한국어 부분언어에 대한 문법 정의 및 GLR 파싱[1)]

김지현, 정병채[0], 이재성
충북대학교 정보컴퓨터교육과, 비즈니스데이터융합학과, 디지털정보융합학과

A grammar definition and the GLR parsing for Korean sub-language

Ji Hyun Kim, Byung Chae Jung[0], Jae Sung Lee
Dept. of Information & Computer Education, Dept. of Business & Data Convergence, Dept. of Digital Informatics & Convergence, Chungbuk National University

요 약

최근 한국어를 배우는 외국인의 증가로 '외국어로서의 한국어 학습(KFL)'에 대한 관심이 늘고 있다. 본 논문에서는 외국인을 위한 한국어 교재에서 사용된 회화 문장으로부터 문장 패턴을 분석하고 이를 기반으로 한국어 부분 언어 문법을 정의한다. 대개 부분 언어 문법은 간단하고 배우기 쉬우므로 외국어로서의 한국어 학습자들이 쉽게 한국어로 의사소통을 할 수 있을 것이다. 특히, 본 논문에서는 이 부분 문법이 컴퓨터로 해석될 수 있도록 문법을 정의하였고, 이를 자동 어휘분석기 생성기(flex)와 자동 파서 생성기(bison)을 이용해 기본적인 검증을 하였다.

주제어: 한국어 부분 언어 문법, GLR 파싱, 외국어로서의 한국어 교육

1. 서론

최근 한류 열풍에 의해 외국 이민자의 유입이 증가하고 있다. 따라서 한국어를 배우고 싶어하고, 사용하고자 하는 외국인들이 증가하고 있다[2]. 배우고자 하는 외국어가 모국어와 많이 다른 경우 이를 배우는 것이 어렵고, 일정 수준 이상의 의사소통을 위해서는 많은 시간을 필요로 한다. 하지만 이해가 쉽고 배우기 간단한 부분언어를 개발한다면, 그리고 이것을 컴퓨터로 해석할 수 있도록 한다면 외국어를 쉽게 학습할 수 있을 것이다.

부분언어 문법은 제한된 영역에서 보다 명확하고, 쉽게 의사전달을 하기 위해 개발되었으며, 대표적으로 AECMA[3]가 있다. 이는 항공관제에서 사용하는 널리 알려진 통제언어이다. 이러한 통제 언어에서는, 사용되는 단어 수, 모호성을 가진 어휘의 사용, 수동태의 사용 등을 제한하고, 20단어 이내의 짧은 문장의 사용을 권한다. 이렇게 간략화된 영어를 국제표준화 하기 위한 움직임이 나타나고 있다[4].

CFG는 파싱에 사용되는 잘 알려진 문법 중 하나이며 기본적인 언어구조를 정의하는데 편리하다. 하지만 자연언어에서 CFG를 사용할 경우, 대개 모호성이 많아, 파스트리를 여러 가지로 생성해 낸다. 가능한 한 모호성을 줄이고 적은 수의 트리를 만들어 내기 위해 문맥 속성을 이용하여 제약을 가한 문법으로 PSG, HPSG, LFG 등이 있다. KPSG는 PSG을 한국어에 적용한 것이다[5]. 그럼에도 불구하고 부분언어처럼 언어의 구조가 간단할 경우, CFG는 기본 문형 정의에 사용될 수 있다.

영어는 어휘가 문장에서 사용된 위치에 따라 그 문법적 성분이 결정되는 언어로, CFG와 같이 구성요소의 순서에 따라 적용되는 규칙이 결정되는 문법이 적용되기 쉽다. 하지만 한국어는 문장요소의 위치가 자유롭고, 조사로 문장 성분을 나타낸다. 또한 주어나 목적어 등의 생략이 빈번하고, 모호성이 높아 이를 CFG로 표현하는 것은 매우 불편하고, 효율이 떨어진다. 이에 자유로운 어순을 반영할 수 있는 ECFG를 제안하기도 했다[6, 7]. 이러한 방법은 주로 문형을 문장 성분(주어, 동사, 목적어)을 중심으로 분류한 것이다. 반면에 서술어 중심 분류는 서술어에 따라 문장의 유형으로 분류하고 사용하는 방식이다[8]. 하지만 실제적으로 이것을 파싱에 체계적으로 사용하기는 어려운 점이 있다.

