내포문의 단문 분할을 이용한 한국어 구문 분석

(Korean Syntactic Analysis by Using Clausal Segmentation of Embedded Clause)

이 현 영 † 이 용 석 ††
(Hyeonyeong Lee) (YongSeok Lee)

요 약 한국어 문장은 대부분 주절과 내포문을 가지는 복문으로 구성되어 있다. 따라서 복문에 나타나는 하나 이상의 용언으로 인해 구문 분석 과정에서 다양한 구문 애매성이 발생한다. 이들 중 대부분은 내포문의 수식 범위로부터 발생되는 구 부착의 문제 때문이다. 이런 구문 애매성은 내포문의 범위를 정해서 하나의 구문 범주의 기능을 가지도록 하면 해결할 수가 있다. 본 논문에서는 내포문의 범위를 정하기 위해서 문형과 한국어의 구문 특성을 이용한다. 먼저, 내포문에 있는 용언의 문형 정보가 가질 수 있는 필수격을 최대로 부착하여 내포문의 범위를 정하고 이를 이용해서 복문을 내포문과 주절로 분할한다. 그리고 한국어의 구문 특성을 이용해서 분할된 내포문의 기능을 하나의 구문 범주인 체언구나 부사구로 변환한다. 이렇게 함으로써 복합문의 구성 형태가 단문 구조로 변환되기 때문에 내포문의 범위에 의한 구 부착의 문제가 쉽게 해결된다. 이것을 본 논문에서는 내포문의 단문 분할이라고 한다. 본 논문에서 제안한 방법으로 1000 문장을 실험한 결과 문형과 단문 분할을 이용하지 않은 방법보다 구문 애매성이 88.32% 감소되었다.

키워드 : 구 부착의 문제, 내포문의 단문 분할, 문형 정보, 구문 분석

Abstract Most of Korean sentences are complex sentences which consisted of main clause and embedded clause. These complex sentences have more than one predicate and this causes various syntactic ambiguities in syntactic analysis. These ambiguities are caused by phrase attachment problems which are occurred by the modifying scope of embedded clause. To resolve it, we decide the scope of embedded clause in the sentence and consider this clause as a unit of syntactic category. In this paper, we use sentence patterns information(SPI) and syntactic properties of Korean to decide a scope of embedded clause. First, we split the complex sentence into embedded clause and main clause by the method that embedded clause must have maximal arguments. This work is done by the SPI of the predicate in the embedded clause. And then, the role of this embedded clause is converted into a noun phrases or adverbial phrases in the main clause by the properties of Korean syntax. By this method, the structure of complex sentence is exchanged into a clause. And some phrases attachment problem, which is mainly caused by the modifying scope, is resolved easily. In this paper, we call this method clausal segmentation for embedded clause. By empirical results of parsing 1000 sentences, we found that our method decreases 88.32% of syntactic ambiguities compared to the method that doesn't use SPI and split the sentence with basic clauses.

Key words : Phrases attachment problems, Clausal segmentation of Embedded clause, Sentence patterns information, Syntactic analysis

† 학생회원 : 전북대학교 컴퓨터정보학과
lhy0730@nate.com
†† 종신회원 : 전북대학교 컴퓨터정보학과 교수
yslee@chonbuk.ac.kr
논문접수 : 2007년 1월 22일
심사완료 : 2007년 12월 8일

정보과학회논문지 : 소프트웨어 및 응용 제35권 제1호(2008.1)

1. 서 론

한국어는 단문보다는 내포문(종속절)을 포함하는 복문의 구성 형태를 가진다. KIBS[1]를 분석한 결과 두 개 이상의 용언으로 구성된 문장은 90.4%나 차지했다. 따라서 단문에서는 모든 체언구나 부사구를 용언에 부착하면 되지만 내포문을 포함하는 문장에서는 부사구나 체언구를 어떤 용언과 결합하느냐에 따라 많은 구문 애매성이 발생하게 된다.

가) 철수가 [학교에 가는 순이를] 보았다.

문형) 가다 : N이 N에 V, N이 N로 V, N이 V
보다 : N이 N을 V

예를 들면, 문장 가)는 체언구 "학교에"가 용언 "가다"나 "보다" 모두에 부착할 수가 있다. 그러나 문형 정보와 관형절의 특성을 이용하여 용언에 부착되는 체언구를 제약하면 체언구 "학교에"는 용언 "가다"에 부착된다[2]. 따라서 관형절의 범위를 "학교에 가는 순이를"로 제약하여 하나의 단문으로 분할할 수가 있다. 이렇게 분할된 관형절은 전체 문장에서 목적어 구실을 하는 체언구의 기능을 가진다. 즉, "보다"의 문형 "N이 N을 보다"에서 "N을"에 해당된다. 따라서 위의 문장 가)는 내포문을 포함한 복문의 구조에서 "철수가 [학교에 가는 순이]를 보았다"는 단문 형식으로 변환된다.

이와 같이 내포문에 포함된 용언의 문형 정보를 이용하여 지역적으로 단위화된 구에 체언구나 부사구의 기능을 부여하는 것을 본 논문에서는 단문 분할이라고 정의한다. 이렇게 함으로서 복합문의 구성 형식이 단문의 형식으로 변환되는 원리를 이용하여 문장 내의 부착 문제를 단문내의 부착 문제로 축소한다. 예로 위의 가)문장에서는 관형절 "학교에 가는 순이를"을 단문 분할한 후에 목적어의 기능을 가지도록 하면 복문의 구조에서 "철수가 순이를 보았다"라는 단문 형식으로 변환되는 것이다. 이런 원리를 이용하면 용언이 여러 개 사용된 문장에서도 부착의 문제를 쉽게 해결하여 구문 애매성을 줄일 수가 있다. 본 논문에서는 한국어 문장에서 아주 많이 출현되는 복문 구조의 문장을 구문 분석에서 상대적으로 처리하기 간편한 단문 구조로 변형하여 처리함으로써 구문 분석의 오류를 최소화하는 구문 분석기를 제안한다.

