Extracting Syntactic Features from a Korean Treebank

Jungyeul Park
UFR Linguistique
Laboratoire de linguistique formelle
Université Paris VII - Denis Diderot
jungyeul.park@linguist.jussieu.fr

Abstract

In this paper, we present a system which can extract syntactic feature structures from a Korean Treebank (*Sejong* Treebank) to develop a Feature-based Lexicalized Tree Adjoining Grammars.

1 Introduction

In a Tree Adjoining Grammar, a feature structure is associated with each node in an elementary tree (Vijay-Shanker and Joshi, 1991). This feature structure contains information about how the node interacts with other nodes in the tree. It consists of a top part, which generally contains information relating to the super-node, and a bottom part, which generally contains information relating to the sub-node.

In this paper, we present a system which can extract syntactic feature structures from a Treebank to develop a Feature-based Lexicalized Tree Adjoining Grammars. Several works have been on extracting grammars, especially using TAG formalism proposed. Chen (2001) has extracted lexicalized grammars from English Penn Treebank and there are other works based on Chen's procedure such as Nasr (2004) for French and Habash and Rambow (2004) for Arabic. Xia *et al*. (2000) developed the uniform method of a grammar extraction for English, Chinese and Korean. Neumann (2003) extracted Lexicalized Tree Grammars from English Penn Treebank for English and from NEGRA Treebank for German. However, none of these works have tried to extract syntactic features for FB-LTAG.

We use with *Sejong* Treebank (SJTree) which contains 32 054 *eojeol*s (the unity of segmentation in the Korean sentence), that is, 2 526 sentences. SJTree uses 43 part-of-speech tags and 55 syntactic tags (Sejong Project 2003).

2 Extracting a Feature structure for FB-LTAG

FB-LTAG grammars eventually use reduced tagset because FB-LTAG grammars contain their syntactic information in features structures. For example, NP_SBJ syntactic tag in LTAG is changed into NP and a syntactic feature <case=nominative> is added. Therefore, we use actually a 13 reduced tagset for FB-LTAG grammars compared with a 55 syntactic tagset for an LTAG without features. From full-scale syntactic tags which end with _SBJ (subject), _OBJ (object) and _CMP (attribute), we extract <case> features which describe argument structures in the sentence.

Alongside <case> features, we also extract <mode> and <tense> from morphological analyses in SJTree. Since however morphological analyses for verbal and adjectival endings in SJTree are simply divided into EP, EF and EC which mean non-final endings, final endings and conjunctive endings, respectively, <mode> and <tense> features are not extracted directly from SJTree. In this paper, we analyze 7 non-final endings (EP) and 77 final endings (EF) used in SJTree to extract automatically <mode> and <tense> features. In general, EF carries <mode> inflections, and EP carries <tense> inflections. Conjunctive endings (EC) are not concerned with <mode> and <tense> features and we only extract <ec> features with its string value. <ef> and <ep> features are also extracted with their string values. Some of non-final endings like *si* are extracted as <hor> features which have honorary meaning. In extracted FB-LTAG grammars, we present their lexical heads in a bare infinitive with morphological features such as <ep>, <ef> and <ec> which make correspond with its inflected forms.

Proceedings of the 8th International Workshop on Tree Adjoining Grammar and Related Formalisms, pages 133–136,
Sydney, July 2006. ©2006 Association for Computational Linguistics

<det> is another automatically extractable feature in SJTree and it is extracted from both syntactic tag and morphological analysis unlike other extracted features. For example, while <det=-> is extracted from dependant nouns which always need modifiers (extracted by morphological analyses), <det=+> is extracted from _MOD phrases (extracted by syntactic tags). From syntactic tag DP which contains MMs (determinative or demonstrative), <det=+> is also extracted. See Table 1 for all the extractable features from SJTree.

Feature	Description	Values
<case>	a case feature assigned by predicate	nom(inative), acc(usative), attr(ibut)
<det>	determiner, modifier	+/-
<mode>	mode	ind(icative), imp(erative), int(errogative), exc(lamatory)
<temps>	tense	pre(sent), past, fut(ure)
<ep>, <ef>, <ec>	a feature marked for different ways of instantiating mode and tense	string values like *eoss*, *da*, *go*, etc.
<hor>	honorific	+/-

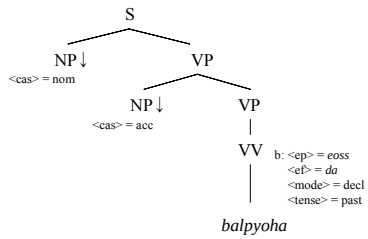
Table 1. Extractable Features from SJTree

