Grammar Sharing Techniques for Rule-based Multilingual NLP Systems

Marianne Santaholma

University of Geneva, ETI/TIM/ISSCO

40, bd du Pont-d'Arve, CH-1211 Geneva

Marianne.Santaholma@eti.unige.ch

Abstract

Rule-based multilingual natural language processing (NLP) applications such as machine translation systems require the development of grammars for multiple languages. Grammar writing, however, is often a slow and laborious process. In this paper we describe a methodology for multilingual and multipurpose grammar development based on grammar sharing. This paper presents the first step towards a language independent core grammar used for recognition, analysis and generation of English, Japanese and Finnish used in a domain specific spoken language translation system. The paper focuses on the grammar architecture and rule writing principles. Evaluation on analysis and generation has shown that two thirds of the rules are shared between these three typologically different languages.

1 Background

Grammar is a central component of many natural language processing (NLP) applications including grammar checkers, rule-based machine translation, and speech recognition systems. It formally describes the structure of a language and the way in which linguistic units such as words are combined to produce sentences in the language (Abeillé, 2000). Hence they are essential for tasks like the analysis and generation of languages. NLP grammars differ from each other, among other things, in their coverage, in the grammar formalisms used and the linguistic theories on which they are based. NLP grammars can further be categorized on whether they are used for processing spoken or written language.

A grammar writer is often confronted by both linguistic and purely practical issues. Firstly, language is a complex system and the development of a grammar requires a solid theoretical base in order to capture all the phenomena of a language relevant to a particular type of application. From the point of view of implementation a grammar without a firm theoretical basis remains difficult to maintain and expand systematically. Additionally one bad design decision can have unforeseen consequences later in other parts of the grammar.

Due to the complex nature of languages grammar writing, and consequently grammar-based system development, is time consuming and expensive. This is naturally even more so the case when developing multilingual applications. A multilingual spoken language translation system might be regarded as one of the "worst" cases since it implies not only the development of grammars for multiple languages but also for multiple purposes: speech recognition, analysis and generation. For these practical reasons grammar-based methods are often complemented or even replaced by statistical methods. During the last decade the increased availability of data, including spoken data, in multiple languages has indeed favoured the development and use of such alternative methods. However, when necessary data is not available, as it is often the case with "minor" languages at the beginning of a project, grammar writing remains as the only realistic option. Additionally when reliable and more predictable results are prioritized over robustness,

Joakim Nivre, Heiki-Jaan Kaalep, Kadri Muischnek and Mare Koit (Eds.)
NODALIDA 2007 Conference Proceedings, pp. 253–260

linguistic rule based processing is usually preferred (for example in speech recognition Rayner et al, 2005).

Different approaches to lessen the development burden of grammars have been implemented. For the case of multilingual grammars these approaches include *domain specific grammar development*, *grammar adaptation* (also called '*grammar porting*') and *grammar sharing*. The first concerns grammars that cover a certain domain specific language, a sublanguage (Kittredge, 2003). Compared to the standard, general language, sublanguages make use of limited vocabulary, syntax and semantics. Given this narrower extent, it is possible to produce a relatively complete linguistic description of the sublanguage structures. Consequently these grammars are less ambiguous and they perform generally better compared to general grammars. Sublanguage grammars for several languages are quicker to develop; however, since the coverage of grammar remains significantly restricted, porting them to new domains can be labourious (Kittredge, 2003).

Instead of limiting the coverage, the second approach, grammar adaptation, reuses the information from an already existing grammar of a language in building a grammar for a new language. The existing grammar rules for the use in same application and preferably of a closely related language are adapted for this new language. This approach has been applied in different types of grammar formalisms and applications (among others by Alshawi et al. 1992; Kim et al. 2003; Santaholma, 2005) and it appears to represent a reduction of effort regardless of the languages used and the linguistic framework adopted. However, the approach still requires a separate grammar for each new language, and thus a fair amount of development work.

