최소자원 최대효과의 구문분석

신효필
CRL(Computing Research Laboratory), 뉴멕시코 주립대학
hshin@crl.nmsu.edu

Maximally Efficient Syntactic Parsing with Minimal Resources

Hyopil Shin
Computing Research Laboratory, New Mexico State University

요 약

이 논문은 지역적 동사구 분할에 바탕을 두고 서술어가 문말에 위치하는 언어에 특성에서 기인하는 속성을 반영하는 부분적 그러나 빠른 구문분석에 관해 논한다. 즉 완벽성 보다는 신속함 그리고 신뢰에 바탕을 둔 새로운 한국어 구문분석에 대해 논의한다. 기존의 문법이론 대신 한국어의 형태적 통사적 특성에 기인한, 성분들의 분할(partitions)에 의한 단위(chunks) 분석방법을 제안한다. 근간은 동사구 장벽(VP-barrier) 알고리즘이며, 이 알고리즘은 한 문장안에서의 다양한 동사의 파생접사에 의해 형성되는 관형화, 명사화, 부사화 등의 파생구조와 내포된 동사구(인용문, 종속문 등)에 의해 형성되는 지역적 동사구내에서 그 성분들의 논리적인 분할을 구성하고 다시 그 다음 요소와 체계적으로 결합하는 관계로 확장하여 가능한 구조들을 생성한다. 다시 언어의존적인 발견적 규칙(heuristics)들을 점수화하여 가장 높은 점수의 단위구조를 적격한 구조로 선택한다. 이 방법은 하위범주화 및 의미정보를 사용하지 않는, 빠른 구문분석이 요구되는 시스템을 위해 고안되었으며, 집단적인 노력이 아닌 개인적인 노력 및 최소의 자원으로도 최대의 효과를 얻을 수 있다는데 그 의의가 있다.

서 론

전통적으로 구문분석은 구구조문법이나 의존문법을 바탕으로 하여 이루어져 왔다. 언어의 특성상 한국어 구문분석은 의존문법이 선호되어 왔으며, 이에 대한 많은 연구가 진행되고 있다. 구문분석은 언어의 통사적 고찰 외에 의미정보, 하위범주화 정보 등 체계적인 기초자료들을 필요로 한다. 그러나 이런 자원들을 구축하는 것도 또한 오랜 시간과 광범위한 연구를 필요로 한다.

한편 인터넷의 급속한 성장은 기존의 정보검색 뿐만 아니라 교차언어 정보검색, 해당 정보요약, 그리고 더 나아가 원하는 언어로의 번역까지 통합된 시스템의 출현을 야기한다. 즉 기계번역 시스템과 다른 응용 시스템과의 결합이 요구되고 있는 것이다. 이러한 통합된 시스템에서 빠른 처리속도는 가장 절실히 요구되는 요소 중의 하나이나, 전통적인 기계번역 방법은 이러한 요구를 만족시키기에는 아직 부족하다. 기존의 기계번역에서 사용되는 '형태소 분석->구문분석->대상언어 변환-> 대상언어 생성'이라는 패러다임에서 가장 많은 노력이 요구되는 단계는 구문분석이기 때문에 신속하고 신뢰할 수 있는 구문분석에 관한 연구는 무엇보다도 중요하다고 할 수 있다.

본 연구는 완벽성 보다는 신속함 그리고 신뢰에 바탕을 둔 새로운 한국어 구문분석에 대해 논의한다. 기존의 문법이론 대신 한국어의 형태적 통사적 특성에 기인한, 성분들의 분할(partitions)에 의한 단위(chunks) 분석방법을 제안한다. 이 방법은 하위범주화 정보를 사용하지 않고 문장 성분들이 논리적 가능성에 의해 지역적으로 단위화된 후 언어의존적인 발견적 규칙(heuristics)에 의해 적격한 구조가 선택되는 기재로 되어있다. 근간은 신효필 (1999)에서 제안된 동사구 장벽(VP-barrier) 알고리즘이다. 이 알고리즘은 우선, 서술어가 문말에 위치하는 언어의 특성을 바탕으로 한 문장안에서 다양한 동사의 활용(관형화, 명사화, 부사화 등) 으로 형성되는 파생구조 및 내포된 동사구(인용문, 종속문)에 의해 형성되는 지역적 동사구내에서 그 성분들의 논리적인 분할을 구성하고 다시 그 다음 요소와 체계적으로 결합하는 관계로 확장하여 가능한 구조들을 생성한다. 다시 언어의존적인 발견적 규칙(heuristics)들을 점수화하여 이 생성된 구조들에 적용하여 가장 높은 점수의 분할을 적격한 구조로 선택한다. 이 구조를 근간으로 하여 변환과 대상언어로의 생성이 이루어 진다. 이 방법은 하위범주화 및 의미정보를 사용하지 않는, 빠른 구문분석이 요구되는 시스템을 위해 고안되었으며, 집단적인 노력이 아닌 개인적인 노력 및 최소의 자원으로도 최대의 효과를 얻을 수 있다는데 그 의의가 있다.

