

결속 조건 B를 이용한 대명사류의
해석 실행에 관한 연구

홍 성심, 이 용훈 : 충남대학교 영어 영문학과

On The Implementation of Reading Pronominals
using Binding Condition B

Sungshim Hong, Yong-hun Lee:

Dept. of English Language/Literature
Chungnam National University

요 약

우리는 본 논문을 통하여 자연언어 처리 (NLP) 분야에서 가장 그 응용이 소홀했던 지배결속 이론 (Chomsky, 1981, 1982, 1986) 중 대명사류(pronominals)의 해석과 관련된 결속이론의 응용을 시도해 보고자 한다. 이 논문에서 사용된 원리 및 개념은 결속 조건 중 성분통어(c-command), 지시 지표(indexation), 영역조건(Binding Domain)이다. 이러한 접근 방식은 한국어의 대명사류를 해석하려 할 때, 위의 세 개념들에 최소한의 수정을 가하면 가장 경제적으로 실행 가능하다는 점에서 그 가치가 높다고 생각된다.

1.0. 글 머리에

1.1. Universal Grammar (보편 문법)

Chomsky 는 일련의 중요한 연구에서 (1981, 1982, 1986) 인간의 언어현상을 설명함에 있어 한가지 중요한 개념적인 가설을 제시하였다. 그것은, 평범한 지능을 가진 인간이라면 누구라도, 비교적 열악하고 제한적인 언어 습득 환경하에서, 그 복잡한 언어를 비교적 짧은 기간내에 습득한다는 사실을 설명하기 위해 세운 가설이다. 이 가설에는 Universal Grammar (UG)가 있다고 주장되는데, 인간의 언어는 일정한 structural properties를 공유하고 있고, 이 일정한 structural properties 가 바로 UG 이며, 이 UG 를 찾는 과제는 언어학자들과 관련 학문 연구자들의 연구 목적이 된다.

다시 말하면, 첫째, 인간은 적어도 한 가지의 언어를 비교적 용이하게 짧은 기간 내에¹⁾ 습득한다는 사실, 둘째, 노출되는 언어 환경이 그다지 적격하지 (optimal) 하지 않아도 어린이는 자신이 노출된 언어에 고르게 반응하여 그 언어를 습득한다는 사실을 설명하려면, 모든 언어가 일정한 구조적 특질을 (보편 문법, UG) 가지고 있다고 가정한다면, 언어의 습득을 쉽게 설명할 수 있게 된다.

그렇다면, 각각의 언어가 2) 서로 완전히 다르다는 사실은 Chomsky 의 보편 문법의 이론적 테두리 안에서 어떻게 설명할 수 있을까? 이에대한 답으로, 소위, Principles and Parameters (원리 및 매개변인) 들이 있다. 즉, 보편문법의 규칙들 모든 언어가 같지만 그 규칙들이 완전한 statement 가 아니고 incomplete 하게 빈칸이 있어 -정해지지 않은 값이 있어- 언어마다 서로 다른 값을 가진다는 것이다.

The list of UG is, in fact, identical across languages with

1. C. Chomsky (1969) 는 어린이가 영어의 성인 문법 또는 통사 구조 (a full adult grammar) 를 습득하는데 약 5년 내지 10 년의 기간이 걸린다는 주장을 한 바 있다. 흔히, critical age 라고 부르는 사춘기, 즉 약 13 - 14세 까지의 기간, 즉, 약 13년이 걸린다는 기존의 주장과는 달리, 7 - 8세에 이미 syntactic knowledge (통사적 지식) 을 다 습득한다는 주장을 폈는데, 이 주장은 상당한 인정을 받아왔다. Carol Chomsky 의 주장을 the minimal distance principle (MDP) 라고 부른다.

2. 여기서의 말하는 각각의 언어는 이미 안정된 상태 (Chomsky 의 용어로 steady state = Ss) 에 든 언어를 일컫는다. Chomsky 의 모든 언어에 공통적인 보편 문법에서 안정 상태에 든 핵심 문법 (core grammar) 이 발전되어 나오는 과정이 매우 빠르고 완전한데, 이를 설명하기 위한 가설이 바로 Principles and Parameters식 접근인 셈이다. 양 동휘 (1994) 참조

one exception: Some of the statements on this list have blanks in them that languages can fill in with some set of lexical items or structures. (Pesetsky, 1994)

따라서, 모든 자연 언어가 서로 다른 것은, 이 변인의 값이 다르기 때문이며, 그러나 결국 추상적인 수준에서의 모든 언어의 원리들은 (statements) 공통적이다. 이 가설을 받아들이면³⁾, 첫째, 인류 공통적으로 언어 습득의 용이성이 쉽게 설명되고, 둘째, 인간의 다른 인지적 (cognitive aspects) 측면 까지 언어 현상과 함께 설명할 수 있게 된다.

