□ 기술해설 □

통계적 방법을 이용한 구문분석†

서강대학교 서정연*
한국과학기술원 김창현**

목 차

1. 서 론
2. 말뭉치를 이용한 통계적 언어처리
 2.1 구문구조부착 말뭉치
 2.2 문법 규칙의 자동 학습
3. 통계적 구문분석
 3.1 문맥 자유 방식의 통계적 구문분석
 3.2 문맥의존 방식의 통계적 구문분석
4. 한국어에 대한 통계적 구문분석
 4.1 한국어 구문구조 부착 말뭉치
 4.2 통계적 구문분석
5. 결 론

1. 서 론

심리학 연구의 결과들은 인간의 언어 사용이 이전의 언어적 경험을 상대적 빈도수와 같은 형태로 인식하고 있으며, 현재의 언어사용은 이러한 경험적 빈도수에 영향을 받고 있음을 보여주고 있다. 자연어 처리에 통계적 방법을 적용하는 방법론들은 모두가 이러한 인식에 기반하고 있다고 할 수 있다. 구문 분석은 문장을 이루는 구문 요소들 간의 문법적 구조를 밝히는 작업으로서, 이 과정에서 필연적으로 발생하는 중의성을 처리하기 위해 많은 방법론들이 시도되고 있다. 통계적 방법을 이용한 구문분석은 규칙 기반의 방법론에서 다루기 어려운 문제들, 즉, 언어적 지식의 획득 및 확장의 용이성, 견고성(robustness)의 제공, 분석 결과에 대한 선호도의 표현 등에 대한 방법론들을 제공해 준다. 이러한 통계적 구문 분석의 시초는 1979년 T. J. Watson 연구소에서 시도한 통계적 문맥자유 문법(stochastic context-free grammar)이다[7]. 더구나 [7]은 자동으로 규칙을 획득하기 위해 inside-outside 알고리듬을 개발하였다. 이 알고리듬은 통계적 문맥자유문법의 매개변수를 구하기 위해 통계적 유한상태 오토마타의 매개변수 추정을 위한 Baum-Welch 알고리듬을 일반화한 것이다. 그러나 초기의 이 연구에서는 신뢰성 있는 매개변수를 구하기가 쉽지 않았으며, 이후 신뢰성 있는 매개변수를 구하기 위해 많은 연구들이 진행되었다. 또한 문맥자유문법의 언어적 정보 표현의 확장을 위해, 일반적인 통계적 문맥 자유문법에 해당 규칙이 적용될 수 있는 부가적인 언어적 문맥을 사용하는 연구가 진행되었으며, [27,28]이 그 초기의 연구들에 해당한다.

이와는 별도로 언어적 지식을 보다 효과적으로 모델링할 수 있는 문맥자유 문법체계들이 제안되었으며, 대표적으로 Lexicalized Tree Adjoining Grammar(LTAG), Tree-Substitution Grammar(TSB) 등이 있다. 이들은 문법규칙이 표현할 수 있는 언어적 구조 정보를 확장시킨 것이며, 통계적 방법의 적용에 대한 연구들이 진행중이다[33,34].

본 연구에서는 통계적 방법에 대한 연구의

† 본 연구는 과학재단의 목적 기초 과제 "한국어 이해에 나타나는 중의성 문제 처리 모델에 관한 연구"의 부분 지원을 받은 것입니다.

*종신회원
**학생회원

내용에 대해, 문맥 자유 방식의 연구로부터 문맥 의존 방법에 대한 연구까지를 차례로 살펴보기로 한다. 통계적 자연어 처리에서의 신뢰성 있는 매개변수를 구하기 위해서는 대량의 말뭉치가 요구된다. 형태소 분석 분야에서는 이미 대량의 품사부착 말뭉치(tagged corpus)를 이용한 연구가 상당히 진전되었음이 보고되었고, 구문 분석 분야에서도 Pennsylvania 대학의 Penn Treebank 등 통계적 구문분석을 위한 많은 구문트리부착 말뭉치(tree-tagged corpus)들이 구축되어 연구에 이용되고 있다. 영어권에서의 말뭉치 구축 연구와 아울러 한국어의 경우 이제 걸음마 단계인 말뭉치 구축 작업을 살펴보기로 한다. 마지막으로 한국어에 대해 이루어진 통계적 구문분석을 살펴보고자 한다.

2. 말뭉치를 이용한 통계적 언어처리

말뭉치를 이용한 방법으로는 대표적으로 언어 정보의 학습, 통계적 정보의 추출 등을 들 수 있다. [12]에서는 괄호가 매겨진 구문구조 말뭉치로부터 구문분석을 위해 미리 정의된 패턴에 따라 규칙을 학습하고 있으며, 적은 양의 말뭉치를 이용해 좋은 결과를 보이고 있다. 이때, 학습된 규칙은 통계적 정보를 포함하지는 않는다. 그러나 대부분의 말뭉치를 이용한 연구에서는 통계적 정보를 추출하여 사용하고 있으며, 이러한 통계적 정보는 분석 결과에 대한 선호도의 기본으로 이용된다. 또한 말뭉치로부터 자동 추출되는 규칙은 동일한 수준의 정보 자원으로부터 추출되므로 일관성이 유지된다는 장점을 지닌다. 본 장에서는 우선 이러한 통계적 정보의 기반이 되는 구문구조부착 말뭉치에 대해 살펴보기로 한다.

2.1 구문구조부착 말뭉치

언어가 사용되는 실제 내용을 들여다보면, 관찰에 의해서는 불가능하게 여겨지는 문장이나 발화들이 자연스럽고도 당연하게 쓰이는 예들을 종종 발견할 수 있으며, 반대로 문법적인 혹은 특정 언어현상의 설명을 위해 만들어낸 인위적인 문장이, 담화 구조 상의 부적절성이나 참조의 명시성, 또는 그 이외의 다른 언어적 제약에 의해 비문처럼 보이는 경우들도 있다[18]. 이러한 문제들로 인해, 관찰(introspection)에 의한 언어적 현상의 설명은 그 제약을 가질 수밖에 없으며, 언어 현상의 올바른 반영을 위해서는 언어현상을 그대로 표현하고 있는 말뭉치의 필요성이 대두된다. 언어 현상의 올바른 반영뿐 아니라 앞서 언급한 언어처리 상의 여러 필요성에 의해 말뭉치 이용의 요구는 더욱 커지게 된다.

말뭉치는 언어 구조의 제부문을 규정하는 법칙들을 귀납적으로 발견하는 데에 쓰일 수 있다[3]. 말뭉치는 법학, 의학 등의 제한적 언어 사용 행태나 또는 언어 일반의 대표적 사용 방식을 살필 수 있도록 일정한 기준에 의해 수집, 가공된 문서 집단이라고 할 수 있다. 이러한 말뭉치는 그 가공 정도에 따라 단순 문서(raw text), 품사부착 말뭉치(tagged corpus), 구문구조부착 말뭉치(tree-tagged corpus), 혹은 그 이상의 의미구조나 담화구조까지도 부착된 말뭉치로 세분할 수 있다. 그러나 말뭉치의 가공에 드는 비용의 제약으로 인해 단순 문서나 품사부착 말뭉치의 구축된 양에 비해 구문구조부착 말뭉치나 혹은 그 이상의 정보를 표현하는 말뭉치 구축은 상대적으로 적을 수밖에 없다. 본 연구에서는 말뭉치의 대상을 구문구조부착 말뭉치로 제한하기로 한다. 구문구조부착 말뭉치라 하더라도 가공의 정도에 따라 구분할 수 있다. 비교적 많은 정보를 포함하여 자세히 세분된 구문구조부착 말뭉치들로는 Leeds-Lancaster Treebank, Polytechnic of Wales Corpus, Gothenburg Corpus, Susanne Corpus, Nijmegen Corpus 등이 있으며, 대량의 좀 덜 세분된 구문구조부착 말뭉치로는 대표적으로 IBM-Lancaster Associated Press Corpus(100만 단어 수준), ACL/DCI Penn Treebank(수백만 단어 수준) 등이 있다. 또한 단순히 단어들 간의 구문관계의 규정 없이 괄호 넣기만을 말뭉치로 구축할 수도 있다. 각 말뭉치들은 대개 서로 다른 문법 형식을 채택하고 있으며, 따라서 구문구조 형태들이 서로 다르게 표현된다. 구축된 말뭉치들은 명시적인 표현 형태의 구분에 의해 구문 요소들이 괄호

