Handling cross-cutting properties in automatic inference of lexical classes:
A case study of Chintang

Olga Zamaraeva, Kristen Howell, Emily M. Bender

University of Washington

February 26, 2019
Grammar Inference

- Overarching Goal: to bring the benefits of *Grammar Engineering* to descriptive and documentary linguists

- How: automate the process of defining implemented grammars

- Short-term Goal: to create a feedback loop that brings some benefits before grammars are broad coverage
Precision (implemented) Grammars

- Bidirectional: parse and generate
- Produce syntactic and semantic representations
- Prioritize precision: the proportion of correct parses
- Uses Include:
  - Linguistic hypothesis testing (Müller, 1999; Bender, 2008; Fokkens, 2014)
  - Comparing analyses over corpora (Bender, 2010)
  - Creating treebanked corpora (Bender et al., 2012)
Precision (implemented) Grammars

```
rules.tdl
head-comp := head-comp-phrase.
subj-head := head-phrase.

irules.tdl
head-sg := plural-suffix :=
  %suffix (* s).

adj-head := plural-suffix :=
  %suffix (* s).

bare-n := plural-suffix :=
  %suffix (* s).

matrix.tdl
norm-top-lex-item := lex-item &
  [ SYNSEM [ LOCAL.CONT [ NOUN.COMMON-ITEM ] ] ].

eng.tdl
head-spec-phrase := basic-head-spec-phrase & head-final &
  [ NON-HEAD-DTR, SYNSEM.OPT - ].

norm-head :=
  [ SYNSEM [ LOCAL.CONT.HOOK.INDEX [ PNG.PER 3rd ],
  INFLECTED, NUM-FLAG - ] ].

lsg-pronoun-lex := no-spr-lex &
  [ SYNSEM, LOCAL.CONT.HOOK.INDEX [ PNG [ PER 1st, NUM sg ] ] ].
```

```
sb-hd_mc_c
   sp-hd_n_c   hd-cmp_u_c
   the_1 n_sg_ilr v_pst_olr sp-hd_n_c
   the cat_n1 chase_v1 the_1 n_sg_ilr
   cat chased the dog_n1
dog

_the_q _cat_n1 _chase_v1 _the_q _dog_n1
```

Zamaraeva, Howell and Bender
Handling cross-cutting properties in automatic inference of lexical classes
The Task at Hand: Handling Cross Cutting Categories

- We automatically infer a precision grammar from a corpus of Interlinear Glossed Text (IGT).
- Previous work (Bender et al., 2014) inferred separate grammars that supported:
  - morphological inflection
  - case and argument requirements
- We integrate the two to infer **lexical classes** that inflect and have transitivity/case-frame requirements.
Methodology: Morphotactic Inference (MOM)  
(Wax, 2014; Zamaraeva et al., 2017)

- Target the morpheme-segmented line of IGT
- Build a graph of stems and affixes
- Collapse items with overlapping edges

Latin Verbs: *am-are* (‘to love’), *port-are* (‘to carry’), *port-o* (‘I carry’), *cap-io* (‘I take’), *fac-io* (‘I do’), and *fac-ere* (‘to do’)

![Diagram showing morphotactic inference for Latin verbs am-are, port-are, cap-io, fac-io, and fac-ere]
Methodology: Case and Valence Inference

- Infer case system based on the relative frequency of grams in the corpus (e.g., ergative-absolutive) (Bender et al., 2014; Howell et al., 2017)
- Infer transitivity for individual verbs in the corpus

Sambok biyuŋ abhuŋsanduthoe.

```
sambok biu-yaŋ a-bhuŋs-a-dhend-u-tha-u-e
millet seed-ADD 2S/A-pile.up-PST-TEL-3P-TEL-3P-IND.PST
NOUN NOUN VERB
```

‘You also piled up the millet seeds.’ [ctn] (Bickel et al. 2013)

- Assign case frame according to transitivity
  - eg. intransitive verbs: subj: abs
  - transitive verbs: subj: erg, obj: abs
Methodology: Creating Integrated Lexical Classes

- Include case frame information in the morphotactic inference system’s input
  - In order to maximize examples of morphological patterns, verbs for which case frame inference failed are included in a ‘dummy’ category
- The morphotactic inference system infers inflectional classes and checks to see if a verb’s case frame is compatible with a lexical class before adding it
Case Study

We evaluate the benefits of this integration by replicating the case study used in the previous stage: Bender et al. 2014
AGGREGATION Pipeline

1. IGT Data Corpus (eg. Flex, Toolbox)
   → Conversion to Xigt
   → Xigt Corpus

2. Enriched Xigt Corpus
   → IGT Enrichment (INTENT)
   ← Grammar Inference

3. Grammar Inference
   → Choices File

4. Parsing Software (ACE, LKB)
   ← Precision Grammar
   ← Grammar_CUSTOMIZATION (Grammar Matrix)

5. Treebanking Software ([tsdb()])
Case Study: Chintang

- Chintang (ISO639-3: ctn)
- Sino-Tibetan language of Nepal
- ~5,000 speakers (Bickel et al., 2010; Schikowski et al., 2015)
- Relatively free word order, although V-final orders are most common (Schikowski et al., 2015)
- Ergative-absolutive case system, with exceptions (Stoll and Bickel, 2012; Schikowski et al., 2015)
Case Study: Dataset

- Chintang Language Resource Program (CLRP; https://www.uzh.ch/clrp/)
- 10,862 instances of Interlinnear Glossed Text (IGT)
- Split into train (8863 IGT), dev (1069 IGT) and test (930 IGT) sets
- Annotation is very detailed

Unisaŋa khatte mo kosi? moba.

