Multilingual Grammar Development via Grammar Porting

Roger Kim
Palo Alto Research Center
3333 Coyote Hill Rd.
Palo Alto, CA 94304 USA
rkim@parc.com

Mary Dalrymple
Dept. of Computer Science
King's College London
Strand, London WC2R 2LS UK
mary@dcs.kcl.ac.uk

Ronald M. Kaplan
Palo Alto Research Center
3333 Coyote Hill Rd.
Palo Alto, CA 94304 USA
kaplan@parc.com

Tracy Holloway King
Palo Alto Research Center
3333 Coyote Hill Rd.
Palo Alto, CA 94304 USA
thking@parc.com

Hiroshi Masuichi
Corporate Research Center
Fuji Xerox Co.
Kanagawa 259-0157 JAPAN
hiroshi.masuichi@fujixerox.co.jp

Tomoko Ohkuma
Corporate Research Center
Fuji Xerox Co.
Kanagawa 259-0157 JAPAN
ohkuma.tomoko@fujixerox.co.jp

Abstract

In this paper, we investigate using an existing deep Lexical-Functional (LFG) grammar to develop a new one for a typologically similar language. In particular, we ported the Japanese ParGram grammar to Korean with promising results after only two months.

1 Introduction

The Parallel Grammar project (ParGram) is an international collaboration aimed at producing broad-coverage computational grammars for a variety of languages (Butt et al., 1999; Butt et al., 2002). The grammars (currently of English, French, German, Japanese, Norwegian, and Urdu) are written in the framework of Lexical Functional Grammar (LFG) (Kaplan and Bresnan, 1982; Dalrymple, 2001), and they are constructed using a common engineering and high-speed processing platform for LFG grammars, the XLE system (Maxwell and Kaplan, 1993). These grammars, as do all LFG grammars, assign two levels of syntactic representation to the sentences of a language: a surface phrase structure tree (called a *constituent structure* or c-structure) and an underlying matrix of features and values (the *functional structure* or f-structure). The c-structure records the order of words in a sentence and their hierarchical grouping into phrases. The f-structure encodes the grammatical functions, syntactic features, and predicate-argument relations conveyed by the sentence. F-structures are meant to encode a language universal level of analysis, allowing for cross-linguistic parallelism at this level of abstraction.

The ParGram project attempts to test the LFG formalism for its universality and coverage and to see how far parallelism can be maintained across languages; previous ParGram work (and much theoretical analysis) has largely confirmed the universality claims of LFG theory. The f-structures produced by the grammars for similar constructions in each language have the same major functions and features, with minor variations across languages (e.g., the f-structures for French nouns have a gender feature but that distinction is not marked in English f-structures). This uniformity has the computational advantage that the grammars can be used in similar applications and that machine translation (Frank, 1999) can be simplified.

We have found that it takes roughly two person-years of effort to construct for a new language a grammar that approximates existing grammars in terms of coverage and accuracy (see (Riezler et al., 2002) for a discussion of the coverage and accuracy of the current English grammar). This suggests that the deep-grammar construction task is not as difficult as many people have suggested, and indeed may require less effort than would be needed to produce training materials for automatic learning procedures for shallower grammars. Nonetheless, we are exploring methods for reducing the linguistic effort that grammar construction requires. The approach described here investigates the difficulty of converting a grammar of one language into a grammar of a typologically similar language. In this investigation, we started with the ParGram grammar of Japanese and used that as the basis for a grammar of Korean.

Typologically similar (but not necessary genetically related) languages are those that not only allow for similar f-structures (as LFG theory suggests is the case with all languages) but also have similar c-structure to f-structure mappings. Whether or not Japanese and Korean are genetically related (an issue that is in some dispute; see (Sohn, 1999) for some discussion), they are typologically similar in at least the following ways: they both are verb final, have relatively free word order, use postpositions to mark grammatical functions, and exhibit rampant pro drop.[1]

The creation of the current Japanese grammar (Masuichi and Ohkuma, 2003) involved two person-years of work at Fuji Xerox. The grammar has broad coverage, providing parses for 97% of sentences in a real-world test suite with good accuracy. Experiments done to test accuracy of the parses show that the ParGram Japanese grammar is comparable to standard Japanese dependency parsers; however, the ParGram grammar provides linguistically more detailed information than basic dependency relations. The Japanese grammar has 50 annotated phrase structure rules which compile out to finite-state machines with a total of 346 states, 1224 arcs, and 1702 disjuncts.