로 묶인 형태와 숫자로 연결된 형태로 나뉠 수 있다. 다음은 괄호로 묶인 경우와 숫자로 연결된 경우의 각 예이다.

IBM-Lancaster Associated Press Corpus (Spoken English Corpus Treebank)
SK01 3 v
[Nr Every-AT1 three-MC months_NNT2 Nr], _, [here_RL [P on_II [N Radio-NN1 4_MC N]P]], _, [N I_PPIS1 N][V present_VV0 [N a_AT1 programme_NN1 [Fn called_VVN [N Workforce_NP1 N]Fn]N]V].

Polytechnic of Wales (POW) Corpus :
189
Z 1 CL FR RIGHT 1 CL 2 C PGP 3 P IN-THE-MIDDLE-OF 3 CV 4 NGP 5 DD THE 5 H TOWN 4 NGP 6 & OR 6 DD THE 6 MOTH NGP H COUNCIL 6 H ESTATE 2 S NGP HP WE 2 OM 'LL 2 M PUT 2 C NGP DD THAT 2 C QQGP AX THERE 1 CL 7 & AND 7 S NGP H WE 7 OM 'LL 7 M PUT 7 C NGP 8 DQ SOME 8 H TREES 7 C QQGP X TRERE

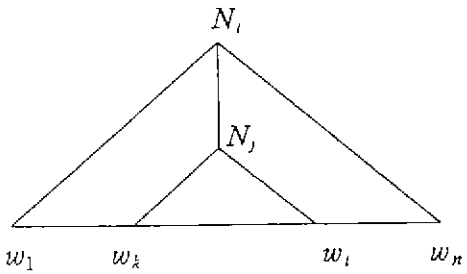
숫자 표현 방식의 장점으로는 하나의 구성성분(constituent)을 이루는 요소들이 문장 상에 연속적으로 나타나지 않더라도 손쉽게 표현할 수 있다는 것이다. 명시적이지는 않지만 좀 더 중요한 구분 요소로는 표현된 문법 정보의 세밀도이다. 가장 단순한 형태로는 구문 요소들의 이름을 전혀 부착하지 않은 채 괄호로 묶기만 한 형태가 될 수 있으며, 구문 요소들의 세분 정도에 의해 그 세밀도가 다르다. 심지어는 생략현상, 대용어 등도 나타낼 수 있다.

2.2 문법 규칙의 자동 학습

구문구조부착 말뭉치로부터 자동으로 문법을 추출하는 것은 단순하다. 문맥자유 구구조문법 형식을 이용하여 말뭉치의 구문 구조가 부착되어 있다면, 각 문장의 부모-자식 관계를 이루는 모든 관계를 구구조 규칙으로 추출한 후, 추출된 모든 규칙들에 대해 중복성을 제거하면 되는 것이다. 다른 문법 체계를 이용하더라도 이와 유사한 과정을 거치게 된다. 그러나, 여기서 한가지 의문점이 발생할 수 있다. 즉, 구문구조 말뭉치 구축을 위해 이용된 문법 규칙을 왜 다시 추출하려고 하는가 하는 것이다. 그러나, 실제로는 구문구조부착 말뭉치와 이를 분석하기 위한 문법 규칙 간에는 그러한 순환적 과정이 적용되지 않는다. 처음 구문구조 말뭉치 구축을 위해 이용되는 구문 규칙은 계산적으로 정확히 형식화되지 않으며 단지 구문구조 부착을 위한 지침일 뿐이기 때문이다. 또한 이렇게 구축된 구문구조부착 말뭉치로부터 추출할 수 있는 문법체계는 다양할 수도 있다. 즉, 구구조문법 형식을 이용해 구축된 구문구조부착 말뭉치로부터도 다른 형식의 문법체계, 이를테면 의존문법 형식의 문법 규칙을 약간의 변형을 거쳐 추출할 수도 있다. 그러나, 통계적 언어처리에서는 말뭉치를 이용한 규칙의 학습에서는 신뢰성 있는 확률 매개변수를 구하는 것이 무엇보다도 중요하다.

Inside-outside알고리듬[7]은 통계적 문맥자유문법의 매개변수 추정에 이용되며, 통계적 유한상태 오토마타의 매개변수 추정을 위한 Baum-Welch 알고리듬[8]의 확장된 형태이다. Inside-outside 알고리듬은 추정치 최대 알고리듬(expectation maximization algorithm)으로써, 확률값 추정을 위한 확률문법의 초기 상태로 임의의 초기값을 할당받는다. 확률 매개변수의 학습을 위해 학습말뭉치의 확률을 증가시키도록 규칙의 확률값을 반복적으로 변경

$$\beta_j(k,l) = P(w_{k..l} | N^j_{k,l})$$
$$= \sum_{q,p,m} P(N^j \rightarrow N^p N^q)\, \beta_p(k,m)\, \beta_q(m+1,l)$$



시킨다. 기본적으로 문법을 촘스키 정규형(Chomsky Normal From)이라고 가정하고, inside-outside 알고리듬을 소개하기로 한다. 임의의 비단말기호 N^j가 문장 내의 단어 w_k... w_l들을 생성해 낼 확률을 내부확률(inside probability)이라고 하며, 다음과 같은 재귀적 수식으로 유도된다. 그림은 내부 확률에 대한 도식이다.

내부확률과 유사하게, 문장 내의 단어 w_k ... w_l들을 생성하는 비단말기호 N^j와 그 이외의 외부단어열 w_1 ... w_{k-1}, w_{l+1} ... w_n이 생성될 확률을 외부확률(outside probability)이라고 정의하며, 그 식은 다음과 같다.

$$\alpha_j(k,l) = P(w_{1,k-1}, N^j_{k,l}, w_{l+1,n})$$
$$= \sum_{k,p,q} \alpha_{p(k,l)} P(N^p \rightarrow N^q N^l) \beta_q(k, k-1)$$
$$+ \sum_{m,p,q} \alpha_{p(k,m)} P(N^p \rightarrow N N^q) \beta_q(l+1, m)$$

각각의 규칙에 부여되는 확률은 문서로부터 자동으로 추출될 수 있는데, 학습 문서가 대상 언어를 대표한다고 가정하면 그 문서에서 규칙이 사용된 상대적 빈도 값을 해당 규칙의 확률로 볼 수 있다. 이때 문서 자체에는 규칙이 몇 번 쓰였는지가 명시적으로 드러나지 않으므로, 여기서 제시된 내부확률과 외부확률에 의해서 확률적으로 그 규칙이 몇 번 쓰였는지를 계산한다. 따라서 구문정보가 없는 문서로부터 규칙의 확률값을 학습시키는 것이 가능하다. 그러나 구분구조부착 말뭉치로부터 규칙이 쓰인 빈도수를 추출할 수 있다면 다음과 같이 단순히 규칙의 확률을 구할 수 있다.

$$P_e(N^i \rightarrow \zeta^j) = \frac{Count(N_i \rightarrow \zeta^j)}{\sum_k Count(N_i \rightarrow \zeta^k)}$$

즉, 특정 규칙의 확률은 해당 규칙의 발생 빈도를 동일한 비단말노드를 확장하는 모든 규칙의 발생 빈도로 나눈 것이 된다.

