Dependency-based Evaluation of Syntactic Parsers

Yusuke Miyao
University of Tokyo

Joint work with Kenji Sagae, Takuya Matsuzaki, and Jun’ichi Tsujii
Background (1/2)

• Diversity of frameworks for syntactic parsing
  – Phrase structure parsers
    • Reranking parser [Charniak & Johnson 2005]
    • Lexicalized PCFG parser [Charniak 2000; Collins 1997]
    • State-splitting parser [Petrov & Klein 2007]
  – Dependency parsers
    • MST parser [McDonald & Pereira 2006]
    • Shift-reduce parser [Nivre 2005; Sagae & Tsujii 2007]
  – Deep linguistic parsers
    • LFG parser [Cahill et al. 2002; Kaplan et al. 2004]
    • CCG parser [Clark & Curran 2004]
    • HPSG parser [Oepen et al. 2004; Malouf & van Noord 2004; Miyao et al. 2004]
Background (2/2)

- Different evaluation criteria for different frameworks
  - Bracketing accuracy for phrase structure parsers
  - Attachment accuracy for dependency parsers
  - LFG parsers were evaluated with PARC 700 DepBank [King et al. 2003]
  - CCG parsers were evaluated with CCGBank [Hockenmaier et al. 2002]
  - HPSG parsers were evaluated with HPSG treebanks [Miyao et al. 2004; Oepen et al. 2004]

- Performances of these parsers differ?
- Framework-independent evaluation is possible?
Topic of this talk

• Comparative evaluation of Penn Treebank (PTB)-based phrase structure parsers and an HPSG parser
  – C&J: reranking parser [Charniak & Johnson 2005]
  – Charniak: lexicalized PCFG parser [Charniak 2000]
  – Enju: HPSG parser [Miyao et al. 2004]

• Approach: converting parser output into dependency-based syntactic representations
Parser evaluation schemes (1/2)

• Accuracy of phrase structures (brackets)
  – Most popular in the evaluation of PTB parsers
    [Collins 1997; Charniak 2000; Charniak & Johnson 2005]

• Problems
  – Only surface structures are evaluated
  – Different grammar theories suppose different phrase structures
Parser evaluation schemes (2/2)

• Accuracy of labeled dependency relations
  – Closer to application needs
  – Deeper structures can be evaluated

• Standard in CoNLL shared tasks

• Popular in the evaluation of deep parsers, while using different gold-standard data [Kaplan et al. 2004; Burke et al. 2004; Clark & Curran 2004; Miyao et al. 2004]
Dependency-based evaluation

• We focus on two schemes of labeled dependency relations
  – Grammatical Relations (GR) [Briscoe & Carroll 2006]
  – Stanford Dependency (SD) [de Marneffe et al. 2006]

• Labeled dependencies among words
Grammatical Relations (GR)

• Originally developed for better parser evaluation than bracketing accuracy [Carroll et al. 1998]
• PARC 700 DepBank [King et al. 2003] (subset of PTB Section 23) was re-annotated
• Used for evaluation of shallow/deep parsers
  – RASP [Briscoe & Carroll 2006]
  – PTB parsers [Preiss 2003]
  – CCG parser [Clark & Curran 2007]

Example: *He runs fast*  
(ncsubj runs He _)  
(ncmod _ runs fast)
GR: example

• GR represents long-distance dependencies
• They market cable-TV on the very grazing opportunities CNN seeks to discourage.

... 
(cmod _ opportunities seeks)
(ncsubj seeks CNN _)
(ncsubj discourage CNN _)
(dobj discourage opportunities)
(xcomp to seeks discourage)
GR: hierarchy

- Relation types are organized in a hierarchy
- Non-leaf nodes allow for soft-matching of relation types
GR: evaluation metrics

• Microaverage precision/recall/f-score
  – Aggregate all relations including non-leaf types
  – Frequent relation types mainly affect the scores

• Macroaverage precision/recall/f-score
  – Average of accuracies for all relation types
  – Infrequent relation types can affect the scores
Stanford Dependency (SD)

• Originally designed for extracting useful relations from PTB-style phrase structures [de Marneffe et al. 2006]
• Provided as a program for converting PTB-style trees into labeled dependency relations
  – Attached to the Stanford parser [Klein & Manning 2003]
• Recently used for evaluating PTB parsers on biomedical text [Clegg et al. 2007; Pyysalo et al. 2007]

Example: *He runs fast*  
ssubj(runs-2 He-1)  
amod(runs-2 fast-3)
SD: example

• SD represents some long-distance dependencies, but not perfectly

• *They market cable-TV on the very grazing opportunities CNN seeks to discourage.*

...  
nsubj(seeks-10, CNN-9)  
rcmod(opportunities-8, seeks-10)  
aux(discourage-12, to-11)  
xcomp(seeks-10, discourage-12)
Example: GR and SD

Regulators also ordered CenTrust to stop buying back the preferred stock.