2 Porting Syntactic Rules

2.1 Direct Porting

The ParGram LFG grammars consist of phrase structure rules that are annotated with information about the corresponding f-structures. The creation of these annotated phrase structure rules forms the bulk of the effort in creating deep grammars for a new language. Thus, if we can port the annotated phrase structure rules for one language into the grammar of another, significant time can be saved. To do this, the two languages must be typologically similar, as is the case of Japanese and Korean.

The word order possibilities required no modification between the Japanese and Korean grammars. The basic verb final order of Japanese could be carried over to Korean, along with free ordering of preceding arguments and adjuncts, including markings for prefered word orders (e.g., subject preceding object). Sample orders covered by the grammars are shown in (1) and (2).

(1) a. Ayuko ga gakusei ni hon wo ageta.
Ayuko NOM student DAT book ACC gave
'Ayuko gave the student a book.' (Japanese)

b. gakusei ni hon wo Ayuko ga ageta.

c. hon wo Ayuko ga gakusei ni ageta.

(2) a. Myungwoni ga haksaeng ehgeh
Myungwoni NOM student DAT

chaek ul juuttda.
book ACC gave
'Myungwoni gave the student a book.' (Korean)

b. haksaeng ehgeh chaek ul Myungwoni ga juuttda.

c. chaek ul Myungwoni ga haksaeng ehgeh juuttda.

The structures for the Korean sentence in (2a) are shown in Figures 1 and 2. The corresponding Japanese structures for (1a) are shown in Figures 3 and 4. The structures are identical except for the lexical items (see section 3 on morphological differences). In the version of the Korean grammar shown here, the system works on Latin transliterated Korean, but it is currently being adapted to use Hangul script.

Similarly, the rules for topicalization could also be ported without modification. In Japanese, noun phrases are marked as topicalized by the postposition *ha*. Topicalized noun phrases may have certain postpositions before the final *ha*, as in (3a). However, nominals with postposition case markers such as *wo*, *ga*, or *no* cannot be topicalized by *ha*. Instead, the postposition is dropped and only

[1] See (Paik and Shirai, 2001) who exploit this similarity in machine translation.

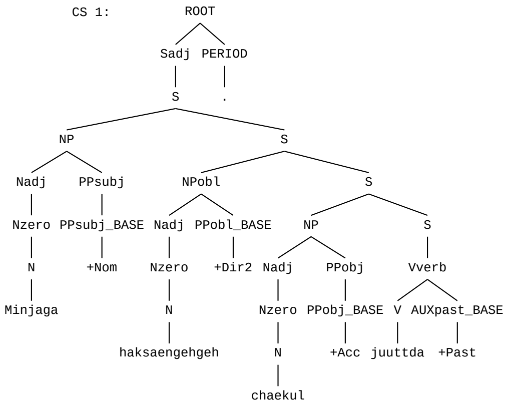
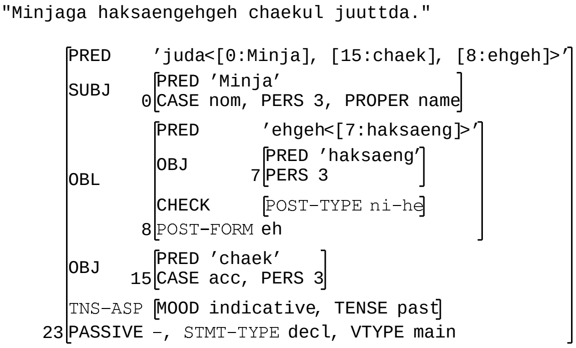