Inside-outside 알고리듬은 이미 존재하는 문법 규칙에의 확률값 할당을 위해 사용될 수 있으며, 확률 문법규칙의 자동 학습에도 이용할 수도 있다. 그러나 Inside-outside 알고리듬은 매개변수값의 지역 최적해(local optimum)는 보장하지만 전역 최적해(global optimum)는 보장하지 못한다. 따라서, 확률 문법규칙의 자동 학습에 이용될 경우 심각한 문제가 발생할 수 있다. [17]에서는 제한된 문맥자유문법의 자동학습에 대한 실험 결과를 보여주고 있다. 여기에서는 문법 규칙에 형태적 제약을 가하여 모든 가능한 규칙을 생성한 후 inside-outside 알고리듬을 적용하여 확률값을 구한다. 규칙의 초기값을 임의로 생성하여 얼마나 다양한 문법 규칙 집합들이 생성되는지를 시험한다. 만일 생성되는 규칙집합이 모두 동일하다면 지역 최적해가 아닌 전역 최적해가 생성되는 것으로 볼 수 있으며, 규칙집합이 다양하게 나타난다면 많은 지역 최적해가 생성되는 것으로 볼 수 있다. 이 실험에서의 두 개의 규칙집합의 동일성은 다음과 같이 정의되었다. 즉, 동일한 규칙집합이 생성되고 동일한 비단말노드를 확장하는 규칙들의 확률값 크기의 순서가 동일하다면 두 문법규칙집합은 동일하다고 정의된다. 실험결과 너무나 많은 지역 최적해가 있음이 나타났다. 300번의 규칙집합 학습의 결과 300개의 서로 다른 규칙집합이 나타난 것이다. 즉, 모든 학습이 지역 최적해를 생성해 낸 것이다. 따라서, inside-outside 알고리듬을 이용한 확률의 학습에는 학습을 올바르게 진행시킬 수 있는 추가적 정보가 필요하다.

3. 통계적 구문분석

언어 모형화에 통계적으로 추출된 확률값을 이용하여 중의성 해소 및 견고성 획득을 얻으려는 많은 방법론들이 연구되고 있다. 문법규칙의 확률값을 추출할 때의 정보에 의해 크게 문맥자유적인 방법과 문맥 의존적인 방법으로 분류될 수 있다. 문맥자유적인 방법에서는 규칙의 확률이 주위의 상황에 무관하게 항상 일정하며, 문맥의존적인 방법에서는 규칙의 확률이 주위의 문맥에 따라 달라진다. 본 장에서는 이 두 가지 방법을 이용하는 통계적 구문 분석의 연구들에 대해서 살펴보기로 한다.

3.1 문맥 자유 방식의 통계적 구문분석

대표적인 문맥자유 문법으로는 확률 문맥자유 구구조문법을 들 수 있다. 확률 문맥자유

구구조문법은 기본적인 문맥자유 구구조문법에 단순히 해당 문법규칙이 발생할 확률을 첨가한 문법이다. 이러한 확률 문법에서는 기본적으로 동일한 비단말노드를 확장하는 규칙들이 갖는 전체 확률값으로 1이 할당된다. 다음은 이러한 확률 문맥자유 구구조문법의 한 예이다.

S→NP VP	0.8	prep→like	1.0
S→VP	0.2	verb→time	0.1
NP→noun	0.4	verb→flies	0.4
NP→noun PP	0.4	verb→like	0.5
NP→noun NP	0.2	noun→time	0.2
VP→verb	0.3	noun→flies	0.3
VP→verb NP	0.3	noun→bullets	0.5
VP→verb PP	0.2		
VP→verb NP PP	0.2		
VP→verb	0.3		

예로, 비단말노드 S를 확장하는 두 규칙의 확률 합은 0.8+0.2=1이 된다. 이러한 확률 문법에서는 임의의 문장의 생성 가능성이 확률로 표현되며, 임의의 문장의 생성 확률은 해당 문장의 모든 가능한 구문구조의 확률 합으로 정의된다. 하나의 문장이 가질 수 있는 구문구조가 하나 이상이 될 수 있기 때문이다. 각 구문구조 생성 확률은, 해당 구문구조의 생성에 참여한 모든 문맥자유 문법규칙의 확률 곱으로 정의될 수 있다. 이것은 각 규칙들의 적용이 문맥 자유, 즉 서로 독립이라는 가정에 기반하고 있기 때문이다. 문장 "Time flies like bullets"에 대해 위의 확률 문법으로 분석된 결과 중의 하나가 다음과 같다고 하자.

위 파스 생성에 이용된 규칙들을 prefix 트리 운행으로 모두 기술해 보면 S→NP VP 0.8, NP→noun NP 0.2, noun→time 0.2, NP→noun 0.4, noun→flies 0.3, VP→verb NP 0.3, verb→like 0.5, NP→noun 0.4, noun→bullets 0.5이며, 따라서 위 파스의 생성 확률은 각 규칙들의 단순확률 곱인

$$0.8\times0.2\times0.2\times0.4\times0.3\times0.3\times0.5\times0.4\times0.5=1.152\times10^{-4}$$

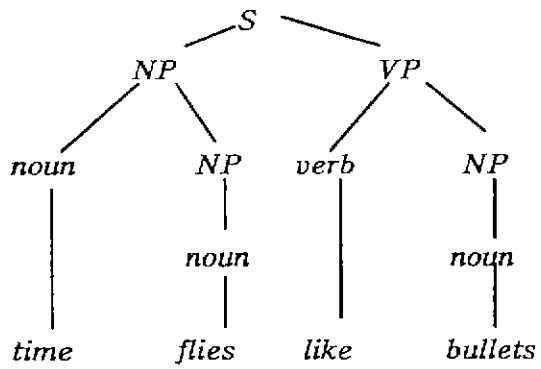
이 된다. 각 파스에 확률을 부여함으로써 얻을 수 있는 명시적인 효과는 중의성 해소의 측면이다. 문장의 여러 가능한 구문 구조에 대해 확률값에 의한 순서가 제공됨으로써, 각 구문구조들에 대한 해석 상의 상대적 우선순위의 기준이 제공될 수 있다. 그러나 이러한 일정 기준에 의한 우선 순위가 실제로는 그렇게 대단한 표현력을 보여주지는 못한다. 확률문법이 제공하는 우선 순위의 기본 원칙은 문법에 의해 표현되는 전체 언어 구조 중 많이 발생하는 구조를 갖는 결과들을 선호하겠다는 것이다. 그러나, 실제의 문장 구조는 전체 구조의 발생 가능성 못지 않게 각 문장에서 발생하는 어휘에 많은 영향을 받는다. 다음의 두 문장을 살펴보자.

They put the bird in the house.
They like the bird in the house.

