형태소 단위의 한국어 확률 의존문법 학습

최 선 화[†] · 박 혁 로[††]

요 약

본 논문에서는 코퍼스를 이용한 확률 의존문법 자동 생성 기술을 다룬다. 한국어의 부분 자유 어순성질과 문장의 필수적 성분의 생략과 같은 특성으로 인하여 한국어 구문 분석에 관한 연구들에서는 주로 의존문법을 선호하고 있다. 본 논문에서는 기존의 어절단위 학습방법에서는 학습할 수 없었던 어절 내의 의존관계를 학습할 수 있는 형태소 단위의 학습 방법을 제안한다. KAIST의 트리 부착 코퍼스 약 3만 문장에서 추출한 25,000문장의 Tagged Corpus을 가지고 한국어 확률 의존문법 학습을 시도하였다. 그 결과 초기문법 2,349개의 정확한 문법을 얻을 수 있었으며, 문법의 정확성을 실험하기 위해 350개의 실험문장을 Parsing한 결과 69.77%의 파싱 정확도를 보였다. 이로서 한국어 어절 특성을 고려한 형태소 단위 학습으로 얻어진 의존문법이 어절 단위 학습으로 얻어진 문법보다 더 정확하다는 사실을 알 수 있었다.

Korean Probabilistic Dependency Grammar Induction by morpheme

Seon Hwa Choi[†] · Hyuk Ro Park[††]

ABSTRACT

In this thesis, we present a new method for inducing a probabilistic dependency grammar (PDG) from text corpus. As words in Korean are composed of a set of more basic morphemes, there exist various dependency relations in a word. So, if the induction process does not take into account of these in-word dependency relations, the accuracy of the resulting grammar may be poor. In comparison with previous PDG induction methods, the main difference of the proposed method lies in the fact that the method takes into account in-word dependency relations as well as inter-word dependency relations. To access the performance of the proposed method, we conducted an experiment using a manually-tagged corpus of 25,000 sentences which is complied by Korean Advanced Institute of Science and Technology (KAIST). The grammar induction produced 2,349 dependency rules. The parser with these dependency rules showed 69.77% accuracy in terms of the number of correct dependency relations relative to the total number dependency relations for best-1 parse trees of sample sentences. The result shows that taking into account in-word dependency relations in the course of grammar induction results in a more accurate dependency grammar.

키워드 : 의존문법(dependency grammar), 문법학습(grammar induction)

1. 서 론

구문분석은 문장의 구조를 밝힘으로써 문장의 명확한 의미를 포착하는데 도움을 준다. 효과적인 구문분석을 위해서는 해당 언어를 잘 기술하는 언어규칙의 집합인 문법이 필요하다. 이러한 문법의 습득을 위한 기본적인 정보원으로서, 텍스트 코퍼스가 대량으로 구축되어 이용되고 있다. 이 코퍼스로부터의 문법의 획득은 언어 전문가에 의해 수동적으로 이루어지거나 학습알고리즘을 통해 자동적으로 이루어진다.

언어전문가에 의한 수동적 지식의 획득은 작은 규모의, 제한된 분야에 대한 문법 구축으로는 비교적 성공 가능성이 있으며, 사람의 직관에 아주 가까운 정확한 문법을 얻을 수 있다. 그러나 기술하고자 하는 언어 현상의 규모가 커질수록, 분야의 제한이 없어질수록, 수동적 지식의 획득에는 어려움이 많다. 또한, 수동적 문법 구축은 문법의 확장 및 관리가 힘든 전형적인 지식획득 병목현상을 나타내는 어려운 작업으로 알려져 있다. 이의 대안으로, 코퍼스로부터 자동적으로 문법을 학습하기 위한 연구가 많이 시도되어 왔다[4, 7, 8, 12, 21]. 이는 코퍼스를 이용하여 통계적 정보를 이용한 문법 학습, 혹은 문법 추론으로, 하나의 연구분야를 이루고 있다. 이런 방식의 문법 학습은 여러 가지 장점을 갖는다. 지식의 획득 및 확장이 용이하고 습득된 통계정보로부터 문장 분석 결과의 적합성을 우선순위화 할 수 있는 등 모호성 처리가 자연스러우며, 잘못된 언어현상이나 비 자연스러운 문장에 대해서도 분석에 실패하지 않고 나름의 보유지식에 비추어 최적의 결과를 내주어주는 등의 장점이 있다.