Figure 1: Korean c-structure for (2a)



```
"Minjaga haksaengehgeh chaekul juuttda."

   PRED      'juda<[0:Minja], [15:chaek], [8:ehgeh]>'
   SUBJ       PRED 'Minja'
            0 CASE nom, PERS 3, PROPER name
              PRED      'ehgeh<[7:haksaeng]>'
              OBJ         PRED 'haksaeng'
   OBL                  7 PERS 3
              CHECK       POST-TYPE ni-he
            8 POST-FORM eh
   OBJ        PRED 'chaek'
           15 CASE acc, PERS 3
   TNS-ASP   MOOD indicative, TENSE past
 23 PASSIVE -, STMT-TYPE decl, VTYPE main
```

Figure 2: Korean f-structure for (2a)

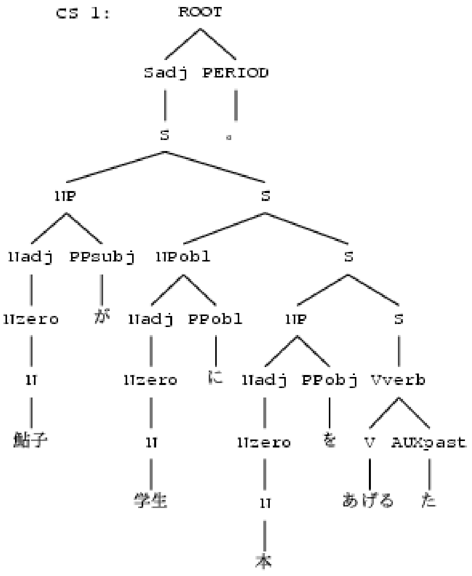
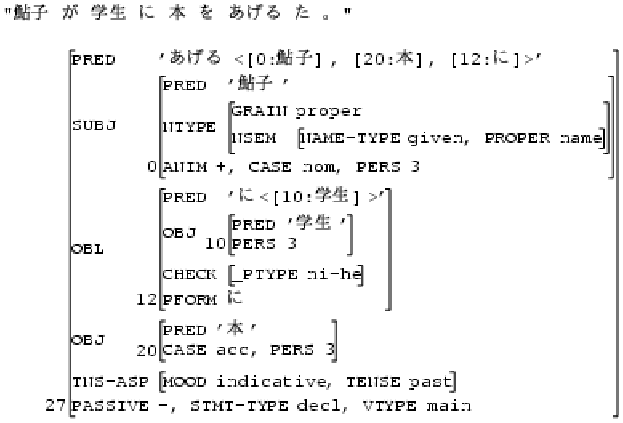


Figure 3: Japanese c-structure for (1a)



```
"鮎子 が 学生 に 本 を あげる た 。"

   PRED     'あげる<[0:鮎子], [20:本], [12:に]>'
             PRED  '鮎子'
   SUBJ      NTYPE  GRAIN proper
                    NSEM  NAME-TYPE given, PROPER name
           0 ANIM +, CASE nom, PERS 3
             PRED  'に<[10:学生]>'
             OBJ     PRED '学生'
   OBL            10 PERS 3
             CHECK  PTYPE ni-he
          12 PFORM に
   OBJ       PRED '本'
          20 CASE acc, PERS 3
   TNS-ASP  MOOD indicative, TENSE past
 27 PASSIVE -, STMT-TYPE decl, VTYPE main
```

Figure 4: Japanese f-structure for (1a)

ha appears, as in (3c). In addition, these phrases are marked in the f-structure as to their topic status; this f-structure information controls their syntactic distribution in the sentence. The corresponding topicalizing postposition in Korean is *un/nun*, with the allomorph *un* following vowel-final nominals and *nun* following consonant-final nominals, as in (3b). Just as with the Japanese *ha*, the Korean topicalizer also cannot cooccur with postpositions marking the basic grammatical functions, as in (3d).