2.0. 결속 이론의 도입과 응용

2.1. 결속 이론 (Binding Principles)

이러한 전체적인 문법체제 내에서 Chomsky 의 접근은 영어 명사구 (noun phrases)의 분포를 통사적, 구조적으로 기술한다. 명사구의 분류는 [+/-a, +/-p]의 자질을 조합함에 따라 pronominal 인지, anaphor 인지, 아니면 소위 anaphoric pronominal 인지에 따라 결정된다.

따라서, 우리는 본 논문에서, 기본적으로는 Chomsky 의 GB framework 내 (1981, 1982, 1986) 에서, 영어 대명사류 (pronominals)를 해석함에 있어서 그 지시 대상 (referent)을 순수 syntactic parser 로 찾아 보려 한다. 이때 도입되어 응용된 GB 개념은 (i) 결속조건 *Binding Condition (b)*, (ii) 성분 통어 *c-command* (iii) 지표 이론 *indexation* 등으로 이들이 각각 본 연구 실행도상에서 어떤 function 이나 procedure 로 쓰이는지는 통사 이론적 소개 후에 곧 뒤따르는 절에서 논의하겠다.

먼저, 영어 대명사류의 분포와 해석은 GB 이론의 결속 조건 (Binding Conditions), 특히 결속 조건 (B) 로 설명되는데, 다음의 예문을 참고해 보자.

3. D. Pesetsky (1994) 는 최근 "how languages differ?" 라는 질문에 대한 답으로 Chomskyan UG (Principles and parameters) 접근과는 다른 Optimality Theory 를 주장하고 있다. 여기서, 그는 자연언어가 서로 다른 것은 다른 변인의 값의 문제가 아니라, 우선 순위 (priority-setting) 의 문제라고 제안하였다.

(1) simple sentences

- a. He left
- b. John washed/behaved/controled/shaved him

우선 우리는 a 의 예문에서 'he' 가 어떤 지시대상을 가리키든 지 아무런 구조적 제약이 없다. 결속 조건 (B)는 이문장의 대명사에 대해 아무런 조건도 부과하지 않는다. 그러나, 예문 b 에서의 'he'를 해석함에있어서는 명백한 제약이 있다. 그것은, 대명사 'him' 4) 은 절대로 'John' 은 가리킬 수 없다는 점이다. 즉, discourse universe 상의 어떤 인물도 -그 syntactic feature 인 인칭 (person), 수(number)와 성(gender)의 일치가 있으면 - 'him'을 가리킬수 있으나 'John' 만은 제외된다. 5)

그렇다면, 단문이 아니고 내포가 있는 경우도 고려해 본다.

(2) embedded sentences

- a. John thinks (that) he is smart
- b. He thinks (that) John is smart
- c. John thought (that) he loved Mary.
- d. John thought (that) Mary loved him

물론, 위 예문중 a,c,d 에서 'he'(또는 'him) 는 'John'이 가 능한 지시대상이 된다. 여기서도 b의 경우는 'he'가 'John'을 가리킬 수 없다. 따라서, 우리는 소위 대명사에 부과되는 통사적 제약조건이 있음을 알수 있다.

4. 여기서 이 대명사의 격 (Case, case) 에 관심을 가질 독자도 있을수 있겠다. 그러나 본 논문에서는 격의 문제를 취급하지 않고, 목적격 (accusative case) 든 주격 (nominative case) 든 그 대명사의 해석에만 관심을 두겠다. 단순화시켜 언급해보면, lexical item (여기서 'HE') + accusative Case = 형태소적으로 가시화된 주격의 'he' 가 나타나고, lexical item ('HE') + nominative Case = 형태소적으로 가시화된 목적격 대명사 'him' 이 나타난다. 이 morphological realization 은 morphosyntactic rules로서 가능하다고 가정하겠다.

5. 물론, 다음의 예문에서도 'he' 는 'John'을 지시할수 없다. 이를 다루는 조건은 별도의 조건 결속 조건 (C) 이 필요한데, 본 논문에서는 다루지 않겠다.

He washed/behaved/controlled/shaved John

(3) inside NP

- a. [John loves [his mother]]
- b. [[His mother] loves John]
- c. [[John's picture of him] would be on sale]
- d. [John thought [[a picture of him] would be on sale]]

위 예문들의 경우는 대명사 + 소유격 (genitive Case) = 'his' 이 된 경우로 이때에도 일정한 결속 조건이 있음을 보여준다. 즉, a 와 b d 의 경우는 'his' 가 'John' 과 동일 지시가 가능하지만 c 의 경우는 가능하지 않다. 이것이 Chomsky 문법에서 오래동안 논의된 어떤 특정 통사 범주가(IP 또는 S, NP) 다른 통사범주와 (VP, PP 등) 구별되는 동기가 된다. 즉 우리는 여기서 전통적 국부영역범주인 IP 와 NP를 그 binding domain 으로 삼는다. 본 논문에서 도입, 응용된 자세한 GB 이론 상의 개념들은 2.3 절에서 실행화를 위한 pseudo-code 로 보이겠다.

