Analysis Techniques for Korean Sentences based on Lexical Functional Grammar

Deok Ho Yoon, Yung Taek Kim
Department of Computer Engineering
Seoul National University
Seoul, Korea

ABSTRACT

The Unification-based Grammars seem to be adequate for the analysis of agglutinative languages such as Korean, etc. In this paper, the merits of Lexical Functional Grammar is analyzed and the structure of Korean Syntactic Analyzer is described. Verbal complex category is used for the analysis of several linguistic phenomena and a new attribute of UNKNOWN is defined for the analysis of grammatical relations.

1. Introduction

In these days, various kinds of Unification-based Grammars are developed and widely researched[1,2]. Lexical Functional Grammar(LFG)[3,4] is one of them and seems to meet well for the grammatical characteristics of Korean.

We have developed a Korean natural language parser, KOSA(KOrean Syntactic Analyzer) which is based on the LFG. It is the analysis part of the KEMTS(Korean-English Machine Translation System) which is our current machine translation system.

In this chapter the grammatical characteristics of Korean and the merits of LFG formalism are presented.

1-1. The Grammatical Characteristics of Korean

Korean which is classified into the Ural-Altaic languages and belongs to the agglutinative languages is greatly different in the linguistic structures from the Indo-European languages such as English.

Korean adopts a short-clause as the unit of the spacing words. One short-clause is constructed by the concatenation of one or more morphemes of individual lexical categories. The concatenation is restricted by word conjoin conditions.

The most common patterns of short-clauses are 'verb(suffix)$^+$' and 'noun(postnoun)$^+$'. In such patterns, morphemes belonging to verb or noun bring the major informations. But because Korean is an agglutinative language, such morphemes have no conjugation and cannot have auxiliary informations freely. In Korean, such auxiliary informations are expressed by suffixes or postnouns which follow verb or noun, and their informations have an important role on the analysis of Korean[10].

Suffixes represent grammatical informations such as modality, tense, mood, voice, and etc. In Korean, agreement rules about gender, number or person are not developed well, but various idiomatic expressions of complex patterns are widely used.

The major function of the postnoun is to show the grammatical relation(GR) between an NP and a verb. Unlike the Indo-European languages in which the GR information is directly obtained from the structure of the sentence, in Korean postnoun tells the GR. So there is no need to distinguish NP and PP, and the order of NPs does not

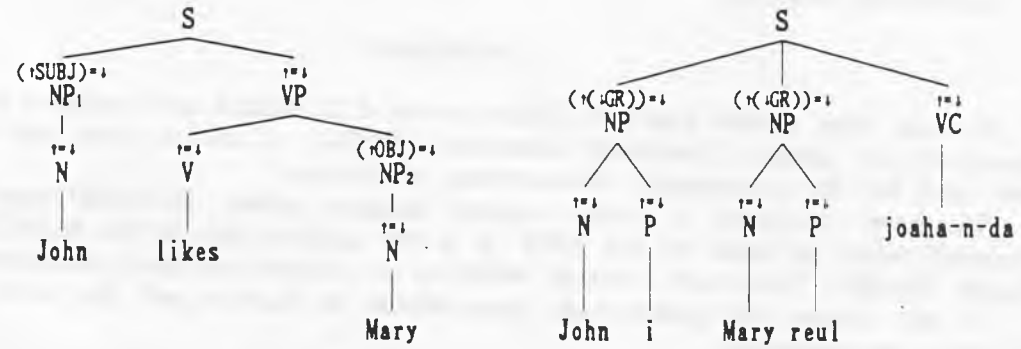
affect on the meaning. This brings on the relatively free word order of Korean.

When postnoun with other kind of information is used, the postnoun with the GR information is omitted frequently. To analyze such cases, inferences using various knowledges and heuristics are required.

1-2. The Merits of LFG for Korean Analysis

LFG has several merits for the analysis of Korean sentences. Some of them comes from the fact that Korean is not a well structured language.

