Grammar Customization with the LinGO Grammar Matrix

1 Tutorial Content

This tutorial provides an overview of the LinGO Grammar Matrix customization system[1] (Bender et al., 2002; Bender and Flickinger, 2005; Drellishak, 2009), a free web-based tool that can be used as an easy entry point into developing broad-based grammars for those unfamiliar with grammar engineering and as a time-saving device for those who are.

Grammar engineering is of interest for both natural language processing applications and linguistic research. For NLP, syntactic structure is becoming increasingly important to a variety of tasks, including MT (e.g., Quirk et al. 2005) and ASR (e.g., Collins et al. 2005), and grammar engineering provides an alternative to manual treebank construction as a way to capture the knowledge required to automatically assign syntactic structure to natural language text. For linguistic research, especially syntactic research in unification-based frameworks, grammar engineering can be used to compare analyses and test them for consistency in relation to analyses of other phenomena. However, developing broad-coverage grammars is time-intensive, and can be prohibitively so in many situations. The LinGO Grammar Matrix is intended to reduce the costs of creating broad-coverage precision grammars.

The Grammar Matrix customization system is a web-based service which elicits typological descriptions of languages and outputs customized grammar fragments suitable for sustained development into broad-coverage grammars. The created grammars use the formalism of Head Driven Phrase Structure Grammar (Pollard and Sag 1994, HPSG), provide bidirectional mappings between surface strings and semantic representations in the format of Minimal Recursion Semantics (Copestake et al. 2005, MRS), and can be run and further developed within the LKB grammar development environment (Copestake 2002).

We intend this tutorial to be of interest to computational linguists of various stripes. Researchers in statistical NLP may find it interesting as a view into a structure-based approach to cross-linguistic variation. Experienced grammar engineers may find this overview interesting for cross-framework comparison and/or the construction of multilingual resources similar to the Grammar Matrix but representing different frameworks. Theoretically-oriented syntacticians can use the Grammar Matrix customization system for linguistic hypothesis testing (Bender, 2008), while typologists may be interested in it as a means of investigating the interaction of phenomena cross-linguistically.

The tutorial will consist of two parts. In the first part, we will demonstrate the web-interface of the Grammar Matrix customization system, illustrating how to use the typological questionnaire to capture subtle linguistic facts and maximize the size of the starting grammar fragment produced by the system. At the end of the first part, we will generate grammars from the filled-out questionnaire. The second part of the tutorial will provide a demonstration and instructions on how to continue the development of the customized starter grammar. These will include explanation of type description language (TDL), the formalism in which the grammars are defined and the LKB grammar development environment, as well as general suggestions about grammar development projects.

This tutorial is an advanced tutorial in the sense that participants should have some background knowledge in linguistics. The first part of the tutorial is accessible to most people working with natural language and technology. The second part of the tutorial, which focuses on grammar engineering, requires some knowledge of formal grammars for natural language (preferably HPSG or another unification based grammar).

[1] http://www.delph-in.net/matrix/customize/matrix.cgi

2 Outline

Part 1 Grammar Customization

(a) General introduction to the Matrix customization system

(b) The Matrix questionnaire: a step by step overview of how to fill out the questionnaire

(c) Customizing grammars: actual grammars will be created from the filled out questionnaires

Break

Part 2 Grammar development with the LKB

(c) An introduction to type description language (TDL)

(d) An overview of the created grammar

(e) Regression testing/grammar profiling

(f) Extending the grammar: Implementing new phenomena with LKB

(g) Large scale grammar development

3 Speakers at the Tutorial

- Emily M. Bender, University of Washington
  Department of Linguistics, University of Washington
  Box 354340
  Seattle WA 98195-4340, USA
  email: ebender@u.washington.edu
  homepage: http://faculty.washington.edu/ebender
  tel: +1 206 543-6914, fax: +1 206 685-7978
  Background: Assistant Professor, Department of Linguistics, Adjunct Assistant Professor, Department of Computer Science and Engineering, and Director, Professional Master's Program in Computational Linguistics. Principle Investigator on the LinGO Grammar Matrix project. PhD Linguistics (2000) Stanford University.

- Antske S. Fokkens, Saarland University
  Department of Computational Linguistics, Saarland University
  Postfach 15 11 50
  66041 Saarbrücken, Germany
  e-mail: afokkens@coli.uni-saarland.de
  homepage: http://www.coli.uni-saarland.de/ afokkens/
  tel: +49 681 302 70019, fax: +49 681 302 4700
  Background: Researcher and teacher at Saarland University. Developer of the Matrix Customization System as part of her PhD research. MsC Language Science and Technology, Saarland University (2007). MA Linguistics (specialization informatics), University of Bordeaux III (2005).

- Safiyyah Saleem
  Department of Linguistics, University of Washington
  Box 354340
  Seattle, WA 98195-4340
  email: ssaleem@u.washington.edu
  tel: +1 206 543 2046 fax: +1 206 685 7978
  Background: Student at University of Washington. Developer of Matrix Customization System as a part of her MA Research. MA Applied Linguistics, Georgia State University(2007)

Primary contact: Antske Fokkens (afokkens@coli.uni-saarland.de)

References

Bender, Emily, Flickinger, Dan and Oepen, Stephan. 2002. The Grammar Matrix: An Open-Source Starter-Kit fir the Rapid Development of Cross-linguistically Consistent Broad-Coverage Precision Grammars. In *Proceedings of the Workshop on Grammar Engineering and Evaluation at the 19th Conference on Computational Linguistics*, pages 8 – 14, Taipei, Taiwan.

Bender, Emily M. 2008. Grammar Engineering for Linguistic Hypothesis Testing. In Nicholas Gaylord, Alexis Palmer and Elias Ponvert (eds.), *Proceedings of the Texas Linguistics Society X Conference: Computational Linguistics for Less-Studied Languages*, pages 16–36, Stanford: CSLI Publications.

Bender, Emily M. and Flickinger, Dan. 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.

Collins, Michael, Roark, Brian and Saraclar, Murat. 2005. Discriminative Syntactic Language Modeling for Speech Recognition. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 507–514, Ann Arbor, Michigan: Association for Computational Linguistics.

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

Copestake, Ann, Flickinger, Dan, Pollard, Carl and Sag, Ivan. 2005. Minimal Recursion Semantics: An Introduction. *Research on Language and Computation* 3(4), 281–332.

Drellishak, Scott. 2009. *Widespread But Not Universal: Improving the Typological Coverage of the Grammar Matrix*. Ph. D.thesis, University of Washington.

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

Quirk, Chris, Menezes, Arul and Cherry, Colin. 2005. Dependency Treelet Translation: Syntactically Informed Phrasal SMT. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 271–279, Ann Arbor, Michigan: Association for Computational Linguistics.