2. 관련 연구

기존의 단문 분할 방법은 문장이 복잡해질 때 구조적,

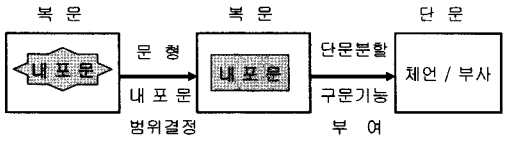


그림 1 내포문의 단문 분할에 의한 복문의 단문화

의미적 애매성을 해소하기 위해 공기 정보나 패턴 정보, 부가적인 한국어의 구문적 특성을 이용한 방법[3-6]들이 연구되어 왔다. [3]은 동사구 장벽 알고리즘을 이용하여 지역적인 동사구 내에서 가능한 결합을 구성하여 영한 번역에 필요한 구를 생성하는 방법이다. 이 방법은 언어 의존적인 규칙 개발이 필요할 뿐만 아니라 제한된 수의 문장 분석에서도 82% 정도의 정확성 밖에 보이지 않으며 부사화는 다루지 않고 있다. 또한 가장 오른쪽에 위치하는 본동사를 중심으로만 설명되기 때문에 문장의 내포 관계나 생략된 성분들의 보충 관계 등을 설명하지 못하고 있다. [4]는 한국어 문장을 같은 의미를 가지는 여러 개의 단문으로 분할했고 [5]는 관형형 용언 바로 다음의 용언이나 이유, 시간 등을 나타내는 구 바로 다음에서만 구간 분할을 시도했으며 [6]은 대등 접속문을 구간 분할하여 각 구간 별로 구문 분석을 수행한 후 통합하여 전체 문장에 대한 구문 구조를 구한다.

그러나 다음과 같은 점에서 본 논문과 차이가 있다. 본 논문에서는 문형 정보와 휴리스틱을 이용하여 내포문을 분할하며 내포문의 범위는 기존 논문에서 다루었던 관형절만이 아니라 부사절과 명사절을 포함한다. 또한, 분할된 내포문이 문장 내에서 하나의 구문적 기능을 가진다는 것이다. 예를 들면 다음의 나)문장에서 명사절 "영희가 효녀임을"은 문장 전체에서 목적어의 기능을 한다. 또한 다)문장에서는 부사절 "소리도 없이"가 부사의 기능을 한다. 이와 같이 분할된 내포문이 하나의 구문적 기능을 하며 의미 분석이나 기계 번역을 위해서는 이러한 정보가 분석 결과에 반영되어야 한다는 점이다.

나) 철수는 [영희가 효녀임을] 안다. - 목적어
다) 바람이 [소리도 없이] 분다. - 부사

또한 기존의 연구는 체언구를 중심으로 용언의 패턴 정보를 분류했기 때문에 부사가 필수적으로 필요한 경우에는 부사구 부착의 문제가 발생할 수 있다. 예문 라)에서 부사 "성가시게"는 용언 "굴다"와 "보다"에 부착이 가능하다. 그러나 본 논문에서는 부사구도 고려한 문형 정보를 사용하여 이런 부사구 부착 문제를 해결한다.

라) 철수가 [성가시게 구는 영희를] 보았다.
문형 : N이 ADV-게 굴다.

3. 한국어의 특성 : 구문 분석 관점에서

3.1 한국어의 형태론적 특성

한국어는 형식 형태소가 많이 발달하였고 여러 형태소들이 결합하여 하나의 구문적 단위를 이루는 경우가 많다. 이러한 형태소 열은 형태론적 애매성과 구문론적 애매성의 원인이 된다. [7]에서 제안한 구문 형태소는 구문 분석을 위한 하나의 단위가 되기 때문에 구문 분석의 효율을 높일 수 있다.

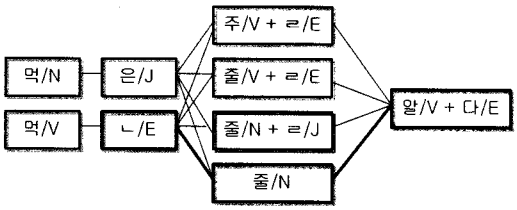


그림 2 "먹은 줄 알다"의 형태소 결과

그림 2는 "먹은 줄 알다"에 대한 일반적인 형태소 분석 결과로서 8개의 형태론적 애매성이 발생한다. 그러나 [7]이 제안한 구문 형태소를 이용하면 "ㄴ 줄 알다"라는 형태소열이 결합하여 '추측'이라는 양상 정보(modality)로 표현된다. 따라서 "먹다/pvg [추측]"이라는 하나의 결과만을 얻을 수가 있다. 이와 같이 구문 형태소는 형태론적 애매성을 해결하는데 도움을 준다. 따라서 본 논문에서는 구문 형태소를 구문 해석의 입력 데이타로 사용한다.

3.2 한국어의 구문적 특성

한국어는 생략이 자주 발생하고 자유 어순을 갖는 비구조적 언어이다. 또한 용언에 따라 다양한 격조사를 요구한다. 따라서 문장의 구조를 파악하기 위해서는 정형화된 구문적 정보만을 이용할 수는 없다.

*1) 철수가 귀찮게 군다. 2) 철수가 군다.**
3) 철수가 순이를 대표로 뽑다.
*4) 철수가 순이를 대표에게 뽑다.**

1)과 2)에서 '군다'는 자동사이므로 주어만을 필수 성분으로 간주할 수 있다. 그러므로 1), 2)는 옳은 문장으로 분석이 된다. 그러나 '군다'라는 용언은 '어떠하게'라는 의미를 가지는 부사를 문장의 필수성분으로 요구한다. 따라서 2)는 의미적으로 올바른 문장이 아님을 알 수가 있다. 또한 3)과 4)에서 '뽑다'는 "~로"라는 조사가 올 수 있지만 "~에게"라는 격조사는 타당하지 않다. 이러한 현상은 '군다'나 '뽑다'라는 용언에 한정된 것이 아니다.