The first merit is the fact that the primitives of LFG are the grammatical relations (GRs) such as SUBJ, OBJ, etc., but not the phrases such as NP, VP, etc. In English, the GRs of NPs can be detected from the order in the phrase tree. For example, we can see that NP_1 is the SUBJ of S and NP_2 is the OBJ of S from the c-structure for English in Fig.1-a, but this is not permitted for Korean as shown in Fig.1-b, because of the free word order of NPs. LFG offers a convinient way to analyze the implicit GRs, and more extended analysis methods will be proposed in chapter 4.



(a) Fig-1. GR of NPs in two C-structures (b)

The second merit is the fact that postnouns and suffixes in Korean can be easily and efficiently analyzed with lexical rules.

Also LFG provides convinience of invoking the inference mechanisms with grammatical devices and constraint conditions for various purposes such as the determination of UNKNOWN attributes.

In the design of KOSA, we tried to maximize such merits of LFG. Following chapters will describe the structure of KOSA and the techniques that we adopt.

2. The Structure of KOSA

Korean Syntactic Analyzer, KOSA is a Korean parser based on LFG. It analyzes a Korean sentence and extracts the grammatical informations in the form of an f-structure. The output of KOSA can be used in various applications. KOSA has developed as the analysis module of a Korean-English Machine Translation System, KEMTS and the output of KOSA is used as the intermediate structures for translation.

KOSA consists of three modules: LexAnal, CstrAnal and FstrAnal. Fig-2 shows the block diagram of KOSA. Each section describes the structure of each module.

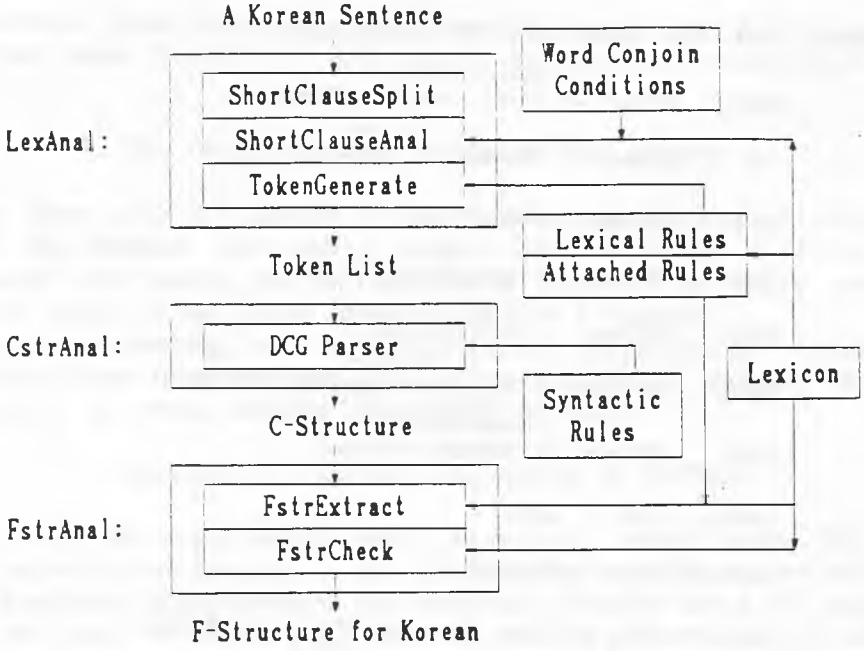


Fig-2. Block Diagram of KOSA

2-1. The Structure of LexAnal Module

LexAnal module analyzes a Korean sentence into the token strings and consists of three phases: ShortClauseSplit, ShortClauseAnal and TokenGenerate.

The ShortClauseSplit phase splits a Korean sentence into a number of short-clauses using blanks and punctuation symbols as the delimeters. This phase can be constructed easily as a simple finite state automata.

Each short-clause is analyzed into morphemes in the ShortClauseAnal phase. As shown in section 1-1, the concatenations of morphemes are restricted by the word conjoin conditions which check the lexical categories, the phonology and the semantics. Although the word conjoin conditions seem to be complicated, they are just simply some local rules which deal only adjacent morpheme pairs. So this phase can be implemented as an automata, too.