(ncsubj ordered Regulators _)  nsubj(ordered-3, Regulators-1)
(ncsubj stop CenTrust _)       advmod(ordered-3, also-2)
(ncsubj buying CenTrust _)     dobj(ordered-3, CenTrust-4)
(ncmod _ ordered also)         aux(stop-6, to-5)
(xcomp to ordered stop)        xcomp(ordered-3, stop-6)
(xcomp _ stop buying)          partmod(stop-6, buying-7)
(dobj buying stock)            prt(buying-7, back-8)
(det stock the)                det(stock-11, the-9)
(passive preferred)            amod(stock-11, preferred-10)
(ncsubj preferred stock obj)   dobj(buying-7, stock-11)
(ncmod _ stock preferred)      (ncmod prt buying back)
(ncmod prt buying back)        (dobj ordered CenTrust)
Previous works on GR/SD evaluation

• RASP
  – Evaluated on GR [Briscoe & Carroll 2006]

• PTB parsers
  – Evaluated on GR (older version, Susanne) [Preiss 2003]
  – Evaluated on SD (GENIA and BioInfer) [Clegg et al. 2007; Pyysalo et al. 2007]

• CCG parser
  – Evaluated on GR [Clark & Curran 2007]

• LFG parsers
  – Evaluated on Parc 700 DepBank [Kaplan et al. 2004; Burke et al. 2004]
Our approach

• Format conversion: convert parser output into GR/SD, without changing the original parsers
  – SD-to-GR
  – HPSG-to-GR, HPSG-to-SD, HPSG-to-PTB
Our approach

• Use gold standard data (PTB and the HPSG treebank) when developing conversion programs
  – Accuracy obtained by converting a gold treebank indicates the quality of format conversion

• Conversion from SD to GR illuminates the difficulty in the translation between the two
HPSG-to-GR/SD conversion

• Basic strategy: mapping predicate-argument relations into GR/SD
• Before/after mapping, fix systematic disagreements of annotation policies
  – Add/remove dependency relations
  – Change lexical heads
  – Reduce coordinated relations
  – Construction-specific conversion
Regulators also ordered CenTrust to stop buying back the preferred stock.

(ncsubj ordered Regulators _)
(dobj ordered CenTrust)
(xcomp to ordered stop)
(ncsubj stop CenTrust _)
(xcomp _ stop buying)
Add/remove dependency relations

• Add:
  – Relative-antecedent relations
  – Text adjuncts

• Remove:
  – Internal structure of named entities
  – Punctuations
  – Subjects of auxiliary verbs
Change lexical heads

• Some constructions needed changes of lexical heads

John came here ten years ago

• Heads of noun phrases (still remaining unfixed)
Reduce coordinated relations

- Shared arguments in coordinated phrases are reduced to one relation

Mary likes apples but hates oranges

Mary likes apples but hates oranges
Construction-specific conversion

• Quotation
• Copula
• Small clause
• Number expressions
• Relative clauses
• ...
• ~ 1,300 line Perl code
Experiments

• GR
  – 560 sentences from GR version of PARC 700 DepBank
  – Measure: microaverage, macroaverage

• SD
  – The same set of sentences as GR
  – Measure: accuracy

• Evaluated parsers
  – C&J [Charniak & Johnson 2005]
  – Charniak [Charniak 2000]
  – Enju [Miyao et al. 2004]

• We also show results for other parsers when available
  – RASP [Briscoe & Carroll 2006], C&C [Clark & Curran 2007], Stanford parser
## Conversion accuracy

<table>
<thead>
<tr>
<th>Conversion</th>
<th>Precision</th>
<th>Recall</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>SD→GR</td>
<td>80.84</td>
<td>69.16</td>
<td>74.54</td>
</tr>
<tr>
<td>HPSG→GR</td>
<td>87.49</td>
<td>86.79</td>
<td>87.14</td>
</tr>
<tr>
<td>CCG→GR [Clark &amp; Curran 2007]</td>
<td>86.86</td>
<td>82.75</td>
<td>84.76</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Conversion</th>
<th>Precision</th>
<th>Recall</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>HPSG→SD</td>
<td>83.43</td>
<td>81.44</td>
<td>82.42</td>
</tr>
<tr>
<td>PTB→SD (Stanford conversion)</td>
<td>100.00</td>
<td>100.00</td>
<td>100.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Conversion</th>
<th>Precision</th>
<th>Recall</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>HPSG→PTB</td>
<td>98.12</td>
<td>98.07</td>
<td>98.09</td>
</tr>
</tbody>
</table>
GR evaluation