1 한국어 의존적 특성

한국어는 교착어이며, 서술어가 문말에 위치하는 언어이다. 어간에 굴절, 파생 접사들이 붙어 다양한 문법관계들이 표현된다. 그 중에서도 동사[1]의 굴절 및 파생은 아주 다양한 형태로 나타난다. 이 파생접사는 동사를 관형어, 부사어, 그리고 명사로 각각 전성시키는 기능을 한다. 서술어의 활용에 의해 시제, 서법, 상 등의 문법관계가 표현된다.

한국어 언어 자료, 특히 말뭉치에 관해서 세심히 고찰해 보면, 의외로 서술어 하나로 이루어진 문장은 극히 드물다는 것을 알 수 있다. 형용사의 서술적 속성에 기인하여 나타나는 관형절, 동사에서 파생된 부사, 그리고 명사화 어미 'ㅁ/기' 등에 의해 실현되는 명사화 등, 원래의 동사가 파생접사에 의해 다른 성분으로 전성되어 문장안에 다양하게 내재되어 있다. 이런 고찰이 지역적 동사구들을 중심으로 단위화(chunk)하는 방법론의 기초가 된다.

이 논의에서는 한국어 문장들은 일련의 자질(features) 들의 연쇄로 구성되며, 형태소 분석시 동사구들은 형태소적 자질과 품사에 의해 인식되고 다음의 파생 그리고 굴절에 의한 동사구(verb phrase)들이 성분들의 단위화의 장벽이 되는 지역적 구의 근간을 이루는 것으로 파악한다.

1. 관형/관계구(Adnominal phrase)
2. 명사화구(Nominalized Phrase)
3. 등위접속구(Conjunctive Phrase)
4. 종속구(Subordinate Phrase)
5. 인용구(Quotative Phrase)

특히 관형화, 명사화와 같은 파생과 관련하여, 부사화는 제외하고, 이들 요소의 전성된 성분이 아니라 원래 성분을 중시한다. 즉 이들을 동사구로 파악하고, 이 동사를 중심으로 지역적인 성분들을 분할하고 단위화하여 생성된 구조에 언어의존적 규칙들을 적용하여 최적의 분할구조를 선택한다. 이 선택된 구조는 변환과정과 생성을 거쳐 영어로 번역된다. 이 방법은 신효필(1999)에서 제시된 동사구 장벽 알고리즘(VP-barrier algorithm)을 근간으로 한다.

2 동사구 장벽(VP-barrier) 알고리즘

형태소 분석시 각각의 어절의 기본형 (base), 표면형 (surface), 품사, 그리고 자질(features)[2] 등이 도출된다. 이 기본형을 바탕으로 사전 검색이 이루어지고 한국어에 대응되는 역어가 등재되며, 사전 검색에 실패한 경우 한국어의 전사화 (transliteration)[3]에 의해 한국어 철자가 알파벳으로 표기되어 역어로 등재된다.[4] 이 결과는 동사구 장벽을 근간으로 하는 구문 분석 및 변환의 입력이 된다.

구문분석시 가장 먼저 행해지는 것은 명사구 단위화(noun phrase chunking)이다. 한국어 명사구는 다음의 규칙을 근간으로 다양하게 이루어 진다.[5]

(1) a. 기본 명사구
(PRO-의 | ADN | AP | (NUM CL-의?) | VP-rel)*N
b. 확장 명사구
NP NUM CL
NP(plural) N(중)-의 NUM CL
(NP-의)* NP | PP
(N)*N

명사구의 경계를 정하는 것도 복잡하고 세밀한 관찰을 요한다. 대부분의 경우 명사구의 오른쪽의 경계는 조사의 출현이(관형격 조사는 제외) 단서가 되지만 왼쪽 경계는 구분이 쉽지 않다. 위의 명사구 구성 중 '관형절 + 명사' 구성(AP, VP-rel)은 명사구 인식시 배제된다. 이는 원래 성분인 동사구로 파악하고 그 확장에서 분할이 인식된다. 명사구 인식에 관한 더 세밀한 관찰도 여전히 필요하다.