Grammar adaptation can be seen as an approach that exploits the common features of languages in NLP grammar development. This is taken further in the third approach, grammar sharing. Instead of just reusing the information of an existing grammar, the grammar rules are actually shared between different languages. This is motivated by the fact that languages are structured by similar underlying principles and hence languages share structure and properties at least to some extent (for an overview see Comrie, 1981; Croft, 1990). There is a vast amount of research done in the field of linguistic universals and language typologies comparing the properties of different languages, and the results of this research are exploited in NLP grammar development different ways by different grammar development projects, including ParGram (Butt et al., 2002) and Grammar Matrix (Bender et al., 2002).

To the best of our knowledge, the actual grammar sharing approach, where rules are directly shared between several languages, has only been implemented for closely related languages, such as Romance languages (Bouillon *et al*. 2006). Grammar sharing has both linguistic and practical advantages. Linguistically more coherent analyses are obtained when rules are written to be used for several different languages. On the practical, system development level the approach contributes to reuse of code and hence to the reduction in the number of rules linguists have to write. Furthermore modifications and debugging are carried out just on one grammar instead of several.

This paper presents a grammar architecture and rule writing principles for development of parameterized core grammar rules for languages that represent different types of languages: English, Finnish and Japanese. These rules are developed for a domain specific spoken language translation system and used for recognition, analysis and generation of these languages.

The rest of this paper is organised as follows. Section 2 briefly presents the Regulus toolkit used for grammar development, and the MedSLT system for which the language independent core grammar is implemented. Section 3 describes the grammar development principles with examples and section 4 summarizes the preliminary evaluation results. Section 5 presents our conclusions.

2 Tools and application

2.1 Regulus grammar development toolkit

The parameterized core grammar rules for English, Finnish and Japanese are implemented using the Regulus grammar development platform. Regulus is an Open Source toolkit for the development of unification grammars for spoken language (Rayner et al, 2006; Regulus, 2007). The main components include an environment for writing and debugging typed unification grammars, tools to support corpus-based specialization of general

grammars, and a compiler which is used to turn unification grammars into Context-free Grammar (CFG) language models that are often used in speech recognition.

This compilation into CFG models imposes certain restrictions on the possible grammar formalism that can be employed. Firstly only finite feature-value pairs are allowed in the grammar rules and secondly the features cannot take any complex values. Hence the theoretically stable grammar formalisms that provide detailed syntactic and semantic analysis like Head Phrase Structure Grammar (Pollard & Sag, 1994) and Lexical functional grammar (Kaplan & Bresnan, 1982) are difficult to implement in the context of current speech recognition systems. For more details of Regulus grammars see Rayner et al, 2006.

2.2 The application: Spoken language translation system – MedSLT

The core rules implemented on Regulus are used in the medical domain spoken language translation system, MedSLT (MedSLT, 2007; Bouillon et al, 2005), which is developed to translate doctor-patient examination dialogues. Typical dialogues consist of medical examination questions about the intensity, location, duration or quality of pain, factors that increase/decrease the pain, medical/therapeutic processes and family history of the patient. The syntactic coverage mainly consists of yes-no questions where the patient's response can either be affirmative or negative. Content wise coverage is divided in subdomains based on specific symptoms (for example, headaches or chest pain).

The desired features of this type of medical domain translator include reliability of translation, flexibility in use and rapid portability to new languages and medical domains. These requirements have significantly influenced the MedSLT architecture. To obtain the reliability of translations and flexibility in use, the basic architecture adopted in MedSLT is a compromise between fixed-phrase translation and rule-based linguistic methods complemented by statistical language modelling as backup (Bouillon et al, 2005). Despite its hybrid architecture the heart of the MedSLT is the linguistic Regulus grammars.

To overcome the common difficulties of multilingual grammar development discussed in the introduction, a number of solutions have been implemented in MedSLT. First of all, one single grammar of a language is automatically compiled by Regulus into the different formalisms needed in all the major components of the translator: speech recogniser, parser and generator. Another significant feature is that a general grammar of a language can be automatically specialized using Regulus for different domains (Rayner et al, 2006). In this way the system combines the advantages of general grammars (applicable in a wide range of domains) and domain specific grammars (less ambiguous).