본 논문에서는 한국어를 이용한 의사소통에서 가장 필수적인 부분언어를 정의하고, 이에 대한 문형을 CFG로 정의하며 이를 GLR 파싱 방법으로 파싱할 수 있도록 한다. CFG의 구조를 정의하고, 속성을 추가하여 제약하는 기존 연구들과는 달리, 본 논문에서는 한국어 부분언어를 제안하여 어순을 제약하고, 이를 CFG 구조로 표현했다는 점에서 차별점을 갖는다. 한국어 부분언어는 외국인을 위한 한국어 교재에서 쓰인 문장을 이용하여 추출하였고, 문장의 구조는 품사 수준에서 정의하였다.

다음으로 2장 관련연구에서는 CFG를 이용하여 개발한 한국어 파서의 문형들을 살펴본다. 3장에서는 기본적인 의사소통이 가능한 수준의 한국어 부분언어를 정의하고, 이를 반영한 CFG를 제안한다. 4장에서는 제안된 CFG를 파싱할 수 있는 GLR 파싱 방법을 제시한다. 이어 5장은 실험 결과, 6장은 결론으로 마무리한다.

2. 관련 연구

CFG는 구성요소의 순서에 따라 문법적 성분이 결정되기 때문에 영어와 같이 위치에 따라 문법적 성분이 결정되는 언어에 강점이 있다. 하지만 한국어는 부분자유어순으로 문장요소들의 위치가 유동적이므로 이를 CFG로 모두 표현하려면 경우의 수가 매우 많아 비효율적이다.

이러한 문제점을 해결하기 위해 자유로운 어순을 반영

1) 이 논문은 [1]의 일부를 확장 보완한 것임.

하여 파싱하는 ECFG를 제안하기도 했다[6, 7]. 나동렬[6]에서는 문장(S)에서 반복적으로 사용되는 체언구(PP)와 부사구(ADVP)를 쉽게 표현하기 위해 그림 1과 같은 CFG를 제안했다. 또한 그림 2와 같이 반복 표시 *를 이용하여 간단하게 표현하였다.

```
S -> PP S
S -> ADVP S
S -> VP
```

그림 1. 간단한 한국어 CFG

```
S -> PP* VP
S -> (PP | ADVP)* VP
```

그림 2. 반복 표현을 이용한 한국어 CFG[6]

양성일[7]은 이를 확장한 ECFG를 제안하며 인용절, 대등절, 종속절 등을 처리할 수 있도록 했다. 문장의 종류를 파악할 수 있는 장치를 하고 이를 문법으로 표현하여 이들 관계를 제약하도록 했다. 예를 들어, 그림 3은 대등절 및 종속절을 처리하는 문법으로 $S_{[ce]}$는 대등절 연결어미로 끝나는 대등절, $S_{[dce]}$는 종속적 연결어미로 끝나는 종속절, ! 는 부정을 의미한다. 즉, 대등절 연결어미로 끝나는 문장이 반복된 후, 대등절 어미가 아닌 문장이 오면, 문장이 종료하는 첫 번째 문장과, 종속절 연결어미로 끝나는 문장이 온 후, 일반 문장이 와서 끝나는 두 번째 문장을 정의한 것이다.

$$S \rightarrow S_{[ce]}^{*}\ S_{![ce]} \mid S_{[dce]}\ S$$

그림 3. ECFG 예[7]