명사구를 먼저 인식하는 것은 핵심어인 명사와 이에 부속하는 여러 수식어, 의존성분들을 결합하여 그 의존관계를 밝히기 위함이다. 따라서 개개의 명사구가 단위화된 후 문장 속에서 남게 되는 성분은 명사구,

---

[1] 이 논의에서 동사는 서술어의 개념으로 사용된다. 따라서 동사는 형용사도 포함하는 개념으로 인식되며, 때에 따라 동사와 서술어를 함께 사용한다.

[2] 형태소 분석결과 도출되는 자질들은 대략 다음과 같다:

Case features: Nominative, Genitive, Accusative, Complementive, Vocative

Adverbial function Features: Comitative, Quotative, Connective, Oblique_Goal, Oblique_Source, Oblique_Destination, Oblique_Direction, Oblique_Status, Oblique_Comparative, Oblique_Means, Discourse_Topic, Discourse_Emphasis, Discourse_Inclusive, Discourse_Additive, Discourse_Exclusive, Discourse_Adversative, Discourse_Contradictive, Discourse_Polite, Discourse_Contingency, Discourse_Similiarity, Discourse_Selective, Discourse_Condition, Discourse_StartingPoint

Predicate Morphological Features: Past, Conjunctive, Subordinative, Connection, Auxiliary_Connective, Future, Adnominalized_Present, Adnominalized_Past, Adnominalized_Future, Nominalized, Adverbialized, Declarative, Interrogative, Imperative, Propositive, Pomisitive, Retrospective, Self_lowering, Honorific.

[3] 한국어 로마자 표기는 Yale 방식에 의해 행해진다.

[4] 형태소 분석 및 사전 검색은 이 논의와 직접 연관이 되지 않기 때문에 자세히 논의하지 않도록 한다. 형태소 분석기는 포항공대의 KOMA를 근간으로 하여 형태/통사적 자질을 출력하도록 수정을 하였다.

[5] 표기는 regular expression에 기초한다. PRO는 대명사류(지시대명사 포함)를 나타내며, ADN은 관형어를 지칭한다. 관형어는 언어학적 개념 양화사(quantifiers)를 포함한다. AP는 형용사에서 파생된 관형구성이며, 이와 대조되는 동사구 관형화(VP-rel)도 명시되어 있다. NUM 과 CL은 각각 수사와 분류사(classifier)를 나타낸다. 분류사에 해당하는 것으로는 '-명, -인, -개, -권, -자루, -잔, -병, -달, -시간, -시' 등이 있다. '?'는 수의적인 출현을 의미한다.

부사, 동사 등이다. 이 중에서 부사는 단위로 뭉쳐질 때 동사와 결합되므로 실제로는 명사구와 동사 두 주요 성분만 남게 된다.

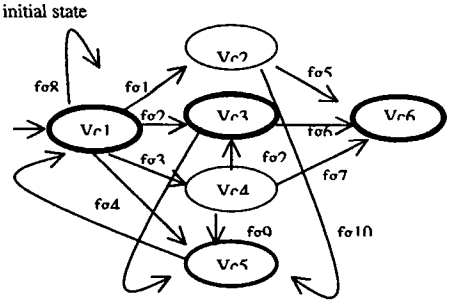
2.1 동사 결합과 장벽

형태소 분석후 한 문장안에 여러 개의 동사가 인식되면 서술어가 문말에 위치하는 언어의 특성상 가장 오른 쪽 동사가 본 동사로 간주된다. 한편 한국어에서 동사가 인접하여 나타날 수 있는데, 이는 파생접사에 의해 다른 문장성분으로 파생되는 경우외에 조동사 구성, 그리고 등위 구성에 의해서도 생성된다.

(2) ...추측해 보아도 크게
V_auxconn V_subordinative V_advz

어긋남이 없을 듯하다.
V_nomz V_adnom V_derive_to_verb

이 문장에서는 근원적으로 모두 여섯 개의 동사가 파생되거나 굴절되어 연접한다. 이 중에서 '본동사+ 조동사' 구성이나 '등위' 구성 등은 한 단위로 묶여져 취급되어야 하며 다음의 동사구와 분리되어야 한다. 이와 같이 하는 이유는 동사군을 인식하고 분리하는 것이 지역적 동사구의 오른쪽 경계를 인식하는 데 필요하기 때문이다. 이와 같이 인접한 동사들을 결합시키고 분리하기 위해 다음의 오토마타를 구성한다.