The most significant drawback of the MedSLT approach, in terms of grammar development, remains, however, the laborious and time-consuming development of the general grammars. One solution to this problem is to share grammars between languages. This approach has been investigated by Bouillon et al. (2006) by developing parameterized rules for Romance languages French, Spanish and Catalan. Bouillon et al. (ibid) concluded that only few language specific rules were needed and that the recognition and generation results were equally good for all these three languages. We are extending this approach to non related languages. In the next section we describe the principles defined to develop these common rules.

3 MedSLT Regulus core grammar architecture and rule writing principles

3.1 Structure of grammar

The MedSLT Regulus core grammar consists of modules that form a three level inheritance structure. This is illustrated in Figure 1. "Language independent rules", the most generic level contains the parameterized language-independent rules that are stored in the "*Common core*" module. This module is shared with all the languages, and its information is inherited by all the lower levels (see section 3.2 for details).

The second level contains separate modules for different language families. According to language typology research one of the evident reasons for similarities between languages is that they are related (Comrie, 1981). In the core grammar we use this fact to reduce redundancy in rule writing. Hence, for example, the properties which are not common for all languages but typical for all Germanic languages, like English and Swedish, would be

stored in the "*Germanic languages*" module. Finally the language specific information is stored in language specific modules, i.e. in separate English, Finnish, and Japanese modules.

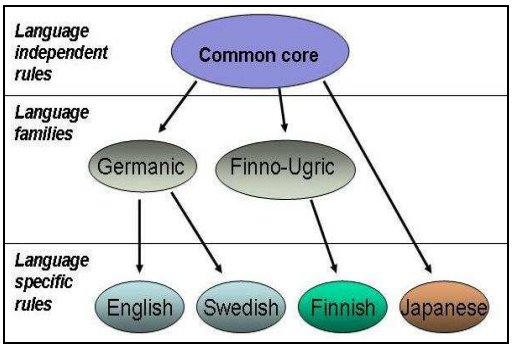

Figure 1: Structure of grammar

3.2 Rule writing principles

The methodology assumes two levels of syntactic representations: constituents and lists of feature-value pairs. In order to divide the constituents and feature-value pairs into the common and language specific modules, the linguistic phenomena necessary to express the concepts of MedSLT diagnosis questions (see Section 2.2) were first defined by analysing the MedSLT corpora of different languages. In a second step the structures used to express the extracted phenomena were compared between languages and the common properties for each structure were extracted. Some general rule writing principles were defined both for the constituents and the features. These principles are presented in the following.

3.2.1 Constituent level

Order of constituents. The basic order of constituents differs remarkably between languages. English and Finnish belong to SVO languages while Japanese is a verb final language, SOV. The first rule writing principle applied (at the constituent level) is that the order of constituents is expressed in a neutral way in the common rule set, "Common core". Instead of hard coding some specific constituent order, only the possible constituents are given, and the order itself is specified by language specific rules. The common rules are parameterized by macros as illustrated in the following.

Example 1 shows a simplified **vbar**[1] rule for phrases containing an auxiliary verb (**aux**) and a main verb (**verb**), for example "*The pain **has been** in the front of the head*". In Finnish and English the auxiliary precedes the main verb, while in Japanese the auxiliary (marker) is at the end of the sentence after the main verb. These constituents are given in the common rule: `vbar → aux verb.` Furthermore the macro `@vbar_aux_vbar` in the common rule points to language specific rules that define the order of these constituents in each language. (The macros follow the Regulus macro writing definition (Rayner et al., 2006)). In Finnish and English the macro points to the identical language specific rule `macro(vbar_aux_vbar(AuxV, V),(AuxV, V))` that defines that **auxiliary verb** (with semantic value **AuxV**) should precede the **Verb** (with the semantic value **V**) ***(AuxV, V)***. The Japanese rule expresses the reverse constituent order by ***(V, AuxV)***.

```
COMMON RULE
vbar:[sem=concat(AuxV, V),
vform=finite] -->
@vbar_aux_vbar(
        aux:[sem=AuxV, vform=finite,
participle_vform=Participle_form],
        verb:[sem=V,
vform=Participle_form]
        ).
*ENG + FIN* (aux+verb)
macro(vbar_aux_vbar(AuxV, V), (AuxV,
V)

*JAP* (verb+aux)
macro(vbar_aux_vbar(AuxV, V), (V,
AuxV)
```

Example 1. Constituent order

Variety of constituents. Besides the order of constituents also the variety of constituents varies between languages. Similarly to the constituent order, also the range of constituents is parameterized in the common rules by using macros. This is illustrated by Japanese and English/Finnish noun phrases (Example 2).