-u-nisaŋa           khatt-e  mo    kosi-i?     mo-pe
3sPOSS-younger.brother-ERG.A take-IND.PST DEM.DOWN river-LOC DEM.DOWN-LOC

‘The younger brother took it to the river.’ [ctn] (Bieckel et al., 2013)
# Grammars

Comparison choices files come from Bender et al. (2014)

In all grammars, all arguments can drop

<table>
<thead>
<tr>
<th>Grammar</th>
<th>Lexicon</th>
<th>Morphology</th>
<th>Word Order</th>
<th>Case System</th>
</tr>
</thead>
<tbody>
<tr>
<td>BASELINE</td>
<td>full form</td>
<td>none</td>
<td>default</td>
<td>default</td>
</tr>
<tr>
<td>ORACLE</td>
<td>Toolbox</td>
<td>hand-defined</td>
<td>v-final</td>
<td>erg-abs</td>
</tr>
<tr>
<td>FF-AUTO-GRAM</td>
<td>full form</td>
<td>none</td>
<td>v-final</td>
<td>erg-abs</td>
</tr>
<tr>
<td>MOM-DEFAULT-NONE</td>
<td>inferred</td>
<td>inferred</td>
<td>default</td>
<td>default</td>
</tr>
<tr>
<td>INTEGRATED</td>
<td>inferred</td>
<td>inferred</td>
<td>default</td>
<td>erg-abs</td>
</tr>
</tbody>
</table>

default word order: free
default case: none
## Lexical Information in each Grammar

<table>
<thead>
<tr>
<th>Grammar</th>
<th># verb entries</th>
<th># noun entries</th>
<th># verb affixes</th>
<th># noun affixes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORACLE</td>
<td>899</td>
<td>4750</td>
<td>233</td>
<td>36</td>
</tr>
<tr>
<td>BASELINE</td>
<td>3005</td>
<td>1719</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FF-AUTO-GRAM</td>
<td>739</td>
<td>1724</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>MOM-DEFAULT-NONE</td>
<td>1177</td>
<td>1719</td>
<td>262</td>
<td>0</td>
</tr>
<tr>
<td>INTEGRATED</td>
<td>911</td>
<td>1755</td>
<td>220</td>
<td>76</td>
</tr>
</tbody>
</table>
# Results

<table>
<thead>
<tr>
<th>Grammar</th>
<th>lexical coverage</th>
<th>parsed</th>
<th>correct</th>
<th>readings</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORACLE</td>
<td>116 (12.5%)</td>
<td>20 (2.2%)</td>
<td>10 (1.1%)</td>
<td>1.35</td>
</tr>
<tr>
<td>BASELINE</td>
<td>38 (0.4%)</td>
<td>15 (1.6%)</td>
<td>8 (0.9%)</td>
<td>27.67</td>
</tr>
<tr>
<td>FF-AUTO-GRAM</td>
<td>18 (1.9%)</td>
<td>4 (0.4%)</td>
<td>2 (0.2%)</td>
<td>5.00</td>
</tr>
<tr>
<td>MOM-DEFAULT-NONE</td>
<td>39 (4.2%)</td>
<td>16 (1.7%)</td>
<td>3 (0.3%)</td>
<td>10.81</td>
</tr>
<tr>
<td>INTEGRATED</td>
<td>105 (11.3%)</td>
<td>32 (3.4%)</td>
<td>15 (1.6%)</td>
<td>91.56</td>
</tr>
</tbody>
</table>
Error Analysis: INTEGRATED system

- The vast majority of sentences that failed did not succeed at lexical analysis
  - Of those the majority failed because an affix or stem was not in the training data
- The remaining 73 sentences did not succeed at syntactic analysis
  - 6 contained an NP who’s case was not compatible with the verb’s case frame (such as a locative NP modifier)
  - 23 did not contain a word that the grammar analysed as a verb
  - 44 contained multiple verbs
### Error Analysis: INTEGRATED vs. ORACLE

<table>
<thead>
<tr>
<th></th>
<th>Morpheme not in Grammar</th>
<th>Morphemes Can’t Combine</th>
<th>Words Can’t Combine</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORACLE</td>
<td>46</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>INTEGRATED</td>
<td>58</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
Ambiguity

**INTEGRATED** has high ambiguity due to multiple entries for verbs and affixes

(1) **cuwa** mai-yu-th-a-k-e
    water **NEG-be.there-NEG-PST-IPFV-IND.PST**
    ‘Was there water?’

Zamaraeva, Howell and Bender UW
Handling cross-cutting properties in automatic inference of lexical classes
Future Work

- Add non-inflecting lexical rules to MOM
- Infer alternate case frames for verbs
- Extend this methodology to other morpho-syntactic features
- Extend syntactic inference to account for more phenomena
Conclusion

- Integrating morphological and syntactic inference improves coverage
  - With better coverage, inferred grammars can help linguists discover patterns of the combinatorics in their data
- This methodology can be extended to other morpho-syntactic features
- We are scaling up quickly! If you have an IGT corpus and want an inferred grammar, come talk to us.
Conclusion

• Integrating morphological and syntactic inference improves coverage
  • With better coverage, inferred grammars can help linguists discover patterns of the combinatorics in their data
• This methodology can be extended to other morpho-syntactic features
• We are scaling up quickly! If you have an IGT corpus and want an inferred grammar, come talk to us.

Thank you!
Acknowledgements

- This material is based upon work supported by the National Science Foundation under Grant No. BCS-1561833. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

- We would like to thank Alex Burrell for providing code from a previous project that we built on in order to infer transitivity.

- We thank previous AGGREGATION research assistants for converting the corpus into Xigt and enriching it with INTENT.
References I


References II