Vc1, Vc3, Vc6: final state and **VP-barriers**
Vc2, Vc4: not final state
Vc5: return to initial state, **V-barriers**
Vc2-Vc6, Vc2-Vc5, Vc3-Vc6, Vc3-Vc5, Vc4-Vc6, Vc4-Vc5: verbal chunking
fg(feature groups) ={aux_connective, subordinative, connective, conjunctive, adnominalized, nominalized, adverbialized, derive_to_copula}
vfg(verbal features except fg) ={aspect, modality, voice}
fg1={conjunctive},
fg2={adnominalized}
fg3={aux_connective}
fg4={nominalized, Quotative}
fg5={subordinative, connective, vfg}
fg6={derive_to_copula, derive_to_verb}
fg7={subordinative, connective, conjunctive, vfg}
fg8={adverbialized}
fg9={nomialized, quotative}
fg10={derive_to_copula+nominalized, derive_to_copula+quotative}

그림 1. 동사결합과 장벽 오토마타

동사 연쇄가 나타날 때 이 오토마타에 의해 동사들이 결합되고 분리된다. 이 중 **동사장벽**(V-barrier, vc5)은 **동사구장벽**(VP-barrier, Vc1, Vc3, Vc6)과 구분되어야 한다. 동사장벽은 동사구라는 점과 파생구조에서도 기인할 수 있다는 점에서는 동사구장벽과 같지만, 명사화(nominalization)된 구나 인용문과 같이 본동사의 논항으로 사용되는 구조는 다음의 인접요소와 결합관계를 갖지 못하고 명사구와 같은 역할을 하는 것으로 파악한다. 즉 관형절이나 종속절이 문장안에서 그 다음 요소와 맺는 관계와 구분하기 위해서이다. 이 동사장벽은 지역적 장벽을 구성하지 못하고 다시 원래의 출발 상태(Vc1)로 돌아가는 구성이다.

이 오토마타에 의한 (2)의 연쇄는 다음과 같은 동사구([ ])와 동사군(< >)으로 분리된다.

(3) [..Vc1-<Vc4-Vc6>] [Vc1-Vc1-Vc5-Vc1-<Vc3-Vc6>]

2.2 지역적 동사구에서의 분할

하위범주화 정보를 사용하지 않는 구문분석은 특히 구성성분들의 경계를 파악하는 데 많은 어려움을 야기한다. 이 동사구-장벽 알고리즘에서 오른쪽 경계는 동사군 형성과 장벽 오토마타에 의해 구분된 동사군이다. 그러나 왼쪽 경계는 하위범주화정보 없이는 구분짓기가 불가능하다. 명시적인 하위범주화를 사용하지 않는 본 논의에서는 [Beale 1998]에서 제안된 절 단위화(clause chunking) 방법을 근간으로 한다.

한국어에서 동사는 문말에 위치하기 때문에, 단위화는 문말 동사에서 왼쪽 요소로 진행된다. 일반적으로 새로운 단어는 그 이전 구에 속하거나 아니면 새로운 구를 형성한다. 예를 들어 다섯 단어가 있는 문장에서 마지막 넷 단어의 가능한 분할은 다음과 같다. [Beale 1998]

(((W2 W3) W4) W5)

즉, W2 와 W3 은 결합하여 절을 형성한다. 이 절은 다시 W4 와 결합하여 절을 형성하고 다시 W5 와 결합하여 새로운 절을 구성한다. 다음은 W1 이 이에 더해질 수 있는 모든 가능한 경우이다.

1. (((W1 W2 W3) W4) W5): W1 이 (W2 W3)에 연결
2. ((W1 (W2 W3) W4) W5): W1 이 ((W2 W3) W4)에 연결
3. (W1 ((W2 W3) W4) W5): W1 이 (((W2 W3) W4) W5)에 연결
4. ((((W1) W2 W3) W4) W5): W1 이 (W2 W3)단계에서 새 절을 구성
5. (((W1) (W2 W3) W4) W5): W1 이 ((W2 W3) W4)단계에서 새 절을

구성

6. ((W1) ((W2 W3) W4) W5): W1이 (((W2 W3) W4) W5)단계에서 새 절을 구성

그러나 이런 일반적인 분할 규칙은 모든 가능한 단위 결합을 생성하기 때문에 다음과 같은 기하급수적인 수의 단위를 생성한다.

$$2 \times \left(2^{n-1} \times \frac{(n-2)(n-1)}{2} \right)$$

예를 들어 한 문장에 10 개의 문장요소가 있다면, 총 18,432 가지의 가능한 분할을 갖는다. 그러나 동사구 장벽 알고리즘을 바탕으로 한 분할 방법은 이러한 결점을 극복한다. 위와 같은 분할 및 단위화는 일차적으로 동사구 내에서 지역적으로 일어나고 다시 이 동사구들이 인접한 오른쪽 요소들과 체계적으로 결합한다. 한 문장안에 널리 분포되어 있는 관형절, 명사화된 절, 인용문, 종속절 등 지역적 동사구 들은 자체내에 기껏해야 몇 개의 요소만을 갖기 때문에 그 가능한 결합은 제한되어 있다. 다음의 예문과 간략히 표기된 형태소 분석 결과를 살펴 보자.