- Microaverage

<table>
<thead>
<tr>
<th>System</th>
<th>Precision</th>
<th>Recall</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>C&amp;J (PTB→SD→GR)</td>
<td>79.08</td>
<td>67.46</td>
<td>72.81</td>
</tr>
<tr>
<td>Charniak (PTB→SD→GR)</td>
<td>78.41</td>
<td>67.68</td>
<td>72.65</td>
</tr>
<tr>
<td>Enju (HPSG→GR)</td>
<td>83.57</td>
<td>81.73</td>
<td>82.64</td>
</tr>
<tr>
<td>RASP [Briscoe &amp; Carroll 2006]</td>
<td>77.66</td>
<td>74.98</td>
<td>76.29</td>
</tr>
<tr>
<td>C&amp;C [Clark &amp; Curran 2007]</td>
<td>82.44</td>
<td>81.28</td>
<td>81.86</td>
</tr>
</tbody>
</table>

- Macroaverage

<table>
<thead>
<tr>
<th>System</th>
<th>Precision</th>
<th>Recall</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>C&amp;J (PTB→SD→GR)</td>
<td>60.20</td>
<td>47.97</td>
<td>53.39</td>
</tr>
<tr>
<td>Charniak (PTB→SD→GR)</td>
<td>59.39</td>
<td>48.08</td>
<td>53.14</td>
</tr>
<tr>
<td>Enju (HPSG→GR)</td>
<td>77.87</td>
<td>71.10</td>
<td>74.33</td>
</tr>
<tr>
<td>RASP [Briscoe &amp; Carroll 2006]</td>
<td>62.12</td>
<td>63.77</td>
<td>62.94</td>
</tr>
<tr>
<td>C&amp;C [Clark &amp; Curran 2007]</td>
<td>65.61</td>
<td>63.28</td>
<td>64.43</td>
</tr>
</tbody>
</table>
SD evaluation

• Accuracy

<table>
<thead>
<tr>
<th></th>
<th>Precision</th>
<th>Recall</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>C&amp;J (PTB→SD)</td>
<td>88.36</td>
<td>88.45</td>
<td>88.40</td>
</tr>
<tr>
<td>Charniak (PTB→SD)</td>
<td>87.05</td>
<td>87.10</td>
<td>87.07</td>
</tr>
<tr>
<td>Enju (HPSG→SD)</td>
<td>77.38</td>
<td>74.54</td>
<td>75.93</td>
</tr>
<tr>
<td>Enju (HPSG→PTB→SD)</td>
<td>87.13</td>
<td>87.16</td>
<td>87.14</td>
</tr>
<tr>
<td>Stanford parser</td>
<td>85.36</td>
<td>83.16</td>
<td>84.25</td>
</tr>
</tbody>
</table>
Comparison of GR gold standard and the converted HPSG treebank

<table>
<thead>
<tr>
<th>Category</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Remaining disagreements</td>
<td>107</td>
</tr>
<tr>
<td>text adjunct</td>
<td>35</td>
</tr>
<tr>
<td>argument/modifier distinction</td>
<td>34</td>
</tr>
<tr>
<td>lexical heads</td>
<td>25</td>
</tr>
<tr>
<td>POS</td>
<td>7</td>
</tr>
<tr>
<td>attachment ambiguity</td>
<td>6</td>
</tr>
<tr>
<td>Conversion errors</td>
<td>36</td>
</tr>
<tr>
<td>named entity</td>
<td>15</td>
</tr>
<tr>
<td>number expression</td>
<td>6</td>
</tr>
<tr>
<td>coordination</td>
<td>6</td>
</tr>
<tr>
<td>others</td>
<td>9</td>
</tr>
<tr>
<td>Errors of the HPSG treebank</td>
<td>14</td>
</tr>
<tr>
<td>noun phrase structure</td>
<td>10</td>
</tr>
<tr>
<td>others</td>
<td>4</td>
</tr>
<tr>
<td>Errors of GR data</td>
<td>13</td>
</tr>
</tbody>
</table>
Discussion

• Conversion quality >> parser performance
• Format conversion is non-trivial
  – Even conversion from SD to GR is difficult
  – Because of differences of lexical heads, POSs, etc.
• How to define annotation policies in a framework-independent way?
  – Without any framework, annotation decisions can be arbitrary
  – Un dokumented decisions complicate the development of format conversion (arg/mod distinction, lexical heads, POS, etc.)
  – There are a lot of linguistically uninteresting constructions in real text
Conclusion

• Framework-independent dependency-based evaluation of syntactic parsers was described
  – Grammatical Relations
  – Stanford Dependency
• The deep parsers obtained higher scores in GR, while the shallow and deep parsers got similar scores in SD
• Accuracy figures are largely affected by the quality of format conversion
• Sources of difficulties in format conversion are presented
• Framework-independent evaluation is possible, but format conversion is much more critical
Future work

• Evaluation on other corpora
  – Susanne (GR data is available, although the annotation policy is slightly different)
  – Biomedical domain

• Better schemes for parser evaluation
  – For easier format conversion
    • Ambiguous/soft matching
    • Multiple reference annotations
  – For deeper representation

• Evaluation of dependency parsers

• Evaluating the impact of differences of parsing accuracy on applications