Typological coverage and descriptive precision in grammar engineering

Scott Drellishak, U. Washington (sfd@u.washington.edu)
Emily Bender, U. Washington (ebender@u.washington.edu)
Dan Flickinger, CSLI Stanford (danf@csli.stanford.edu)
Jeff Good, MPI EVA (good@eva.mpg.de)

Overview

- Introduce the Grammar Matrix
- Discuss how a typologically-informed model of coordination has been implemented in a machine-readable grammar
- Address some potentially interesting typological claims that came out of the implementation

Grammar Matrix

- What kind of description?
  - Formal/machine-readable grammars
  - Intended to work for parsing strings so as to assign semantic representations to them
  - Also intended to work generating strings representing linguistic forms from semantic representations

Grammar Matrix

- The Grammar Matrix (Bender et al. 2002) is a machine-readable grammar "starter kit"
- Based on lessons learned from building machine-readable grammars within Head-Driven Phrase Structure Grammar (Pollard and Sag 2004; Sag, Wasow, and Bender 2003)

Grammar Matrix

- Originally, these grammars were designed narrowly for particular languages (e.g., English or Japanese)
- However, it was noticed that many linguistic "universals" were being duplicated across grammars
- It, therefore, made sense to "factor out" what was common to all the grammars so they all could make use of a common foundation

Grammar Matrix

- Some of the relevant "universals"
  - Words and phrases combine to make larger phrases
  - The meaning of a phrase is determined by the words in the phrase and how they are put together
  - Heads of phrases determine which types of arguments they require, and how they combine semantically with those arguments

Grammar Matrix

- What does the Matrix look like?

```
headed-phrase := phrase &
   [ SYNSEM.LOCAL [ CAT.HEAD head & #head,
                    AGR #agr ],
     HEAD-DTR.SYNSEM.LOCAL local &
          [ CAT.HEAD #head,
            AGR #agr ] ].
```

Grammar Matrix

- Or a little prettier...

$$\textit{headed-phrase}: \begin{bmatrix} \textit{phrase} & \\ \text{SYNSEM.LOCAL} & \begin{bmatrix} \text{CAT.HEAD} & \boxed{1}\textit{head} \\ \text{AGR} & \boxed{2} \end{bmatrix} \\ \text{HEAD-DTR.SYNSEM.LOCAL} & \begin{bmatrix} \textit{local} & \\ \text{CAT.HEAD} & \boxed{1} \\ \text{AGR} & \boxed{2} \end{bmatrix} \end{bmatrix}$$

"In any headed phrase, the mother's HEAD value (part of speech and related characteristics) and agreement features come from the head daughter."

Grammar Matrix

- In addition, work has been done within the Grammar Matrix to deal with grammatical phenomena which differ across languages in common ways
  - Negation
  - Word order
  - Yes-no questions
  - Coordination
  - ...

Grammar Matrix

- Such aspects of grammars are handled via modules that extend the core of the Matrix
- Modules consist of
  - Statements of rules describing common grammatical strategies
  - Utilities which take input from the user and configure the modules in an appropriate way for a given language

Grammar Matrix

- Why (we think) the grammar matrix might be good for typology
  - Integrates typological variation with precise descriptions
  - In time could help automate typological classification
  - Allows typologists and descriptive linguists to more easily cooperate with computational linguists

Grammar Matrix

- Computational typology of this sort is still very new
- The typological coverage is, in many respects, quite simplistic
- However, the Matrix has been, and continues to be, tested on more languages in the context of grammar engineering classes

Coordination

- One aspect of grammar for which a Grammar Matrix module has been developed is coordination
- What we mean by "coordination" here is a structure that combines elements of like or similar category into a single larger element
- The current coordination module does not yet make it easy to formalize all types of coordination

Coordination

- Aspects of coordination we are trying to handle at this stage
  - Formal aspects of coordination marking (e.g., "conjunctions", special morphological forms, etc.)
  - Different strategies for different phrase types
  - How often coordination markers appear in the coordinate structure (e.g., monosyndeton vs. polysyndeton)