(4) 해방 이후에 전적으로 우리말로 우리의
np np_adv np_adv np_adv np_gen

학문을 건설해야 할 필요에 직면하여
np_acc v_sub v_adn np_adv v_sub

이를 위한 노력이 여러 분야에서
np_acc v_adn np_nom adn np_adv

이루어져 왔다.
v_auxcon v_aux

모두 17 개의 요소가 있는 이 문장은 어말 동사로부터의 일반적인 단위화를 이룬다면 처리하기 어려울 정도의 가능한 결합을 구성한다. 그러나 지역적으로 이런 결합을 이룬다면 사정은 달라진다. 명사구 결합과 동사군 형성후에 다음과 같은 자질 연쇄와 지역적 동사구 장벽들이 형성된다.

(5) NP1 NP2 NP3 NP4
| | | |
해방이후에 전적으로 우리말로 우리의학문을

Vc1 NP5 Vc2 NP6
| | | |
건설해야할 필요에 직면하여 이를

Vc3 NP7 NP8 Vc4
| | | |
위한 노력이 여러분야에서 이루어져왔다

표기의 간략상 위의 구조를 알파벳으로 대치하면 다음과 같다.

(6) {NP1 NP2 NP3 NP4 Vc1] NP5 Vc2] NP6 Vc3] NP7 NP8 Vc4}
| | | | | | | | | | | |
A B C D v1 E v2 F v3 G H v

이 문장은 주동사는 Vc4이며 3개의 지역적 동사구로 구성되어 있다. 동사구 장벽이론의 중요한 언어 의존적 고찰은 한 지역의 동사구 내의 성분들은 그 지역적 술어에 속하거나 또는 문장의 주동사의 성분이 되지, 인접한 다른 지역적 동사구의 성분이 되지 못한다는 점이다. 따라서 분할에 있어 'a1 a2 (a3 Head)]' 구조는 a3는 지역적 동사 Head에 속하고 a1, a2는 이 지역적 구조가 아닌 본동사에 직접 속한다.

분할은 각 지역적 동사구에서 가능한 모든 구성을 이루도록 행해진다. 편의상 위의 (5)의 구조를 알파벳으로 대치한 (6)의 구조로 설명하도록 한다.

(7) A B C D v1] E v2] F v3] G H v}
a. A B C D (v1)] E (v2)] F (v3)] G H v}
b. A B C (D v1)] (E v2)] (F v3)] G H v}
c. A B (C D v1)]
d. A (B C D v1)]
e. (A B C D v1)]

위의 지역적 가능한 분할 구조에서 본동사와 그 성분들(G,H)은 분할 될 필요가 없다. 왜냐하면 이 성분들은 당연히 본동사에 속하기 때문이다. 분할은 각 지역적 동사구에서 그 모든 가능한 경우를 생성한다. 하위범주화 정보를 이용하지 않기 때문에 이러한 분할은 필수불가결하다. ( )로 묶여지는 성분들이 하나의 단위(chunk)가 된다. 첫번째 구(v1이 동사)는 다섯 가지의, 두번째 구(v2가 동사)는 두 가지의, 세번째 구(v3이 동사)도 두 가지의 가능한 논리적 결합을 이룬다.

2.2 지역적 동사구의 확장

지역적 동사구의 요소들은 위의 (7)과 같이 자체적으로 가능한 논리적 분할에 의해 단위화 된 후, 각각의 구성들은 왼쪽에서 오른쪽으로 그 다음 동사구에 있는 첫 요소들과 결합되어 그 구조가 확장된다. 결합되는 그 다음 요소는 언제나 명사구 아니면 동사이다. 다음 성분이 명사구이면 이전의 동사구는 그 명사를 수식하는 관형절이거나, 아니면 그 명사와 직접적인 관계가 없는 종속절이다. 다음 성분이 동사인 경우는 이전의 동사구가 종속절인 경우와 인용문과 같이 보충절인 두 경우가 가능하다. 따라서 크게 다음 요소에 내포되는 경우(관형절)와 단순히 병치되는(종속절, 또는 보충절) 두 가지의 경우가 각각의 지역구와 그 다음요소 사이에 가능함을 알 수 있다. 위의 (7)의 예에서 3 개의 지역적 동사구는 각각 다섯 가지, 두 가지, 세 가지의 가능한 지역적 분할을 갖는다. 첫번째 지역적 구의 다섯 가지 분할은 각각 그 다음 구의 첫 번째 요소와 결합할 때 두