TokenGenerate phase generates the token strings from the morphemes. In this phase, some morpheme patterns are combined into one complex token. Among some kinds of complex tokens, verbal complex(VC) tokens are the most important. Typically a verb and its following suffixes are combined into one VC token. But there also exist more complex VC token types, and they are discusses in chapter 3. By generating complex tokens, many local linguistic phenomena can be excluded from the CstrAnal/FstrAnal modules. Because these modules analyze the global relationship among the sentence constituents, the approach of combining morphems can greatly enhance the efficiency. This phase is implemented as the recursive pattern rewriting rules.

2-2. The Structure of CstrAnal Module

The syntactic rules of the CstrAnal module are shown in Fig-3, and these rules are enough to analyze most Korean sentences. Complex tokens are dealt like the simple tokens according to their lexical categories. Each syntactic rule has functional schemata showing the method of unification. By adding these functional schemata to each branch

of the phrase trees, the c-structures are constructed.

```
                                (↑GR)=↓  ∈(↑ADJ)      =↓
(S1)     S[Type] -> ( NP    AVP )*  V[Type]

                                 ∈↑       ∈↑
(S2)     S[Type] -> S[connective] S[Type]

                                 =↓       =↓
(NP1)    NP[Type] -> N  P[Type]

                                      =↓          =↓
(NP2)    NP[Type] -> S[nominative] P[Type]

                                 ∈(↑ADJ)  =↓
(NP3)    NP[Type] -> ADJ  NP[Type]

                                          (↑GR)=↓          =↓
(NP4)    NP[Type] -> NP[possesive/conjunctive] NP[Type]

                                 ∈(↑XADJ)
                                (↑UNKNOWN)=↑  =↓
(NP5)    NP[Type] -> S[modify] NP[Type]

                                 =↓
(AVP1)   AVP -> ADV

                                 =↓
(AVP2)   AVP -> S[adverb]
```

Fig-3. The Syntactic Rules of KOSA

(S1) shows the structure of a simple sentence and (S2) shows the coordinative sentences. (NP1) and (NP2) show the basic structures of NPs and (NP3)-(NP5) show the constituents which can modify the NPs. With above rules, postnouns are combined with nouns(or nominal clauses) at the lowest level of the c-structure, but this has no problem because the postnouns supply only the auxiliary informations.

The unhierarchical syntactic rule (S1) makes the forms of c-structures flat and brings on much ambiguity especially on the position of NPs. So above rules examine context-sensitive constraints to decrease the ambiguity. The applications of rules are restricted by the context-sensitive informations in the bracket. But this approach is not enough to prohibit the ambiguity of NP's position. To resolve such ambiguity, the possibility for the unification of f-structures should be examined.

This module is implemented with the DCG(Definite Clause Grammar) parser[5] on PROLOG.

2-3. The Structure of FstrAnal Module

The FstrAnal module consists of two phases: FstrExtract and FstrCheck.

Because CstrAnal module results much ambiguity, FstrAnal module should cover the task of filtering out illegal c-structures as well as the task of analyzing the f-structures. Two phases of this module, will function as a two-level filter and generate the result f-structures from correct c-structures only.

FstrExtract phase extracts the f-structures of the input sentence from the c-structures by the bottom-up unification algorithm[3,6]. The complexity of the unification algorithm in KOSA is not heavy, and is the level of general unification algorithm for LFG formalism. Even though the grammatical characteristics of Korean are not reflected well by the unification algorithm, they are reflected through the lexicon informations and the functional schemata shown in section 2. Attached rules are used to extract the functional schemata for the verbal complex tokens in this phase. Chapter 3 will describe the functions of the attached rules.

FstrCheck phase examines the extracted f-structures whether they are grammatical or not. Grammatical devices and constraint conditions of LFG are utilized for KOSA, but some constraint conditions are modified and extended in order to solve Korean

linguistic phenomena. Some heuristics to the determine the unknown GR values of NPs are used in this phase. Section 4-2 will describe the modifications/extensions and the heuristics.