Particles (including case particles, topic particles, and postpositions) are very frequent in Japanese and they have various functions in Japanese syntax. In a noun phrase case parti-

[1] See Rayner et al., 2006 for detailed description of Regulus grammars.

cles are used to mark subcategorized verbal arguments for which English and Finnish apply other linguistic means (word order, inflectional case). Consequently Japanese requires a `case_particle` constituent that the other two languages do not. Hence the common noun phrase (**np**) rule in Example 2 is formed of a **nbar** and a macro `@case_particle`. As in the case of constituent order, a macro specifies the rule in different languages: in English and Finnish the macro `@case_particle` takes the value "empty" (**_**) and in Japanese the language specific rule introduces the particle constituent:
`particle:[sem=Particle, @noun_head_features(Head)].`

```
COMMON RULE

np:[sem=@np_nbar(Nbar, Particle),
@noun_head_features(Head)] -->

nbar:[sem=Nbar, @noun_head_features(Head)],
@case_particle(particle:[sem=Particle,
@noun_head_features(Head)], _).

*ENG + FIN* (empty)
Macro(case_particle(Yes, No), No)

*JAP* (constituent case_particle)
Macro(case_particle(Yes, No), Yes).
```

Example 2. Variety of constituents

3.2.2 Feature-value pairs

In Regulus grammars, as in other constraint-based grammars, the feature-value pairs encode the fine-grained information, e.g. about the number, person, subcategorization, and semantic categories. As both the required feature-value pairs and the values that the features can take differ between languages, they are parameterized in the language independent rules. The basic principal applied is that the features that differ between languages, like agreement, are generalised under the head feature macros. The head features are the features that are provided by the heads (like noun or verb) of the compositional grammatical constituents such as noun phrases and verb phrases. These are referred as head feature macro rules such as `@noun_head_features(Head)`. Furthermore these macros point to the language specific rules where the needed language specific features are defined. This can be illustrated by noun and verb head features that include, e.g. agreement features.

Agreement is a highly language specific system. In English subjects and predicates agree in person and number. Finnish has number, person, and case agreement between subject and predicate, and Japanese doesn't apply any of these agreement features. The rule in Example 3 shows a simplified declarative sentence rule. The sentence (**s**) consists of a noun phrase (**np**) and of a verbal phrase (**vp**) (`s -> np vp`). The agreement features are parameterized in the **np** by the macro `@noun_head_features(Head)` and in the **vp** by `@verb_head_features(Head)` that point to language specific information.

```
COMMON RULE

s:[sem=concat(Np, Vp),
@verb_type(Type)] -->
        np:[sem=Np,
sem_np_type=SemType,
@noun_head_features(Head)],
        vp:[sem=Vp,
subj_sem_np_type=SemType,
@verb_head_features(Head),
vform=finite, inversion=false,
@verb_type(Type)].
```

Example 3. Phrasal head features

In the case of Finnish, the **noun_head_features** macro evokes the language specific rule:
`macro(noun_head_features ([Number,Person,Case]), [number=Number, person=Person, case=Case]).`

This specifies that **noun_head_features** include the features called `number` (singular/plural), `person` (1/2/3) and `case` (inflectional case). The **verb_head_features** macro evokes a similar language specific rule:
`macro(verb_head_features( [Number,Person,Case]), [number=Number, person=Person, subj_n_case=Case]).`