(4) ECM 구문들

- a. John believes [him to be smart]

위 예문은 소위 예외적 격을 받는 (Exceptional Case Marking) 명사구가 있는 구문이다. 여기서 대명사 'him' 은 결코 'John'을 지시할 수 없다. 그런 뜻이라면, 'John believes himself to be smart'의 예문이 해당된다.

결국, 우리는 GB 이론상의 순수 통사적 제약 조건만을 응용해서 영어 대명사류의 해석을 실행시켜보려는 것이다.

2.2. X-Bar 이론

본 연구에서 우리는 대명사류를 해석 실행함에 있어 전반적인 수형도 또는 구조도를 X-bar Theory 체제⁷⁾ 안에서 규명하려 시도하였

7. 전통적인 phrase structure rules 에 의한 phrase-marker 대신에 이를 보다 일반화시킨 Chomsky (1986) 의 체제로서 모든 구 구조가 (phrase structures) 일반화된 형태를 갖추고 있음을 나타낸다.

한편, NLP 연구에서는, 소위 Context-free Grammar 가 가장 그 formalism을 실행하기 용이한 초기의 통사이론이었으므로 많은 학자들의 응용과 시도가 있었다. 그러나, 우리는 이미 인간의 언어를 CFG 로 실행하기에 문제가 많음도 알고 있다. 바로 이 phrase structure grammar 가 CFG 의 일종이라 말할수 있다. 홍성심 (1992), P. Norvig(1992) 참조

다. 이 일반화된 규칙은 다음과 같다.

$$\begin{array}{l}
 X'' \quad \text{---} > \quad WP \quad X' \\
 * X' \quad \text{---} > \quad X' \quad Y'' \text{ 8)} \\
 X' \quad \text{---} > \quad \quad X \quad Y'' \text{ 9)}
 \end{array}$$

여기서 X' 앞에 있는 * 는 adjunct 규칙이며, 여러 개의 adjunct 가 나타날 수도 있다. 위 도식에서 X'' (XP) 는 최대 투사 범주이며, 이 최대 투사범주는 그보다 한 bar 가 적은 intermediate projection (중간 투사 범주) 으로, 또 이 중간 투사범주는 다시 핵과 그 complement 인 또다른 최대 투사범주로 쓰인다. 여기서 X 대신에 N, V, P, A, 등이 대치되면 각각 NP, VP, PP, AP 등이 된다.

parsing 을 위한 GB tree 를 build 할때 사용된 parsing 은 일종의 Top-down method 로서 구체적인 function 은 appendix에 명시해 놓았다. 이 GB tree 는 기본적으로 각각의 입력문장을 단어로 끊어서 (여기서 variable 이름은 token 이다) 그 단어의 품사를 확인하고 X-bar 체제에 맞춰 구조도(또는 수형도) 를 형성한다. 이 procedure 의 입력정보는 struct node 의 형태이며, 출력정보도 struct node 이다. 그 구체적인 실행 case 는 appendix 에서 보이겠다.

2.3. 도입 및 응용된 GB 개념들

2.3.1. Binding Domain (국부 영역)¹⁰⁾

Chomsky 의 결속 이론에 나타난 결속영역은 사실 여러가지의 정의가 있고 그에 따른 경험적 장. 단점이 있으나 여기서 모든 정

8. 이런 notation 대신 다른 표기 방법을 쓰기도 하고 학자에 따라서는 X'', X', X 대신 여러 투사 단계가 여러 개 있는 체제도 있다. bar notation 대신, 정수로 X^{n, n'} 등을 쓰기도 한다.

9. 여기서 소위 head 인 X 가 initial 인 영어와 같은 언어 (head-initial) 와 head 가 final인 한국어같은 언어 (head-final) 의 parameter setting 의 논의가 가능하다. 즉, X-bar 전체의 체제를 바꾸지 않고도 parameter 의 값을 +/- 로, 마치 스위치를 on/off 하듯이, 바꾼다면 서로 상이한 두 언어가 설명될 수 있다. H.Lasnik & Uriagereka (1988)

10. 여기서 Binding Domain 을 결속 영역이라 하지 않고 그저 국부 영역이라 한것은 이 국부 조건이 (또는 locality Domain) 어떤 특정 정의를 채택한 것이 아니고 Chomsky (1973, 1980) 식의 국부 영역의 개념을 벗어나지 않는다는 의미에서 generic 한 이름으로 사용한 것임을 밝혀 둔다

의의(definitions) 동기는 논의하지 않겠다. 다만, 어떤 일정한 국부적인 영역의 개념이 영어 또는 한국어의 대명사류를 해석함에 필요하다는 것만은 확실하다. 이 국부적인 영역을 가장 단순한 version 인 NP, tensed IP로 잡겠다.

