05)</td>
<td>91.7</td>
</tr>
</tbody>
</table>
Meaningless Progress.....
Junior should defuse any bombs in the room immediately.

(S (NP-SBJ (NNP Junior))
(VP (MD should)
 (VP (VB defuse)
  (NP (NP (NNS bombs))
   (PP-LOC (IN in)
    (NP (DT a)
     (NN room)))))
  (ADVP-TMP (RB immediately)))))

NP-TMP
  NP
    PP-LOC
      in
      NP
        the
        room
Treebank Tree with Null Elements

- \text{believe(SOMEBODY, shoot(SOMEONE, Who))}
A “Perfect” Parseval Analysis..

• $\text{believe}(\text{SOMEBODY}, \text{shoot (SOMEONE, Who)})$

Without null element, can’t tell what the semantic role of Who was, or even the predication it’s part of.
A Complete Parsing System

- Ryan Gabbard, Seth Kulick, and Mitch Marcus. Fully Parsing the Penn Treebank. HLT/NAACL 2006
- Ryan Gabbard, Null Element Restoration, PhD Dissertation, Nov 2010
Step 1: Recovering Function Tags
# Treebank II Function Tag Set

<table>
<thead>
<tr>
<th>Syntactic</th>
<th>Semantic</th>
<th>Miscellaneous</th>
</tr>
</thead>
<tbody>
<tr>
<td>SBJ Subject</td>
<td>TMP Temporal</td>
<td>CLF It-cleft</td>
</tr>
<tr>
<td>LGS Logical</td>
<td>LOC Location</td>
<td>HLN Headline</td>
</tr>
<tr>
<td>Subject</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PRD Predicate</td>
<td>DIR Direction</td>
<td>TTL Title</td>
</tr>
<tr>
<td>PRD Predicate</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DTV Dative</td>
<td>EXT Extent</td>
<td></td>
</tr>
<tr>
<td>DTV Dative</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PUT Locative of</td>
<td>BNF Benefactive</td>
<td>TPC Topic</td>
</tr>
<tr>
<td>“put”</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VOC Vocative</td>
<td>MNR Manner</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOM Nominal</td>
<td></td>
<td>CLR</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADV Non-specific</td>
<td>CLR Closely-</td>
<td></td>
</tr>
<tr>
<td>adverbial</td>
<td>Related</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
A Sample PCFG

1. s $\Rightarrow^5$ np vp
2. s $\Rightarrow^{25}$ s pp
3. s $\Rightarrow^{15}$ v np
4. s $\Rightarrow^{10}$ v pp
5. vp $\Rightarrow^2$ v np
6. vp $\Rightarrow^4$ v pp
7. vp $\Rightarrow^4$ v
8. np $\Rightarrow^1$ n n
9. np $\Rightarrow^4$ det n
10. np $\Rightarrow^3$ np pp
11. np $\Rightarrow^2$ n
12. n $\Rightarrow^4$ arrow
13. n $\Rightarrow^2$ flies
14. n $\Rightarrow^4$ time
15. v $\Rightarrow^4$ flies
16. v $\Rightarrow^4$ like
17. v $\Rightarrow^2$ time
18. p $\Rightarrow^{1.0}$ like
19. det $\Rightarrow^{1.0}$ a
20. pp $\Rightarrow^{1.0}$ p np
An Example Derivation

This example adapted from Jelinek et. al. 1991.
The Key Trick in ‘95: Lexicalizing the CFG

New problem: Estimating the probability of
S(bought) → NP(yesterday) NP(Marks) VP(bought)
Solving the Sparse Data Problem with Independence Assumptions

Instead of

\[ S(\text{bought}) \rightarrow \text{NP}(\text{yesterday}) \text{ NP}(\text{Marks}) \text{ VP}(\text{bought}) \]

1. Generate Head, given lexicalized parent:
   \[ S(\text{bought}) \rightarrow \ldots \text{VP}(\text{bought}) \ldots \]

2. Generate other children given parent & head:
   \[ S(\text{bought}) \rightarrow \text{NP}( ) \text{ NP}( ) \text{ VP}(\text{bought}) \]

3. Lexicalize other children, given parent, head & node type:
   \[ S(\text{bought}) \rightarrow \text{NP}(\text{yesterday})\text{NP}(\text{Marks}) \text{ VP}(\text{bought}) \]
To augment the parser output with function tags

1. Find the code in Collins Model II that removes function tags
2. Remove it!

(From Collins & Koo '05)
## Results for Function Tag Groups & Overall

<table>
<thead>
<tr>
<th></th>
<th>Overall w/ CLR</th>
<th>Overall w/o CLR</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Collins 2 +FTags</strong></td>
<td>88.95</td>
<td>90.78</td>
</tr>
<tr>
<td>Blaheta, 2003</td>
<td></td>
<td>88.28</td>
</tr>
<tr>
<td>Jijkoun &amp; de Rijke, 2004</td>
<td>88.50</td>
<td></td>
</tr>
</tbody>
</table>