In English the `case` and `subj_n_case` features are ignored by language specific declarations. The language specific macros corresponding to the Finnish ones are

```
macro(noun_head_features
([Number,Person]),
[nuber=Number,person=Person])
```
and
```
macro(verb_head_features
([Number,Person]),
[number=Number,person=Person]).
```

In Japanese all three features are ignored. The examples 4, 5 and 6 show the Finnish, English and Japanese declarative sentence rules after these language specific features have been applied.

```
s:[sem=concat(Np, Vp),
@verb_type(Type)] -->
np:[sem=Np, sem_np_type=SemType, person=Person, number=Number, case=Case],
vp:[sem=Vp, subj_sem_np_type=
SemType, inversion=false, person=Person, number=Number,
subj_n_case=Case, vform=finite,
@verb_type(Type)].
```

Example 4. Agreement in Finnish

```
s:[sem=concat(Np, Vp),
@verb_type(Type)] -->
        np:[sem=Np,
sem_np_type=SemType, person=Person,
number=Number],
        vp:[sem=Vp,
subj_sem_np_type=SemType, inver-
sion=false, person=Person, num-
ber=Number, vform=finite,
@verb_type(Type)].
```

Example 5. Agreement in English

```
s:[sem=concat(Np, Vp),
@verb_type(Type)] -->
        np:[sem=Np,
sem_np_type=SemType],
        vp:[sem=Vp,
subj_sem_np_type=SemType, inver-
sion=false, vform=finite,
@verb_type(Type)].
```

Example 6. Agreement in Japanese

4 Evaluation

To evaluate the common parameterised rules the MedSLT core grammar was tested on analysis and generation of MedSLT English, Finnish and Japanese examination questions. Instead of a large lexical coverage, the focus was on the covered linguistic phenomena. The test corpora contained MedSLT sentences including the variety of phenomena presented in Table 1. The aim of the evaluation was to find out how many rules were shared with all three languages and how many language specific rules were necessary in order to analyse and generate the corpora for each language.

To cover these phenomena in English, Finnish and Japanese total of 65 rules were written[2]. The number of rules used per language varies: English uses 54 out of 65 rules, including 3 language specific rules, Finnish uses 56 of 65 and has no language specific rules. Japanese makes use of 51 common rules and has 5 language specific rules.

Covered phenomena	
Sentence types	declarative, yn-question, wh-question, subordinate "when" clause
Tenses	present, past(imperfect), present perfect, past perfect
Aspects	Continuous
Verb subcategorisation	transitive, intransitive, predicative (be+adj), existential (there+be+np),
Determiners	article, number, quantifier
Adpositional modifiers	prepositional, postpositional
Adverbial modifiers	verb modifying and sentence modifying adverbs, comparison
Pronouns	personal, possessive, dummy pronouns
Adjective modifiers	predicative, attributive, comparison

Table 1. Covered phenomena

As Table 2 summarizes, two thirds (43/65) of parameterized rules are used by all three languages. Additionally in total 22 % of rules

[2] This number includes the language independent parameterized rules and languages specific rules.

are used for two languages. Only 12% of rules are strictly language specific.

Languages	No of rules	%
ENG + FIN + JAP	43	66
FIN + JAP	7	22
ENG + FIN	6	
ENG + JAP	1	
ENG	3	12
FIN	0	
JAP	5	
TOTAL	**65**	**100**

Table 2. Rules summarized

Furthermore, the grammar has altogether 57 declared features, 30 of them are common for all three languages. English ignores 13 features, Finnish 15 and Japanese doesn't make use of 19 of total 57 features. The used features vary significantly between languages depending on the typological character of language. Important features, like different case-features in Finnish, are ignored in English, while Japanese omits features including the agreement features like number and person that are significant in English and Finnish.

Based on the above presented figures we can conclude that the defined grammar architecture and rule writing principles captures the cross linguistic similarities and variations efficiently both on constituent and feature-value level.

5 Conclusion

This paper has presented a methodology for more efficient multipurpose and multilingual grammar development for typologically different languages based on rule sharing. The common parameterized rules were developed and tested on English, Finnish and Japanese on medical sublanguage. Evaluation showed that two thirds of rules were shared by all languages when parsing and generating the MedSLT medical examination questions.