이와 같이 한국어는 부사나 특별한 격을 수반하는 용언이 많이 존재한다. 이러한 용언의 경우 나머지 격을 보조적인 의미로 파악하기 때문에 문장의 올바른 의미를 파악하기 어렵거나 애매성 발생의 원인이 된다[2,8]. 따라서 이러한 용언들의 구조적 유형을 어떤 틀로 제약할 필요가 있다. 이를 문형이라고 한다. 한국어의 경우, 구문 분석에서 이러한 문형 정보의 이용은 필수적이라고 여겨진다[2].

3.3 한국어의 구조적 특성

한국어 문장은 용언 중심의 언어로 모든 체언구나 부사구는 용언의 지배를 받는다. 또한 2개 이상의 용언으로 구성되는 복문의 구조가 많다. 예로 KIBS[1]의 16,195 문장을 분석해 본 결과 중문과 복문의 구성 비율[1]이 표 1과 같다.

표 1 문장의 구성 비율

문장 수	중문	복문			단문
		관형절	명사절	부사절	
16,195	12,334	10,661	1,912	2,013	1,551
100 %	76.2 %	65.9 %	11.8 %	12.4 %	9.6 %

중문은 단문과 단문이 대등적 연결어미나 종속적 연결어미 등에 의해 연결된 문으로 내포문의 범위는 연결어미를 가지는 용언까지로 하면 된다. 중문에서는 생략이 문제가 될 수 있지만 전체 주절에 대한 처리를 할 때 생략된 부분을 보완한다[6]. 다음의 예문 마)에서는 용언 "먹다"의 문형에 의해 "철수가"가 "먹다"의 주어로 인식을 하면 "부르다"의 주체는 생략되었음을 알 수가 있다. 이 경우는 전체 주절의 의미상의 주어를 "철수가"로 설정하며 의미 분석이나 기계 번역에서는 이를 활용한다.

마) (철수가 밥을 먹고) 노래를 부른다.
문형 : N이 N을 먹다.
N이 N을 부르다.

그러나 주절과 내포문으로 구성되는 복문은 내포문의 범위를 어떻게 결정하느냐에 따라 체언구나 부사구 부착의 문제가 발생된다. 이를 해결하기 위하여 본 논문에서는 문형 정보를 제약조건으로 사용하는 단문 분할 방법으로 내포문의 범위를 정하고 주절과의 관계와 기능에 따라 명사절, 관형절, 부사절로 분류한다[2]. 이를 정리하면 다음의 표 2와 같다.

1) 한 문장에서 종속절이 여러 개인 경우에는 각각 계산함
예) 밥을 먹은 철수를 보기가 쉽다 — 관형절, 명사절

표 2 내포문의 구분

구분	세분화
명사절	인용 명사절
	'ㅁ/기' 명사절
	'지/냐' 명사절
관형절	관계 관형절
	동격 관형절
	의존 관형절
부사절	부사성 활용 어미에 따라

4. CFG 기반의 문형

4.1 문형의 분류

한국어의 문장은 보어와 수식어로 구성되어 있다. 보어는 문장을 구성하기 위해 반드시 필요한 성분을 말한다. 수식어는 보어 성분을 꾸며주는 역할을 하며 생략할 수가 있다. 따라서 문형을 결정하기 위해서는 보어와 수식어를 구별하는 기준이 필요하다[8,9]. 이를 위해 본 논문에서는 다음과 같은 기준을 사용한다.

1) 용언의 통사/의미적 충족성 : 보어는 용언이 지니고 있는 통사적, 의미적 요건을 충족시켜야 한다. 예를 들어 "철수가 성가시게 군다"라는 문장에서 "군다"라는 용언은 "adv-게"라는 부사구가 의미적으로 반드시 필요함을 알 수 있다.
   - 철수가 성가시게 군다
   - 철수가 군다*
2) 생략 불가성 : 보어로 쓰이는 단어가 생략될 경우, 문장의 구조나 의미를 파악하기 힘들다. 따라서 보어는 절대 생략될 수 없다.
   - 철수가 성가시게 군다
3) 반복 불가성 : 보어는 필요한 것만 추가되며 반복할 수 없다. 특정한 필수격으로 사용되는 보어는 한 문장에서 용언에 대하여 중복되어 나타날 수 없다.
   - 철수가 밥을 먹는다
4) 도치 불가성 : 순서를 도치시켰을 때 문장이 성립하지 않으면 보어이다. 한국어에서 도치는 빈번하다. 그러나 도치되면 문장의 의미가 변하는 경우, 이는 문장의 용언에 대한 보어로 간주할 수 있다. 아래 두 번째 예제의 경우, 문어체 문장에서는 첫 번째 문장처럼 사용하는 것이 일반적이다.
   - 철수가 순이를 며느리로 삼았다.
   - 철수가 며느리로 순이를 삼았다*
5) 종속절의 구조 : 종속적 연결어미에 의해 문장의 서술기능을 보완한다.
   - 나는 "철수가 학교에 간다"라고 말했다

문형만을 가지고 한국어 문장을 구문 분석하기에는 몇 가지 문제점이 있다. 한국어는 같은 의미적 계층을 가지는 용언이라도 개개의 용언에 따라 문형이 다르다[2,8]. 만일 문형이 같다고 하더라도 명사에 대한 제약이 있다. 따라서 문형과 더불어 명사에 대한 제약도 고려하여야 한다. 예를 들어 감각동사인 "맡다, 보다"의 경우 "N이 N을 V"라는 문형을 가지며 주어는 객체를 나타내는 명사가 될 수 있다. 그러나 목적어에 오는 명사들은 용언에 따라 제약이 따른다. '보다'의 경우에는 '구체물'이 오지만 '맡다'의 경우에는 '추상물'이거나 '냄새'라는 유일한 명사를 요구한다. 이러한 명사에 대한 의미지표는 문형에 대한 제약으로 반드시 필요하다.