이를 실행함에 있어서, 우리는 stack 의 push/pop 을 이용하여, input string 을 단어별로 나누어, 품사대로 구별하고, 이를 각각의 stack에 저장하는 방법을 사용하였다.

즉, Binding Condition (B) 를 적용함에 있어서는, GB tree 를 만들때, pronominaldl1 위치를 전역변수 (global variable) 로 기록, 저장한후 tree root로 부터 pronominal 이 있는 node 까지 찾아 가면서 IP 나 NP가 나오면 stack에 저장시켜 놓는다. 그후 stack 에서 차례로 pop 하면서 Binding Domain 을 찾아 간다. 만일 pop 한 node가 IP이면, 그 IP 안에 tense 가 있는지를 검사하여 tense가 있으면 그범위를 국부 영역으로 삼고, 없으면, 다시 pop 하여 그를 국부 영역으로 삼는다. 이 GB 이론의 개념들은 각각 function 으로 처리되고 있는데, 이 국부 영역의 경우 입력값과 return value 는 없다. pseudo-code 는 appendix 에 따로 첨부하였다.

2.3.2. indexation (지표지시)

pronominal 이 포함되어 있는 binding domain 안에서 명사를 찾아 그 명사를 입력값으로 하여, search2_noun_table() 로 noun table에서 명사의 격 형태를 찾아 값을 돌려 받은후에, 그명사가 pronominal 이 가리키는 명사로서의 자격이 있는지를, 받은 격 형태와 pronominal 을 비교하여 알아낸다. 일단 자격이 있으면 전역변수 (global variable)에 명사의 위치를 저장시킨후, 성분뮌어 (c-command)의 여부를 시험할때 사용한다.

예를 들어, pronominal 이 'he'이고 명사가 'John'이면, function 인 search2_noun_table (John)은 'he' 돌려주고, return value 와 pronominal인 'he' 가 일치하므로 co-index로 본다. 만약, pronominal 이 'he'이고, 명사가 'Mary' 이면 search2_noun_table (John) 은 return value와 pronominal 'he' 가 일치하지 않으므로 Binding 의 co-index 를 만족시키지 못하는 것이 된다.

2.3.3. c-command (성분 뮌어)

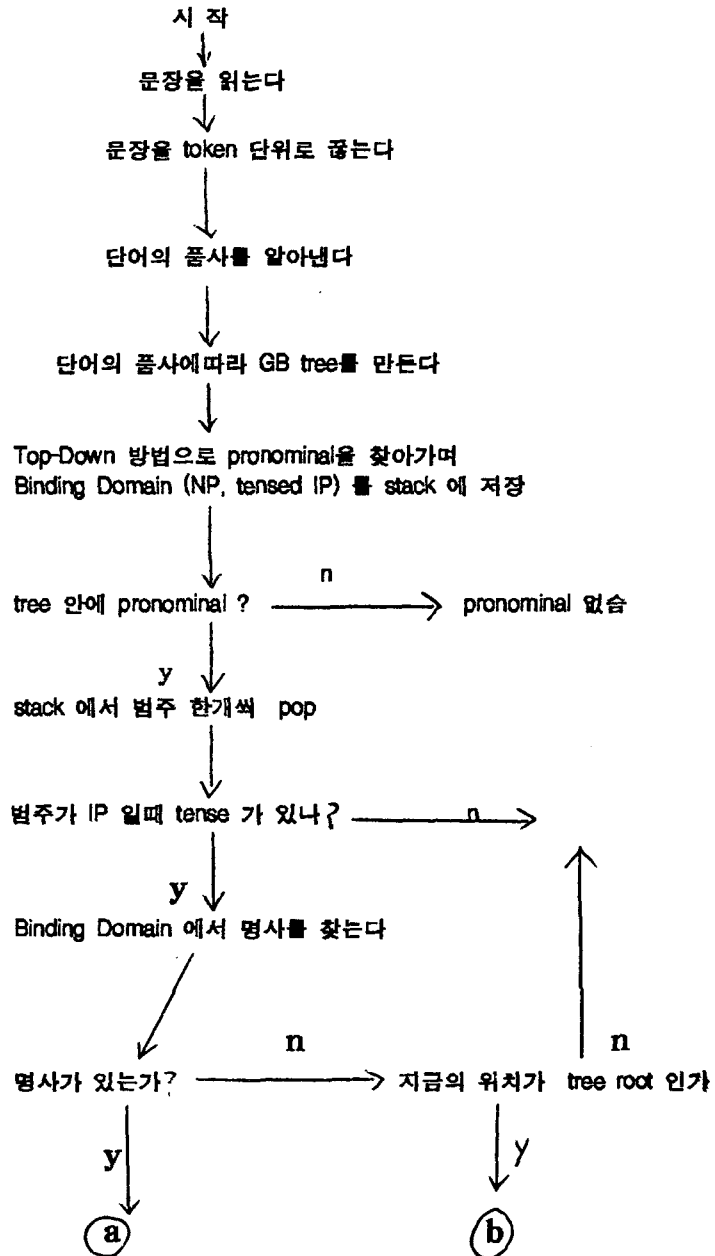
Binding domain내 에서, pronominal을 binding 할수 있는 명사를 찾아 갈 때에 first 와 second를 설정해 놓는다. second 는 first 를, first 는 현재 위치(*S) 를 따라 가게 만들어 놓고, pronominal 을 binding 할 수 있는 명사를 찾으면, second의 오른쪽 pointer를 이용하여 그 오른쪽 tree에 전역 변수 (global variable)로 선언해 놓은 pronominal이 있는지를 검사한다.. 그런 pronominal이 있으면 ON 을, 없으면 OFF 를 return 한다.