(3) a. kinoo made ha
yesterday until TOPIC
'until yesterday (topic)' (Japanese)

b. uhjeh kaji nun
yesterday until TOPIC
'until yesterday (topic)' (Korean)

c. *kinoo ga ha ⟹ kinoo ha

d. *uhjeh ga nun ⟹ uhjeh nun

Nominal internal structure was also ported directly from the Japanese grammar to the Korean. This includes the analysis of adjectival, nominal, and postpositional modifiers of the head noun. For example, the rules used to produce the analysis for the Japanese complex nominal in (4a) were ported directly to produce the analysis of the Korean nominal in (4b).

(4) a. Ayuko-no ookii e hon
Ayuko-GEN big picture book
'Ayuko's big picture book' (Japanese)

b. Myungwoniui kun kurim chaek
Myungwoni-GEN big picture book
'Myungwoni's big picture book' (Korean)

Similarly, the rules building oblique noun phrases, i.e., noun phrases with postpositions that serve as oblique arguments of verbs, were ported directly. An example in Japanese is shown in (5a) with the corresponding Korean phrase in (5b).

(5) a. ooki ie ni
big house in
'in the big house' (Japanese)

b. kun jib eh
big house in
'in the big house' (Korean)

A rule fragment for these oblique noun phrases from the Japanese grammar is shown in (6). For illustrative purposes, we have shown a very simple rule; the annotations on most rules are significantly more complicated, which is why grammar porting is so desirable for bootstrapping grammar development.

(6) NPobl ⟶
{ Nadj: (^ OBJ)=!
PPobl: ^ =!
|AN: (^ OBJ)=!
PPobl: ^ =!
(!CHECK POST-TYPE)=c 'to-ni'
| ...}.

In the first disjunct in Rule (6) (from { to |), the NPobl consists of an Nadj which is the OBJ of the corresponding f-structure (Nadj: (^ OBJ)=!) followed by a PPobl postposition which is the head of the corresponding f-structure (PPobl: ^ =!).[2] The second disjunct (from | to |) is similar except that the PPobl is restricted to postpositions of the type `to-ni` and the OBJ of this postposition is an AN instead of the usual Nadj. Note that `to-ni` is a purely formal symbol whose spelling echos the surface forms of Japanese. The corresponding Korean form can be marked with this value to satisfy the constraint in this rule, or its spelling can be changed to make it more suggestive of the Korean realization. Other disjuncts are found in this rule, indicated here by |...}.

Other parts of the grammar that could be ported without change include the implementation of pro-drop for subjects and objects. Examples of sentences with pro-dropped subjects for Japanese and Korean are seen in (7). The analysis corresponding to the Korean sentence is shown in Figures 5 and 6.

(7) a. jitensha de ie ni kaeru.
bicycle by home to return
'(I/You/He/She/We/They) return home by bicycle.' (Japanese)

b. jajungu ro jib eh dorakanda.
bicycle by home to return
'(I/You/He/She/We/They) return home by bicycle.' (Korean)

Pro-drop is analyzed by optionally providing a null pronominal subject and/or object for each verb frame that subcategorizes for these functions. If an overt subject or object is found in the clause, then the pro-drop option is not chosen because the PRED of the overt subject would fail to unify with the PRED of the optional pronominal subject. However, if there is no overt subject, then the pro-drop option must be chosen because otherwise the subcategorization requirements of the verb would not be met. The null-anaphor template NA that provides the pronominal arguments is shown in (8a), where GF is a grammatical function value that is passed in by the verbal template. (8b) shows the expansion of the template for pro-dropped SUBJs.

(8) a. NA(GF) =
@(DEFAULT
(^ GF PRED) (^ GF PRON-TYPE) 'pro' null).

b. (^ SUBJ PRED)='pro'
(^ SUBJ PRON-TYPE)=null

[2] In the XLE grammar development platform, the ^ corresponds to the traditional LFG ↑ and the ! corresponds to the traditional LFG ↓.