맡다 : 주체가 냄새를 맡다 - 추상물, 냄새
보다 : 주체가 TV를 보다 - 구체물

이러한 의미지표를 가장 일반적으로 표현하는 것이 공기 정보이다. 그러나 코퍼스로부터 추출한 공기 정보는 자료부족 문제를 야기할 수 있다. 이는 용언과 체언이나 부사에 대한 부분적인 공기 관계만을 추출할 수 있음을 의미한다. 본 논문에서 구축한 문형은 [8,9]를 근간으로 보어와 수식어를 구별하여 기본 문형을 만들었으며 자료 부족 문제를 해소하기 위해서 명사와 용언 및 부사와 용언에 대한 의미 표지는 연세 한국어 사전[10]의 분류를 참조하였다.

4.2 문형의 예

문장의 중심인 용언을 기준으로 문형을 설정하였다. 용언은 다시 동사와 형용사, 서술격 동사로 구별하였고 [1]과 [10]을 활용하여 동사가 31개, 형용사가 8개, 서술격 동사가 5개로 총 44개의 문형을 설정하였다. 그림 3은 본 논문에서 분류한 문형의 일부이다.

V1) N(이/는/은/가)+V	A1) N(이)+A
V2) N(이)+N(에/에게)+V	A2) N(이)+N(에)+A
V3) N(이)+N(로/으로)+V	A3) N(이)+N(와)+A
:	:
V29) N(이)+S(기/음)+V	A6) N(이)+N(로)+A
V30) N(이)+S(기로)+V	A7) N1(이)+N(로)+N2(이)+A
V31) N(이)+S(기로)+N(에게/와)+V	A8) N1(이)+N(와)+N2(이)+A

그림 3 문형의 일부

4.3 문형 정보의 사전 구조

하나의 용언은 여러 문형을 가질 수가 있다. 또한 같은 문형이라도 용언에 따라 요구하는 체언이 다르다. 따라서 이러한 정보를 문형에 대한 의미제약으로 사용한다[10]. 본 논문에서는 문형과 의미지표를 다음과 같은 구조로 저장하여 사용한다. 의미지표는 문형에 오는 보

어의 수만큼 사용할 수가 있다.

문형의 사전 표기 :

단어 [&문형 [^의미지표]*]*

&, ^ : 문형과 의미지표의 구분자

* : 0번 이상 반복

문형 정보는 중첩을 포함하여 8762개의 엔트리(형용사:2337, 동사:6425)에 대해 문형 사전을 구축하였으며 하나의 엔트리당 평균 문형수는 1,41개(형용사;1.14, 동사:1.68)를 가진다. 가장 많은 문형을 가지는 용언은 "제시하다"로 8개의 문형(V1, V2, V3, V11, V12, V13, V26, v27)을 가지며 형용사의 경우에는 "가깝다", "관계없다", "떳떳하다" 등 16개의 엔트리가 3개씩의 문형을 가진다. 문형의 총 갯수는 동사가 10,771개이며 형용사는 2,664개이다. 빈도를 [문형의 수/문형의 총 갯수]로 표현하면 그림 4는 상위빈도를 가지는 문형의 일부이다.

동사	빈도	갯수	형용사	빈도	갯수
V1	31.08%	3,348	A1	83.1%	2,213
V11	35.49%	3,823	A5	10.1%	270
V2	12.54%	1,351	A2	4.5%	119
V12	6.0%	647			

그림 4 상위 빈도의 문형정보

예를 들어, 품사가 형용사인 "가깝다"는 A1, A2, A3 문형을 가진다. "가깝다"가 A1 문형일 때는 "N이"에 "장소"나 "사람"과 관련된 명사가 오고 A2 문형일 때는 "N이"에"새벽, 연말, 때, 사람, 시간"가 오고 "N에"는 "장소, 사람, 수"등에 해당되는 명사가 온다. 그리고 "A3"에서는 "N이"에 "사람"이 오고 "N와"에도 "사람"이 온다. 이를 문형 사전에 표기하면 다음의 그림 5와 같다. 사전 표기에 사용된 명칭은 [10]을 따르며 "사람"의 경우에는 "인명"과 사람임을 나타내는 모든 명사(삼촌, 이모 ... 등)가 포함된다.

```
가깝&A1^1장소,사람&A2^1장소,새벽,연말,때,사람^2장소,사람,수&A3^1사람^2사람
```

