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:

  $\text{Filling}_{\text{VBG}}$ out_{RP} $\text{detailed}_{\text{VBN}}$ $\text{forms}_{\text{NNS}}$ about_{IN} these_{DT} $\text{individuals}_{\text{NNS}}$ ...

- Tag with Brill tagger

Results for Extraction Method-1

Tagger	*Precision*	*Recall*	*F-score*
Penn	0.872	0.781	0.824
Brill	0.889	0.172	0.288

- 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:

  [VP Filling] [PRT out] [NP detailed forms] [PP about] [NP 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

Chunker	*Precision*	*Recall*	*F-score*
Penn	0.880	0.812	0.845
TiMBL	0.814	0.672	0.736

- 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

Chunker	*Precision*	*Recall*	*F-score*
TiMBL$_{SEQ_1}$	0.767	0.609	0.679
TiMBL$_{SEQ_2}$	0.233	0.828	0.364

- Reasonable recall, precision for TiMBL$_{SEQ_1}$
- Good recall, bad precision for TiMBL$_{SEQ_2}$

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

System	*Precision*	*Recall*	*F-score*
Basic	0.744	0.766	0.755
+verb	0.745	0.719	0.732
+particle	0.719	0.875	0.790

- 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

VPC type	*Precision*	*Recall*	*F-score*
Intrans	0.697	0.500	0.582
Trans	0.820	0.578	0.678