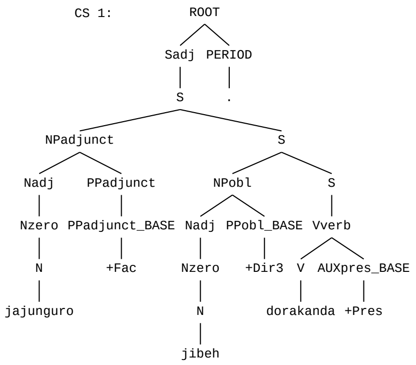
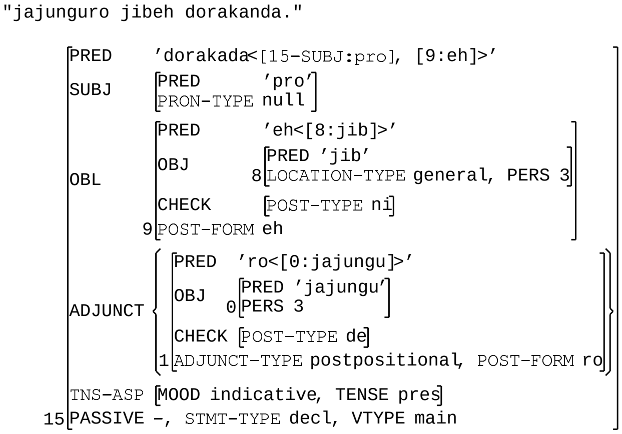


Figure 5: Pro-drop: Korean c-structure for (7b)



```
"jajunguro jibeh dorakanda."

     PRED     'dorakada<[15-SUBJ:pro], [9:eh]>'
     SUBJ     [PRED      'pro'
              [PRON-TYPE null ]
              [PRED      'eh<[8:jib]>'
              [OBJ        [PRED 'jib'
     OBL      [          8[LOCATION-TYPE general, PERS 3]
              [CHECK      [POST-TYPE ni]
             9[POST-FORM eh
                [PRED  'ro<[0:jajungu]>'
                [OBJ   [PRED 'jajungu'
     ADJUNCT {[      0[PERS 3
                [CHECK [POST-TYPE de]
               1[ADJUNCT-TYPE postpositional, POST-FORM ro
     TNS-ASP  [MOOD indicative, TENSE pres]
   15PASSIVE -, STMT-TYPE decl, VTYPE main
```

Figure 6: Pro-drop: Korean f-structure for (7b)

The ability to drop postpositional case and discourse function markers was also ported directly. In a standard SOV (or OSV) sentence, if the accusative case marker is dropped, but the nominative case marker is not, as in (9b), the sentence is given only one parse with the caseless noun phrase taking the object function and the nominative case marked noun phrase being the subject of the sentence. The same holds when only the nominative case marker is dropped but the accusative is not, as in (9c). When both case markings are dropped, as in (9d), the sentence is given two parses with each noun phrase being the subject in one parse and the object in the other. The Japanese equivalents of (9) receive the same analyses.

(9) a. Minjaga Taesunul boattda.
Minja-NOM Taesun-ACC saw
'Minja saw Taesun.' (Korean)

b. Minjaga Taesun boattda.

c. Minja Taesunul boattda.

d. Minja Taesun boattda.
'Minja saw Taesun.' (preferred due to default word order)

2.2 Modification of Rules

Although the typological similarities between the languages are striking, there are a few places where the annotated phrase structure rules had to be altered. The most important of these was in the analysis of sentential negation. Both Korean and Japanese have a type of affixal negation, shown in (10). However, Korean can encode negation as an adverb, similar to English, as in (11) (Kim, 2000). This construction is not found in Japanese and hence had to be added to the Korean grammar as part of the port.

(10) a. Taro ga modoranai.
Taro NOM return-NEG
'Taro isn't returning.' (Japanese)

b. Minja-nun doraka jianihanda.
Minja-TOP return-PRE -NEG
'Minja isn't returning.' (Korean)

(11) Minja-nun ani dorakanda.
Minja-TOP NEG return-PRES
'Minja isn't returning.' (Korean)

The structure for the Korean adverbial negation in (11) is shown in Figures 7 and 8. The annotated phrase-structure rules had to be modified to allow for an optional ADVneg in initial position in Vverb. The corresponding affixal negation construction in Japanese (10a) is shown in Figures 9 and 10 (Korean affixal negation in (10b) is identical). Note that although the c-structures differ between the languages, the f-structures are very similar with negation being an ADJUNCT to the main predicate and having ADJUNCT-TYPE neg.