Coordination

- Monosyndeton:
  John, James, and Matthew
- Polysyndeton (Polish):
  Tomek i Jurek i Maciek przyjechali do Londynu.
  "Tomek and Jurek and Maciek went to London."
  (Example from Haspelmath (to appear: 11))
- Omnisyndeton (Abun, West Papuan):
  Mbos e ndabu e ndam ga sye ne e an fowa sino.
  pigeon & dove & bird REL big DET & 3PL forbidden all
  "Pigeons, doves and birds that are big, they are all forbidden (for women to eat)." (Berry and Berry 1999:96)

Implementation

- Two problems
  - Parsing and generating indefinite numbers of conjuncts
  - Parsing and generating the three different patterns of syndeton

Implementation

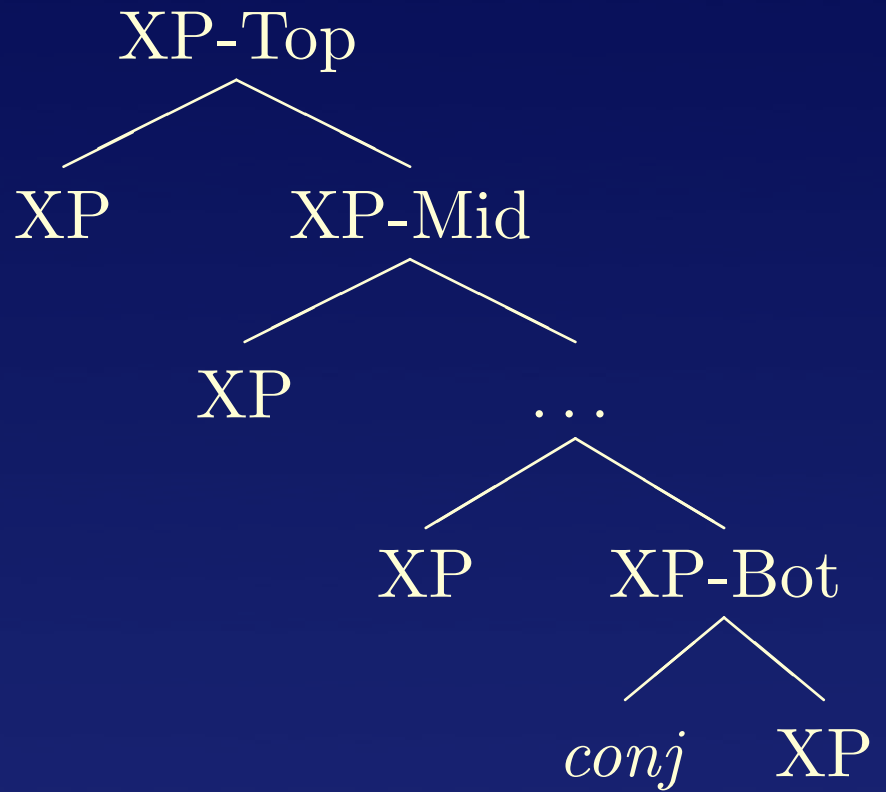
- One way of formalizing "unlimited" numbers of conjuncts
  XP → XP+ *conj* XP
  XP → XP *conj* (XP *conj*)+
- Another way
  XP-Top → XP XP-Mid
  XP-Mid → XP XP-Mid
  XP-Mid → XP XP-Bot
  XP-Bot → *conj* XP
  (for monosyndeton)

Implementation

- We choose the latter for two reasons
  - The parsing and generating system we use, the LKB (Copestake 2002), doesn't allow rules of the first type
  - It makes it easier to create rules to deal with deal with issues like agreement clash resolution in coordinate structures