영어의 경우 전치사구가 명사구에 부착될 확률이 동사에 부착될 경우보다 더 높다고 보고되고 있다. 따라서 위의 두 문장을 단순 문맥자유 확률문법으로 분석할 경우 동일한 구조를 갖게 되며, in the house가 두 문장 모두에서 bird에 부착되는 분석 결과가 더 높은 확률을 가질 것이다. 그러나 실제로 두 문장을 구분짓는 중요한 요소는 개별 동사와 명사, 전치사 in과 전치사 내부의 명사 간의 의미적인 측면이다. 문맥자유 방식에 적용된 규칙 간의 독립이라는 가정에 기인한 이러한 문제는 위와 같은 전치사구 부착 문제 등에 많은 영향을 미친다.

3.1.1 통계적 정보의 이용

본 장에서는 말뭉치로부터 추출한 통계적 정보를 이용하여 올바른 구문구조를 할당하려는



연구들에 대해서 살펴보기로 한다. 초기의 연구로써는 [39]에서 mutual information을 사용하여 구(phrase)의 경계를 결정하려 하였다. 단어간의 예상도에 의한 지역 최적해(local minima)가 구의 경계와 잘 일치한다는 것이 주된 가정이었다. 즉, 두 단어를 중심으로, 왼쪽 단어를 포함한 왼쪽 단어열과 오른쪽 단어를 포함한 오른쪽 단어열이 비교적 낮은 mutual information을 갖는다면 이 두 단어가 두 구(phrase)의 경계단어가 된다는 것이다. [30]에서는 [39]의 생각을 심화하여, 대량의 말뭉치를 이용하여 실험하였다.

[20]에서는 simulated annealing 기법을 이용하여 문장을 분석하였다. 우선, 임의의 구문구조를 입력으로 하여 이 구조의 질을 평가할 수 있는 평가함수가 정의된다. 그리고, 비단말 노드의 이름을 변경하거나 트리를 재구성하는 등의 일련의 변경 사항들이 정의된다. 마지막으로 simulated annealing 기법을 이용하여, 탐색공간을 이동하며 구문분석을 수행한다.

[13]에서는 점수를 갖는 문맥자유규칙을 자동으로 학습하기 위해 분포분석기법을 사용한다. a→b c와 같은 문법규칙의 점수는(a,b,c는 모두 품사임), 품사 a가 자신의 좌, 우 단어들과 갖는 분포와, 품사쌍 b c가 갖는 이들의 좌, 우 단어들과의 분포 간의 상대적 엔트로피(relative entropy)에 의해 결정된다. 따라서, pronoun→determiner noun과 같은 규칙은, 대명사(pronoun)와 명사구(determiner noun)가 갖는 분포가 유사하므로 높은 점수를 받게 될 것이다. 구문분석은 유사도를 최대로 하도록 두 쌍의 품사를 하나의 품사로 만드는 규칙을 반복적으로 적용하게 된다.

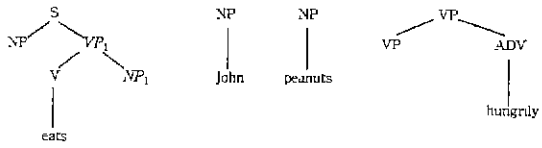
[32,35]에서는 inside-outside 알고리듬이, 매우 미약한 초기 문법지식에도 불구하고, 문서의 괄호 넣기를 비교적 높은 정확률로 수행하는데 이용될 수 있음을 보여주었다.

[9]에서는 inside-outside 알고리듬이, 언어학자들에 의해 쓰여진 문법을 확률문법으로 변경하는데 이용되었다.

3.1.2 지역정보의 확장

앞에서 살펴본 문맥자유 구구조문법의 가장 큰 단점은 어휘가 고려되지 않으며, 지역정보(local information)가 효과적으로 표현되지 못한다는 것이다. 이러한 단점을 보완하는 문법 체계로 등장한 것들이 있으며, 부분 트리를 기본 문법규칙의 단위로 사용하는 Tree-substitution Grammar와 Tree-Adjoining Grammar, 어휘 자체에 구문 속성을 표현하는 Lexicalized Tree-Adjoining Grammar (LTAG), Link Grammar, Categorial Grammar 등이 있다. 특히 어휘 자체가 문법 체계에 반영될 경우 유용한 통계적 모델의 구축이 가능하다. 물론, 충분한 매개변수 추정을 위한 말뭉치가 전제되어야 한다.

[33,36]에서는 LTAG를 사용하는 확률모델을 기술하고 있다. LTAG는 기본 트리에 하나의 어휘가 포함되어 있어, 해당 어휘의 지역정보(구문적 성향)를 표현하고 있다. 따라서, 문맥자유 구구조문법과는 달리 구문구조의 확률연산이 어휘를 반영하게 된다. 이때의 확률은 트리 결합확률로 정의되며, 따라서 각 기본트리는 부속(adjunction)이나 대치(substitution) 연산 수행에 대한 확률을 갖는다.



즉, 위와 같은 기본트리가 있을 때, VP_1 노드는 자신에 부속될 수 있는 VP를 근(root)으로 하는 모든 트리들과의 부속 (adjunction) 연산에 대한 확률 합이 1이 된다. 마찬가지로 NP_1도 자신을 대체할 수 있는 NP를 근으로 하는 모든 트리들과의 대체(substitution) 연산 확률 합이 1이 된다.

$$1.\ \sum_{\alpha \in I} P_I(\alpha) = 1$$

$$2.\ \forall_{\alpha \in I \cup_A} \forall_{\eta \in S(\alpha)} \sum_{\alpha \in I} P_S(S(\alpha, \alpha, \eta)) = 1$$

$$3.\ \forall_{\alpha \in I \cup_A} \forall_{\eta \in S(\alpha)} \sum_{B \in A \cup \{none\}} P_A(A(\alpha, \beta, \eta)) = 1$$

I는 기본트리집합, A는 부속트리집합을 의

미하며, $s(\alpha)$는 트리 α에서 대체가 가능한 모든 노드의 집합을, $a(\alpha)$는 트리 α에서 부속이 가능한 모든 노드의 집합을 의미한다. 또한, $P_I(\alpha)$는 처음 구문분석을 시작할 때 선택되는 기본트리의 확률을, P_S는 대체확률을, P_A는 부속확률을 의미한다. [36]에서는 통계적 LTAG의 확률매개변수 추정을 위해 변형된 inside-outside 알고리듬을 제시하고 있다. 통계적 모델에서는 고려되는 인자가 클수록 추정해야 할 매개변수가 많아진다. 따라서, 어휘를 포함하는 경우 신뢰성있는 매개변수의 추정을 위해서는 많은 말뭉치가 필요하게 된다. 이 때문에 LTAG를 이용한 통계적 구문분석 방법의 성공적 실험 결과는 아직 보고되고있지 않다.