지금까지 대부분의 코퍼스 기반 문법 자동 학습은 구 구조문법 형식의 문맥자유문법 학습에 치중되어 왔다[2, 3, 12, 16]. 이 구 구조문법은 어순이 비교적 고정적인 영어와 같은 언어의 문법을 작성하는 데에 효과적으로 적용되어 왔다. 이와는 상대적으로, 한국어나 일본어, 터키어, 러시아어와 같이 (부분적으로)자유 어순의 성격을 가진 언어의 문법 기술에는 의존문법이 더 효과적일 수 있다. 의존문법은 문

† 준 회 원 : 전남대학교 대학원 전산학과
†† 종신회원 : 전남대학교 전산학과 교수
논문접수 : 2002년 2월 8일, 심사완료 : 2002년 10월 11일

장 내의 임의의 두 단어 사이의 지배-피지배 관계를 정의함으로써 문법을 기술하므로, 구 구조문법에 비해 단어의 발생 순서를 덜 제약적으로 표현할 수 있어서 빈번하게 일어나는 생략과 피지배소 단어들의 발생 순서 뒤바뀜에 효과적으로 대응할 수 있기 때문이다.

본 논문에서는 코퍼스를 이용한 의존문법의 통계적 자동 학습을 목표로 한다. 특히, 어절 단위로 학습했던 기존 연구[21]와는 달리 어절 내부의 의존관계까지 학습하는 형태소 단위 학습방법을 실험하여 얼마나 정확한 의존문법이 학습되는지 보인다.

본 논문의 구성은 다음과 같다. 2장에서는 의존문법 학습과 관련된 기존 연구들을 살펴보고, 3장에서는 한국어 의존문법 학습의 특징을 소개하고 4장에서는 확률 의존문법 학습 알고리즘을 제시한다. 그리고 5장에서는 한국어 확률 의존문법 학습을 실험하고 그 결과를 분석하여 마지막으로 6장에서는 결론을 맺는다.

2. 관련 연구

2.1 의존 문법 자동 학습

의존문법은 크게 두 가지로 나누어 볼 수 있다. 하나는 단어 순서를 명시하는 것과 단어 순서에 무관한 것이다. Carroll[8]은 인사이드-아웃사이드 알고리즘을 이용하여 단어순서를 명시하는 의존문법의 자동학습을 실험하였다. 그가 사용한 문법 규칙의 형태는 다음과 같다.

$$\overline{X} \rightarrow \alpha X \beta$$

□는 단말노드이고, α와 β는 비 단말노드들의 열로서 빈 열일 수 도 있다. Carroll은 초기 문법으로 아무것도 없는 빈 상태에서 시작한다. 한 문장씩 차례로 분석에 필요한 규칙을 첨가하여 그렇게 수정된 문법을 반복적으로 인사이드-아웃사이드 알고리즘을 이용하여 확률값 재추정을 하는 방식으로 문법학습이 이루어진다. 이 연구는 결국 의존문법 학습이라기보다는 제한된 구 구조 문법 형식을 빌린 의존문법을 학습함으로써 문법 탐색 공간을 줄이는 효과를 얻고자 한 시도라고 볼 수 있다.

어순이 가변적인 경우의 언어를 위해서는 단어순서와 무관한 의존문법이 더 적합할 수 있다. 단어 순서와 무관한 의존문법은 다음과 같은 형식으로 규칙을 기술한다. 여기에서 x는 y의 지배소, y는 x의 피지배소라고 말하고 그 의존관계의 기능적 역할은 f로 표현된다.

$$x \rightarrow y(f)$$

이와 같이, 의존문법 규칙은 오직 단어와 단어 사이의 지배-피지배 관계와 각 관계의 기능적 역할을 표현할 뿐이다. 의존문법은 구 구조문법과 달리, 문장을 구성요소(Constituents)단위로 나누지 않고, 대신, 단어와 단어 사이를 연결하는 문법적인 관계를 구별함으로써 문장을 분석한다[13]. 이러한 형태의 의존문법 확률 파라미터의 재추정을 위한 시도는 이승미[21]에 의해 시도된바 있다. 의존관계 집합을 표현하는 단위요소로 완결-링크와 완결-링크 열을 정의하고 이를 이용하여, 단어간 의존관계의 확률 값 학습을 시도하였다. 구성요소의 대표 지배소들 간의 의존관계만을 고려하여 학습한 결과 의존관계 정확도는 62.82%로 나타났다. 하지만, 이 방법은 초기 문법 3,080개로 학습을 시작하여 학습 후 78.18%가 줄어든 627개의 문법을 얻었다. 복잡한 구문구조를 표현하기에는 문법의 개수가 너무도 적다. 본 논문에서는 이승미[21]의 인사이드-아웃사이드 재추정 알고리즘을 이용하여 어절 내부 의존관계까지 학습한다. 이로서 한국어의 한 어절이 가질 수 있는 어절 내부의 여러 개 의존관계까지 고려하는 학습을 통해 정확한 의존문법을 생성하기 위한 시도를 하였다.