이것은 is_binding ()라는 function (함수)¹¹⁾ 속에 c-command ()라는 별도의 함수로 선언해 놓고 입력값이 없고, pronominal 의 위치 와 명사의 위치가 모두 전역 변수이기 때문에 return value 는 ON(1) 또는 OFF(0) 이다.

11. 우리가 사용한 언어가 C 이므로, 사실상 Pascal 같은 언어의 function/procedure 구별이 여기서는 의미가 없다. 즉, C 에서는 그저 subroutine으로 Pascal 에서의 function/procedure 가 하는 과제를 모두 function 이라 부르되 같은 작업을 하게 할 수 있다.

2.4. 실행 연구 구성도

본 실행 연구의 구성도의 일부는 다음과 같다.



2.5. 본 연구의 시사점 및 추후 연구의 방향

2.5.1. 한국어 대명사의 해석에 미치는 시사점

'himself' 또는 '자기' '자기자신' 같은 재귀 대명사류 (reflexives or anaphors) 는 지배 결속이론내에서 많은 논란이 있어왔지만, pronominals는 영어나 한국어에서 공히 그 분포가 어느정도 정렬되어 있다고 생각된다. 홍성십 (1985) 은 pronominals 는 영어와 한국어가 동일한 분포에 있음을 k-binding condition으로 parameterize 했으므로, 이를 받다들인다면, 우리가 본연구에서 한국어 대명사류의 해석을 실행하고자 할때, GB tree 만 다르게 build 하면 된다. 즉, 전체적인 X-bar scheme 에서, head-final parameter 만 설정하면 된다.¹²⁾

2.5.2. 한계점

본 연구는 보편 문법을 따르면서 한국어가 영어와 다른 문법 체제를 가지고 있지 않다는 가설을 전제로 하고 있다. 즉, 변인이 되는 값의 switch 만을 on/off 로 차별화 할뿐, 기본적으로 구조적 특질들을 공유하고 있다는 것이다. 따라서 모편문법에서 매개변인의 값이 매겨지면서 핵심문법이 되므로 자연언어의 다양성이 이 매개변인에 기인하는 셈이다.¹³⁾

따라서 만일 이 가설자체를 부인한다면, 본 연구는 그 바탕을 잃게 된다. 그러나, 인간의 다른 인지 작용과 언어의 관계를 이해하고, 규명하려는 시도에서, 언어 습득의 현상을 설명하기위한 가장 적격 한 접근이 바로 보편문법-매개변인 접근임을 받아 들이는 것은 바람직한 연구방법이다.¹⁴⁾

이 실험연구는, 또한, C 언어로 쓰여졌다는 점이 취약점으로 지적될 수도 있다. Pascal 및 LISP 로도 실행될 수있으리라 믿어 그 연구가 현재 진행중에 있음을 여기서 밝혀 둔다.

12. 여기서 물론, 한국어를 romanize한 상태를 뜻한다. 또한 실제 한국어의 대명사 해석 실험 연구는 별도의 연구과제로 남겨 둔다.

