Research on Language and Computation (2005) 3:131–138
DOI 10.1007/s11168-005-1293-1

Introduction

Shared Representation in Multilingual Grammar Engineering

1. Multilingual Grammar Engineering

Multilingual grammar engineering is the development of implemented grammars for parsing and/or generation of natural language texts explicitly set in a multilingual environment. This field of endeavor was the topic of a workshop at ESSLLI 2003 (in Vienna), and the workshop, in turn, was the inspiration for the current volume, Our purpose here is to bring together papers from different multilingual grammar engineering efforts which all address one particular issue, namely, questions of shared representations across grammars: Which levels of representation should be shared? To what extent should the representations be shared? What are the implications (engineering and linguistic) of such mismatches as are allowed? How can the process of standardization be designed to encourage experimentation and new discoveries within particular language grammars while still maintaining consistent standards? What kinds of evaluation methods are available to test grammars or grammar outputs for conformity with the standards?

1.1. Why multinational grammar engineering?

The practice of multilingual grammar engineering is motivated by both practical and scientific considerations. From a practical point of view, multilingual grammar engineering is a means of reusing existing technology, along two dimensions. In the first instance, grammars for multiple different languages written in the same formalism and with standardized output representations allow grammar engineers to reuse the same grammar development environments, parsers, generators, and down-stream applications for different languages. In the second instance, a grammar for one language can often reuse or adapt analyses developed for analogous phenomena in a grammar for another language. Work in the KPML framework (e.g., Bateman et al., 2005) has a long tradition of reuse of resources across languages (including typologically diverse ones) at the level of pragmatic and discourse information for both strategic and tactical generation. In addition, recent work in the Pargram (Butt et a1., 1999, 2002) and Lin

GO Grammar Matrix (Bender et al., 2002; Flickinger and Bender, 2003) projects suggests that syntactic structure as well is amenable to reuse across (again, even typologically diverse) languages.

This suggests the scientific motivation for multilingual grammar engineering: to the extent that implemented grammars serve as a means of linguistic hypothesis testing, multilingual grammar engineering can serve as means of testing hypotheses about language universals. This testing takes two forms first, by maintaining test suites which reflect the phenomena analyzed within the grammar, a grammar engineer can use regression testing to ensure that as analyses of new phenomena are added they interact properly with the existing analyses (Oepen et al., 2002). Second, by developing the grammar against corpora of naturally occurring text, the grammar engineer can test the hypotheses encoded in the grammar against aipproximation of the actual state of the language (Baldwin et al., 2004). Multilingual grammar engineering allows crosslinguistic hypotheses to also be subjected to these two stringent kinds of testing.

Not surprisingly, the benefits and motivations for multilingual grammar engineering have implications for the design and extent of shared representations. These are discussed in section 2 below.

1.2. Different approaches to multilingual grammar engineering

In the contributions to the ESSLLI workshop and the papers in this volume, we find four basic approaches to grammar engineering. The first, exemplified by the ParGram project and the paper by King et al. in this volume, involves the coordinated parallel development of grammars of different languages and at different sites. The commonalities across these grammars are maintained as they all develop, and can serve to jump-start new grammars as the project takes on new languages. The second, exemplified by the LinGO Grammar Matrix project and the paper by Borthen and Haugereid in this volume, involves abstracting a cross-linguistic core grammar on the basis of a small number of existing broad coverage grammars, and then using that core grammar as the basis for new grammars. As the core grammar is updated and extended, these changes are propagated to existing grammars. A third approach involves multilingual grammar engineering in the context of pairs or sets of closely related languages. In this case, resources developed for one language are ported to another. This is exemplified by the paper by Smrž in this volume, as well as by ongoing ParGram work (Kim et al., 2003) porting an existing Japanese grammar to Korean, and recent steps toward developing a common core Scandinavian Matrix grammar. The fourth approach aims to allow the expression of any similarities between any groups of languages without expecting any particular constraint or construct to be universally valid across languages, while still maintaining the goals of re-use and parallel development. This approach is exemplified in the context of multilingual generation (both strategic and tactical) by the paper by Bateman et al, in this volume. Finally, the work of Cahill et al. (this volume) represents a related strand of research. This work takes a multilingual perspective on the abstraction of wide-coverage LFG grammars and lexical resources from tree or dependency banks, addressing the knowledge acquistion bottleneck in multilingual constraint-based grammar development.