We have shown that the defined grammar architecture that a) has a modular structure (a language independent module and language specific modules) and that b) assumes two levels of syntactic representations (constituents and feature-value pairs) that are both parameterized and generalized in the common rule level, captures efficiently the similarities and differences of typologically different languages.

Acknowledgements

I would like to thank Pierrette Bouillon and Manny Rayner for their advice. The MedSLT system is developed at TIM/ISSCO, University of Geneva, and funded by Swiss National Science Foundation.

References

Abeillé A. 1993. Les nouvelles syntaxes: grammaires d'unification et analyse du français. A. Colin, Paris.

Alshavi H (ed). 1992. *The core language engine*. Cambridge, Massachusetts: the MIT press.

Bender E, Flickinger D, Oepen S. 2002. The Grammar Matrix. An Open Source Starter-Kit fort he Rapid Development of Cross-Linguistically Consistent Broad-Coverage Precision Grammars. In *Proceedings of the 19th International Conference on Computational Linguistics*, Taipei, Taiwan, p. 8-14.2002.

Bouillon B, Rayner M, Chatzichrisafis N, Hockey B A, Santaholma M, Starlander M, Isahara H, Kanzaki K, Nakao Y. 2005. A Generic Multi-Lingual Open Source Platform for Limited-Domain Medical Speech Translation. In *Proceedings of EAMT 2005*, Budapest, Hungary, pp. 50-58.

Bouillon P, Rayner M, Novellas Vall ., Nakao Y, Santaholma M, Starlander M, Chatzichrisafis N (2006). Une grammaire multilingue partagée pour la traduction automatique de la parole. In *Proceedings of Traitement Automatique des Langues Naturelles*, 10 - 13 avril 2006, Leuven, Belgique.

Butt M, Dyvik H, King T H, Masuichi H, and Rohrer C. 2002. The Parallel Grammar Project. In *Proceedings of COLING-2002 Workshop on Grammar Engineering and Evaluation*. pp. 1-7.

Comrie B. 1981. *Language Universals and Linguistic Typology*. University of Chicago Press, USA.

Croft W. 1990. *Typology and universals*. Cambridge University Press.

Kaplan R & Bresnan J. 1982. Lexical-Functional Grammar: A formal system for grammatical representation. In Joan Bresnan (editor), *The Mental Representation of Grammatical Relations*, pp. 173-281. Cambridge, MA: The MIT Press.

Kim R, Dalrymple M, Kaplan R, King T H, Masuichi H, Ohkuma T. 2003. Multilingual Grammar development via Grammar Porting. In *Proceedings of the ESSLLI 2003 Workshop on Ideas and Strategies for Multilingual Grammar Development.* Vienna, Austria, pp.49-56.

Kittredge R. 2003. Sublanguages and controlled languages. In *Mitkov, Ruslan (ed.), The Oxford Handbook of Computational Linguistics*, pp. 430-447. Oxford University Press, Oxford.

MEDSLT 2007. https://sourceforge.net/projects/medslt/. As of 15 Mars 2007.

Pollard C & Sag I. 1994. *Head-Driven Phrase Structure Grammar*. University of Chicago Press, Chicago.

Rayner M, Bouillon B, Chatzichrisafis N, Hockey B A, Santaholma M, Starlander M, Isahara H, Kanzaki K, Nakao Y. 2005. A Methodology for Comparing Grammar-Based and Robust Approaches to Speech Understanding. In *Proceedings of Eurospeech-Interspeech*, 4-8, September, 2005, Lisboa, Portugal.

Rayner M, Hockey B A. and Bouillon P. 2006. *Regulus. Putting Linguistics into Speech recognition*. Stanford University Center for the Study of language and information, Stanford, California.

REGULUS.2007.https://sourceforge.net/projects/regulus/. As of 15 Mars 2007.

Santaholma M. 2005. Linguistic representation of Finnish in a limited domain speech-to-speech translation system. In *Proceedings of the 10th Conference on European Association of Machine Translation, 2005*, Budapest, Hungary, pp. 226-234.