3. The Introduction and Usage of VC category

In English, there is the VP category which consists of all sentence constituents except the subject of the sentence. But such a category can't be found in Korean because of the free word order among the NP constituents including the subject constituents. So Korean verb seems to be directly governed by the S category.

Verbs are ususally combined with suffixes or another morphemes into complex tokens in TokenGenerate phase. In this chapter, various usages of the VC category which means the lexical category of verbal complex tokens will be shown.

3-1. Analysis of Auxiliary Informations in Suffixes

In Koean, there are many suffixes with complex and various usages. But most of them does not affect on the meaning of the verb supplying only the auxiliry informations. So when the FstrExtract phase extracts the functional schemata for a VC token which consists of a verb and its following suffixes, the auxiliary informations of suffixes are appended to the functional schemata of the verb.

For example, Korean word '*meok-eot-da*' means 'ate'. '*meok*' is a verb which means 'eat', '*eot*' is a past-tense suffix, and '*da*' is a ending suffix for descriptive sentences. The FstrExtract phase appends these informations from lexicon like below.

```
vc([v(meok),f(eot,tense),f(da,final)]) :
    (↑PRED) = 'EAT<(↑SUBJ)(↑OBJ)>'
    (↑TENSE) = PAST
    (↑MODE) = DESC
```

3-2. Analysis of Idiomatic Expressions

Koean has many idiomatic expressions on the predicate part. If idiomatic expressions are analyzed in CstrAnal/FstrAnal modules, the c-structures and the functional schemata can become much more complicated. So KOSA combines each idiomatic expression into one VC token in TokenGenerate phase, and obtains their functional schemata from the attached rules in FstrExtract phase. This approach greatly diminishes the overhead of CstrAnal and FstrAnal modules.

For example, a Korean idiomatic predicate '*meok-eul soo eop-da*' consists of three short-clauses and five morphemes. It means 'cannot eat', and can be thought as 'eat' with auxiliary information of negative possibility. So KOSA, combines this expression into one VC token and the attached rule adds the functional schemata, (↑POSSIVILITY)= '-' to those from lexicon. Below is the result token and functional schemata.

```
vc([v(meok),f(eul,modify),n(soo),v(eop),f(da,final)]) :
    (↑PRED) = 'EAT<(↑SUBJ)(↑OBJ)>'
    (↑MODE) = DESC
    (↑POSSIBILITY) = '-'
```

3-3. Analysis of Duplicated Constituents Expressions

Some Korean sentences have duplicated subjects or duplicated objects. This phenomenon is called as duplicated constituents problem, and KOSA analyzes the typical case of this problem using VC category.

For example, in Korean '*Cheolsoo-ga ki-ga keu-da*' means 'Cheolsoo is tall'. Because

postnoun '*ga*' is a subject marker, there exist two subjects '*Cheolsoo-ga*' and '*ki-ga*'. As '*ki*' means 'height' and '*keu*' means 'big', '*kei-ga keu*' means 'be tall'. In Korean, the verb, '*ki-keu*' which means 'be tall' is also used. Like this, many Korean adjective verbs are often expressed in the form of a subject and following simple adjective verb. So KOSA combines '*ki-ga keu-da*' into one VC token, and the attached rule interprets it just like '*ki-keu-da*'. Similar method is applied to verbs which require duplicated objects.

3-4. Analysis of Passive/Causative Expressions

In Korean, passive/causative expressions are all represented using suffixes. For example, '*meok-hi-da*' means 'be eaten', and '*hi*' is a suffix showing passiveness. Similarly '*meok-i-da*' means 'let ... eat', and '*i*' is a suffix showing causativeness.

KOSA combines such an expression into one VC token, and obtains the functional schemata for this token using the methods proposed by Kaplan[7,8].