13. N. Chomsky (1991) 는 인간 언어는 매개변인이 작용하는 어휘부를 빼면 동일하다고 주장한다.

...
If this approach is correct (principles and parameters), then within syntax (excluding phonology) there are no rules for particular languages and no construction-specific principles.

14. 이러한 맥락에서 연구해온 NLP 학자들중 가장 대표적으로 R. Berwick 이 있다.

3.0. 글을 나오면서

본 연구는 그 응용이 소홀했던 Chomsky 의 GB 통사 이론중 결소 조건 (B)를 실행화 해 본다는데 그 가장 큰 의의를 둔다. 성분통어, 지시지표, 국부영역 의 세 중요 개념을 함수로 만들어 전체 실행화 program에 영향을 주지 않도록 하였다. 15)

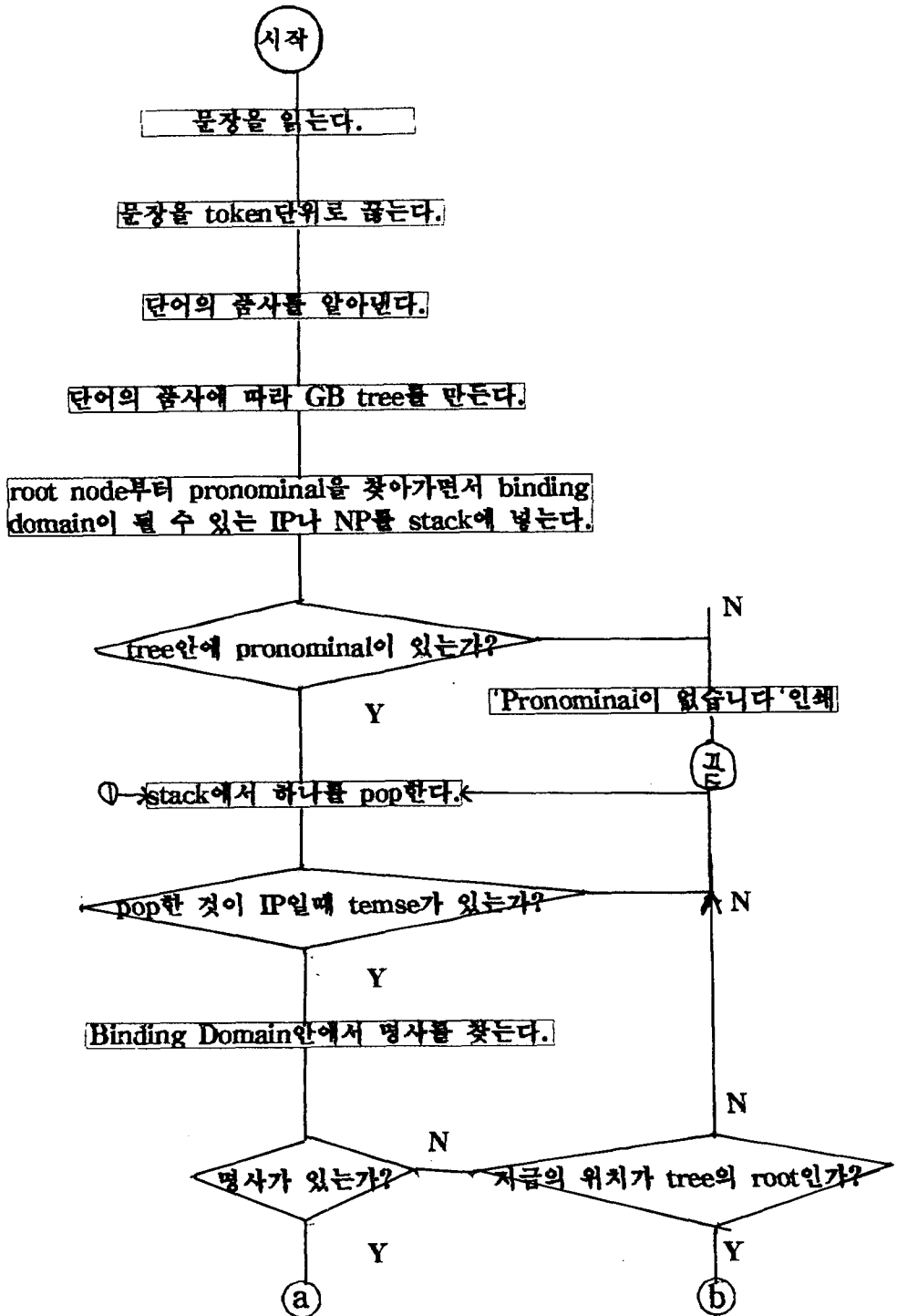
보편문법내에서 영어대명사의 지시대상을 구조적으로 해석할 수 있다면, 한국어 대명사의 지시 대상을 찾아 해석하는 실행화는 경제적으로 이루어질 수 있다.