Once again, the particular approach to multilingual grammar engineering has implications for the design and extent of shared representations, These are discussed in the next section.

2. Implications for Shared Representations

Multilingual grammar engineering requires an implicit or explicit commitment to a shared level or levels of representation. Some such sharing is required if the projects on different languages are to exchange technological resources. However, which levels of representation are shared, and the extent to which they are shared, differs across multilingual grammar engineering efforts. In particular, the motivation for the project (section 1.l) and the type of project (section 1.2) each have implications for shared representations. This section tales each of these issues in turn.

2.1. Implications from the motivation for multilingual grammar engineering

At a basic practical level, grammars which are interpreted by the same parsers (and generators) must be implemented in the formalism interpreted by these engines. To the extent that the parsers and generators remain agnostic about the particulars of grammars, this only requires that the grammars share the general form of their representations, and does not constrain the content to be at all similar.

To the extent that the grammars are intended to be interchangeable with respect to down-stream applications, there may be further con-straints on the level of representation corresponding to the output of a parser (or input to a generator). For example, the LinGO Grammar Matrix (Flickinger and Bender, 2003) implements the algebra for Minimal Recursion Semantics developed in (Copestake et ai., 2001). Thus the outputs that the LKB (Copestake, 2002) parser produces with a Matrix grammar are well-formed MRS representations. The well-formedness constraints on MRS representations are semantically motivated, but also form part of the interface conditions to downstream applications, such as creativity support in document creation (Uszkoreit e al., 2004), automated customer email response (Siegel and Bender, 2002), or machine translation (Oepen et al., 2004). To the extent that such interface conditions can be kept to just the MRS well-formedness conditions. Matrix grammars should be reusable across different application settings, and down-stream applications reusable with different Matrix grammars. For applications involving natural language understanding, however, further standardization of the content of MRS representations is probably necessary. Likewise, the standardized f-structures of LFG grammars promote reuse of back-end systems and KPML grammars take as input standardized Sentence Plan Language (SPL) representations (Kasper, 1989) base on text-planning input, and then realize surface forms which based on the SPL representations.

The other practical motivation for multilingual grammar engineering is reuse of analyses across grammars. This consideration encourages shared representations not just at the level of the interface to further applications, but also grammar-internally. The sharing of specific features and values across f-structures in ParGram grammars represents not only a confluence of output representations but also a sharing of analyses within the grammars (see King et al., this volume). Likewise, Matrix-based grammars share not only their output representations, but also a common set of features and types implementing valence, agreement, long-distance dependencies, etc, Many of these similarities are required in order to build the standardized output representations (e.g., the implementation of the linking of syntactic and semantic arguments), but others may be more purely syntactic. Finally, a grammar-porting approach aims to reuse particular analyses as much as possible, by only adapting those which do not work for the target language.

Finally, the testing of hypotheses about language universals (and typological variation) brings a conflicting set of pressures to the design of shared representations. On the one hand, any hypothesized universal would of necessity be shared across all of the grammars. On the other hand, the practice of testing science through engineering requires giving engineers enough flexibility to create a working system. It can be quite expensive (in time and effort) to change a fundamental decision that has implications throughout a grammar. The larger a grammar gets, the more inertia it tends to have in this sense. Thus parallel grammar development requires a high level of commitment on the part of the participating grammar engineers. We believe that most practical efforts in this space will end up moving towards broad semantic (dependency) uniformity across all languages, and syntactic uniformity across subsets of languages or particular aspects of languages.

2.2. Implications from the type of multilingual grammar engineering

Somewhat separately from the motivations for multilingual grammar engineering, the particular approach to grammar engineering employed has implications for the sharing of representations. This section will review the strategies outlined above and the types of sharing that are evident in the projects using the strategies. In many cases, it is hard to separate the effect of the approach to multilingual grammar engineering from the effect of the linguistic theory or framework adopted, but we hope that this tour is illuminating nonetheless.