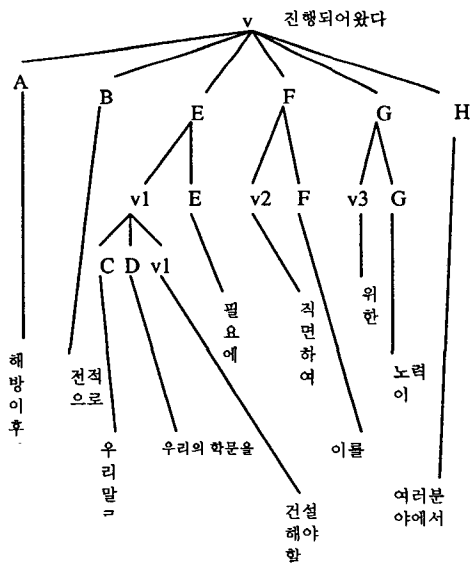
가지 가능성-내포(관형절), 병치(종속, 보충)- 에 의해 10(=5*2) 개의 확장된 구를 구성한다. 두번째 구는 두 가지의 가능한 구성이 있으므로 실제로는 20(=2*(5*2)) 가지의 경우가 생긴다. 이와 같이 하면 이 지역적 구들이 결합될 때 ((((5*2)*2))*2))*2))*2=160 가지의 가능한 분할들이 생성됨을 알 수 있다. 이 분할은 모든 가능한 분할을 생성하는 일반적인 단위화에 의해 생성되는 92,160 가지보다 현저히 적은 숫자이다.

지역적으로 단위화된 구가 다음 구와 결합하는 과정을 간략히 살펴 보면 다음과 같다. 편의상 첫 지역적 구로는 (7)-c 를 다른 두 지역적 구조로는 각각의 첫번째 경우를 예로 든다.

(8) a. A B (C D v1)] E (v2)] F (v3)] G H v}
b. A B ((C D v1) E) (v2)
c. A B ((C D v1) E) ((v2) F) (v3)
d. A B ((C D v1) E) ((v2) F) ((v3) G) H v

a는 첫번째 구의 단위화가 (7)-c의 구성임을 나타내고 그 다음 요소 E에 내포되는 구조(b), 이 구조가 v2에 병치되고 v2가 F에 내포되는 구조(c), 그리고 다시 이 구조가 v3와 병치되고, v3이 G에 내포되는 구조(d)이다. 이 예는 160 여 가지의 가능성 중 하나이며, 내포는 같은 구로 ( )로 결합되지만, 병치의 경우는 그렇지 않음을 알 수 있다. ( )로 결합되지 않은 성분들은 전체 서술어에 귀속되는 성분이다. 위의 구조는 다음과 같은 구문분석 수형도로 표시될 수 있다.

그림 2. 구문분석 수형도



그러나 위의 구조는 적격한 구조가 아니다. 적격한 구조는 다음의 언어 의존적 휴리스틱스(heuristics)에 의해 결정된다.

2.3 언어의존적 휴리스틱스

단위화는 맹목적인 논리적 가능성에 의해 가능한 결합들을 생성하기 때문에 여전히 중의적인 분석결과가 도출된다. 이러한 분할들은 언어의 특성에 기인한 휴리스틱스 (heuristics)에 의해 더 여과되어야 한다. 본 논의에서는 이러한 규칙을 바탕으로 벌점을 부여하는 방식을 채택한다. 따라서 가장 높은 점수를 갖는 분할이 적격한 구조로 선택된다. 다음은 채택되고 있는 몇 예비적인 한국어 휴리스틱이다.

1. 같은 구에 같은 격 자질이 나타나면, 벌점 0.3 을 부여한다.
2. 동사가 [adnominalized] 자질을 갖고 그 다음 명사에 내포되지 않은 분할에, 벌점 0.8 을 부여한다.
3. 동사가 [subordinative] 혹은 [conjunctive] 자질을 갖고 그 다음 명사에 내포되어 있으면, 벌점 0.8 을 부여한다.
4. 첫번째 지역적 동사구에서 주어가 없고 목적어가 있는 분할에, 벌점 0.2 를 부여한다.
5. 첫번째 지역적 동사구에서 주어, 목적어가 없는 분할에, 벌점 0.2 를 부여한다.
6. 주어가 주동사에 분할되지 않으면, 벌점 0.4 를 부여한다.
7. 주어, 목적어 외의 부사어 기능의 어구가 주어 목적어에 선행하는 분할에 벌점 0.3 을 부여한다.