Korean does not need features <person> or <number> as in English. Han *et al*. (2000) proposed several features for Korean FBLTAG which we do not use in this paper, such as <adv-pp>, <top> and <aux-pp> for nouns and <clause-type> for predicates. While postpositions are separated from *eojeol* during our grammar extraction procedure, Han *et al*. considered them as "one" inflectional morphology of noun phrase *eojeol*. <aux-pp> adds semantic meaning of auxiliary postpositions such as only, also etc. which we can not extract automatically from SJTree or other Korean Treebank corpora because syntactically annotated Treebank corpora generally do not contain such semantic information. <top> marks the presence or absence of a topic marker in Korean like *neun*, however topic markers are annotated like a subject in SJTree which means that only <case=nominative> is extracted for topic markers. <clause-type> indicates the type of the clause which has its values such as main, coord(inative), subordi(native), adnom(inal), nominal, aux-connect. Since the distinction of the type of the clause is very vague except main clause in Korea, we do not adopt this feature. Instead, <ef> is extracted if a clause type is a main clause and for <ec> is extracted for other types.

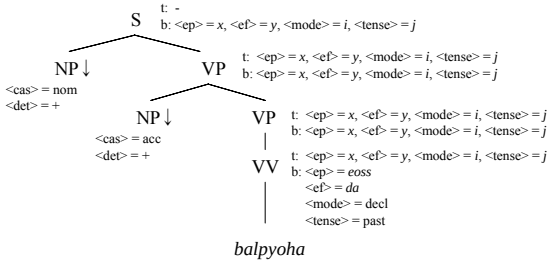
3 Experimentations

The actual procedure of feature extraction is implemented by two phases. In the first phase, we convert syntactic tags and morphological analysis into feature structure as explained above (see Table 2 for our conversion scheme for syntactic tags and see Table 3 for morphological analyses). In the second phase, we complete feature structure onto nodes of the "spine (path between root and anchor, node in an initial tree and path between root and foot node in an auxiliary tree)". For example, we put the same feature of VV bottom in Figure 1a onto VV top, VP top/bottom and S bottom because nodes in dorsal spine share certain number of feature of VV bottom. The initial tree for a verb *balpyoha.eoss.da* ('announced') in (1) is completed like Figure 1b for a FB-LTAG.

(1) 일본 외무성은 즉각 해명 성명을 발표했다.
ilbon *oimuseong.eun*
Japan ministy_of_foreign_affairs.Nom
jeukgak *haemyeng* *seongmyeng.eul*
immediately elucidation declaration.Acc
balpyo.ha.eoss.da
announce.Pass.Ter
'The ministry of foreign affairs in Japan immediately announced their elucidation'



a. First phase



b. Second phase

Figure 1. Extracted FB-LTAG grammar for *balpyoha.eoss.da* ('announced')

Table 4 shows the results of experiments in extracting feature-based lexicalized grammars. See Park (2006) for the detail extraction scheme.

4 Evaluations

Finally, extracted grammars are evaluated by its size (see Figure 2) and its coverage (see Table 5). The number of tree schemata is not stabilized at the end of the extraction process, which seems to indicate that the size of Treebank is not enough to reach the convergence of extracted grammars. However, the number of tree schemata appearing at least twice and three times (threshold = 2 and 3) in Treebank is much stabilized at the end of the extraction process than that of tree schemata appearing only once (threshold = 1).

The coverage of extracted grammars is calculated not only by the frequency of tree schemata but also by the number of tree schemata.

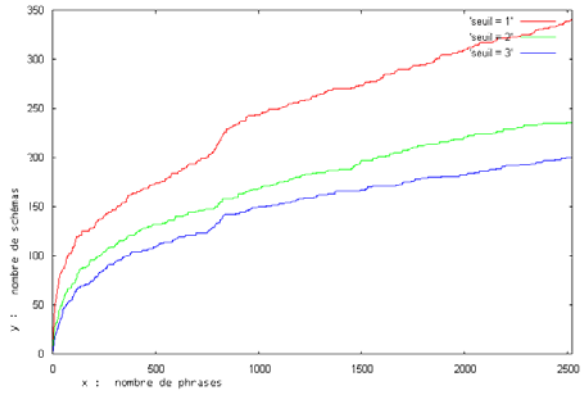


Figure 2. Size of tree schemata

We manually overlap our 163 tree schemata for predicates, which contain 14 subcategorization frames with 11 subcategorization frames of a FB-LTAG grammar proposed in Han *et al.* (2000) to evaluate the coverage of hand-crafted grammars [1]. Our extracted template grammars cover 72.7 % of their hand-crafted subcategorization frames[2].

