Extracting the Unextractable: A Case Study on Verb-Particles

Timothy Baldwin
Objectives

• to develop a method to extract verb particle constructions (VPCs) from unannotated corpora of arbitrary size

• to use both linguistic constraints and statistical tendencies in the extraction process

• (ultimately) to determine subcategorisation information at the same time as extracting VPCs
Definitions

- **Intransitive verb-particle construction**: (head) verb, particle(s) and no complement (e.g. *fall back, lie down*).

- **Transitive verb-particle construction**: (head) verb, particle(s) and an NP complement in either the split (e.g. *hand the paper in*) or joined configuration (e.g. *hand in the paper*).
Extraction Method-1: POS Tag-based

- Dedicated “particle” (=intransitive preposition) POS tag in Penn Treebank POS tagset

- Possible to extract VPCs by locating each particle and searching for the governing verb:

  Filling\textit{VBG} out\textit{RP} detailed\textit{VBN} forms\textit{NNS} about\textit{IN} these\textit{DT} individuals\textit{NNS} ...

- Tag with Brill tagger
Results for Extraction Method-1

<table>
<thead>
<tr>
<th>Tagger</th>
<th>Precision</th>
<th>Recall</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Penn</td>
<td>0.872</td>
<td>0.781</td>
<td>0.824</td>
</tr>
<tr>
<td>Brill</td>
<td>0.889</td>
<td>0.172</td>
<td>0.288</td>
</tr>
</tbody>
</table>

- Results of VPC extraction over WSJ, without subcat information
- Recall calculated over set of 200 VPCs randomly-selected from the Alvey Tools VPC data, of which 64 were attested in the corpus (@ mean token frequency of 3.5)
Reflections on Method-1

- Good results for original Penn Treebank tagging, less convincing for Brill tagger
- Particle tag precision/recall for Brill tagger: 0.838/0.103
- High precision, low recall
Extraction Method-2: Chunk-based

- Dedicated “particle” (=intransitive preposition) chunk tag in CoNLL-2000 chunk tagset
- Possible to extract VPCs by locating each particle and searching for the governing verb:

  \[ [\text{VP} \text{ Filling}] [\text{PRT} \text{ out}] [\text{NP} \text{ detailed forms}] [\text{PP} \text{ about}] [\text{NP} \text{ these individuals}] \]

- Chunk with TiMBL, using the Brown corpus as training data (chunk-level F-score of 0.903)
Results/reflections for Extraction Method-2

<table>
<thead>
<tr>
<th>Chunker</th>
<th>Precision</th>
<th>Recall</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Penn</td>
<td>0.880</td>
<td>0.812</td>
<td>0.845</td>
</tr>
<tr>
<td>TiMBL</td>
<td>0.814</td>
<td>0.672</td>
<td>0.736</td>
</tr>
</tbody>
</table>

- Again, good results for original Penn Treebank chunking, marginally worse for TiMBL-based chunking
- Particle chunk F-score: 0.734
- Recall better but could still improve — cause: particle chunks not identified as such
Extraction Method-3: Chunk Grammar-based

- Use simple chunk grammar to identify exemplars which are compatible/incompatible with a VPC analysis
- Search over singleton prepositional chunks, as well as adverbial chunks where the head is contained in a canonical set of particle types
- Disallow VPCs for which negative evidence is found, allow all VPCs for which positive evidence is found
• Example VPC-compatible chunk sequences:
  VP NP PRT ,
  VP PRT SBAR[if]

• Example VPC-incompatible chunk sequences:
  VP ADVP PRT
  VP PRT NP[PRP]

• Ambiguous chunk sequences:
  VP PRT NP
  VP NP PRT NP
## Results/reflections for Extraction Method-3

<table>
<thead>
<tr>
<th>Chunker</th>
<th>Precision</th>
<th>Recall</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>TiMBL&lt;sub&gt;SEQ₁&lt;/sub&gt;</td>
<td>0.767</td>
<td>0.609</td>
<td>0.679</td>
</tr>
<tr>
<td>TiMBL&lt;sub&gt;SEQ₂&lt;/sub&gt;</td>
<td>0.233</td>
<td>0.828</td>
<td>0.364</td>
</tr>
</tbody>
</table>

- Reasonable recall, precision for TiMBL<sub>SEQ₁</sub>
- Good recall, bad precision for TiMBL<sub>SEQ₂</sub>
Extraction Method-4: System Combination

- Different methods proposed, each with particular strengths and weaknesses
- Combine proposed method into integrated method using second-tier classifier (TiMBL)
- Extra feature: single-word nominalised/adjectival form of VPC in corpus (e.g. changeover, dried-up)
- Training data: annotated VPC data from Brown corpus
Results/ reflections on Method-4

<table>
<thead>
<tr>
<th>System</th>
<th>Precision</th>
<th>Recall</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic</td>
<td>0.744</td>
<td>0.766</td>
<td>0.755</td>
</tr>
<tr>
<td>+- verb</td>
<td>0.745</td>
<td>0.719</td>
<td>0.732</td>
</tr>
<tr>
<td>+- particle</td>
<td>0.719</td>
<td>0.875</td>
<td>0.790</td>
</tr>
</tbody>
</table>

- Best results when particle added in as feature
- Higher F-score than any of the component methods (but lower than the methods based on the Penn Treebank annotation)
Getting Subcategorisation Information

• While the results to date are promising, ideally we would like to be able to extract subcat information (= lexical type) at same time as getting VPC data

• Same basic method, but partition extraction process into intransitive and transitive VPCs, tailor the individual extraction methods to be able to differentiate between them

• Extra feature: does head verb have intransitive/transitive usage?
## Preliminary Results

<table>
<thead>
<tr>
<th>VPC type</th>
<th>Precision</th>
<th>Recall</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intrans</td>
<td>0.697</td>
<td>0.500</td>
<td>0.582</td>
</tr>
<tr>
<td>Trans</td>
<td>0.820</td>
<td>0.578</td>
<td>0.678</td>
</tr>
</tbody>
</table>