2.2 인사이드-아웃사이드 알고리즘

코퍼스로부터 자동으로 문법을 학습하기 위해 필요한 확률 파라미터 재추정 알고리즘 중 대표적인 것은 Baker[11]에 의해 처음으로 제안되고 [6]에서 보다 실제적으로 정의되면서 그 뒤로 많은 연구자들에 의해 이용되어온 인사이드-아웃사이드(Inside-Outside) 알고리즘을 들 수 있다. 이는 Baum-Welch 재추정 알고리즘의 일종으로 촘스키 정규형인 확률적 문맥 자유 문법에 적용되도록 되어 있고, 기대치-최대화 알고리즘의 특별형태로, 학습코퍼스가 주어졌을 때, 학습코퍼스의 확률이 높아지도록 반복적으로 파라미터들의 확률 값을 조정함으로써 더 나은 확률 추정 값을 구하는 알고리즘이다. 이 알고리즘은 반복적으로 수행되면서 학습 코퍼스의 확률 값을 국부적 최대치로 만드는 확률 파라미터 값으로 수렴하는 것을 보장한다. 즉, 학습코퍼스가 언어현상을 충분히 반영하는 대표적인 코퍼스일 때, 이 학습 코퍼스의 확률을 높이면 높일수록 다른 텍스트에 대한 확률도 높아질 것이라는 가정에 기반 한 알고리즘이다[5].

이 재추정 알고리즘은 다음과 같이 학습 코퍼스의 엔트로피를 최소화 혹은 확률 값을 최대화하는 방향으로 진행된다.

1. 이전-교차-엔트로피 = 0.0
2. 문법 규칙 및 확률값 초기화
3. 확률값 재추정 및 새-교차-엔트로피 계산
4. (이전-교차-엔트로피 ≈ 새-교차-엔트로피)일 때까지 반복
   A. 이전-교차-엔트로피 = 새-교차-엔트로피
   B. 확률값 재추정 및 새-교차-엔트로피 계산

이때 각 확률 파라미터, 즉 각 규칙의 확률값을 재조정하는 부분은 다음과 같이 표현할 수 있다. $N^i \rightarrow \zeta^j$가 N^i의 j번째 생성 규칙을 나타낸다고 할 때, $N^i \rightarrow \zeta^j$의 확률 추정 값은

$$p_e(N^i \rightarrow \zeta^j) = \frac{C(N^i \rightarrow \zeta^j)}{\sum_k C(N^i \rightarrow \zeta^j)}$$

로 정의되고, 각각의 규칙 사용 빈도수는 아래 식과 같이 유도된다.

$$\begin{aligned} C(N^j \rightarrow N^p N^q) &= \sum_{Tree, W_{1,n}} P(Tree, W_{1,n}) O_{cc}(N^j \rightarrow N^p N^q, Tree) \\ &= \sum_{k,l,m} p(N^j_{k,l}, N^p_{k,m}, N^q_{m+1,l} \mid W_{1,n}) \\ &= \frac{1}{p(W_{1,n})} \sum_{k,l,m} \alpha_j(k,l) p(N^j \rightarrow N^p N^q) \beta_p(k,m) \beta_q(m+1,l) \end{aligned}$$

여기서 β는 인사이드 확률을 나타내는데, $\beta_j(k,l)$은 j번째 비 단말노드가 k부터 ℓ까지의 단어열을 생성할 확률을 말한다.

$$\begin{aligned} \beta_j(k,l) &= P(W_{k,l} \mid N^j_{k,l}) \\ &= \sum_{p,q,m} P(N^j \rightarrow N^p N^q) \beta_p(k,m) \beta_q(m+1,l) \end{aligned}$$