[14]에서는 확률 문법으로 TSG(Tree Substitution Grammar)를 이용하였다. TAG와 유사하게 기본 규칙의 단위는 트리가 되나, 트리의 결합 연산은 대체만이 가능하다. [14]는 구문구조가 부착된 말뭉치로부터 모든 가능한 문법규칙(부분트리)을 추출하며, TSG에서의 확률 매개변수는 문맥자유 구구조문법과 마찬가지로 부분트리의 발생 확률로 정의된다. TAG에서는 하나의 구문트리를 생성해 내는 유도트리(derivation tree)가 하나 이상 존재할 수 있으며, 따라서, 가장 높은 확률을 갖는 최적 구문트리를 찾기 위해서는 지수승(exponential)의 시간이 소요된다. [14]에서는 이를 피하기 위해 최적해를 추정하는 Monte Carlo 기법[25]이 이용되었다. 실험에 이용된 말뭉치는 Penn Treebank 중의 Air Travel Information System(ATIS) 말뭉치를 수작업으로 수정한 750 문장으로써, 문장 유형이 질문과 명령형 뿐인 비교적 단순한 문장구조의 구문구조 부착 말뭉치이다. 학습용으로 사용된 675개 문장의 구문구조로부터 모두 40만개의 규칙(부분구문트리)이 대치 확률과 함께 추출되었으며, 75개의 문장에 대해 실험되었다. 평가 기준으로 완전일치(exact match)를 사용하였음에도 상당히 높은 정확률을 보였다. 그러나, 제한된 영역의 단순한 구문구조를 갖는 말뭉치에 대한 실험이었음에도 규칙의 수가 상당히 많았던 점을 감안하면 좀더 복잡한 구문구조를 갖는 영역의 말뭉치에 대해서는 정확률에 대한 결과를 예측할 수 없다. 학습용 말뭉치가 대상 영역의 구문구조를 대표할 정도가 되어야 위와 같은 실험 결과를 얻을 수 있으며, 이를 위해서는 많은 학습용 말뭉치가 필요하고, 또한 여기서 추출되는 구문규칙 또한 상당히 많아 질 것이다. 그러나, 비교적 간단한 구문구조를 갖는 제한된 영역에서의 통계를 이용한 성공적 구문분석은 주목할 만하다.

3.2 문맥의존 방식의 통계적 구문분석

문맥자유 방식의 통계적 구문분석에서는 규칙 적용의 확률이 문맥에 관계 없이 항상 일정하다. 그 대표적인 문제점으로는 앞에서 이미 지적한 바와 같이 더 자주 발생하는 구조가 선호될 뿐 실제의 문맥 내에서의 상대적인 규칙 선호도가 반영될 수 없다는 것이다. 따라서 문맥의존 방식의 구문분석에서는 규칙 확률에 문맥을 반영하자는 것이다. 여기서의 문맥은 해당 규칙의 적용에 영향을 미칠 수 있는 임의의 조건상황을 의미한다. 이해를 돕기 위해 우선 규칙 적용 시 어휘정보를 문맥정보로 이용하는 간단한 모델을 살펴보자. 구구조문법에서의 한 구성성분을 이루는 단어들 중 첫 번째 단어가 규칙 적용에 많은 영향을 미친다고 가정할 수 있다[6]. 즉, 규칙의 적용 확률이 첫 번째 단어에 따라 달라지는 $P(A\rightarrow B|C,w)$형태를 가지게 된다. 규칙 적용 확률은 다음과 같이 계산될 수 있다.

$$P(A\rightarrow B|C,w)=\frac{Count(A\rightarrow B|C,w)}{\sum_{A\rightarrow\alpha\in G}Count(A\rightarrow\alpha|C,w)}$$

이때, $Count(A\rightarrow B|C,w)$는 품사가 C인 단어 w가 규칙 $A\rightarrow B$의 첫 번째 단어로 발생한 횟수이고, $\sum_{A\rightarrow\alpha\in G}Count(A\rightarrow\alpha|C,w)$는 규칙의 부모 비단말노드가 A이고, 이 구성성분의 첫 번째 단어로 C가 발생한 규칙의 횟수이다. 다음은 이 문맥에 의한 규칙의 적용 확률을 계산한 결과이다[6].

결과적으로 규칙이 특정 단어에 의존적으로 적용되는 효과가 발생한다. 예를 들어 말뭉치 내에서 단수명사는 단독으로 명사구를 이루는 경우가 드물며, 반면에 복수명사는 다른 명사

	the	house	peaches	flowers
NP→N	0	0	.65	.76
NP→N N	0	.82	0	0
NP→NP PP	.23	.18	.35	.24
NP→ART N	.76	0	0	0

	ate	bloom	like	put
VP→V	.28	.84	0	.03
VP→V NP	.57	.1	.9	.03
VP→V NP PP	.14	.05	.1	.93

를 수식하는 경우가 드물게 나타난다. 이러한 관찰은 위의 명사 house, peaches에 그대로 나타난다. 단수명사 house가 단독으로 명사구를 이루는 규칙 NP→N에 나타날 확률은 0이고, 복수명사 peaches가 다른 명사를 수식하는 규칙 NP→N N에 나타날 확률도 0으로 나타나고 있다. 동사의 경우 문맥의존 방식에 의해 subcategorization 효과를 얻을 수 있다. 자동사 bloom인 경우 VP→V NP나 VP→V NP PP와 같이 목적어를 취하는 규칙에 발생할 확률은 V→NP에 비해 상당히 작으며, 전치사구를 필요로 하는 put의 경우 VP→NP에 비해 VP→NP PP에 나타날 확률이 상당히 크다.

문맥의존 구문분석에서는, 어떠한 적절한 문맥을 사용하면 어느 정도의 정확도 향상을 기대할 수 있는가 하는 것이 주요 관심사가 된다. 예를 들어, 전치사구 부착에는 동사의 종류뿐 아니라 전치사의 종류, 그리고 전치사구 내의 명사의 종류도 영향을 미치며, 이러한 문맥의 증가는 정확도의 향상을 가져올 수 있다. 그러나, 신뢰도 있는 확률값의 획득을 위해 더 많은 학습 말뭉치를 요구하게 된다. 문맥자유 구문분석에서는 규칙 상호간의 독립성이 가정되었으며, 방금 살펴본 형태에서도 규칙 상호간에는 여전히 독립이 가정되고 있다. 즉, 해당 구성성분의 첫 번째 단어를 문맥으로 이용하는 것은 규칙들 상호간의 독립성에 전혀 영향을 주지 못하기 때문이다. 또한 어휘 자체를 기반으로 하여 신뢰성 있는 확률 매개변수들을 구하기 위해서는 대량의 말뭉치가 필요하게 된다. 전치사는 그 수가 제한되어 있으며, 또한 전치사의 종류에 따라 구문구조에 미치는 영향이 서로 틀리다. 따라서, 전치사나 관사, 접속사와 같이 그 수가 제한되어 상대적으로 말뭉치에 자주 발생하는 단어들은 어휘 자체를 이용하더라도 신뢰성 있는 확률 매개변수를 얻을 수 있을 것이다. 그러나, 상대적으로 발생 빈도가 낮은 명사나 동사와 같은 단어들에 대해서는 일정한 범주로 묶어준다든가 하는 방법이 요구된다.

[27]에서 사용한 문맥의존 구문분석기인 Pearl은 구문구조에 확률분포를 적용할 때, 중의성 해결을 위해 문맥 정보를 고려하는 확률모델이 중의성 해결을 위해 자동으로 학습될 수 있음을 보여주고 있다. 이 실험에서는, 규칙기반의 중의성 해결을 시도하는 문법을 이용하여, 규칙 기반의 중의성 해결 정보들을 통계적 모델에 의한 가장 높은 확률을 갖는 일련의 문법 규칙들을 선택하는 알고리듬으로 대체한다. 여기서 이용된 규칙기반의 문법의 언어적 제약사항들로는, 규칙 VP→V PP를 적용할 때 PP의 전치사가 of이거나 동사가 eat일 때에는 이 규칙을 적용할 수 없다는 등의 정보를 가지고 있다. 물론 선택 제약에는 특정 어휘 뿐 아니라 어휘범주(품사)를 이용하기도 하며, 구문구조에서의 빈자리(gap) 생성에 관한 제약도 포함된다.