그림 5 "가깝다"의 문형 사전 표기

4.4 문형을 제약조건으로 하는 CFG

본 논문에서는 구문 분석을 위한 기본 틀로 구구조 문법에 기반한 조건 단일화 방법을 이용하였다[11]. 이는 구구조 규칙의 간결함과 구구조 의존적 언어의 특성을 조건 단일화 제약을 통해 문장을 분석하는 것이다. 아래는 구구조 규칙 "SV -> NP SV"가 적용되기 위해 필요한 제약들을 문형 정보 및 의미 지식을 이용하여 기술한 예를 보여주고 있다. NP는 체언구를 의미하여 SV는 용언이 포함된 문장을 의미한다. "철수가 밥을 먹었다"라는 문장을 분석하기 위해서는 우측의 <NP>에 "철수가"가 할당되며 x1의 값을 가지고 "밥을 먹었다"가 우측의 <SV>가 되며 x2의 값을 가지며 이 값을 이용하여 좌측의 <SV>로 단일화하게 된다. 먼저 x2를 x0로 단일화 한 후에 x1(철수가)의 격조사가 주격((x1 jform) =c jcs)이면 아래의 문장으로 진행하고 주격이 아니면 '*or*'의 역할에 의해 ((x1 jform) =c jco)의 문장으로 스킵(skip)하여 단일화를 진행한다. '((x0 subj) = *defined*)'의 문장은 x0(밥을 먹었다)의 주격(subj)이 이미 정의되어 있는지 체크하여 정의가 되어 있으면 x0의 문형 정보가 v6이나 v10인지를 검사((x0 sp-info) =c (*or* v6 v10))하는 과정을 진행한다. '*or*'의 역할은 격조사가 주격(jcs)일 수도 있고 목적격(jco)일 수도 있는 것처럼 여러 항목의 조건을 검사하기 위해 사용된다.

```
(<SV> -> (<NP> <SV>)    ;;; CFG r Rule
 ((x0 = x2)
  ((x0 svalue) <= (+ (x1 svalue) (x0 svalue)))
   (*or*
     (((x0 subcat) =c pvgi)
       (*or*
          (((x1 jform) =c jcs)
             (*or*
                (((x0 subj) = *defined*)
                ((x0 sp-info) =c (*or* v6 v10))
                 (*or*
                   (((x0 subj jform) =c jxc)
                    ((x0 comp) = x1))
                   (((x0 subj jform) =c jcs)
                    ((x0 comp) = (x0 subj))
                    ((x0 subj) = *remove*)
                    ((x0 subj) = x1))))
                (((x0 subj) = *undefined*)
                 ((x0 subj) = x1)))
                 ((x0 svalue) <= (+ (x0 svalue)
8)))))
          (((x1 jform) =c jco)
             (*or*
                (((x0 sp-info) =c v2)
                 ((x0 dest) = *undefined*)
                 ((x0 dest) = x1))
                (((x0 sp-info) =c (*or* v26 v27))
                 ((x0 about) = *undefined*)
                 ((x0 about) = x1))
```

그림 6 문형을 제약조건으로 하는 조건 단일화 기반 CFG의 예

5. 구문 애매성 해결

5.1 단문 분할 알고리즘

복문에서 내포문의 범위를 정하여 단문으로 분할하는 과정은 두 단계로 구성된다. 1단계는 필수격 결합 단계로 문형이 가지는 필수격들을 최대로 만족할 때까지 체언구나 부사구들을 결합한다. 2단계는 필수격이 모두 채워진 이후의 과정으로 필수격의 충돌이 발생할 때까지 남아있는 체언이나 부사구들을 용언에 부착한다. 이는 지역적으로 최대의 범위를 가지는 단문으로 분할을 하기 위한 것이지만 내포문에 부착된 체언구가 본용언의 필수격으로 사용되는 경우가 있기 때문에 분리해서 처리한다. 이를 pseudo code 표현하면 그림 7과 같다.

```
1) 용언을 만날 때까지 좌에서 우로 분석을 수행한다.
2) 용언을 만나면 관형형 어미를 가지는지 검사한다.
3) 관형형 어미이면 다음 체언구를 입력한다
4) 관형절의 유형을 파악한다
5) 문형의 필수격을 모두 채울 때까지 입력 정보와 단일화한다
   - 필수격을 모두 채운 시점까지를 단문으로 분할
6) 필수격의 충돌이 발생할 때까지를 내포문의 범위로 한다.
```

그림 7 단문 분할 과정의 pseudo code 표현

이 과정을 알고리즘으로 나타내면 그림 8과 같다.

```
clausal_segment()
{
    while (문장의 끝) {
      word = input_token( ) ;  // 구문 형태소 입력;
      if (word == 용언) {
         if (용언의 어미 == 관형형 어미) {
           input_next_token( );  // 체언구를 입력
           chk_rel_type( ) ;     // 관형절의 유형파악;
         }
         // 필수격이 충돌하거나 채워질 때까지 분할
         while (문형 정보의 필수격 충돌이 발생하거나
                입력 정보가 없을 때까지) {
           unify( );     // 용언과 단일화
           if (문형의 필수격) high_score 할당
           eles if (문형의 필수격 아니고
             문장의 필수격이 다 채워짐) low_score 할당
                else mid_score 할당
         }
         set_function( );  //분할된 단문의 기능 결정
    }  // 입력 문장의 끝
}
```

그림 8 단문 분할 과정을 알고리즘으로 표현한 예

5.2 단문 분할

내포문을 포함하는 문장에서 단문 분할 알고리즘을 적용하여 복문의 구조가 단문 형식으로 변환되는 과정을 관형절을 예로 들어 살펴본다. 기술의 편의를 위해 체언구는 'NP'로, 부사는 'ADV'로, 용언 중에서 관형형은 'PVM'으로 나머지는 'VP'로 표현하며 입력 순서에 따라 번호를 할당해서 사용한다.

NP1 ADV NP2 NP3 PVM1 NP4 VP1

바) 철수가 (자주 영수와 학교에서 싸우는 영희를) 보다.

문형) 싸우다 : N이 N와 V

보다 : N이 N을 V, N이 N을 N로 V

사) 철수가 거울로 (영수와 학교에서 싸우는 영희를) 보다.