[1] Our extracted tree schemata contain not only subcategorization frames but also some phenomena of syntactic variations, the number of lexicalized trees and the frequency information while Han *el al.* (2000) only presents subcategorization frames and some phenomena.

[2] Three subcategorization frames in Han *el al.* (2000) which contain prepositional phrases are not covered by our extracted tree schemata. Generally, prepositional phrases in SJTree are labeled with _AJT which is marked for adjunction operation. Since there is no difference between noun adverbial phrase and prepositional phrases in SJTree like [S *na.neun* [NP_AJT *ojeon.e* 'morning'] [NP_AJT *hakgyo.e* 'to school'] *ga.ss.da*] ('I went to school this morning'), we do not consider _AJT phrases as arguments.

5 Conclusion

In this paper, we have presented a system for automatic grammar extraction that produces feature-based lexicalized grammars from a Treebank. Also, we evaluated by its size and its coverage, and overlap our automatically extracted tree schemata from a Treebank with a manually written subcategorization frames to evaluate the coverage of hand-crafted grammars.

References

Alexis Nasr. 2004. *Analyse syntaxique probabiliste pour grammaires de dépendances extraites automatiquement*. Habilitation à diriger des recherches, Université Paris 7.

Chunghye Han, Juntae Yoon, Nari Kim, and Martha Palmer. 2000. *A Feature-Based Lexicalized Tree Adjoining Grammar for Korean*. IRCS Technical Report 00-04. University of Pennsylvania.

Fei Xia, Martha Palmer, and Aravind K. Joshi. 2000. A Uniform Method of Grammar Extraction and Its Application. In *The Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-2000)*, Hong Kong, Oct 7-8, 2000.

Günter Neumann. 2003. A Uniform Method for Automatically Extracting Stochastic Lexicalized Tree Grammar from Treebank and HPSG, In A. Abeillé (ed) *Treebanks: Building and Using Parsed Corpora*, Kluwer, Dordrecht.

John Chen. 2001. *Towards Efficient Statistical Parsing Using Lexicalized Grammatical Information*. Ph.D. thesis, University of Delaware.

Jungyeul Park. 2006. *Extraction automatique d'une grammaire d'arbres adjoints à partir d'un corpus arboré pour le coréen*. Ph.D. thesis, Université Paris 7.

K. Vijay-Shanker and Aravind K. Joshi. 1991. Unification Based Tree Adjoining Grammar, in J. Wedekind ed., *Unification-based Grammars*, MIT Press, Cambridge, Massachusetts.

Nizar Habash and Owen Rambow. 2004. Extracting a Tree Adjoining Grammar from the Penn Arabic Treebank. In *Proceedings of Traitement Automatique du Langage Naturel (TALN-04)*. Fez, Morocco, 2004.

Sejong Project. 2003. *Final Report of Sejong Korean Treebank*. Ministry of Education & Human Resources Development in Korea.

Anchor	Tree type	Syntactic tag	Node type	Conversion example
verb	α	NP_SBJ	subst	NP [<cas> = nom <det> = +]
verb	α|β	VP, VP_MOD	-	VP [<ep> <ef> <mode> <tense>]
anchored by _MOD phrase	β	NP|NP_CMP|NP_MOD |NP_OBJ||NP_SBJ	root	NP [<det> = +]
postposition	α	NP_SBJ	root	NP [<cas> = nom]
postposition	α	NP_SBJ	subst	NP [<cas> = NONE]

Table 2. Conversion example for syntactic tags

Verbal ending	Ending type	Conversion example
eoss	EP	<ep> = *eoss*, <tense> = past
si	EP	<ep> = *si*, <hor> = +
da	EF	<ef> = *da*, <mode> = ind

Table 3. Conversion example for morphological analyses

	# of lexicalized tree (α + β)	Average frequencies per lexicalized tree	# of tree schemata (α + β)	Average frequencies per tree schemata
G	12 239 (7 315 + 4 766)	3.26	338 (109 + 229)	118.1

Table 4. Results of experiments in extracting feature-based lexicalized grammars

	Coverage of grammars by the frequency of tree schemata			Coverage of grammars by the number of tree schemata		
Threshold	1	2	3	1	2	3
60 % of training set	60.75 %	60.7 %	60.66 %	81.66 %	83.83 %	83.5 %
90 % of training set	91.14 %	91.14 %	91.11 %	95.86 %	98.3 %	96.5 %

Table 5. Coverage of grammars: 60% of training set (1511 sentences) and 90% of training set (2265 sentences)