For '*meok-hi-da*' and '*meok-i-da*', the attached rule for passiveness/causativeness transforms the functional schemata of '*meok-da*' like below.

```
vc([v(meok),f(da,final)]) :                  => vc([v(meok),f(hi,pass),f(da,final)]) :
    (↑PRED) = 'EAT<(↑SUBJ)(↑OBJ)>'                  (↑PRED) = 'EAT<(↑OBL_AGT)(↑SUBJ)>'
    (↑MODE) = DESC                                   (↑MODE) = DESC

vc([v(meok),f(da,final)]) :                  => vc([v(meok),f(i,cause),f(da,final)]) :
    (↑PRED) = 'EAT<(↑SUBJ)(↑OBJ)>'                   (↑PRED) = 'LET<(↑SUBL)(↑OBJ2)(↑XCOMP)>(↑OBJ)'
    (↑MODE) = DESC                                   (↑XCOMP PRED) = 'EAT<(↑SUBJ)(↑OBJ)>'
                                                     (↑XCOMP SUBJ) = (↑OBJ2)
                                                     (↑XCOMP OBJ) = (↑OBJ)
                                                     (↑MODE) = DESC
```

4. Determination Techniques of Grammatical Relations

The GR of Korean NPs are mainly determined by the postnouns. The GR value of P is transmitted by '↑=↓' to the NP, and indirectly used by '(↑(↓GR))=↓'[9].

But sometimes the GRs of NPs cannot be determined by the postnouns for two reasons. One reason is the omission of the postnoun showing the GR value. Another reason comes from the relation between the relative clauses and the antecedents.(Relative clause precede its antecedent, in Korean.) Here the antecedent has a role as an NP in the relative clause. But the postnoun of the antecedent shows only the GR for main clause, and the GR for relative clause is unknown.

Even in such cases, we should find the hidden GRs for correct analysis. This chapter describes the determination techniques of such unknown GRs.

4-1. Introduction of UNKNOWN Attributes

Because the heuristics to determine the unknown GR value should refer to the global relations among the VC and another NPs, the f-structure of the sentence should be able to be extracted before the heuristics are invoked. So we have introduced the UNKNOWN attribute to represent the temporary GR values. It is inserted and used during the FstrExtract phase, and changed to the correct GR value by the heuristics in FstrCheck phase.

The UNKNOWN is inserted by two methods. When the postnoun showing the GR value is omitted, the 'null' postnoun whose lexicon information has the functional schemata, '(↑GR)=UNKNOWN' is inserted in TokenGenerate phase. By the functional schemata, UNKNOWN becomes the attribute representing the NP whose GR is unknown. For the relative clause, syntactic rule (NP5) in section 2-2 inserts the UNKNOWN attribute whose value is the f-structure for the antecedent to the relative clause.

4-2. Extension of Constraints for UNKNOWN

There are several grammatical devices and constraint conditions in LFG, but some of them are used in modified or extended forms for the effective use of UNKNOWNs.

Because Korean sentences can have multiple NPs with unknown GR values, f-structure with multiple UNKNOWN attributes should be permitted and the consistency constraint should be relaxed. KOSA has solved this problem without any change of the unification algorithm by attaching index numbers to the UNKNOWN attributes as $UNKNOWN_1$, $UNKNOWN_2$,... when they are inserted.

The completeness/coherence constraint sould be extended for sentences with multiple UNKNOWNs. This extension is similar to that stated in [8], but the number of UNKNOWN attributes can be more than one here. So the extended completeness/ coherence constraint is as following: The number of UNKNOWN attributes should be less than or equal to the number of unsaturated grammatical functions of the PRED value for the intermediate f-structures, and should be equal for the final f-structures.

4-3. Heuristics for GR-Determination of the UNKNOWNs

For the complete analysis, the hidden GR values of the UNKNOWN attributes should be determined. KOSA uses three heuristics to determine them.

First is the simple mapping method. If there is only one UNKNOWN attribute in an f-structure and one unsaturated grammatical function, the GR value of the UNKNOWN is determined as the unsaturated grammatical function.

If the number of the UNKNOWN attributes is N(more than one), there should be also N unsaturated grammatical functions. Then they can be matched in N! different ways. To select the most proper mapping, two heuristics are used.

One heuristic is the agreement-point comparison method. The lexicon informations for nouns contain the semantic markers. They are transmitted to the values of UNKNOWN attributes. Each unsaturated grammatical function has the agreement-point information for each semantic feature on range [-1.0..1.0]. This is also given from the lexicon. For each mapping, the sum of agreement-points is calculated and the mapping of the highest score is selected. Because the number N is not so large, this heuristic does not bring a heavy overhead on examination.

