Grammar Engineering Complements Language Documentation

Emily M. Bender
University of Washington
ebender@uw.edu

1. What is Grammar Engineering?

- The development of grammars-in-software, used in combination with parsing and generation algorithms to:
  - Assign structures (morphological, syntactic and semantic) to input strings
  - Generate strings from input semantic representations
  - Recognize ungrammatical strings
- Dates back to the 1950s
- Finally practical, thanks to advances in computing technology and computational linguistics

2. Why Do Grammar Engineering?

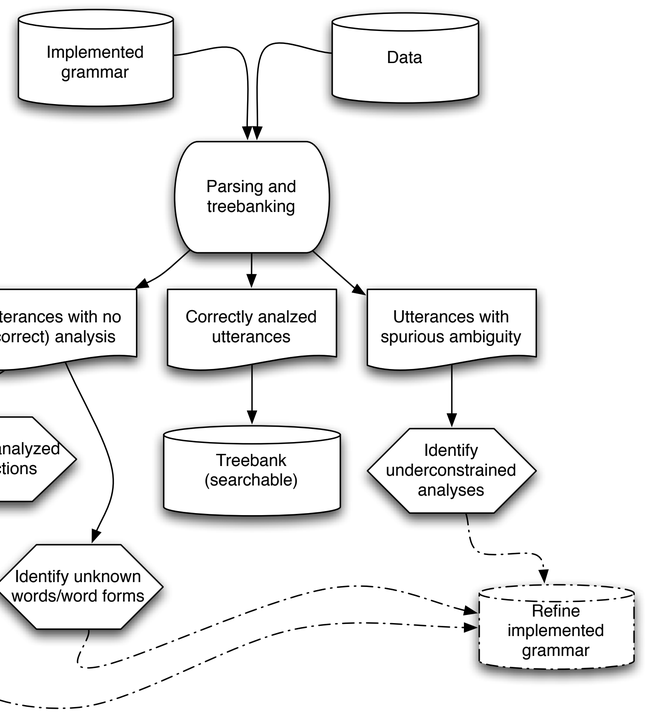
- Automatically apply analyses to large data sets
- Identify unanalyzed constructions
- Identify words or word forms missing from lexicon/morphology
- Identify underconstrained analyses
- Produce a searchable treebank

3. What is the Grammar Matrix?

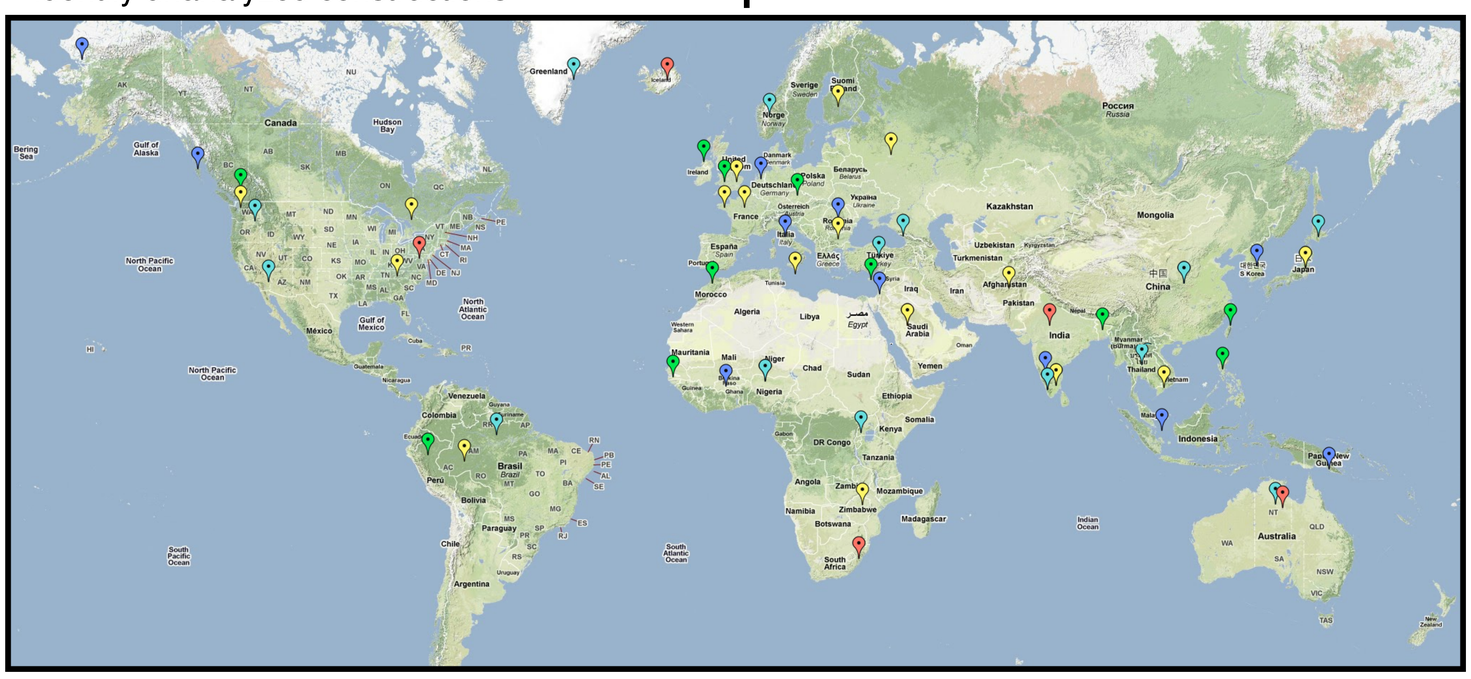
`http://www.delph-in.net/matrix`

- A repository of implemented analyses, including:
  - A core grammar with analyses of general patterns such as semantic compositionality
  - "Libraries" of analyses of cross-linguistically variable phenomena
- Accessible via a web-based questionnaire
- Customization system produces working HPSG grammars from typological descriptions