(a) NP1
(b) NP1 ADV
(c) NP1 ADV NP2
(d) NP1 ADV NP2 NP3
(e) NP1 ADV NP2 NP3 PVM1
(f) NP1 ADV NP2 NP3 PVM1 NP4
(g) NP1 ADV NP2 NP3 [PVM1 NP4]
(h) NP1 ADV NP2 [NP3 PVM1 NP4]
(i) NP1 ADV [NP2 NP3 PVM1 NP4]
(j) NP1 [ADV NP2 NP3 PVM1 NP4]
(k) NP1 [ADV NP2 NP3 PVM1 NP4] VP1
(l) NP1 [[ADV NP2 NP3 PVM1 NP4] VP1]
(m) [NP1 [ADV NP2 NP3 PVM1 NP4] VP1]

바)문장을 구문 분석하는 과정으로, (e)는 용언(PVM1)이 입력되었어도 관형형 어미를 가지므로 단일화(앞의 체언구와 결합) 과정을 거치지 않고 다음의 체언구(NP4)를 입력한다. (g)에서는 관형형 용언과 후행하는 체언구를 단일화하고 관형절의 종류를 파악한 후, 문형 정보를 이용하여 (i)단계까지 필수격을 채워간다. (j)에서는 문형의 필수격이 모두 채워졌지만 필수격의 충돌이 발생하지 않았기 때문에 계속해서 가까운 용언에 부착해 간다. 결국에는 필수격의 중복이 발생하기 전까지 최대의 범위를 가지는 관형절을 하나의 단문으로 분할한다. 그리고 하나의 구문 범주의 기능(여기서는 목적어)을 할당한다. 그러면 복문의 구조가 단문 "철수가 영희를 보았다"의 구조로 변환된다.

5.3 가중치 할당

필수격의 충돌이 발생하기 전까지 모든 부사구나 명사구를 내포문의 용언과 단일화하는 방법은 문제를 야기한다. 예로 앞의 바)처럼 부사가 사용된 경우는 가까운 용언에 부착이 가능하지만 사)와 같이 체언구인 경우에는 두 가지 해석이 모두 가능하기 때문이다. 예문 사)에서는 체언구 '거울로'가 용언 '보다'에 부착되어야 의미적으로 타당하다. 따라서 이런 문제를 해결하기 위해 가중치를 도입한다. 가중치는 필수격일 때에는 높은

가중치(high_score)를 할당한다. 필수격이 아닌 경우에는 문형의 필수격이 다 채워졌으면 낮은 가중치(low_score)를 할당하고, 그렇지 않으면 중간 가중치(mid_score)를 할당한다. 가중치는 언어 의존적 휴리스틱 정보[6,7]와 문형에 의해 다양하게 부여되며 내포문과 주절 모두에 해당된다. 구문 분석 결과로는 가장 높은 가중치를 가지는 파스 트리만을 선택하고 나머지 정보는 그대로 보관한다. 본 논문에서 사용되는 휴리스틱 정보는 다음과 같다.

h1) 문두의 부사는 문장 전체를 수식한다.
h2) 가까운 거리에 있는 단어끼리 묶인다.
h3) 관형절에서는 하나의 필수 성분이 빠진다.
h4) 이중주어 문장에서는 보조사가 무조건 주어가 되나 조사의 형태가 같으면 앞에 것이 주어가 된다.
h5) 문두의 '은/는/ㄴ'은 본용언과 호응관계를 가진다.
h6) 부사절에서는 주어가 생략된다[9].

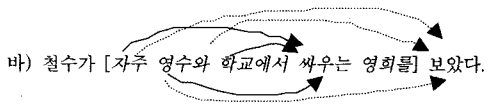
사) 철수가 거울로 (영수와 학교에서 싸우는 영희를) 보았다.
가중치 : 5 (1 5 3 5) -- 19
5 5 (5 3 5) -- 23

실제 시스템에서는 내포문인지의 정보와 휴리스틱 정보 등에 따라 가중치는 여러 가지 값이 사용되며 4.4절의 문법 규칙에서 기술된 "((x0 svalue) <= (+ (x1 svalue) (x0 svalue)))"가 가중치를 계산하는 부분이다. 이해의 편의를 위해 높은 가중치(5), 중간 가중치(3)와 낮은 가중치(1)를 설정해서 예문 자)에 적용시켜 보자. 먼저 내포문 "싸우다"의 문형은 "N이 N와 V"이므로 "영수와"와 "영희가"가 필수격이므로 5를 할당하고 "학교에서"는 필수격이 아니며 필수격이 다 채워지지 않은 상태에서 단일화가 이루어지므로 3을 할당한다. "거울로"는 필수격이 채워진 이후에 단일화를 수행하므로 1을 할당한다. "철수가"에서 필수격의 충돌이 발생하므로 "(거울로 영수와 학교에서 싸우는 영희를)"까지를 단문 분할하고 누적 가중치 14를 가진다. 주절은 "철수가 영희를 보았다"의 형태로 변환되었으며 "철수가"가 "보다"의 필수격이므로 가중치 5를 할당하면 총 누적 가중치는 19가 된다.

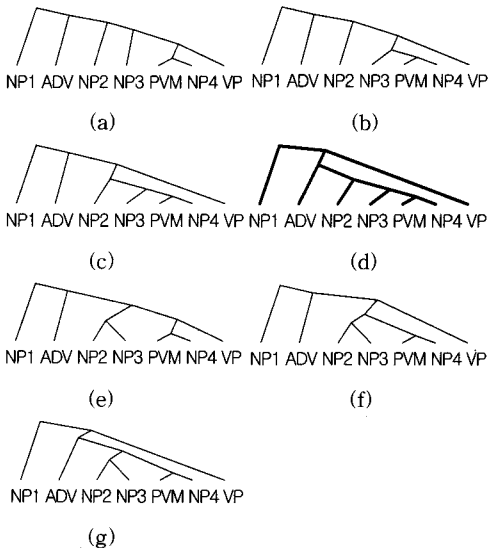
그러나 "보다"의 문형은 "N이 N을 N로 보다"도 있으므로 이 문형을 적용하면 "(영수가 학교에서 싸우는 영희를)"로 단문 분할이 이루어진 중간 결과를 사용하며 이때의 누적 가중치는 13을 가진다. 아울러 "거울로"는 "보다"의 필수격이므로 가중치 5를 할당해서 실행하면 총 누적 가중치는 23을 얻게 된다. 이와 같이 주절의 용언이 2개 이상의 문형을 가질 경우에는 각각의 문형에 대해서 구문 분석을 수행한 후에 누적 가중치가 가장 높은 것을 분석 결과로 생성한다. 이렇게 함으로써 "철수가 거울로 (영수와 학교에서 싸우는 영희를) 보았다"와 "철수가 (거울로 영수와 학교에서 싸우는 영희를) 보았다"는 분석이 모두 생성되지만 높은 가중치를 가지는 "철수가 거울로 (영수와 학교에서 싸우는 영희를) 보았다"는 결과만이 출력된다.