Pearl은 규칙 적용 시의 확률값 할당에 어휘 문맥과 구조문맥을 고려한다. 이때 이용되는 가정은, 문장 S에 대한 하나의 구문트리 T에 이용되는 규칙들에 대해, 각 비단말노드와 그 아들들은, 즉 하나의 문법 규칙은, 비단말노드의 형제와 부모, 그리고 해당 규칙이 적용될 때의 단어 품사를 중심으로 하는 trigram에 의존한다는 것이다. 이를 수식으로 표현하면 다음과 같다.

$$P(T|S) \simeq \prod_{A \in T} P(A \rightarrow \alpha | C \rightarrow \beta A \gamma, a_0\, a_1\, a_2)$$

여기서 C는 A를 지배하는 바로 상위의 비단말노드이며, a_1은 구성성분 A를 이루는 가장 왼쪽 단어의 품사이다. Pearl을 이용한 첫 번째 실험에서는 Voyager direction-finding 영역에서 학습문장 1100개, 실험문장 40개를 추

출하였다. 40개의 문장 중 38문장에 대해 구문트리가 생성되었으며, 이중 35개의 분석 결과가 기존의 규칙 기반의 분석 결과와 대등하였다. 즉, 전체 정확도는 88%이다. 실험 결과는 문맥의존 확률모델이 규칙기반 문법규칙에 표현된 제약사항들을 표현할 수 있음을 보여주는 것이었다. 그러나 적용된 실험 문장들이 단순하고 또한 매우 적은 관계로 실험 결과에 대한 신뢰도를 확신할 수 없다. 또한 동일한 실험 문장을 반복적으로 실험에 이용함으로써 문법확률이 실험문장들에 과도하게 적응되었을 가능성도 있다. 또한, 구문구조에 가장 중요한 역할을 하는 어휘정보가 포함되지 않은 점도 Pearl의 확률모델의 문제점을 보여준다.

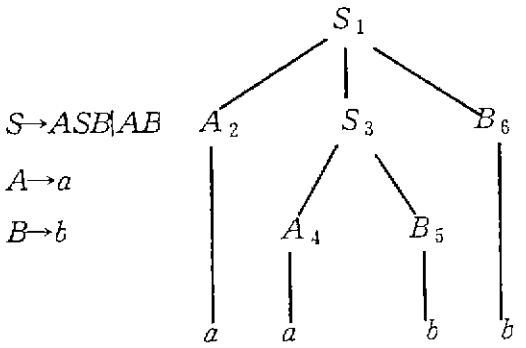
Pearl에 이어 Pearl을 수정, 확장한 문맥의존 확률모델인 Picky는 1000문장의 학습문장과 100개의 문장으로 이루어진 3개의 실험문장 집합을 이용하였다. 이 세 개의 실험문장 집합에 대한 평균 정확률은 Pearl에 비해 약간 높은 89.3%였다.

[5]에서도 품사 trigram과 부모규칙을 문맥으로 하는 문맥의존 구문분석을 수행하였다. Penn Treebank에서 선택된 8000개의 문장으로 이루어진 구문구조부착 말뭉치로부터 규칙과 해당 규칙의 문맥의존 확률을 추출하였다. 구문분석 과정에서는 참여 규칙의 수를 줄이기 위해 규칙의 적용 가능한 갯수와 규칙의 확률에 의해 규칙을 걸러낸다. 이는 다음과 같은 수식으로 표현될 수 있다.

$$\text{if}(Num(C|T_1) > e_1 \;\&\&\; P(A \rightarrow B\,\beta | a_0\, a_1\, a_2) < e_2)$$
$$\textit{then detete the rule}$$

구문분석 시점 T_1에서 적용 가능한 규칙의 개수가 e_1개보다 많고, 규칙의 문맥 확률이 임계치인 e_2보다 작으면 규칙을 제거하게 된다. 실험 문장은 학습말뭉치 내의 111개 문장과 영한기계번역 시스템인 MATES-EK에서 실험 문장으로 사용되었던 50개의 문장을 이용하였다. 전체 실험 문장 중 1문장만이 분석에 실패하였고, 실험 결과 최상위 5개의 분석 결과 내에 올바른 분석 결과가 나타난 정도가 평균 85%였다. 걸러진 규칙은 전체 구문분석 참여 규칙의 61% 정도였으나, 분석의 정확률에 있어서는 거의 차이가 없었다.

[10]의 HBG(history-based grammar)는 구문 분석 과정에서의 임의의 연산 확률은 이전에 발생한 일부 혹은 모든 연산에 의해 영향을 받는다는 가정에 기반하고 있다. HBG에서의 문맥은 해당 구문트리의 가장왼쪽유도구조(leftmost derivation)로 정의된다. 다음과 같은 문맥자유문법을 이용해 문자열 aabb의 구문구조를 그려보면 다음과 같다.



위 구문트리에 대한 가장왼쪽유도구조를 기술해 보면 다음과 같다.

$$S \rightarrow^{\gamma 1} ASB \rightarrow^{\gamma 2} aSB \rightarrow^{\gamma 3} aABB \rightarrow^{\gamma 4} aaBB$$
$$\rightarrow^{\gamma 5} aabB \rightarrow^{\gamma 6} aabb$$

여기서 γ_i는 구문트리 상의 i번째 노드확장을 위해 사용된 규칙이다. 노드 i를 확장하기 바로 전의 문장형태(sentential form)를 t_i^-로 표기할 경우 적용된 규칙의 순서열인 $\gamma_1\,\gamma_2$는 aSB와 일대일 관계를 가지며, 결국 t_3^-과도 동일하다. 따라서 가장왼쪽유도구조와 구문트리 간의 일대일 관계를 이용하면 구문트리 T와 문장 S간에 다음과 같은 결합확률(joining prob.)을 정의할 수 있다.

$$P(T,S) = \sum_{i=1}^{m} p(\gamma_i | t_i^-)$$

실험 결과로 HBG는 올바른 구문트리가 가장 높은 확률을 갖는 분석 결과일 경우는 약 75%였으며, 이 수치는 단순 확률문맥자유문법의 약 60%와 비교하여 상당히 높은 수치로 나타났다. 그러나, 이용 가능한 문맥을 더 증가시

킨 결과 오히려 정확도가 더 떨어졌다. 즉, 문맥이 증가할수록 더 많은 말뭉치가 필요하다는 것을 알 수 있다. HBG는 앞서의 문맥의존 방법과는 달리 상당한 양의 어휘정보를 모델이 포함시켰으며, 구문분석에 결정트리(decision tree) 방법을 도입하였다.

4. 한국어에 대한 통계적 구문분석

앞에서 살펴본 말뭉치 기반의 통계적 구문분석 연구들은 대부분 대량의 구문구조부착 말뭉치와 문법규칙을 이용한 것들이다. 그러나 한국어에 대한 연구에서는 아직 이렇다할 대량의 구문구조부착 말뭉치나 대량의 문법규칙이 없는 실정이다. 최근에서야 구문구조부착 말뭉치 구축에 대한 시도가 진행되고 있으며, 아직도 선행되어야 할 기초적 연구분야가 많은 실정이다. 말뭉치 구축에서는 일관성을 유지할 수 있도록 하는 연구들이 선행되어야 할 것이고, 또한 한국어에 대한 특성을 반영할 수 있도록 적절한 통계 모델이 개발되어야 하며, 대량의 문법규칙도 요구되는 실정이다. 본 장에서는 한국어 구문구조말뭉치 구축에 대한 여러 사항들과 통계적 구문분석에 대한 것들을 살펴보기로 한다.