Other syntactic differences between Korean and Japanese include the existence of double accusative constructions in Korean, as in (12), but not in Japanese.[3] This difference did not require any changes to the annotated c-structure rules since the restriction against double accusatives

[3] In contrast, double nominative constructions are found in both languages and hence were ported directly.

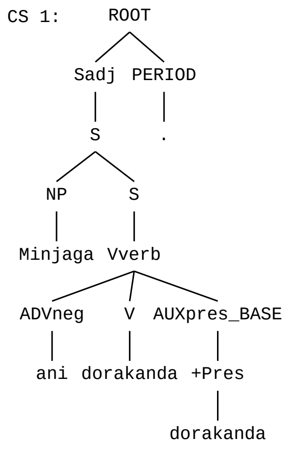
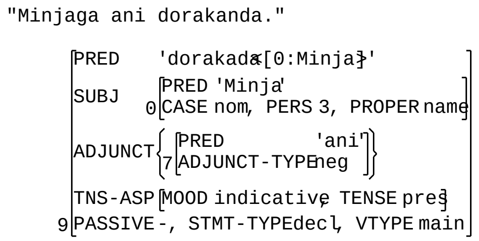


Figure 7: Adverbial negation: Korean c-structure



```
"Minjaga ani dorakanda."

 PRED     'dorakada<[0:Minja]>'
 SUBJ     [PRED 'Minja'
         0[CASE nom, PERS 3, PROPER name]
 ADJUNCT {  [PRED          'ani']
          7 [ADJUNCT-TYPE neg  ] }
 TNS-ASP  [MOOD indicative, TENSE pres]
9PASSIVE -, STMT-TYPE decl, VTYPE main
```

Figure 8: Adverbial negation: Korean f-structure

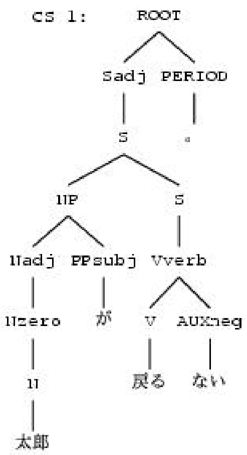
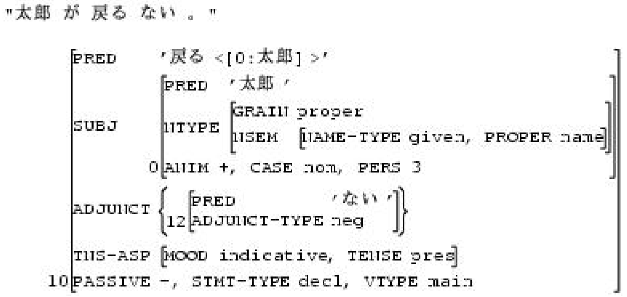


Figure 9: Affixal negation: Japanese c-structure



```
"太郎 が 戻る ない 。"

  PRED      '戻る <[0:太郎]>'
            [PRED  '太郎'
  SUBJ       NTYPE [GRAIN proper
                    NSEM  [NAME-TYPE given, PROPER name]]
           0 ANIM +, CASE nom, PERS 3]
  ADJUNCT  {   [PRED           'ない']
            12 [ADJUNCT-TYPE neg    ] }
  TNS-ASP  [MOOD indicative, TENSE pres]
10PASSIVE -, STMT-TYPE decl, VTYPE main
```

Figure 10: Affixal negation: Japanese f-structure

is not done on the phrase structure but rather in the subcategorization frames of the relevant verbs (see (O'Grady, 2002) for a detailed analysis of this construction).