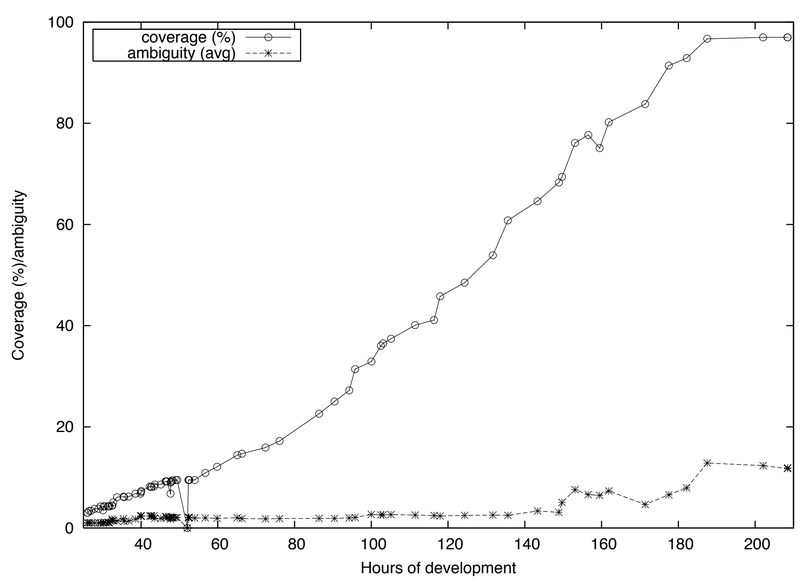
Bender et al. 2002, 2010



Grammar engineering process and outcomes



Languages of grammars created with the customization system. Map image courtesy of Google Maps, location data courtesy of WALS.



Wambaya grammar development curve

4. Sample Treebank Queries

Show me all utterances with. . . :

- two overt arguments of the same verb
- an argument marked by preposition X
- a long-distance dependency crossing two or more clause boundaries
- an overt subject and an implicit object
- noun phrases with two or more modifiers on the same side of the head
- floated quantifiers

5. Why HPSG?

Grammar engineering requires precise, formalized analyses. HPSG (Head-driven Phrase Structure Grammar; Pollard and Sag 1994) is well-suited to grammar engineering in general and grammar engineering for language documentation in particular:

- Lexicalist framework: Most of the information is stored in the lexicon.
- Constraint-based: Linguistic knowledge can be applied in an order-independent fashion, in parsing and generation.
- Surface-oriented: HPSG analyses do not posit ancillary structures.
- Integrated: HPSG analyses map surface strings to semantic representations.
- Broad-coverage: HPSG grammars seamlessly incorporate both broad generalizations and idiosyncrasies; no core/periphery distinction.
- Data-driven: A more bottom-up approach to the exploration of linguistic universals.

6. Opportunities for Collaboration

Grammar engineering does not have to mean more work for the field linguist!

- We believe that the best model involves collaboration between field linguists and grammar engineers.
- The cost of grammar implementation is small compared to the overall cost of language documentation.
  - In 210 hours of development (1/20th of the time Nordlinger spent on the original analysis), Bender (2008) was able to create a grammar that could assign appropriate analyses to 91% of the example sentences from Nordlinger 1998.
- In UW's Linguistics 567 (taught annually in Winter quarter), students each do implemented grammars for different languages. Many of these students would be very interested to work with active field linguists.

7. Related work

Grammar engineering is not limited to HPSG! Here are some other multilingual grammar engineering projects:

- ParGram (LFG): `http://pargram.b.uib.no`
- CoreGram (HPSG): `http://hpsg.fu-berlin.de/Projects/core.html`
- OpenCCG (CCG): `http://openccg.sourceforge.net`
- PAWS (PC-PATR): `http://carla.sil.org/paws.htm`

8. Acknowledgments

This poster represents joint work with the Grammar Matrix development team, including: Dan Flickinger, Stephan Oepen, Scott Drellishak, Laurie Poulson, Antske Fokkens, Michael Wayne Goodman, Kelly O'Hara, Safiyyah Saleem, Joshua Hou and Daniel P. Mills.

This material is based upon work supported by the National Science Foundation under Grant No. 0644097. 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 gratefully acknowledge the support of the Utilika Foundation.

References

Emily M. Bender. 2008. Evaluating a crosslinguistic grammar resource: A case study of Wambaya. In *Proceedings of ACL08:HLT*.

Emily M. Bender, Scott Drellishak, Antske Fokkens, Laurie Poulson, and Safiyyah Saleem. 2010. Grammar customization. *Research on Language & Computation*, pp. 1–50. 10.1007/s11168-010-9070-1.

Emily M. Bender, 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. In John Carroll, Nelleke Oostdijk, and Richard Sutcliffe, editors, *Proceedings of the Workshop on Grammar Engineering and Evaluation at the 19th International Conference on Computational Linguistics*, pp. 8–14. Taipei, Taiwan.

Martin Haspelmath, Matthew S. Dryer, David Gil, and Bernard Comrie, editors. 2008. *The World Atlas of Language Structures Online*. Max Planck Digital Library, Munich. Http://wals.info.

Rachel Nordlinger. 1998. *A Grammar of Wambaya, Northern Australia*. Research School of Pacific and Asian Studies, The Australian National University, Canberra.

Carl Pollard and Ivan A. Sag. 1994. *Head-Driven Phrase Structure Grammar*. Studies in Contemporary Linguistics. The University of Chicago Press and CSLI Publications, Chicago, IL and Stanford, CA.