4.1 한국어 구문구조 부착 말뭉치

한국어에 대한 말뭉치 구축 작업은 최근에야 시작되었으며, 따라서 영어권에서와는 달리 최근에야 말뭉치를 이용한 연구가 활발히 진행되고 있다. 품사부착 말뭉치를 이용한 형태소 중의성 해소에 대한 연구들이 대부분이며, 구문구조부착 말뭉치를 이용한 연구로는 [1]이 보일 뿐이다. 한국어에 대한 말뭉치 구축은 연세대학교 한국어 사전 편찬실이 1989년에 최초로 시작했으며, 1992년에는 한국과학기술원이 13개 주제 분야에서 1천만 어절 이상을 목표로 말뭉치 구축 작업을 시작하였으며, 1993년에 창설된 고려대학교 언어정보연구소가 그 뒤를 이어 말뭉치를 만들고 있다[3]. 구문구조부착 말뭉치 구축은 초기 단계로써, 1996년 현재 한국과학기술원이 10만 어절 분량의 구문구조부착 말뭉치 구축 작업을 진행중이다.

한국어에 대한 구문구조말뭉치 문법 형식으로는 크게 구구조 형식을 이용하는 방법과 의존문법 형식을 이용한 방법을 이용할 수 있다. 그러나 어느 형식을 이용하더라도 기본적인 원칙을 정하고 그 원칙에 충실하게 일관성을 지키는 것이 중요하다. 실제 언어생활에서 사용되는 문장을 살펴보면 구문구조를 결정하기 어려운 것들이 존재한다. 이러한 문장들에 대한 일정한 기준을 세우는 작업도 일관성 유지를 위한 노력에 포함되어야 할 것이다. 문장 "나는 그녀와 함께 갔다."에서 "그녀와"가 수식하는 어절이 "함께"인지 "갔다"인지 결정하기 쉽지 않다. 그러나 문맥에 따라서는 "그녀와"가 "함께"를 수식하는 것이 자연스러울 수도 있고, 혹은 위와 같이 "갔다"를 수식해도 자연스러울 수 있는 경우들이 있다. 이러한 어려운 상황들은 주로 특정 어휘들에 편중되어 나타나는 경향이 있으므로, 구문구조부착 말뭉치의 구축을 시작하기 전에 사전조사가 이루어지는 것이 바람직할 것이다. Penn Treebank의 내부적인 일관성은 23%에 이를 뿐이며, [29]에 쓰인 실험용 자료의 일관성도 50% 정도에 불과하다고 한다. 50%의 의미는, 동일한 문장에 대해 구문구조부착을 두 번을 시킬 경우 두 구조가 같을 확률이 50% 정도라는 것이다[29].

구문구조 말뭉치의 가장 간단한 형태를 괄호 넣기이다. 괄호 넣기는 다음의 예문과 같이 단순히 문장을 구성성분 단위로 묶기만 할 뿐이다.

1) [[철수/nq+는/jx [무척/a 좋아하/pa+ㄴ다/ef]]./s.]
2) [[[[철수/nq]는/jx [무척/a 좋아하/pa]]ㄴ다/ef]./s.]

여기서 이용된 형태소 범주는 [2]을 이용하였다. 1)은 어절단위로 괄호 넣기를 한 것이고, 2)는 형태소 단위로 괄호 넣기를 한 것이다. 그러나 좀 더 풍부한 정보의 구문구조부착 말뭉치를 만들기 위해서는 일관성유지의 노력뿐 아니라 구문관계도 결정을 해야 한다. 위의 예문에서는 "철수는"이 "좋아한다"와 주격 관계를 가질 수도 있고 목적격 관계를 가질 수도 있다. 한국어에서는 특히 보조사의 격 결정 문

제가 어려우므로, 말뭉치 구축 시 보조사만이라도 격을 결정해 주는 방향이 바람직할 것이다.

4.2 통계적 구문분석

[4]에서는 의미정보 대신에 통계정보를 이용할 것을 제안하고 있으며, [1]은 한국어에 통계적 구문분석을 처음으로 제안하고 있다. [1]에서는 괄호로 묶은 약 430문장의 말뭉치를 이용하여 의존문법을 추출하고 확률 매개변수를 계산한다. 실험 결과 10단어 이하일 경우의 문장수준의 정확도는 약 90%에 달하지만 이후 단어 수의 증가에 따라 거의 선형적으로 정확도가 감소하여, 30단어일 경우에는 약 6%의 정확도만을 보인다. 그러나, 의존관계의 수준에서만 살펴보면 10단어 이하일 경우 약 95%의 정확도를 보이고, 30단어일 경우에도 80% 정도의 정확률을 보인다. 이것은 문장이 30단어일 경우 약 5-6개 정도의 구문관계가 틀리게 나타나는 것이다. 그러나 [1]은 괄호 넣기만을 결과로 출력하고 있을 뿐 구문관계에 대한 것은 고려하고있지 않다. 만일 구문관계를 고려한다면 정확률은 많이 떨어질 것으로 보인다.

5. 결 론

본 연구에서는 통계적 구문분석에서의 여러 연구들을 살펴보았다. 그러나 통계적 구문분석의 중요한 문제인 자료부족문제는 언급하지 않았다. 규칙기반의 방법론들에 비해 규칙의 획득 및 확장이 쉽고, 견고성 및 결과의 순위화 등이 용이하지만, 대량의 말뭉치를 필요로 하며, 말뭉치 내의 정보의 일관성을 유지해야 하는 어려움 등이 있다. 통계적 구문분석은 문맥자유 구구조문법을 시작으로 어휘정보를 포함시키는 문법체계의 등장, 문맥 정보를 이용하는 문맥의존 방법론으로까지 연구가 진행되고 있다. 그러나, 고려하는 문맥이 클수록 대량의 말뭉치가 요구된다. 문법규칙의 자동학습에 이용되는 inside-outside 알고리듬은 지역 최적해만을 제공하므로 올바른 확률 매개변수를 얻기 위해서는 추가적인 언어적 정보를 이용해야 할 것이다. 통계적 구문분석의 정확도를 높이기 위해서는 어휘가 고려되어야 하며, 어휘의 고려는 LTAG, Link Grammar, Categorial Grammar 등과 같이 어휘를 규칙에 포함시키는 문법체계의 등장과 규칙적용 시 어휘를 고려하는 문맥의존 방법론으로 해결책이 모색되고 있는 것 같다. 그러나, 충분한 말뭉치의 확보가 선결되어야 한다. 한국어에 대한 통계적 구문분석은 이제 시작 단계이며, 한국어의 특성을 반영할 수 있는 통계적 모델과 아울러, 연구용으로 공유될 수 있는 말뭉치의 구축이 가장 시급한 문제이다.

참고문헌

[1] 김형근, "확률적 의존문법과 한국어 구문분석, "한국과학기술원, 전산학과, 석사학위논문, 1995.

[2] 김재훈, 서정연, "자연언어 처리를 위한 한국어 품사태그", CAIR-TR-94-55, 한국과학기술원 인공지능연구센터, 1994.

[3] 노용균, 박동인, "corpus 구축 및 현황과 한국어 corpus 구축 및 활용의 제문제," 정보과학회지, 12권 8호, pp. 64-71, 1994.

[4] 양재형, 김영택, "다중 지식원을 이용한 한국어의 분석," 정보과학회논문지, 21권 7호, pp. 1324-1332, 1994.