The other heuristic is used when the agreement-points of several mappings are tied at the highest. Although NPs have almost free order in Korean, we can find the common word orders among them. The orders are not indispensable, but usual sentences follow them. So we can use these common word orders to determine the GR values of the UNKNOWNs. To find the order between the NPs without referring to the c-structure, we can utilize the index number attached to the UNKNOWNs.

5. Sentence Analysis Example of KOSA

In this chapter, the analysis steps of KOSA will be illustrated for following example. The first line of the example is the real Korean input, the second line is the input sentence written in Roman alphabet, the third line shows the meanings of morphemes belonging to the noun or verb category, and the last line shows the meaning of input sentence. For easy understanding, we replaced the Korean characters with Italic-style and morphemes belonging to the noun or verb category with English word.

우리가 탄 비행기는 서울에 도착했다.
woori-ga ta-n bihaenggi-neun Seoul-e dochakha-et-da.
we take-on airplane Seoul arrive
The airplane which we took on arrived at Seoul.

5-1. Analysis Result of LexAnal Module

- after ShortClauseSplit phase: five short-clauses are generated

[*'woori-ga'*, *'ta-n'*, *'bihaenggi-neun'*, *'Seoul-e'*, *'dochakha-et-da'*]

- after ShortClauseAnal phase: eleven morphemes are generated

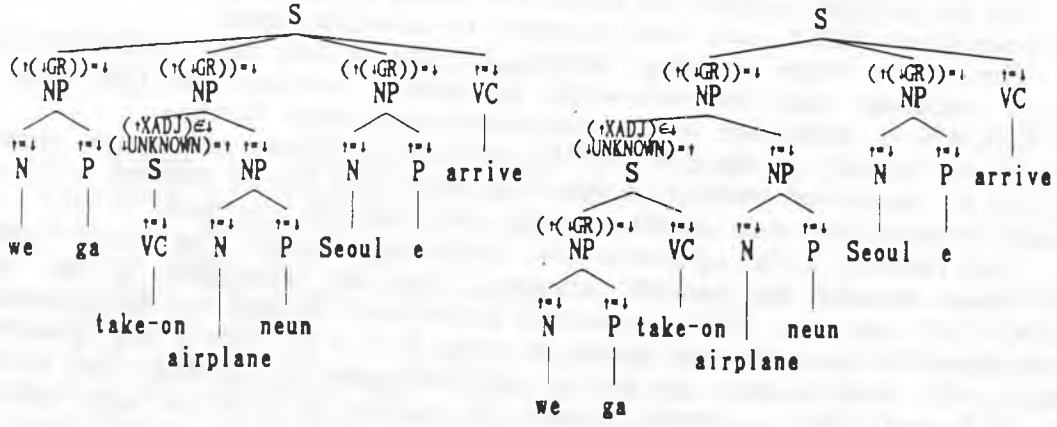
[noun(we), post(*ga*,subj-mark), verb(take-on), suffix(*n*,modify),
noun(airplane), post(*neun*,topic), noun(Seoul), post(*e*,obl_{loc}-mark),
verb(arrive), suffix(*et*,tense), suffix(*da*,final)]

- after TokenGenerate phase: eight tokens are generated

[noun(we), post(*ga*,subj-mark), vc([verb(take-on), suffix(*n*,modify)]),
noun(airplane), post(*neun*,topic), noun(Seoul), post(*e*,obl_{loc}-mark),
vc([verb(arrive),suffix(*et*,tense),suffix(*da*,final)])]