To start with the extreme case of porting a grammar for one language to handle another, closely related language [e.g., Japanese to Korean (Kim et al., 2003), or Czech to Russian (Smrž, this volume)], a large part of the grammar will be shared. Smrž is working in a metagrammar framework (Smrž and Horak, 2000), in which the grammar engineer implements a set of constraints describing rules of the grammar which are then compiled out into the particular rules. In creating a metagrammar for Russian on the basis of one for Czech, Smrž adapts the existing metagrammar rules, keeping most of the analyses and representations intact.

In its parallel grammar development model, the ParGrarn project has developed a shared level of representation (LFG's *f*-structure) which represents relatively 'deep' syntactic representation (grammatical functions), some semantic information (tense and aspect), and some other syntactic information (e.g., case values), King et al, (this volume) describe the process by which the ParGram project develops and maintains this standard, The more semantic aspects of this shared representation support interchangeability with respect to downstream applications, while the more syntactic aspects facilitate the transfer of analyses across grammars both explicitly and by encouraging grammar engineers to reuse existing features rather than create new ones.

The core-grammar model of the LinGO Grammar Matrix emphasizes shared semantic representations and partially shared syntactic representations, while separating the two. In this context, Borthen and Haugereid (this volume) examines grammatical phenomena tied to the presumed cognitive status of the referents of NPs for the hearer. They propose a small set of features argued to be part of semantic (rather than syntactic or pragmatic contextual) representations which distinguish the different kinds of NPs. These features are tested in an implemented grammar for Norwegian in which they capture the distribution of light pronouns as well as of adjectives marked for 'specificity'. The features are further argued to be applicable for a related range of phenomena in English, Dutch, and Turkish. Matrix-based grammars for these languages should be able to adopt

Borthen's analysis relatively easily given the standardized syntax-semantics interface.

The systemic-functional approach of Bateman et al. (this volume) emphasizes cross-linguistic similarities and diffe ences at multiple levels of representation (lexicogrammar, semantics, and genre/register) as well as in the mappings between those levels. Consistent with the functional typological orientation of the work, no particular constraints or analyses are assumed to be shared across all languages. At the same time, any congruences between languages that are found can be represented, and the authors report greater degrees of congruence at the levels of discourse function and semantic hierarchies than in syntactic representations or the mappings between the levels. The methodology reported involves grammar porting on the one hand as new grammars are developed by adapting existing ones, and multilingual representations on the other, as all grammars can be represented within the same resource.

Finally, the work of Cahill et al. (this volume) on abstracting robust LFG-type grammars from treebanks has the potential to share *f*-structure representations across different languages in much the same way that the ParGram grammars do, although this is not directly emphasized. Depending on the degree of similarity between the languages in question and the information encoded in the available treebanks, it may also be possible to share strategies for producing *f* structures on the basis of shallow parses provided by treebank-derived grammars.

3. Conclusions

In this brief introduction, we have explored the motivations for multilingual grammar engineering, different possible approaches, and the implications from both for issues pertaining to shared representations. There seems to be a general tendency across projects to locate the main uniformity within semantic or dependency representations, and to allow more variation (albeit somewhat constrained variation) in the phrase structure representations. From a linguistic point of view, this is not very surprising: languages in general are believed to be very similar in their dependency relations while differing within some relatively constrained range of possibilities in their phrase structure.

Within this general agreement, however, we find that different approaches carve up the space differently. The Grammar Matrix approach aims to integrate constraints on phrase structure with relatively rich semantic constraints both of which will be valid cross-linguistically. The ParGram approach has led to significant advances by focusing on cross-linguistic uniformity at the level of f-structure, with less emphasis on commonality in constraints on phrase structure. In contrast, the metagrammar approach focuses on reusable constraints on phrase structure rather than on the linking with semantic representations. Finally, the systemic-functional approach eschews any hard cross-linguistic constraints while searching for the uniformities it can find, and finding those primarily at the levels of semantic and functional representations. As work progresses on each of these approaches, we expect to gain more clarity about the benefits and challenges presented by the distinct choices of shared representations, measured both by adequacy of the encoded linguistic hypotheses and by tractability in processing the resulting wide-coverage grammar implementations.

References

Baldwin T., Beavers J., Bender E. M., Flickinger D., Kim A., Oepen S., (2004) Beauty and the Beast: What Running a Brad-coverage Precision Grammar over the BNC, Taught us about the Grammar – and the Corpus. Paper presented at the International Conference on Linguistic Evidence: Empirical, Theoretical, and Computational Perspectives, Tubingen, Germany.