[5] 조정미, "통계 정보를 이용한 영어 구문분석," 한국과학기술원, 전산학과, 석사학위논문, 1993.

[6] James Allen, Natural Language Understanding. Redwood City, CA., The Benjamin/Cummings Publishing Company, Inc., 1995.

[7] J.Baker, "Trainable grammars for speech recognition", In Speech communication papers presented at the 97th Meeting of the Acoustical Society of America, 1979.

[8] L.Baum, "An inequality and associated maximization technique in statistical estimation for probabilistic functions for a Markov process", inequalities, 3 : 1-8, 1972.

[9] E.Black, J.Lafferty and S.Roukos, "Devel-

opment and Evaluation of a Broad-Coverate Probabilistic Grammar of English-Language computer Manuals", In the Proceedings of ACL, Newark, Delaware, 1992.

[10] E.Black, F.Jelinek, J.Lafferty, D. Magerman, R.Mercer and S.Roukos, "Towards History-Based Grammars : Using Richer Models for Probabilistic Parsing", In the Proceedings of ACL, Columbus, Ohio, 1993.

[11] E.Black, J.Lafferty and S.Roukos, "Developemnt and Evaluation of a Broad-Coverage Probabilistic Grammar of English-Language Computer Manuals", In the Proceedings of ACL, Newark, Delaware, 1992.

[12] E.Brill, "Transformation-Based Error-Driven Parsing", In the Proceedings of Third International Workshop on Parsing Technologies, Tilburg, The Netherlands, 1993.

[13] E.Brill, M.Marcus, "Automatically acquiring phrase structure using distributional analysis", In Darpa Workshop on Speech and Natural Language, Harriman, N.Y., 1992.

[14] R.Bob, "Using an annotated corpus as a stochastic grammar", In the Proceedings of European ACL, Utrecht, 1993.

[15] P.F.Brown, V.J.D.Pietra, P.V.DeSouza, J. C.Lai, R.L.Mercer, "Class-based n-gram models of ntural language", Computational Linguistics, Vol. 18, No. 4, pp. 467-479, 1992.

[16] G.Caroll, E.Charniak, "Learning Probabilistic Dependency Grammars from Labelled Text", In the Proceedings of the Fall Symposium on Probabilistic Approaches to natural Language, AAAI, 1992.

[17] G.Caroll, E.Charniak, "Two experiments on learning probabilistic dependency grammars from corpora", In Workshop Notes, Statistically-Based NLP Techniques, AAAI, 1-13.

[18] W. Chafe, "The importance of corpus linguistics to understanding the nature of language", In the Proceedings of the Nobel Symposium 82, pp. 79-97, New York, 1992.

[19] Eugene Charniak, Statistical Language Learning, Cambridge, MA., MIT Press, 1993.

[20] R.Haigh, G.Sampson, E.Atwell, "Project APRIL - a progress report", In the Proceedings of the Annual Meeting of the Association for Computational Linguistics, Buffalo, N.Y., 1988.

[21] J.Hammersley, D.Handscomb, Monte Carlo Methods. Chapman and Hall, 1964.

[22] F.Jelinek, J.Lafferty and R.Mercer, "Basic Methods of Probabilistic Context Free Grammars", Technical Report, IBM, Yorktown Heights, 1990. Technical Report RC 16374(72648).

[23] A. Joshi and Y. Schabes, "Tree-Adjoining Grammars and Lexicalized Grammars", M. Nivat (ed.), Definability and Recognizability of Sets of Trees, Elsevier, 1991.

[24] J.Lafferty, D.Sleator, D.Temperly, "Grammatical Trigrams : A Probabilistic Model of Link Grammars," School of Computer Science, Tech. Report CMU-CS-92-181, 1992.

[25] K.Lari and S.Young, "The Estimation of Stochastic Context-Free Grammars Using the Inside-Outside Algorithm", Computer Speech and Language, 4, 1990.

[26] M.Liberman and Y. Schabes, "Statistical Methods in Natural Language Processing", In the Proceedings of European ACL, Utrecht, 1993.

[27] D.Magerman and M.Marcus, "Pearl : A Probabilistic Chart Parser", In the Proceedings of European ACL, Berlin, 1991.

[28] D.Magerman and C.Weir, "Efficiency, robustness and Accuracy in Picky Chart Parsing", In the Proceedings of European

ACL, Newark, Delaware, 1992.

[29] D.Magerman and C.Weir, "Statistical Decision-Tree Models for Parsing", In the Proceedings of ACL, Cambridge, Massachusetts, 1995.

[30] "Parsing a natural language using mutual information statistics", In the Proceedings of 8th National Conference on Artificial Intelligence, 1990.

[31] "Building a Large Annotated Corpus of English : the Penn Trebank", Computational Linguistics, Vol. 29, No. 2, 1993.

[32] F.Pereira and Y.Schabes, "Inside-Outside Reestimation from Partially Bracketed Corpora", In the Proceedings of ACL, Newark, Delaware, 1992.

[33] P.Resnik, "Probabilistic Tree-Adjoining Grammar as a Framework for Statistical NaturalLanguage Processing", In the Proceedings of COLING, Nantes, 1992.

[34] Y.Schabes, "Stochastic Lexicalized Tree-Adjoining Grammar", In the Proceedings of COLING, Nantes, 1992.

[35] Y.Schabes, M.Roth and R.Osborne, "Parsing the Wall Street Journal with the Inside-Outside Algorithm", In the Proceedings of European ACL, Utrecht, 1993.

[36] Y.Schabes, R.Waters, "Stochastic Lexicalized Context Free Grammars", In the Proceedings of Third International Workshop on Parsing Technologies, Tilburg, The Netherlands, 1993.

[37] R.Simmons and Y.Yu, "The Acquisition and Use of Context-Dependent Grammars for English", Computational Linguistics, Vol. 18, No. 4, pp. 391-418, 1992.

[38] Clive Souter, Eric Atwell, "A Richly Annotated Corpus for Probabilistic Parsing," In the Proceedings of the Fall Symposium on Probabilistic Approaches to natural Language, AAAI, 1992.

[39] W. Stolz, A probabilistic Procedure for grouping words into phrases, Language and Speech, 8, 1965.

[40] R.Weischedel, M.Meteer, R.Schwarz, L. Ramshaw and J.Palmucci, "Coping with Ambiguity and Unknown Words through Probabilistic Models", Computational Linguistics, Vol. 19, No. 2, pp. 359-382, 1993.

[41] J.Wright, E.Wrigley and R.Sharman, "Adaptive Probabilistic Generalized LR Parsing", In the Proceedings of Second International Workshop on Parsing Technologies, Cancun, Mexico, 1991.

서 정 연

1981 서강대학교 수학과 학사
1985 미국 Univ. of Texas, Austin 전산학과 석사.
1990 미국 Univ. of Texas, Austin 전산학과 박사.
1990.9～91.1 미국 Texas Austin, UniSQL Inc. Senior Researcher.
1991.3～91.6 한국과학기술원 인공지능연구센타 선임연구원
1991.7～95.2 한국과학기술원 전산학과 조교수
1995.3～96.2 서강대학교 전산학과 조교수
1996.2～현재 서강대학교 전산학과 부교수
관심분야 : 한국어 정보처리, 자연언어처리, 기계번역, 대화처리

김 창 현

1991 홍익대학교 공과대학 전자계산산학과 학사
1993 한국과학기술원 전산학과 석사
1993～현재 한국과학기술원 전산학과 박사과정