SUMMARY

- In the first surface realization (SR) shared task (Bela et al., 2011), no grammar-based systems achieved competitive results, as input conversion turned out to be more difficult than anticipated.
- However, a silver lining of the failure of grammar-based systems in the shared task is that it revealed several problems with the data (right).
- Since then, Narayan & Gardent (2012) have shown that grammar-based systems can be substantially improved with error mining techniques.
- In this paper, inspired by recent work on converting dependency trees (Ambati et al., 2013) and semantic parsing (Kwiatkowski et al., 2010; Artzi & Zettlemoyer, 2013) with Combinatory Categorial Grammar (CCG), we pursue the alternative strategy of inducing a CCG from an enhanced version of the shared task dependencies, with initial experiments showing even better results.

Algorithm

- In contrast to Ambati et al.’s (2013) approach, the algorithm integrates the proposal of candidate lexical categories with the derivational process, making it possible to derive categories involving unattested arguments, such as s, np, dcl, etc.
- It also makes greater use of unary type-changing rules, as with Artzi & Zettlemoyer’s (2013) approach, unlike their approach, though, it works in a broad coverage setting, and makes use of all the combinators standardly used with CCG, including ones for type-raising.
- The algorithm (right) makes use of a seed lexicon that specifies category projection by part of speech, as in (1), as well as a handful of categories for function words, as in (2).
- After lexical instantiation, the algorithm recursively visits each node in the primary dependency tree bottom up (combineEdges), using a local chart (doCombos) at each step to combine categories for adjacent phrases in all possible ways, along the way creating new categories (extendCats and coextendCats) and unary rules (applyNewUnary), as in (3)-(5)

Experiments and Future Work

- We ran the induction algorithm over the standard PTB training sections (02-21), selecting preferred derivations with a simple complexity metric counting steps and shakes, then training a realization ranking model as in previous work (White & Rajkumar, 2012).
- Focusing only on the 90.5% of the sentences for which a complete derivation was found (complete) yields a BLEU score of 0.8616 on the devset.
- Section 23 results of 0.8260 were much higher than Narayan & Gardent’s (2012) score of 0.675 with 38.8% coverage, though still trailing the shared task scores of the top statistical dependency parsers.
- In future work, we hope to close the gap with the top systems by integrating an improved ranking model into the induction process and resolving the remaining representational issues with problematic constructions.

Free Object Relative Example

Economists are divided as to [how much manufacturing strength] they expect to see in tomorrow’s report on industrial production and capacity utilization, also due tomorrow (np1/2,000,6, “deep” representation)

- Here, only one of the two roles played by how much manufacturing strength is captured in the deep representation, making it difficult to linearize this phrase correctly.
- By contrast, below is an experimental version of the shallow representation intended to capture all the syntactic dependencies in the PTB (Kudo to Richard Johansson!), doing so makes it feasible to define a complete CCG.

Augmented Syntactic Dependencies with CCG Derivation

- Categories are divided as to how much manufacturing strength they expect to see in...
- September reports on industrial production and capacity utilization, also due tomorrow...

First SR Shared Task Input


Definitions: edges is a map from dependency graph nodes to their edges, where an edge in a CCG is a pair together with a coverage binary; agenda is a priority queue of edges sorted by the scoring metric; chart manages equivalence classes of edges; see text for descriptions of auxiliary functions such as extendCat and coextendCat below.

Algorithm

- breadthFirst, leaves, unaryRules = 0. For each item in training set:
  1. edge[head] = nonCordinate(leaf), edge[head] = in-sidial(leaf), for node in input graph
  2. combineEdges(), with over of input graph
  3. bestEdge = unpackEdge[leaf], bestEdges ++ bestEdge[head], leaves ++ abstractCat(bestEdge), unaryRules ++ abstractRules[bestEdge], if bestEdge complete
- doCombos(leaf, node)
  1. combinedEdges[leaf, child] for child in node kids
  2. edge[node] = in-sidial(leaf) if node has coord relations, otherwise edge[leaf] = extendCate(extendCat, only) for argument rule
  3. agenda + edge[node], agenda += edge[head] for child in node kids, chart += 0
  4. While agenda not empty:
     (a) next = agenda pop
     (b) chart += next
     (c) doCombos(chart) unless next packed into an existing chart item
  5. edge[node] = chart edges for node filtered for maximal input coverage
- doCombos(chart)
  1. agenda += applyUnary(next), if next is for node
  2. For item in chart:
     (a) agenda += applyUnary(next, item), if next is adjacent to item
     (b) agenda += applyNewUnary(next, item), if next connected to item by a modifier relation.

Outputs: bestEdges, leaves, unaryRules

Table: Information Loss in First SR Shared Task Inputs

<table>
<thead>
<tr>
<th>Model</th>
<th>Exact</th>
<th>Complete</th>
<th>BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>GEN</td>
<td>74.6</td>
<td>79.5</td>
<td>0.4713</td>
</tr>
<tr>
<td>GLOBAL</td>
<td>72.7</td>
<td>79.0</td>
<td>0.4779</td>
</tr>
<tr>
<td>NOSUBJ</td>
<td>71.3</td>
<td>79.2</td>
<td>0.4698</td>
</tr>
<tr>
<td>NOSUBJ-DEP</td>
<td>71.2</td>
<td>79.2</td>
<td>0.4666</td>
</tr>
<tr>
<td>ALL</td>
<td>71.8</td>
<td>79.4</td>
<td>0.4903</td>
</tr>
<tr>
<td>ALL+</td>
<td>76.2</td>
<td>80.5</td>
<td>0.9210</td>
</tr>
<tr>
<td>COMPLETE</td>
<td>48.6</td>
<td>86.8</td>
<td>0.9006</td>
</tr>
<tr>
<td>GEN-DEP</td>
<td>71.0</td>
<td>79.3</td>
<td>0.4698</td>
</tr>
<tr>
<td>GEN-DEP-DEP</td>
<td>70.0</td>
<td>79.2</td>
<td>0.4642</td>
</tr>
</tbody>
</table>

Table: Augmented Syntactic Dependencies

<table>
<thead>
<tr>
<th>Combos</th>
<th>Input</th>
<th>Error</th>
<th>BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>GEN</td>
<td>2.4</td>
<td>79.5</td>
<td>0.4713</td>
</tr>
<tr>
<td>GLOBAL</td>
<td>23.7</td>
<td>79.0</td>
<td>0.4779</td>
</tr>
<tr>
<td>NOSUBJ</td>
<td>31.3</td>
<td>79.2</td>
<td>0.4698</td>
</tr>
<tr>
<td>NOSUBJ-DEP</td>
<td>31.2</td>
<td>79.2</td>
<td>0.4666</td>
</tr>
<tr>
<td>ALL</td>
<td>36.6</td>
<td>80.5</td>
<td>0.9210</td>
</tr>
<tr>
<td>ALL+</td>
<td>48.6</td>
<td>86.8</td>
<td>0.9006</td>
</tr>
<tr>
<td>COMPLETE</td>
<td>3.8</td>
<td>80.4</td>
<td>0.8088</td>
</tr>
</tbody>
</table>

Acknowledgements

Thanks to the anonymous reviewers, Richard Johansen and the University of Sydney Schre Lab for helpful comments and discussion. This work was supported in part by NSF grants IIS-1145655 and IIS-1320318.