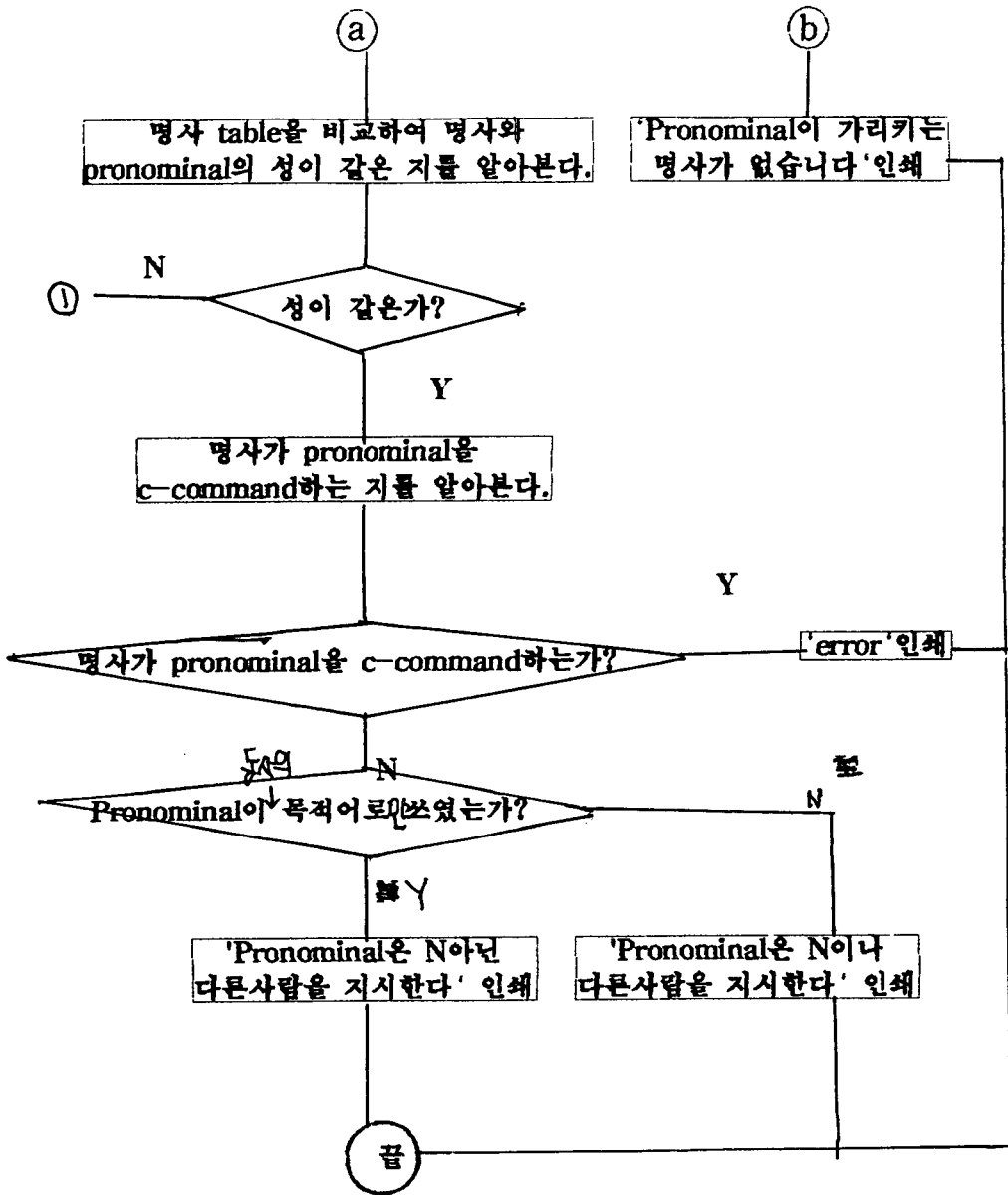
15. 이를 modular programming 이라 한다.