5.4 문형과 단문 분할을 이용한 구문 분석

문형 정보와 단문 분할 원리는 한국어 문장에서 부사구 부착이나 체언구 부착에 의한 부착의 문제를 해결하는 제약 조건으로 사용할 수가 있다. 예로 바)의 문장은 부사구 "자주"나 체언구 "영수와", "학교에서"가 어느 용언에 부착하느냐에 따라 다음과 같이 7개의 애매성을 가지지만 본 논문에서 제안한 방법으로 구문 분석을 수행하면 결과 (d)만이 파스트리로 출력된다.



문형) 싸우다 : N이 N과 싸우다
보 다 : N이 N을 보다



아울러, 문형을 이용하면 이중주어나 이중 목적어 문장을 처리할 수가 있다. 예문 아)는 이중 주어 문장이고, 예문 자)는 이중목적어 문장이다. 자동사나 타동사의 정보만을 이용하면 구문 분석에 실패하지만 문형을 이용하면 구문 분석할 수가 있다.

아) 철수가 돈이 모자란다.
문형) N1이 N2이 V
자) 어머니가 철수를 아침을 굶겼다.
문형) N1이 N2을 N3을 V

6. 실험 및 평가

6.1 실험 및 분석

한국어를 구문 분석하기 위해서 동사, 형용사, 서술격 동사로 용언을 세분하여 문형 사전을 구축하였고, 조건 단일화 기반의 CFG를 이용하여 문법 규칙을 기술하였다. 또한 관형형 어미를 가지는 용언은 후행하는 체언구와 먼저 단일화 연산을 수행하도록 하기 위해서 일반 용언과 분류하여 기술하였다. 이렇게 기술된 문법 규칙은 총 79개이다. 일반적인 구구조 기법보다 문법 규칙의 수가 적은 것은 본 논문에서는 어절 단위의 분석과 구문 형태소, 자질 정보, 대분류된 품사 태그를 이용하고 문법 규칙은 이진 문법으로만 기술하였기 때문이다.

실험을 위해 KIBS에서는 단문 분할이 가능한 10어절 이내의 700문장을 추출하고 초등학교 사회 교과서[12]에서는 300문장을 추출하였다. 추출된 문장은 평균 3.01개의 용언(보조 용언은 제외)을 포함하는 복문의 구조를 가졌다. 구문 애매성을 제약하기 위해서 다음과 같은 3가지 방법으로 실험을 하였으며, 각각의 실험 결과에 따른 평균 파스 트리의 수(구문 애매성의 수)는 표 6과 같았다.

실험1 : 일반적인 구문 분석 방법으로 [11]을 사용
실험2 : 구문 형태소와 문형을 이용한 단문 분할 방법
실험3 : 문형과 지역적으로 단위화한 단문 분할법

표 3 실험에 따른 평균 구문 애매성의 수

실험문장	평균 용언수	어절수[2]	실험1	실험2	실험3
KIBS(700)	3.55	9.62	68.43	18.21	6.93
사회(300)	2.47	11.35	51.25	21.04	7.04
평 균	3.01	10.49	59.84	19.63	6.99

실험3에 의해 생성된 결과를 분석해 보면 신문이나 교과서보다는 소설에서 발췌한 문장에서 오류가 많았다. 이는 "철수의 손잡이가 달린 가방", "꽃이 아름다운 공원을 보았다"와 같이 의미 정보를 요하거나 주어가 생략된 문장이 많았기 때문이다. 또한 사회 교과서에서는 명사들의 나열과 관형격 조사 및 보조사가 많이 사용되었기 때문에 KIBS보다 애매성이 많았다. 명사들의 나열인 경우에는 명사구의 범위에 의해 많은 애매성이 발생하였고, 보조사에 의한 불확실한 격 정보 때문에도 많은 구문 애매성이 발생하였다. 예를 들어 "조선의 궁궐과 가옥", "조선의 역사와 현재"는 같은 구조로 되어 있지만 "조선의 [궁궐과 가옥]"과 "[조선의 역사와] 현재"와 같이 분석 결과가 다른 구조로 되는 경우도 있어서 최상위 결과로 정확하게 분석하는데 어려움이 있었다.

표 4 실험3에 의해 생성된 최상위 구문 트리 결과의 정확도

실험 문장	평균 구문 트리 수	오분석 수	정확도
KIBS(700)	6.93	9	98.71%
사회(300)	7.04	5	98.33%
평 균	6.99	14	98.52%

6.2 실험 평가

본 논문에서 제안한 방법으로 구문 분석을 수행하면 구문 애매성의 수가 평균 59.84개에서 6.99개로 88.32%나 줄어들었다. 이는 테스트 문장이 대체로 10어절 내외의 문장임에도 불구하고 문형을 제약조건으로 한 단문 분할 방법이 한국어의 구문 분석에 매우 효율적임을 보여준다. 또한 분석에 사용된 컴퓨터는 Pentium4이며 CPU는 2.4GHz이고 RAM은 512MB를 사용하였다. 실험에 소요된 시간은 실험1이 평균 2.06초인데 반해서 실험3은 1.09초 소요되었다. 또한 구문 분석한 결과 중에서 올바른 후보가 생성되지 않은 경우는 2문장이었다. 이는 문형 정보만에 의한 구조적인 문제로 다음 절에서 논한다. 따라서 기존의 구문 분석 결과보다는 본 논문에서 제안한 시스템에 의한 구문 분석 결과가 정보 검색이나 기계 번역, 자연어 이해와 같은 응용 시스템에 보다 효율적으로 적용될 수 있다.