(12) Kay-ka haksayng-ul tali-lul mwul-ess-ta.
dog-NOM student-ACC leg-ACC bit
'The dog bit the student on the leg.' (Korean)

In addition to the double accusative construction in (12) in which both accusatives are nominal arguments of the verb, there is another type involving complex predication, as in (13) (see (Lee, 1993) on complex predication in Korean).

(13) Minjaga badakul chungsorul hattda.
Minja-NOM floor-ACC clean-ACC do
'Minja cleaned the floor.' (Korean)

These are similar to the *suru* complex predicates found in Japanese except for the case marking. As such, we hope to extend the Japanese complex predicate analysis to Korean with only minor modification of the phrase structure rules to allow case marking within the complex predicate.

Finally, there are some differences between the languages in the nominal classifier system and in the multiple marking of quantification;[4] we have yet to explore these.

3 Porting Lexicons and Morphologies

Unlike the annotated c-structure rules, the lexical items differ significantly between Japanese and Korean. However, once the lexical item head

[4] We would like to thank an anonymous reviewer for bringing the multiple quantification to our attention.

words are changed, the information for many entries can remain the same. For example, the entry for the Japanese accusative postposition *o* is identical to that of the Korean accusative postposition *ul/rul* other than the form of the postposition, e.g., both assign accusative case. This was similar for the majority of closed and open class items. Thus, by using a Japanese to Korean dictionary to translate the head words in the lexicon, a detailed lexicon can be semi-automatically created for Korean.[5] However, at this point in our experimentation, we are still working with a small lexicon for open class items, although all the closed class items have been translated.

In addition to the lexicon, the tokenizer and morphology are done differently in the Japanese and Korean grammars. The Japanese grammar uses the ChaSen tokenizer (Asahara and Matsumoto, 2000) to insert token boundaries and determine certain part of speech information. Since the Korean writing system puts spaces between words, similar to English, a tokenizer was ported from the English grammar. The tokenizer inserts token boundaries between the space-delimited words and around punctuation. The resulting tokens are then fed into a finite-state morphology (FSM) for Korean (Beesley and Karttunen, 2003). In the current version of the grammar, this FSM works on Latin transliterated Korean like the rest of the grammar, but existing Korean FSMs can easily be incorporated into the grammar. Despite these differences between the Japanese and Korean morphologies, the same annotated phrase structure rules could be used. For example, as seen in Figure 3, the Japanese case markers *ga* and *o* are treated as separated words of category PPsubj and PPobj. Figure 1 shows that the Korean case markers are morphological tags +Nom and +Acc, but theses tags are of category PPsubj and PPobj and hence interact with syntactic rules just as in Japanese.

In addition to these lexicon and tokenization and morphology preprocessing steps, some minor changes to the core annotated phrase structure rules were needed in the domain of suffix syntax. For example, the Japanese grammar allows both orders for the location suffix in conjunction with the topic suffix. However, in Korean, the only possible order is for the location suffix to be followed by the topic suffix (e.g., *eh nun*); thus, the rule had to be further constrained for the Korean grammar.

4 Conclusion

We are encouraged by our success in this preliminary investigation into using an existing deep grammar to develop a new one for a typologically similar language. With only two person-months of effort, we found that major parts of the Japanese LFG grammar could be carried over unchanged into the Korean grammar. Most of the core annotated phrase structure rules remain the same, and it seems that many lexical items can be ported merely by changing the head-word of the Japanese entry to its Korean equivalent. New finite-state machines for tokenization and morphological analysis had to be created and incorporated into the system, as was to be expected.

Much work remains to be done to bring Korean coverage and accuracy up to the level of the Japanese and other ParGram grammars. This work will focus on peripheral syntactic rules and expansions to both the lexicon and morphology, with relatively little modification anticipated for the rules of major syntactic constructions. Based on our current rate of progress, we estimate that the Korean grammar will reach a level comparable to the current Japanese with a total of eight months of effort, about a third of the time we would have expected it to take to develop a Korean grammar from scratch. Of course, a final assessment of grammar-porting effectiveness will eventually also require systematic, corpus-based evaluations of coverage and accuracy.