5-2. Analysis Result of CstrAnal Module

- after CstrAnal module: two alternative c-structures are generated as below



5-3. Analysis Result of FstrAnal Module

- functional schemata of morphemes obtained from lexicon

noun(we): (↑PRED) = 'PRO'
(↑NUM) = PLURAL
(↑PERS) = 3

noun(airplane): (↑PRED) = 'AIRPLANE'

noun(Seoul): (↑PRED) = 'SEOUL'

verb(take-on): (↑PRED) = 'TAKE-ON<(↑SUBJ)(↑OBJ)>'

verb(arrive): (↑PRED) = 'ARRIVE<(↑SUBJ)(↑OBL_{LOC})>'

post(*ga*): (↑GR) = SUBJ

post(*neun*): (↑TOPIC) = '+'

post(*e*): (↑GR) = OBL_{LOC}

suffix(*n*): (↑MODE) = MODIFY

suffix(*et*): (↑TENSE) = PAST

suffix(*da*): (↑MODE) = DESC

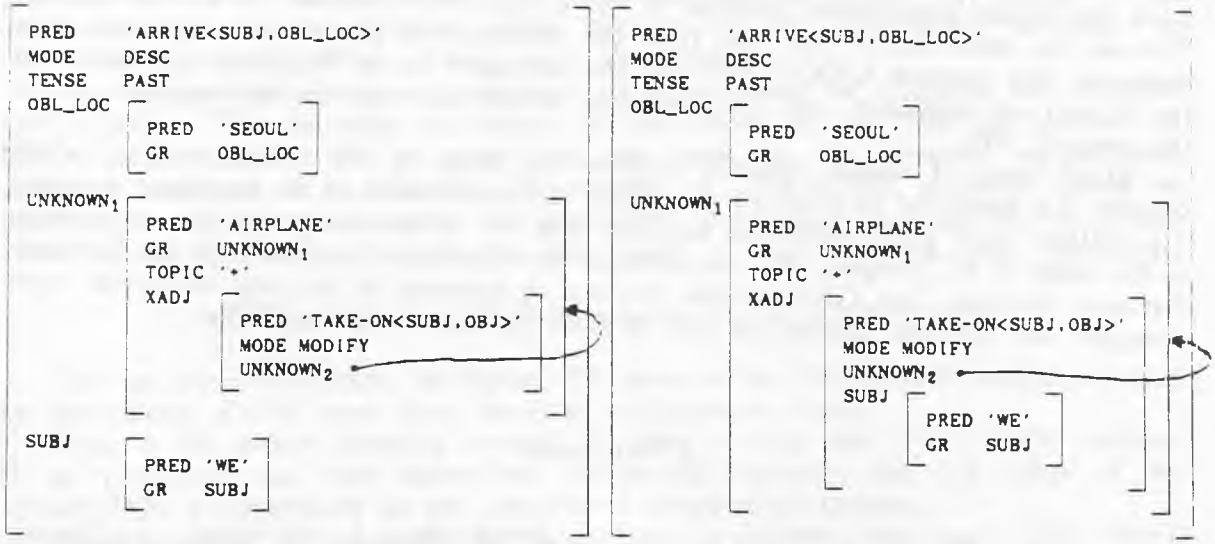
- functional schemata of complex tokens obtained by lexical rules

vc([verb(take-on), suffix(*n*,modify)]):
(↑PRED) = 'TAKE-ON<(↑SUBJ)(↑OBJ)>'
(↑MODE) = MODIFY

vc([verb(arrive),suffix(*et*,tense),suffix(*da*,final)]):
(↑PRED) = 'ARRIVE<(↑SUBJ)(↑OBL_{LOC})>'

(↑TENSE) = PAST
(↑MODE) = DESC

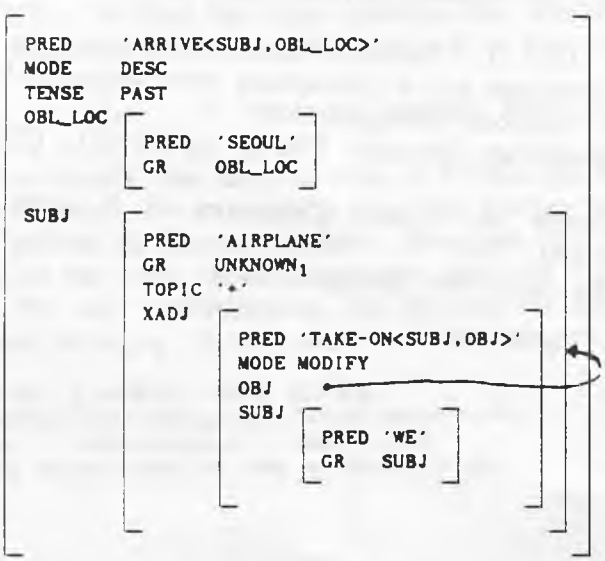
- after FstrExtract phase: two alternative f-structures are generated as below