문장의 구조적 유형을 공식화한 틀을 문형이라 하며 주로 서술어에 따라 구분된다[8]. 이용석[11]은 이러한 문형을 CFG로 구현하여 처리하였다. 이러한 문형의 일부 예는 그림 4와 같다.

```
N(이/ 는/ 은/ 가) + V
N(이) + N(에/ 에게) + V
N(이) + N(로/ 으로) + V
N1(은) + N2(이) + V   // 근심되다, 생각나다, 욕심나다
N(이) + A             // 아름답다, 붉다 ...
N(이) + N(에) + A     // 유리하다, 능하다, 해롭다 ...
N(이) + N(와) + A     // 같다, 다르다, 동일하다...
```

그림 4. 문형 예[11]

3. 한국어 부분언어 정의

한국어는 회화체에서의 생략이 매우 빈번하다. 문장 내에서 주어나 목적어 등을 생략하거나 어절내에서 격조사를 생략하는 등 여러 가지 경우가 있다. 이러한 생략에도 의사소통이 가능한 것은 구문분석으로는 모호성이 많으나 의미나 화용 분석에서는 모호성을 해결할 수 있기 때문이다. 따라서 구문분석에서는 생략에 의한 모호성에도 불구하고, 구문분석의 결과를 다음 단계인 의미분석이나 화용분석으로 보내 줄 수 있어야한다. 또, 외국인들에게는 격조사의 생략이 훨씬 편리한 언어 용법일 수 있다. 가능하면 구성적 요소를 반영하여 이에 익숙한 외국어 학습자들에게 쉽게 적응할 수 있도록 하는 것도 한 방법일 수 있다.

이 논문에서는 이러한 특성을 반영하여 외국인이 이해하기 쉽고, 또한 CFG로 표현이 쉽게 되는 부분 언어 문법을 개발한다. 이를 이루기 위한 기준은 다음과 같다.

1) 어순은 주어 목적어 술어의 순으로 제약한다.
2) 부사와 같이 임의의 위치에서 수식하는 어절은 가능하면 위치를 정해 주어 혼동이 없도록 한다.
3) 외국인들이 조사 사용에 어려움을 겪는 것을 고려하여 조사가 생략된 한국어 문형을 개발하되, 대표적 외국어 어순에 맞도록 한다.
4) 구문구조가 불명확한 경우에 한 해, 조사 사용을 필수화하여 구문구조가 드러날 수 있도록 한다.
5) CFG 표현에서 한국어 조사나 어미 형태를 고려하여 문법 표현을 한다.
6) 문법 내에 어휘나 태그를 사용하여 문법범주를 세분화하여 표시한다. 어휘는 직접 문법에 사용할 수도 있지만, 가능하면 태그로 대신한다.

조사가 생략된 문형을 개발할 경우, 조사를 생략한 채 어순을 변화시키면 잘못된 구조의 문장이 나올 수 있다. 즉, 한국어는 어순이 자유로와 SOV, OSV 등의 구조가 가능하여 조사를 생략해도 이해가 가능하지만, 외국인들에게는 (1)의 원문이 아래의 (3)처럼 밥이 나를 먹었다는 뜻으로 전혀 다르게 해석될 수 있다.

- 나는 밥을 먹었다. I ate steamed rice (1)
- 나 밥 먹었다. I ate steamed rice (2)
- 밥 나 먹었다. Steamed rice ate me (3)

조사를 사용할 경우, 격조사 정보가 상위 문법 정보에 포함되어 있어야 한다. 즉, 기존의 명사에 주격조사가 붙어 '주어' 태그가 되거나, 동사 어간에 연결어미가 붙어 '연결서술어'와 같은 형태로 아래와 같이 표현하여 문법 범주를 세분화할 필요가 있다.