Bateman J. A., Kruijff-Korbayová I., Kruijff G.-J. (2005) Multilingual Resource Sharing Across Both Related and Unrelated Languages: An Implemented, Open-Source Framework for Practical Natural Language Generation. *Research on Language and Computation*, 3, pp. 191–219.

Bender E. M., Flickinger D., Oepen S.(0002) The Grammar Matrix: An Open-Source Starter-Kit for the Rapid Developme of Cross-Linguistically Consistent Broad-Coverage Precision Grammars. In Carroll J., Oostdijk N., Sutcliffe R. (eds), *Proceedings of the Workshop on Grammar Engineering and Evaluation at the 19th International Conference on Computational Linguistics,* Taipei, Taiwan pp, 81–14.

Butt M., Dyvik H., King T. H., Masuichi H., Rohrer C. (2002) The Parallel Grammar Project. In Carrol J., Oostdijk N., Sutcliffe R. (eds.), *Proceedings of the Workshop on Grammar Engineering and Evaluation at the 19th International Conference on Computational Linguistics*. Taipei, Taiwan, pp. 1–7.

Butt M., King T. H., Niño M.-E., Segond F. (1999) *A Grammar Writer's Cookbook.* CSLI, Stanford, CA.

Copestake A. (2002) *Implementing Typed Feature Structure Grammars*. CSLI Publications, Stanford, CA.

Copestake A., Lascarides A., Flickinger D. (2001) An Algebra for Semantic Construction in Constraint-based Grammars. In *Proceedings of the 39th Meeting of the Association for Computational Linguistics*. Toulouse, France.

Flickinger D., Bender E. M. (2003) Compositional Semantics in a Multilingual Grammar Resource. In Bender E. M., Flickinger D., Fouvry F., Siegel M. (eds.), *Proceedings of the ESSLLI 2003 Workshop "Ideas and Strategies for Multilingual Grammar Development"*. Vienna, Austria, pp. 33–40.

Kasper R. T. (1989) A Flexible Interface for Linking Applications to PENMAN's Sentence Generator. In *Proceedings of the DARPA Workshop on Speech and Natural Language.*

Kim R., Dalrymple M., Kaplan R. M., King T. H., Masuichi H., Ohkuma T. (2003) Multilingual Grammar Development via Grammar Porting. In Bender E. M., Flickinger D., Fouvry F., Siegel M. (eds.), *Proceedings of the ESSLLI 2003 Workshop "Ideas and Strategies for Multilingual Grammar Development"*. Vienna, Austria, pp. 49–56.

Oepen S., Bender E. M., Callmeier U., Flickinger D., Siegel M., (2002) Parallel Distributed Grammar Engineering for Practical Applications. In Carrol J., Oostdijk N., Sutcliffe R. (eds.), *Proceedings of the Workshop on Grammar Engineering and Evaluation at the 19th International Conference on Computational Linguistics.* Taipei, Taiwan pp. 15–21.

Oepen S., Dyvik H., Lønning J. T., Velldal E., Beermann D., Carroll J., Flickinger D., Hellan L., Johannessen J. B., Meurer P., Nordgård T., Rosén V. (2004) Som åKapp-ete Med Trollet? Towards MRS-Based Norwegian-English Machine Translation. In *Proceedings of the 10th International Conference on Theoretical and Methodological Issues in Machine Translation (TMI)*. Baltimore, MD.

Siegel M., Bender E. M. (2002) Efficient Deep Processing of Japanese. In *Proceedings of the 3rd Workshop on Asian Language Resources and Standardization at the 19th International Conference on Computational Linguistics.* Taipei, Taiwan.

Smrž P., Horak A. (2000) Large Scale Parsin of Czech. In *Proceedings of the Workshop on Efficiency in Large-Scale Parsing Systems, COLING 2000.* Saarbrücken, Universität des Saarlandes, pp. 43–50.

Uszkoreit H., Callmeier U., Eisele A., Schäfer U., Siegel M., Uszkoreit J. (2004) Hybrid Robust Deep and Shallow Semantic Processing for Creativity Support in Document Production. In *Proceedings of KONVENS 2004.* Vienna, Austria, pp. 209–216.

Emily M. Bender
Dan Flickinger
Frederik Fouvry
Melanie Siegel