Learning Verb Argument Structure from Minimally Annotated Corpora

Anoop Sarkar and Wootiporn Tripasai
{anoop,tripasai}@linc.cis.upenn.edu
Department of Computer and Information Science
University of Pennsylvania
Classification of Verb Alternations

- Task: automatic classification of verbs based on their underlying thematic structure

- Verbs that take the same number and category of arguments but assign different thematic roles to them

- Payoff:
  - acquisition of lexical semantic knowledge;
  - improve disambiguation information for lexicalized probabilistic parsers
Classification of Verb Alternations: Application of SF Learning

**Unergative**

**INTRAN:** The horse raced past the barn. \((NP_{agent} \text{ raced})\)

**TRAN:** The jockey raced the horse past the barn. \((NP_{causer} \text{ raced } NP_{agent})\)

**Unaccusative**

**INTRAN:** The butter melted in the pan. \((NP_{theme} \text{ melted})\)

**TRAN:** The cook melted the butter in the pan. \((NP_{causer} \text{ melted } NP_{theme})\)

**Object-Drop**

**INTRAN:** The boy washed. \((NP_{agent} \text{ washed})\)

**TRAN:** The boy washed the hall. \((NP_{agent} \text{ washed } NP_{theme})\)

(Stevenson and Merlo 1997)
The Hypothesis (Merlo and Stevenson 2001)

- All verbs in each class can occur with the same syntactic context as other verbs

- Statistical distributions of syntactic context can be distinguished for each verb

- Identify probabilistic features that pick out verb co-occurrences with particular syntactic contexts and use for classification

- **This work**: application of SF learning to this kind of classifier to see if noisy data with less annotation can be used
Corpus tagged by Adwait Ratnaparkhi’s tagger and then chunked using Steve Abney’s chunker:

Pierre NNP nx 2
Vinken NNP
, ,
61 CD ax 3
years NNS
old JJ
, ,
will MD vx 2
join VB
the DT nx 2
board NN
as IN
a DT nx 3
nonexecutive JJ
director NN
Nov. NNP
29 CD
. .
Features used (cf. Merlo and Stevenson 2001)

1. simple past (VBD), and past participle (VBN)

2. active (ACT) and passive (PASS)

3. causative (CAUS)

4. animacy (ANIM)
New Features used

- **POS features**: part of speech of subject and object head noun

- **SF features**: transitive (TRAN) and intransitive (INTRAN)
Differences in data between current study and (Merlo and Stevenson 2001)

- (Merlo and Stevenson 2001) used an automatically parsed corpus of 65M words. Note that the parser was trained on 1M words of annotated data: the Penn Treebank

- However, not all their features exploited the parse tree structure (they used part-of-speech tags for features such as the causative feature)

- In our study we wanted to explore whether automatic subcat frame learning can replace the use of a full parser.
Learning Subcategorization Frames: TRAN and INTRAN features

- Discover valid subcategorization frames (SFs) for each verb
- Distinguish arguments from adjuncts
- Learning from data *not* annotated with SF information
Methods Used

- Hypothesis Testing using:
  - Likelihood Ratio test
  - T-score test
  - Binomial models of miscue probabilities

- Hypothesis: \[ p(f \mid v) = p(f \mid \neg v) = p(f) \]
Subsets of observed frames

- Iterative algorithm:
  - First use counts for the observed frame $f$ in hypothesis testing
  - If $f$ is rejected as true SF, produce all subsets of $f$
  - Select one subset of $f$ as successor observed frame $s$ which is updated with $f$’s counts
  - Repeat for each $s$ rejected by hypothesis testing
Subsets of observed frames

NP PP(of) PP(by) {2} →
  - NP PP(of) {2}
  - NP PP(by) {4}
  - PP(of) PP(by) {0} →
    - PP(of) {0}
    - PP(by) {4}

NP PP(to) PP(in) {1} →
  - NP PP(in) {1}
  - NP PP(to) {0}
  - PP(to) PP(in) {0} →
    - PP(to) {0}
    - PP(with) {0}
  - PP(to) PP(in) {0} →
    - PP(with) {0}
    - NP {2+1+1}
  - NP PP(with) {1}
Successor Selection

1. Choose the successor frame that results in the strongest preference (lowest entropy across the corpus; exponential in num of frames)

2. Pick the successor frame with highest cumulative frequency at each step (greedy)

3. Random selection

→ Random selection works the best
Experiment

- Data: 23M words of WSJ text chunked

- 76 verbs picked to balance frequency (classes from Levin)