Implementation

- Lots of constituents are parsed/generated



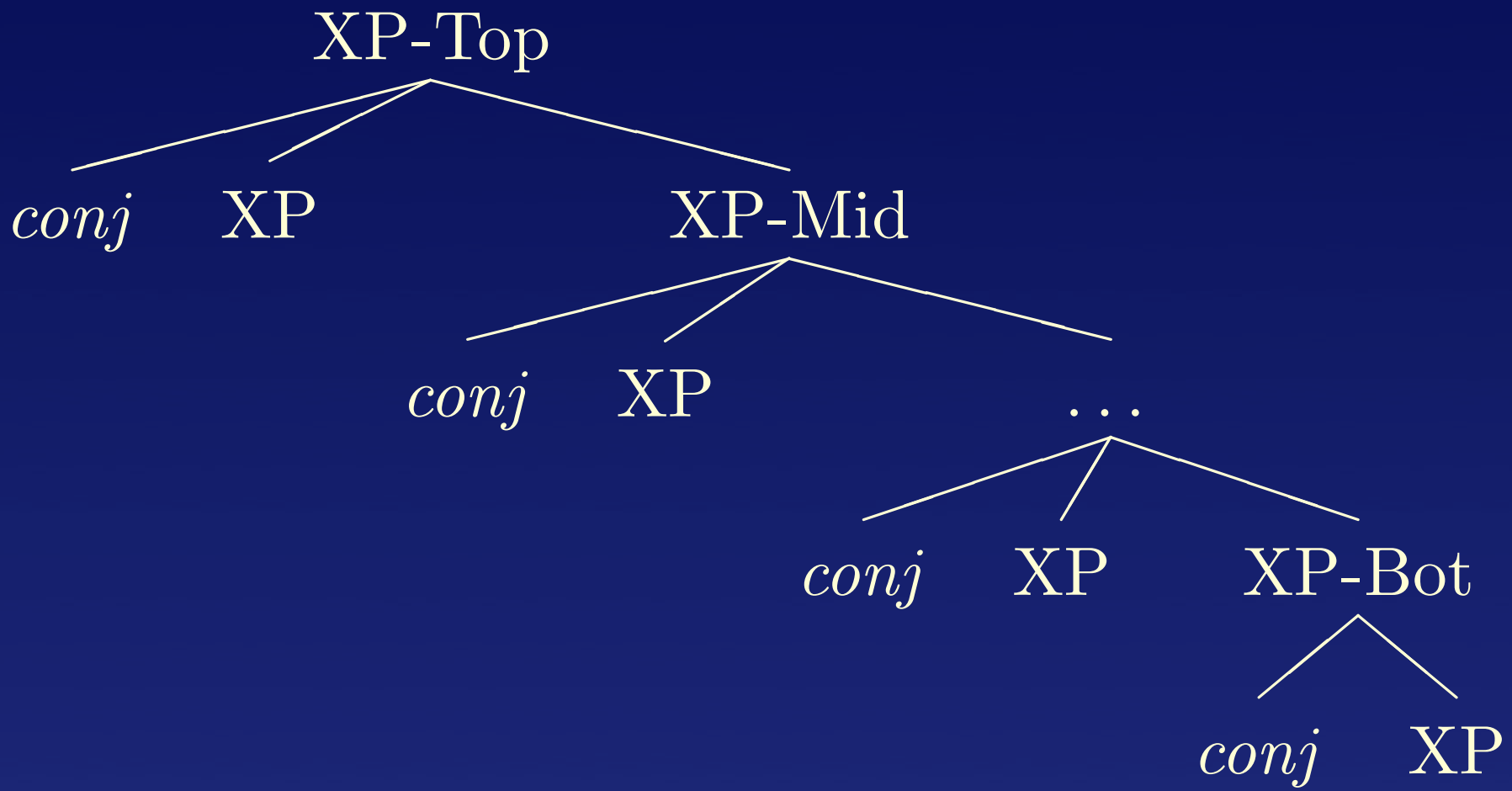
- But, HPSG doesn't put much theoretical significance on tree structures

Implementation

- Monosyndeton
  XP-Top → XP XP-Mid
  XP-Mid → XP XP-Mid
  XP-Mid → XP XP-Bot
  XP-Bot → *conj* XP
- Polysyndeton
  XP-Top → XP XP-Coord
  XP-Coord → *conj* XP-Top
- Omnisyndeton
  XP-Top → *conj* XP XP-Mid
  XP-Mid → *conj* XP XP-Mid
  XP-Mid → *conj* XP XP-Bot
  XP-Bot → *conj* XP

Implementation

- Omnisyndeton tree example



Typology

- The omnisyndeton dilemma: One too many conjunctions
- This means one coordination relation needs to be ignored in our framework
- To handle this we need to posit homophony
  - One "real" conjunction
  - One "expletive" conjunction