- With no drop in basic parser F score!
# Results for Function Tag Groups & Overall

<table>
<thead>
<tr>
<th></th>
<th>Overall w/ CLR</th>
<th>Overall w/o CLR</th>
<th>Syn 56%</th>
<th>Sem 36%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collins 2 +FTags</td>
<td>88.95</td>
<td>90.78</td>
<td>95.76</td>
<td>84.56</td>
</tr>
<tr>
<td>Blaheta, 2003</td>
<td></td>
<td>88.28</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jijkoun &amp; de Rijke, 2004</td>
<td>88.50</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Musillo &amp; Merlo, Oct 2005</td>
<td></td>
<td></td>
<td>96.5</td>
<td>85.6</td>
</tr>
</tbody>
</table>

• With no drop in basic parser F score!
Step 2: Recovering Null Elements
Null Elements for “Wh-movement”

- *T* - marks WH-questions
- Null elements are co-indexed using integer tags.
- To recover Pred-arg structure:
  - Replace null element with co-indexed lexical material

**Predicate argument structure:**
see(your, what)

**Similar approaches for other grammatical phenomenon**
Previous Work Forward

- Integrated with Parsing (Collins Model 3, Dienes&Dubey 03)
- Postprocessing Parser Output
  - Hand Built Rule-based (Campbell 04)
  - Statistical Machine Learning Approaches
    (Johnson 02, Levy&Manning 04, Jijkoun&de Rijke 04)

- Our approach combines:
  - Campbell’s Problem Decomposition
  - Machine Learning

- Into:
  - A pipeline of linear classifiers
The Null Element Replacement Algorithm

- Pipeline & rich linguistic features à la Campbell
- Pipeline of linear classifiers (McCallum’s Mallet)
Aggregate results over all recovered categories

<table>
<thead>
<tr>
<th>System</th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>D&amp;D</td>
<td>78.50</td>
<td>68.08</td>
<td>72.92</td>
</tr>
<tr>
<td>Pres</td>
<td>74.70</td>
<td>74.62</td>
<td>74.66</td>
</tr>
</tbody>
</table>

- Automatically parsed trees from section 23
Present vs. Campbell’s Rule Based System

- Test on gold-standard trees from section 23 using Campbell's metric

<table>
<thead>
<tr>
<th>Category</th>
<th>Present</th>
<th>Campbell</th>
</tr>
</thead>
<tbody>
<tr>
<td>NP *</td>
<td>88.8</td>
<td>86.9</td>
</tr>
<tr>
<td>NP <em>T</em></td>
<td>96.3</td>
<td>96.0</td>
</tr>
<tr>
<td>ADVP <em>T</em></td>
<td>82.2</td>
<td>79.9</td>
</tr>
<tr>
<td>0</td>
<td>99.8</td>
<td>98.5</td>
</tr>
</tbody>
</table>
Wh – placement with factor graphs

That is the book I tried to sell

Insert 0s

That is the book \((WH\ 0)\) I tried to sell

Locate open slots

\((WH\ 0)\) I tried tried/ADV sell/ADV to sell tried/SBJ sell/OBJ sell/ADV
Wh – placement with factor graphs
## Results: two systems for Wh-placement

<table>
<thead>
<tr>
<th>Type</th>
<th>Gold Old</th>
<th>Gold New</th>
<th>Parsed Old</th>
<th>Parsed New</th>
</tr>
</thead>
<tbody>
<tr>
<td>NP <em>T</em></td>
<td>92.8</td>
<td>96.5</td>
<td>85.9</td>
<td>87.9</td>
</tr>
<tr>
<td>ADVP <em>T</em></td>
<td>79.2</td>
<td>86.7</td>
<td>77.7</td>
<td>79.8</td>
</tr>
<tr>
<td>NP *</td>
<td>78.6</td>
<td>82.7</td>
<td>72.2</td>
<td>71.6</td>
</tr>
<tr>
<td>NP * (na)</td>
<td>95.8</td>
<td>96.6</td>
<td>88.3</td>
<td>88.0</td>
</tr>
<tr>
<td>WHNP 0</td>
<td>92.0</td>
<td>90.2</td>
<td>61.5</td>
<td>59.6</td>
</tr>
<tr>
<td>WHADVP 0</td>
<td>71.0</td>
<td>66.7</td>
<td>68.9</td>
<td>61.8</td>
</tr>
</tbody>
</table>
Summary: Full Parsing!

Before…

S
   NP
     patients
   VP
     were
   VBN
     examined

After…

S
   NP-SBJ-1
   VP
     patients
     were
     VBN
     NP-1
     examined

• 20 Function Tags – Mostly syntactic or semantic
  • Syntactic Tags: 95% recovery accuracy
  • Semantic Tags: 84% recovery accuracy
• Traces
  • Passivization, control, raising: ~70% recovered
  • Wh-traces: ~80% recovered