- Learning subcategorization frames for these verbs (puts the verbs into either the TRAN or INTRAN class)
Experiment

- Trained a (decision stub) Boosting classifier and a Decision Tree classifier using C5.0

- Used a 90%-10% training-test split with 10-fold cross-validation

- Tried various combinations of features to find the most informative ones
<table>
<thead>
<tr>
<th>Features</th>
<th>Average error rate from Rule Set</th>
<th>SE</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRAN, INTRAN, VBD, VBN, PASS, ACT</td>
<td>67.7%</td>
<td>0.9%</td>
</tr>
<tr>
<td>TRAN, INTRAN, VBD, VBN, PASS, ACT, CAUS</td>
<td>40.8%</td>
<td>0.6%</td>
</tr>
<tr>
<td>TRAN, INTRAN, VBD, VBN, PASS, ACT, ANIM</td>
<td>36.9%</td>
<td>1.0%</td>
</tr>
<tr>
<td>TRAN, INTRAN, VBD, VBN, PASS, ACT, PART OF SPEECH</td>
<td>38.1%</td>
<td>1.1%</td>
</tr>
<tr>
<td>TRAN, INTRAN, VBD, VBN, PASS, ACT, CAUS, ANIM</td>
<td>33.9%</td>
<td>0.8%</td>
</tr>
<tr>
<td>TRAN, INTRAN, VBD, VBN, PASS, ACT, CAUS, ANIM, PART OF SPEECH</td>
<td>37.1%</td>
<td>0.9%</td>
</tr>
<tr>
<td>TRAN, INTRAN, VBD, VBN, PASS, ACT, ANIM, PART OF SPEECH</td>
<td>35.9%</td>
<td>1.7%</td>
</tr>
<tr>
<td>TRAN, INTRAN, VBD, VBN, PASS, ACT, CAUS, ANIM, PART OF SPEECH</td>
<td>38.3%</td>
<td>1.0%</td>
</tr>
<tr>
<td>Features</td>
<td>Average error rate from Decision Tree</td>
<td>SE</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>---------------------------------------</td>
<td>-----</td>
</tr>
<tr>
<td>TRAN, INTRAN, VBD, VBN, PASS, ACT</td>
<td>49.4%</td>
<td>1.1%</td>
</tr>
<tr>
<td>TRAN, INTRAN, VBD, VBN, PASS, ACT, CAUS</td>
<td>41.1%</td>
<td>0.8%</td>
</tr>
<tr>
<td>TRAN, INTRAN, VBD, VBN, PASS, ACT, ANIM</td>
<td>37.5%</td>
<td>0.8%</td>
</tr>
<tr>
<td>TRAN, INTRAN, VBD, VBN, PASS, ACT, PART OF SPEECH</td>
<td>39.2%</td>
<td>0.8%</td>
</tr>
<tr>
<td>TRAN, INTRAN, VBD, VBN, PASS, ACT, CAUS, ANIM</td>
<td>33.4%</td>
<td>0.7%</td>
</tr>
<tr>
<td>TRAN, INTRAN, VBD, VBN, PASS, ACT, CAUS, PART OF SPEECH</td>
<td>39.0%</td>
<td>0.7%</td>
</tr>
<tr>
<td>TRAN, INTRAN, VBD, VBN, PASS, ACT, ANIM, PART OF SPEECH</td>
<td>35.8%</td>
<td>1.3%</td>
</tr>
<tr>
<td>TRAN, INTRAN, VBD, VBN, PASS, ACT, CAUS, ANIM, PART OF SPEECH</td>
<td>39.5%</td>
<td>1.0%</td>
</tr>
</tbody>
</table>
Results

- Baseline: pick argument structure at random, ER = 65.5%

- (Merlo and Stevenson 2001) measure expert-based upper bound, ER = 13.5%

- (Merlo and Stevenson 2001) obtain ER = 30.2% with 65M words of automatically parsed WSJ text

- Current work: C5.0 classifier (using SF info), ER = 33.4% with 23M words of chunked text (SF info obtained by learning)
Conclusion

- Presented a technique which automatically identifies argument structure for a set of verbs

- This work shows that this task can be accomplished using only tagged and chunked data

- We also showed that a subcategorization frame learning algorithm can be applied to this task

- We achieved an error rate of 33.4% using chunked data which compares favorably with work that used automatically parsed data