<주어> -> <명사> + 주격조사
<연결서술어> -> <동사어간> + 연결어미

한국어 부분언어 문형을 개발하기 위해 국립국어원에

서 발간한 ‘여성결혼이민자와 함께하는 한국어1’ 교재[12]에 수록된 총 1,841문장을 사용하였다. 이 중 존칭형 ‘습니다.’와 같은 문장은 기본형 ‘ㄴ다’, 와 해요체 ‘요’로 바꾸었다. 그 결과 총 989문장이 나왔으며 이를 간소한 문형 개발을 위한 원본 문장으로 사용하였다. 이 문장들을 형태소 분석기 및 태거 프로그램[13]을 사용하여 품사 태그열을 추출하고 이 중 반복되는 문장 패턴(문장의 품사 태그열)을 제거한 186개의 문장 패턴만을 기초자료로 사용하였다. 품사 태그열을 이용하여 문법규칙(grammatical rule)을 정의했지만, 가능하면 태그열이 나온 어절 단위(띄어쓰기 단위)를 고려하여 기술하였다. 그리고 문장에 어순이 주어, 목적어, 부사어에 기술되는 위치가 다르더라도 각각의 체언 뒤에 붙은 조사를 구분하여 주어, 목적어, 부사어에 해당되는 어절 단위로 인식하였다.

4. GLR 파싱 및 실험

CFG의 파싱은 일반적으로 LR 파싱 방법으로 수행하고 대부분 결정적인 프로그래밍 언어의 특성에 따라 LR 파싱 방법은 효과적이다. 문법이 모호하여 두 개 이상의 파스트리가 발생하는 경우, 우선 순위를 두고 우선 순위에 따라 하나를 선택하였다. 하지만 자연언어는 많은 모호성을 가지고 있고, 모호성의 해결은 어휘, 구문, 의미 등의 분석과정을 거치며 해소된다. 본 논문에서는 문법 정의만을 목적으로 하여, 어휘 모호성이 해결된 형태소 태그열을 입력 받아 모호성이 포함된 구문 분석 결과를 내놓는다. 여기서 생성된 모호성이 포함된 구문분석 결과는 상위 단계에서 해결되어야 할 것이다.

모호성이 포함된 파스트리는 LR 파싱으로는 한계가 있기 때문에 파싱 과정을 일반화시킨 GLR 파싱 방법을 사용한다. LR 파싱은 푸시다운 오토마타를 수행하면서 모호성이 발생하면 그중 선호하는 하나의 오토마타 경로만을 선택하여 수행하고, 만약 그 경로가 잘못되면 오류로 종료한다. 반면에 GLR 파싱은 모호성이 발생하면 가능한 모든 오토마타 경로를 수행하다 최종적으로 병합되는 경로를 찾아 그 결과를 출력한다. 따라서 GLR파서는 모호성이 포함된 문장의 구문분석이 어느 정도 가능하다.

본 논문에서는 bison을 이용하여 GLR 파싱이 가능한 부분 문법을 정의하였다[10]. 그림 5는 GLR 파싱이 실행되는 방법을 설명하기 위한 것이며, 격조사를 생략했을 경우, 명사가 주어와 목적어로 파싱될 수 있도록 만든 문법의 예이다.(세종 말뭉치[14]의 품사 태그를 사용함)