- after FstrCheck phase: final f-structure

left alternative: rejected as illegal
<SUBJ,OBL_LOC> : {OBL_LOC,UNKNOWN$_1$,SUBJ}
=> coherency constraint violation
<SUBJ,OBJ> : {UNKNOWN$_2$}
=> completeness constraint violation

right alternative: selected
<SUBJ,OBL_LOC> : {OBL_LOC,UNKNOWN$_1$}
=> UNKNOWN$_1$ turns out to be SUBJ
<SUBJ,OBJ> : {UNKNOWN$_2$,SUBJ}
=> UNKNOWN$_2$ turns out to be OBJ



6. Conclusion

We have introduced the structure of KOSA, a natural language parser for Korean, and discussed some related issues. In the design of KOSA, the overall concept of LFG formalism is adopted, and LFG is confirmed to meet well for the grammatical characteristics of Korean. But some additional concepts for analysis are developed for KOSA further. Among them, the usages of verbal complex category and some heuristics concerned with the UNKNOWN attributes are formulated and discussed. In English, there are similar grammatical functions to the UNKNOWN such as TOPIC. But Korean NPs are far more flexible and free from the restriction of grammatical structures. And sentences with multiple UNKNOWNs are also common. So the heuristics that meet well for Korean are necessary, and the heuristics shown here can also be used to recover the omitted NPs.

Main issues of current research includes the usage of NP tokens, each of which consists of a noun and its following postnouns, and replacement of the functional schemata '(↑(↓GR))=↓' with a GR-determine function. The NP token concept has the same origin as the usage of VC category, and can provides the reduction of overhead for the CstrAnal/ FstrAnal modules. The GR-determine function is expected to be very useful for more complete and efficient analysis of the relations between verbs and NPs.

[References]

1. Sag, I.A., Kaplan, R., Karttunen, L., Kay, M., Pollard, C., Sieber, S., Zaenen, A., "Unification and Grammatical Theory", CSLI, 1987.
2. Sieber, S.M., An Introduction to Unification-Based Approaches to Grammar, pp.5-7, CSLI Lecture Notes No.4, 1986.
3. Bresnan, J.(Ed.), The Mental Representation of Grammatical Relations, Cambridge Mass.: MIT Press, 1982.
4. Kaplan, R.M. and Bresnan, J., "Lexical Functional Grammar: a formal system for grammatical representation", 1982, pp.173-281, in Bresnan, J.(Ed.), The Mental Representation of Grammatical Relations, Cambridge Mass.: MIT Press, 1982.
5. Bresnan, J., "The Passive and Lexical Theory", 1982, pp.3-86, in Bresnan, J.(Ed.), The Mental Representation of Grammatical Relations, Cambridge Mass.: MIT Press, 1982.
6. Wescoat, M.T., "Practical Instructions for Working with the Formalism of Lexical Functional Grammar", pp.1-37, in Bresnan, J.(Ed.) Lexical Functional Grammar, Stanford University, 1987.
7. Pereira, F.C.N. and Waren, D.H.D., "Definite Clause Grammars for Language Analysis - A Survey of the Formalism and a Comparison with Augmented Transition Networks", 1980, pp.231-278, Artificial Intelligence 13.
8. Peter Sells, Lectures on Contemporary Syntactic Theory, pp.135-191, CSLI Lecture Notes No.3, 1985.
9. Ishikawa, A., Complex Predicates and Lexical Operations in Japanese, pp.64-83, University Microfilm International, 1985.
10. 남기심, 고영근, 표준국어문법론, pp.93-104, 탑출판사, 1985.