이상의 휴리스틱으로 각 분할은 점수를 갖게 되고 가장 높은 점수가 할당되는 분할이 적격구조로 선택된다. 최고의 점수가 여러 개의 분할에 나타날 때는 가장 긴 단위 구성을 택하도록 한다. 그러나 위와 같은 휴리스틱과 벌점은 절대적이지 않고 언어현상에 바탕을 둔 상대적인 개념이다. 따라서 정확한 구조로의 분석은 절대적으로 위와 같은 휴리스틱과 정확히 할당된 벌점에 기인한다. 따라서 언어의존적인 위와 같은 휴리스틱을 설정하는 것과 더 조절된 벌점이 정확성을 높이는 중요한 척도가 된다.

3 구 현

이 구문분석 방법은 뉴멕시코 주립대학 CRL(Computing Research Laboratory)의 한-영 기계번역 시스템에 구현되고 있다. 이 기계번역 시스템은 정보검색 및 요약 그리고 기계번역 시스템을 결합한 일종의 통합시스템의 일부이다. 개인 사용자가 영어 질의어를 입력하고 한국어로 번역된 이 질의어 중 해당되는 것을 선택하면, 말뭉치(인터넷

동아일보)에서 해당자료가 검색되고 요약된 후 이 한국어 문장들이 영어로 번역되고 그 결과가 출력된다. 이 시스템은 한번에 10 개씩 요약된 텍스트를 출력하는데 평균 소요시간은 2분 미만(Sun Sparc Station Ultra-4에서) 이다. 그러나 실제 번역에만 소요되는 시간은 이보다 적다.

부록의 그림은 실제의 결과를 보여준다. 개인 사용자가 'America'라는 질의어를 입력하면, 말뭉치에서 이와 관련된 색인된 한국어 어휘가 리스트로 제시되고 이 중 한 가지나 몇 가지를 선택하면 관련된 텍스트들이 선택되고 요약되어 영어로 번역된 결과가 제시된다(부록의 그림 3 참조).

번역의 질(quality)은 아직 더 개선될 필요가 있다. 번역의 질을 높이기 위해서는 구문분석 뿐만 아니라 번역사전 및 변환 그리고 영어 생성에도 더 많은 노력을 기울어야 한다. 위의 예에서도 사전 검색에 실패한 단어들이 한국어 철자를 바탕으로 알파벳으로 전사되고 있으며(그림 3에서 'wunpannunglyek'(운반능력) 등), 영어 표면형 도출에 있어서도 잘못 생성된 형태들(그림 3에서 'agreeded' 등)이 존재한다.

그러나 기존의 방법들이 많은 중의성에 의해 번역에 실패하거나 , 긴 처리 시간을 요구하는 반면 이 방법은 최상의 분할구조를 제시하여 영어구조를 생성하기 때문에 실패율이 낮다. 이러한 시도는 광범위하고 신뢰할 수 있는 언어자료들이 이용가능하지 않을 때 제한된 자원과 노력으로 최대의 효과를 얻는 구문분석을 가능하게 하며, 현재까지의 실행결과는 매우 고무적이다.

4 전망 및 결론

지금까지 살펴 본 동사구 장벽이론을 근간으로 한 분할 및 단위화 시도는 하위범주화 정보없이 일련의 성분들의 결합관계를 구축하려는 시도이다. 서술어가 문말에 위치하는 언어에서, 문장의 핵심이 되는, 가장 오른쪽에 위치하는 본동사로부터 그 가능한 결합을 맹목적으로 구성하는 것보다, 언어 의존적인 특성에 바탕을 두고 문장내에 다양하게 내포되어 있는 지역적인 동사구 내에서 가능한 결합을 구성하여, 그 가능한 단위 구조의 경우를 줄일 수 있으며 또 오분석의 가능성도 낮출 수 있다. 이러한 시도는 맹목적인, 기계적인 단위화에 바탕을 두고 있기 때문에 이를 여과할 수 있는 신뢰할 수 있는 언어의존적인 규칙의 개발이 무엇보다도 중요하다고 할 수 있다. 실제로 제한된 수의 문장의 분석결과에서는 82% 정확성을 보이고 있다.(200 문장의 테스트시).