Typology

- Omnisyndeton (revised)
  XP-Bot → *expletive-conj* XP
- We have chosen the bottom conjunction as the expletive one since it is the easiest to identify
- We are not yet aware of data favoring the choice of one conjunction as expletive over another

Typology

- These expletive conjunctions are implemented in essentially the same way as other expletives
- The difference: They are the only ones (so far) that obligatorily co-occur in the same construction as their non-expletive homophones
- We think the need to treat omnisyndeton differently is interesting—though we're not sure we have found an ideal solution

Typology

- To this point, we've been using simplistic notation like XP → XP *conj* XP
- But as seen above, the Matrix formalism is much more complex and expressive

headed-phrase:

$$\begin{bmatrix} \textit{phrase} \\ \text{SYNSEM.LOCAL} & \begin{bmatrix} \text{CAT.HEAD} & \boxed{1}\textit{head} \\ \text{AGR} & \boxed{2} \end{bmatrix} \\ \text{HEAD-DTR.SYNSEM.LOCAL} & \begin{bmatrix} \textit{local} \\ \text{CAT.HEAD} & \boxed{1} \\ \text{AGR} & \boxed{2} \end{bmatrix} \end{bmatrix}$$

Typology

- Another fact that came out in implementation: Category-neutral rules using devices like "XP" are inadequate
- A language may have the same basic coordination strategy for noun phrases, verb phrases, and sentences
- But, at least in many languages, separate rules are needed for each major phrasal type

Typology

- This is because many aspects of syntax can differ across superficially similar coordination types
  - Noun phrase coordination: Agreement class of whole structure can be different from agreement class of constituent noun phrases
  - Verb phrase coordination: Agreement class of each coordinated verb phrase typically the same

Typology

- Thus, at the level of implementation, even a language with a multi-purpose conjunction like *and* has many different coordination rules

Conclusion

- Grammar engineering has reached a point where its machine-readable descriptions can be more typologically informed
- This is good for grammar engineering
- It also seems like it may be good for typology

Acknowledgments

We would like to thank Stephan Oepen, Laurie Poulson, the Grammar Engineering classes of 2004 and 2005 at UW, and NTT Communication Science Laboratories for their support through a grant to CSLI (Stanford).

References

Bender, Emily M. and Dan Flickinger. 2005. Rapid prototyping of scalable grammars: Towards modularity in extensions to a language-independent core. In Proceedings of the 2nd International Joint Conference on Natural Language Processing IJCNLP-05 (posters/demos), Jeju Island, Korea.

Berry, Keith, and Christine Berry. 1999. A description of Abun: a West Papuan language of Irian Jaya. Pacific Linguistics B-115. Canberra: Pacific Linguistics, Research School of Pacific and Asian Studies, The Australian National University.

Copestake, Ann. 2002. Implementing Typed Feature Structure Grammars. Stanford: CSLI.

Bender, Emily M., Dan Flickinger and Stephan Oepen. 2002. The Grammar Matrix: An Open-source starter-Kit for the rapid development of cross-linguistically consistent broad-coverage precision grammars. *Proceedings of the Workshop on Grammar Engineering and Evaluation at the 19th International Conference on Computational Linguistics*. Taipei, Taiwan. pp. 8-14.

Drellishak, Scott and Emily M. Bender. 2005. A coordination module for a crosslinguistic grammar resource. In S. Müller (Ed.) The Proceedings of the 12th International Conference on Head-Driven Phrase Structure Grammar, 108–128, Stanford. CSLI Publications.

Haspelmath, Martin. To appear. Coordination. In Shopen, Timothy (ed.), *Language typology and l inguistic description*. 2nd ed. Cambridge: Cambridge University Press. (Available at: http://email.eva.mpg.de/~haspelmt/coord.pdf)

Carl Pollard and Ivan A. Sag. 1994. Head-Driven Phrase Structure Grammar. University of Chicago Press.

Sag, Ivan A., Thomas Wasow, and Emily M. Bender. 2003. Syntactic Theory: A formal introduction. 2nd Edition. Stanford: CSLI Publications.