함수명	입력	출력	기능	사용함수
main();			문장입력, parsing, binding여부 검사	initialize(); parse(); is_binding();
initialize();			word table, tense table, verb table, noun table의 초기화	fill_word_table(); fill_tense_table(); fill_verb_table(); fill_noun_table();
fill_word_table();	char*, int		word table에 단어와 품사를 넣는다.	
fill_tense_table();	char*, char*, char*		tense table에 동사와 그 원형과 시제를 넣는다.	
fill_verb_table();	char*, int		verb table에 동사와 동사가 취하는 IP또는 CP를 넣는다.	
fill_noun_table();	char*, int char*		noun table에 명사와 격과 그 격형태를 넣는다.	
parse();		int (1 or 0)	문장을 parsing하여 parsing이 되는지 안되는 지를 알려준다.	get_token(); find_future(); make_gb_tree();
get_token();			문장을 token단위로 끊는다.	
find_future();		int	단어의 품사를 알아낸다.	
deal_det();			단어의 품사가 DET면 전역변수 temp1에 그 단어를 copy한다.	
make_gb_tree();	struct node*	struct node*	token의 종류에 따라 그에 해당하는 GB tree를 차례대로 생성한다.	deal_det(); I_verb(); NP_to_CP(); NP_to_IP(); make_template(); make_tree();
I_verb();			단어의 품사가 V일 경우 tense table을 조사하여 I node아래 시제를 V node아래에 단어를 넣는다.	search_tns_table();

함수명	입력	출력	기능	사용함수
NP_to_CP();			'think'와 같은 동사가 CP를 목적으로 취할때 verb table을 읽어서 처리해준다.	search_verb_table();
NP_to_IP();			'believe'와 같은 동사가 IP를 목적으로 취할때 verb table을 읽어서 처리해준다.	search_verb_table();
make_template();		struct node*	XP, X_bar, X node를 만들고 XP의 pointer를 돌려준다.	
is_binding();			Binding Condition B를 이용하여 대명사와 명사의 관계를 알아본다.	find_pronoun(); find_N(); c_command(); search2_noun_table(); message1(); message2(); message3(); message4();
find_pronoun();		struct node*	tree 중에서 대명사를 찾는다.	find_pronoun(); ipush(); ipop(); ppush(); ppop();
find_N();		struct node*	tree 중에서 명사를 찾는다.	find_N();
c_command();	struct node*, struct node*	int	N이 pronoun을 c_command하는지의 여부를 검사한다.	
make_tree();	struct node*, struct node*, int, int, int		XP의 spec, XP, X_BAR, X, terminal node의 값을 넣는다.	
search_tns_table();	char*	char*	동사의 시제를 table에서 찾는다.	
search_verb_table();	char*	int	동사의 목적어 형태(IP또는 CP)를 table에서 찾는다.	

함수명	입력	출력	기능	사용함수
search1_ noun_table();	char*	int	명사의 격을 table에서 찾는다.	
search2_ noun_table();	char*	char*	명사의 격형태를 table에서 찾는다.	
message1();			'문장중에 pronominal이 없습니다'를 인쇄	
message2();			'문장중에 pronominal이 가리키는 N이 없습니다'를 인쇄	
message3();			'문장중에 pronominal은 제 삼자를 가리킵니다.' 를 인쇄	
message4();			'문장중에 pronominal은 N이나 제 삼자를 가리킵니다.' 를 인쇄	
error();			'Syntax error'를 인쇄	





Primary References

- Allen, J. (1987). Natural Language Understanding. The Benjamin/Cummings Publishing Company, Inc.
- Berwick, R. (1985). The Acquisition of Syntactic Knowledge. MIT Press: Cambridge, Mass.
- Butler, C. (1985). Computers in Linguistics. Blackwell
- Chomsky, N. (1957). Syntactic Structures. Mouton
- Chomsky, N. (1973). "Conditions on Transformations", in Anderson & Kiparsky eds. A Festschrift for Morris Halle, Holt
- Chomsky, N. (1981). Lectures on Government-Binding, Doris
- Chomsky, N. (1982). Some Concepts and Consequences of GB Theory, MIT press, Cambridge
- Chomsky, N. (1986). Knowledge of Language. MIT Press, Cambridge
- Dorr, B.(1992). Machine Translation. MIT Press
- Gazdar, G. & C. Mellish. (1989). Natural Language Processing in LISP Addison & Wesley
- Haegeman, L. (1990) An Introduction to GB Syntax,
- Hong, S. (1985) A & A' Binding in Korean and English:Government & Binding Parameters, Doctoral Dissertation, The Univ. of Connecticut
- Hong, S. (1992). "A Case Study on Sentence-Generation Using a CFG for English/Korean", Language Vol 13, Chungnam National University
- Lasnik & Uriagereka (1988) A Course in GB Syntax, MIT Press, Cambridge
- Pylyshyn, Z. (1985). Computation and Cognition Cambridge. MIT Press
- Radford, A. (1988). Transformational Syntax. Cambridge University Press
- Shank, R & C. Riesbeck. (1981). Inside Computer Understanding. Hillsdale, NJ.
- Stabler, E. (1992). The Logical Approach to Syntax. MIT Press
- Sudkamp, T. (1988). Languages and Machines. Addison & Wesley
- 양동휘 (1994) 문법론, 한국문화사, 서울