6.3 문형 정보만에 의한 구조적 문제

문형 정보를 이용하여 복합문의 구성 형식을 단문으로 바꾸어주는 단문 분할 알고리즘을 이용해서 구문 분석을 수행하면 많은 구문 애매성이 감소되었다. 그러나 문장의 구조적 유형만을 가지고 문형을 설정하면 잘못된 결과를 초래할 수도 있다. 예로 "아름답다"의 문형 정보만을 이용하면 아래의 2)문장은 1)문장과 같이 a)로 분석된다. 그러나 의미적으로는 b)가 타당하다. 이 문제를 해결하기 위해서는 "N이 N이 아름답다"라는 새로운 문형 정보를 추가하는 것을 고려해 볼 수가 있다. 그러나 이 경우에는 1)문장이 b)처럼 해석되는 구문 애매성을 초래한다. 따라서 이를 해결하기 위한 새로운 제약이 필요하다. 본 연구에서는 공기 정보를 이용하여 이를 해결하지만 차후에는 의미지식을 이용한 의미처리를 도입하고자 한다.

1) 철수가 [아름다운 공원을] 보았다.
2) 꽃이 아름다운 공원을 보았다.
   a) 꽃이 [아름다운 공원을] 보았다.
   b) [꽃이 아름다운 공원을] 보았다.
문형) 아름답다 <-> N1이 아름답다
      보다 <-> N1이 N2를 보다

2) 구문 형태소 단위로 형태소 분석된 결과에서의 어절 수

7. 결 론

부분 자유 어순을 가지고 기능어가 발달된 한국어를 구문 분석하기 위해서 문형을 제약 조건으로 하는 조건 단일화 기반의 CFG를 사용하여 단문 분할을 시도하였다. 문형은 내포문을 포함하는 문장에서 부사나 체언구에 의해 발생되는 부착의 문제를 해결하고 필수격을 필요로 하는 용언이나 이중 주어, 이중 목적어의 문제를 해결하는 데 좋은 제약이 됨을 보였다. 또한 문형 정보만으로 해결되지 않는 구문 애매성은 문형의 필수격에 의미를 제약하거나 공기 정보를 이용하여 해결하였다.

본 논문에서는 문형을 이용한 조건 단일화 기반의 CFG를 사용하여 어순이 자유로운 한국어 문장에서 발생하는 많은 구문 애매성을 해결할 수 있음을 보였다. 이는 일본어와 같이 문법을 기술하기 어려운 언어라도 문형만 파악된다면 효율적인 구문 분석이 가능함을 의미한다. 향후 연구 과제로는 지금까지 개발된 문형 규칙이 한국어의 여러 현상을 처리할 수 있도록 개선하는 것이다. 또한 의미 지표를 좀 더 세분화하여 문형에 대한 제약으로 활용하기 위한 연구가 필요하다.

참 고 문 헌

[1] KIBS : Korean Information Base System, http://kibs.kaist.ac.kr/kibs

[2] 이현영, 황이규, 이용석, "문형과 단문 분할을 이용한 한국어 구문 모호성 해결", 제 12회 한글 및 한국어 정보처리 학술대회, pp. 116-123, 2000.

[3] 신효필, "최소자원 최대효과의 구문 분석", 제11회 한글 및 한국어 정보처리 학술대회, pp. 242-247, 1999.

[4] 박현재, 이수선, 우요섭, "의미 정보를 이용한 이단계 단문분할 알고리즘", 제 11회 한글 및 한국어 정보처리 학술대회, pp. 237-241, 1999.

[5] 김광백, 박의규, 나동렬, 윤준태, "구간 분할 기반 한국어 구문 분석", 제 14회 한글 및 한국어 정보처리 학술대회, pp. 163-168, 2002.

[6] 장재철, 박의규, 나동렬, "구간 분할 기반 한국어 대등 접속 구문분석 기법", 제 14회 한글 및 한국어 정보처리 학술대회, pp. 139-146, 2002.

[7] 황이규, 구문 형태소를 이용한 형태소 및 구문 모호성 축소, 전북대학교 박사학위 논문, 2001.

[8] 서울대학교, "한·영동사의 하위범주화와 대응에 관한 연구", 한국전자통신연구소 최종 연구보고서, 1989.

[9] 장석진, 정보기반 한국어 문법, 도서출판 언어와 정보, 1993.

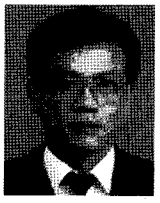
[10] 연세대학교 언어정보개발원, 연세한국어 사전, 두산동아, 1999.

[11] 양승원, 박영진, 이용석, "조건 단일화 기반 PATRII를 이용한 한국어 구문 분석", 한국정보과학회 논문지 Vol.22, No.4, pp. 653-662, 1995.

[12] 교육부, 사회 5-1, 국정교과서주식회사, 1995.

이 현 영

1991년 전북대학교 전산통계학과 졸업 1996년 전북대학교 전산통계학과(석사) 1996년~현재 전북대학교 전산통계학과 박사과정. 관심분야는 한국어처리, 정보검색, 데이타 색인



이 용 석

1977년 서울대학교 전자공학 졸업(공학사). 1979년 한국과학기술원 전산학 졸업(석사). 1995년 일본 국립도쿠시마대학교 지능정보 졸업(공학박사). 1979년 한국표준연구소 선임연구원. 1983년~현재 전북대학교 컴퓨터공학전공 교수. 관심분야는 자연어처리, 정보검색, 음성처리, 데이타 색인