%merge 문은 공통의 경로가 나왔을 경우, 이를 하나의 경로로 병합하는 구문분석 명령이다. bison의 RHS 문법 표기에서 모호성이 있는 비단말 문법기호가 제대로 병합되기 위해서는 그 문법기호 다음에 반드시 모호성이 없는 문법기호가 나와야 병합이 이루어진다. 이를 처리하기 위해, 문법 기술을 다양하게 변형하여 사용하였다. 또, cn은 create_node 모듈의 약칭으로 매개변수를 이름으로 갖는 노드를 만들어 그 포인터를 리턴한다. ac는 add_child 모듈의 약칭으로 첫 번째 매개변수에 두 번째 매개변수의 포인터를 자식 노드로 붙이되 세 번째 매개변수의 위치에 자식노드로 붙인다. 예를 들어, ac($$, $1, 1)은 $1의 노드를 $$의 1번째 자식 노드로 붙인다.

```
PGM   : s END        { Tree_print($1);  printf("OK\n"); return 1;}
      ;
s     : subj vp %merge <stmtMerge> {$$=cn("s"); ac($$, $1, 1); ac($$, $2, 2); }
      | obj vp %merge <stmtMerge> {$$=cn("s"); ac($$, $1, 1); ac($$, $2, 2); }
      ;
subj  : noun JKS     { $$=cn("subj"); ac($$, $1, 1); ac($$, $2, 2); }
      | noun         { $$=cn("subj"); ac($$, $1, 1); }
      ;
obj   : noun JKO     { $$=cn("obj"); ac($$, $1, 1); ac($$, $2, 2); }
      | noun         { $$=cn("obj"); ac($$, $1, 1); }
      ;
noun  : NNG          { $$=cn("noun"); ac($$, $1, 1); }
      | NNP          { $$=cn("noun"); ac($$, $1, 1); }
      ;
vp    : VV EF              { $$=cn("vp"); ac($$, $1, 1); ac($$, $2, 2); }
      | VA EF              { $$=cn("vp"); ac($$, $1, 1); ac($$, $2, 2); }

      ;
```

그림 5. GLR 파싱을 위한 간단한 예

이 문법은 bison에 의해 c파일로 바뀌고, 이를 컴파일하여 부분문법 파서를 만들 수 있다. 이 과정에서 품사 태그를 인식하여 토큰으로 넘겨주는 어휘분석기가 필요한데, 이는 flex를 이용하여 구현하였다[9]. 아래는 이 문법에 따라 실험한 예이다. 즉, 태그열을 입력하여 그에 상응하는 파서 트리를 출력한 것이다. 파스트리의 표현은 일반트리를 이진트리로 변환하여 표현하였다. 즉, 첫 번째 자식노드만 부모의 자식노드로 표현하고, 두 번째부터의 자식노드들은 첫 번째 자식노드의 형제노드(sibling)로 표현하였다. 그림 6은 문장 아래 subj과 vp가 있고, subj 밑에 noun과 JKS가 자식 노드로 있는 그림이다. 이 경우, 모호성이 없으므로 파스트리가 하나이다.

```
D>subglr
NNP JKS VV EF END

s
|
subj--------vp
|           |
noun--JKS   VV----EF
|
NNP
```

그림 6. 모호성이 없는 GLR 파싱 예

그림 7은 격조사의 생략으로 모호성이 있는 경우로 ‘?’ 노드 밑에 가능성이 있는 파스트리가 표현된다. 그림 7의 경우는 obj vp를 갖는 s와 subj vp를 갖는 s의 두 가지가 모두 그려진 결과이다.

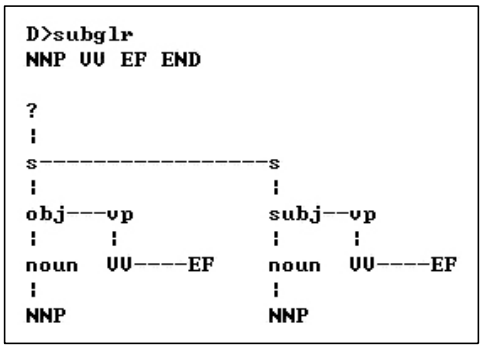

그림 7. 모호성이 있는 GLR 파싱 예

실험은 외국어로서의 한국어 교재[12]에서 일부를 발췌하여 수행하였다. 형태소 품사 태그는 어휘 중의성과 형태소 중의성이 해소된 상태로 입력된 것으로 가정하여, 파서의 성능만을 독립적으로 평가하였다. 따라서, '나는' 이라는 입력이 들어오면 이를 태깅한 결과 '나/NNP + 는/JKC' 가 나온다고 가정하고, 이들 중 NNP+JKC 태그열을 입력으로 파서에 넘긴다. 총 1,841문장에서 동사에서 존칭 생략 등을 통해 문장을 간략화하여 989문장을 추출하였고, 이 문장의 패턴을 정리하여 186개의 문장 패턴(태그열 패턴)을 얻었다. 이를 기반으로 부분 CFG 문법을 개발하고, 이를 이용하여 간략화된 989문장을 테스트해 본 결과 326개의 문장(33%)이 파싱에 성공하였다.