이 방법의 또다른 장점은 기존의 구문분석이론이 긴 문장이나 여러 문장성분이 섞여 있는 경우에, 그 중의성과 가능한 결합으로 분석에 어려움을 겪거나 실패를 하는 경우에 나타난다. 즉 지역적인 동사구내에서 기본적으로 단위화가 이루어지기 때문에 이 분석방법은 실제로 서술어가 하나인 경우보다는 문장속에 다양한 성분-관형절, 명사화절, 인용절, 종속절 -등이 나타나는 경우에 더 효율적이다. 지역적 동사구를 바탕으로 이루어진 단위화는 그 다음 성분들과 제한된, 체계적인 방법으로 결합하기 때문에, 구문분석의 성공율이 더 높다. 이러한 점은 기존의 의존문법이나 구구조문법등을 바탕으로 한 구문분석에 있어서도 부분적 구문분석 방법으로 도입될 수 있다.

참고문헌

신효필. 1999. *The VP-Barrier Algorithm for a robust Syntactic Parsing in Head-Final Languages*, to appear in NLPRS 1999.

Abney Steven. 1991. *Parsing by Chunks. Principle-Based Parsing*: Computation and Psycholinguistics, Kluwer Academic Publishers. 257-278

Beale Steve. 1998. *Expedition: Turkish-English MT.* manuscript. Computing Research Laboratory. New Mexico State University.

Melcuk, Igor A. 1988. *Dependency Syntax: Theory and Practice.* State University of New York Press.

Oflazer Kemal. 1999. *Dependency Parsing with an Extended Finite State Approach.* in ACL `99 proceedings. 254-260.

Ryu Beopmo. 1997. Using Local Dependency for Syntactic Parser of Korean. M.S. Thesis. POSTECH.

Shieber Stuart M. 1984. *The Design of a Computer Language for Linguistic Information.* Proc. of the 10th COLING, p. 362-366. Stanford University.

부록

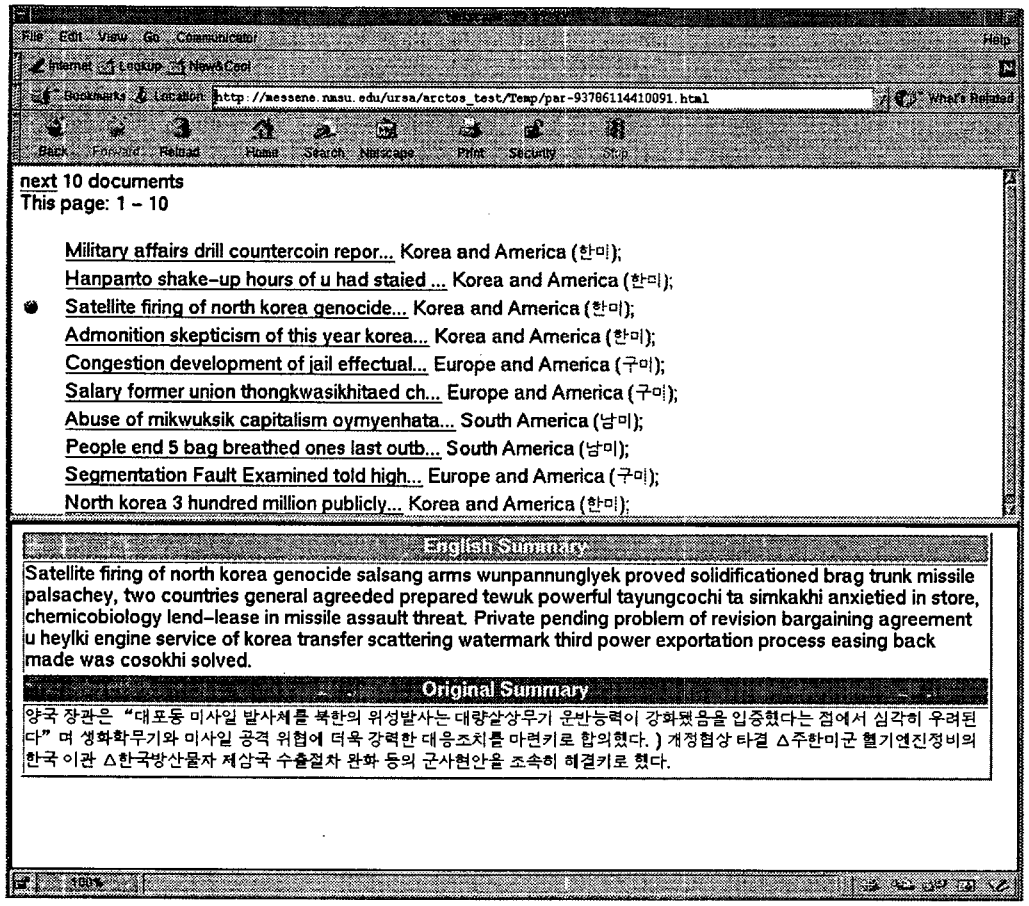


그림 3. 시스템 번역결과