We conclude from this limited experiment that porting LFG grammars across typologically similar languages is an effective method for bootstrapping multilingual grammar development. Outside the domain of LFG grammar development, our experience suggests that grammars written in other formalisms can also be used in similar grammar ports. For example, the Japanese HPSG grammar (Siegel and Bender, 2002) should lend itself to the rapid creation of a Korean grammar, especially if it exploits the concepts developed in the gram-

[5] There has been significant work on Korean/Japanese machine translation, e.g. (Paik and Shirai, 2001), including the development of lexical resources (Paik et al., 2001).

mar MATRIX project (Bender et al., 2002). Thus, we hope that the techniques described here will be exploited more generally in the development of broad-coverage deep grammars for a range of languages.

References

Masayuki Asahara and Yuji Matsumoto. 2000. Extended models and tools for high-performance part-of-speech tagger. In *Proceedings of International Conference on Computational Linguistics (COLING)*, pages 21–27.

Kenneth Beesley and Lauri Karttunen. 2003. *Finite-State Morphology: Xerox Tools and Techniques*. CSLI Publications, Stanford, California.

Emily 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 *Proceedings of the Workshop on Grammar Engineering and Evaluation*, pages 8–14. COLING 2002 workshop.

Miriam Butt, Tracy Holloway King, Maria-Eugenia Niño, and Frédérique Segond. 1999. *A Grammar Writer's Cookbook*. CSLI Publications.

Miriam Butt, Helge Dyvik, Tracy Holloway King, Hiroshi Masuichi, and Christian Rohrer. 2002. The parallel grammar project. In *COLING 2002: Workshop on Grammar Engineering and Evaluation*, pages 1–7.

Mary Dalrymple. 2001. *Lexical Functional Grammar*. Academic Press, New York. Syntax and Semantics, volume 34.

Anette Frank. 1999. From parallel grammar development towards machine translation. In *Proceedings of MT Summit VII*, pages 134–142.

Ronald Kaplan and Joan Bresnan. 1982. Lexical-functional grammar: A formal system for grammatical representation. In Joan Bresnan, editor, *The Mental Representation of Grammatical Relations*, pages 173–281. The MIT Press.

Jong Bok Kim. 2000. *The Grammar of Negation: A Constraint-based Approach*. CSLI Publications, Stanford, California.

Sookhee Lee. 1993. The syntax of serialization in Korean. In Patricia Clancy, editor, *Japanese/Korean Linguistics*, volume 2, pages 447–463. CSLI Publications, Stanford, California.

Hiroshi Masuichi and Tomoko Ohkuma. 2003. Constructing a practical Japanese parser based on Lexical-Functional Grammar. *Journal of Natural Language Processing*, 10(2):79–109. To appear; in Japanese.

John T. Maxwell, III and Ronald M. Kaplan. 1993. The interface between phrasal and functional constraints. *Computational Linguistics*, 19:571–589.

William O'Grady. 2002. Korean case: A computational approach. *Korean Linguistics*, 11:29–51.

Kyonghee Paik and Satoshi Shirai. 2001. Exploiting linguistic similarities for machine translation: A case study of Japanese-to-Korean. In *Proceedings of International Conference on Speech Processing*, pages 737–742.

Kyonghee Paik, Francis Bond, and Satoshi Shirai. 2001. Using multiple pivots to align Korean and Japanese lexical resources. In *NLPRS-2001*, pages 63–70.

Stefan Riezler, Tracy Holloway King, Ronald M. Kaplan, Richard Crouch, John T. Maxwell III, and Mark Johnson. 2002. Parsing the wall street journal using a lexical-functional grammar and discriminative estimation techniques. In *Proceedings of the ACL*.

Melanie Siegel and Emily Bender. 2002. Efficient deep parsing of Japanese. In *Proceedings of the 3rd Workshop on Asian Language Resources and International Standardization*, pages 31–38. COLING 2002 workshop.

Ho-min Sohn. 1999. *The Korean Language*. Cambrige University Press, Cambridge. Chapter 2.4.