5. 결론

부분문법 언어는 주로 제한된 영역에서 명확한 의사전달을 목적으로 한다. 본 논문에서는 외국인의 쉬운 한국어 학습을 위하여 한국어 구문을 단순화시킨 부분문법을 개발하였다. 외국인을 위한 한국어 교육 교재에 나온 문장들을 이용하여 개발하였고, 기본 문형을 추출한 후 GLR 파싱이 가능한 형태로 변형하였다.

본 논문에서는 총 1,841문장을 동사에서의 존칭 생략 등을 이용해 간략화하고 문장 패턴을 정리하여 186개의 문장 패턴(태그열 패턴)을 얻었다. 이를 기반으로 부분 CFG 문법을 개발하고, GLR 파싱의 가능성을 검증하였다. 이 연구는 간소 패턴을 사용한 의사소통이 궁극적인 목적이다. 따라서, 다시 원본 문장과 같은 의미의 전달이 가능하면서 GLR 파싱이 가능한 문장으로 패턴을 변경할 경우, GLR 파싱 성공률은 증가하였고, 이를 계속할 경우, 어느 정도 안정된 형태의 부분 문법을 완성할 수 있을 것이다. 본 논문에서는 기본 문형 패턴의 개발과 GLR 파싱의 가능성을 확인하였으며, 이를 이용하여 문형개발을 계속할 필요가 있다.

참고문헌

[1] 김지현, "외국어로서의 한국어 교재를 위한 한국어 문형," 충북대학교 교육대학원, 교육학석사학위논문, 2013.

[2] 김승권 외, "2009년 전국 다문화 가족 실태 조사 연구," 보건복지부, 법무부, 여성부, 한국보건사회연구원, 2010.

[3] AECMA Simplified English : http://www.techscribe.co.uk/ta/aecma-simplified-english.pdf.

[4] Richard Hodgkinson, A standard for simplified natural language : http://www.tcanz.org.nz/site/tcanz/files/Standards/InternationalStandardsSimplifiedEnglish.pdf.

[5] 김영택, 자연언어처리, 생능출판사, 2001.

[6] 나동렬, "한국어 파싱에 대한 고찰," 정보과학회지 제 12권, 제 8호, pp.33-46, 1994.

[7] 양성일, "확장 문맥 자유 문법과 패턴-액션 규칙을 이용한 한국어 구문 분석에 관한 연구," 연세대학교 대학원 전산과학과, 석사학위 논문, 1999.

[8] 강은국, 조선어 문형 연구, 박이정 출판사, 1996.

[9] flex 2.2.5 : ftp://reality.sgiweb.org/freeware/fw-5.3/fw_GNUflex/GNUflex.html.

[10] GNU bison 2.7 : http://www.gnu.org/software/bison/manual/bison.html

[11] 이용석, "구문분석의 실용화를 위한 구문분석기 개발에 관한 연구," 한국전자통신연구원, 1999.

[12] 국립국어원, 여성결혼이민자와 함께하는 한국어 1 : http://www.korean.go.kr/09_new/edu/kteacher_book01.jsp.2010

[13] 김보겸, 이다니엘, 이재성, "3단계 확률기반 형태소 분석기를 이용한 한국어 품사 태거 구축 방법," 한국컴퓨터교육학회 학술발표대회논문집, 제 16권, 제 1호, pp. 129-134, 2012.

[14] 국립국어원, 21세기 세종계획 최종결과물(2011년 12월 수정판), 2011.