α는 아웃사이드 확률로서, $\alpha_j(k,l)$은 j번째 비단말노드가 k부터 ℓ까지의 단어열을 지배할 때, 1부터 k-1까지의 단어열과 ℓ+1부터 n까지의 외부 단어열이 생성될 확률을 말한다.

$$\begin{aligned} \alpha_j(k,j) &= P(W_{1,k-1}, N^j_{k,l}, W_{l+1,n}) \\ &= \sum_{h,p,q} \alpha_p(h,l) P(N^p \rightarrow N^q N^j) \beta_q(h,k-1) \\ &= \sum_{m,p,q} \alpha_q(k,m) P(N^p \rightarrow N^j N^q) \beta_q(l+1,m) \end{aligned}$$

3. 한국어 의존문법 학습

한국어는 부분 자유 어순을 가지며, 주어나 목적어와 같은 필수적 성분의 생략이 빈번하다. 이와 같은 특성으로 인하여 한국어 구문 분석에 관한 연구들에서는 주로 의존문법을 선호하고 있다[18, 22]. 이는 의존문법이 어순의 자유성에 의한 문제점을 쉽게 해결할 수 있으며 구성요소의 불연속성이나 생략 등과 같은 현상에 큰 영향을 받지 않을 수 있기 때문이다. 하지만 의존문법 자동생성에 관한 연구는 구 구조 문법 자동생성에 관한 연구보다 활발한 연구를 하지 못하고 있다. 의존문법은 구 구조 문법과 달리 문장을 구성요소 단위들로 나누지 않고 단어와 단어 사이를 연결하는 문법적인 관계를 구별함으로 문장을 분석한다[13]. 이러한 형태의 의존문법 확률 파라미터의 재추정을 위한 시도나 문법을 자동 학습하려는 시도가 거의 드물다. 그 중 이승미[21]는 구 구조 문법 학습에 맞도록 설계된 인사이드-아웃사이드 알고리즘을 의존문법에 적합하도록 변형한 인사이드-아웃사이드 알고리즘을 제시하였다.

3.1 한국어의 특성

한국어에서 어절은 띄어쓰기의 단위이며, 하나의 어절은 하나 이상의 실질 형태소로만 구성되거나, 하나 이상의 실질형태소(실질어)와 하나 이상의 형식 형태소(기능어)로 구성될 수 있다. 영어의 경우와 다르게 한국어 한 어절 내에는 문법적인 기능을 달리하는 형태소가 여러 개 올 수 있으며 한 어절 내에서의 의존관계를 달리하고 있다. 예를 들면, 다음과 같다.

"밥을 먹기가 힘들었다."

위 어절 '먹기가'는 실질어 '먹다'에 명사형 전성어미 '기'가 그리고 주격조사 '가'가 결합되어 형성된다. 명사형 전성어미 '기'는 문장 "밥을 먹다"를 명사화하는 역할을 하며 주격조사 '가'는 이러한 '밥을 먹기'를 '힘들었다'의 주어로 만드는 역할을 하고 있다.

〈 표 1 〉 기능어의 분류

기 준		기능어의 종류
구절 간 관계 명시	격조사	주격조사(jcs), 목적격조사(jco), 보격조사(jcc), 부사격조사(jca), 관형격조사(jcm), 공동격조사(jct), 인용격조사(jcr), 접속격조사(jcj), 통용보조사(jxc)
	어 미	대등적 연결(ecc), 종속적연결어미(ecs), 관형사형어미(etm)
구절 내 관계 명시	격조사	서술격조사(jp), 호격조사(jcv), 종결보조사(jxf)
	어 미	명사형어미(etn), 선어말어미(ep), 종결어미(ef)
	접 사	명사파생접사(xsn), 동사파생접사(xsv), 형용사파생접사(xsm), 부사파생접사(xsa)

위의 예제에서도 알 수 있듯이 한국어의 기능어는 크게, 성분을 변화시키는 것과 다른 구절과의 관계를 명시하는 것으로 나누어 볼 수 있다. <표1>[20]은 이러한 기능어의 특성에 따라 분류된 결과를 보여준다. 이의 구분은 그 형식 형태소가 어절에서 어느 위치에 주로 나타나느냐에 따라 구분해 볼 수 있다. 주로 성분의 변화를 유발시키는 형식 형태소는 주로 어절의 제일 끝에 나타나는 경향이 많다[20].

3.2 한국어 의존문법 학습의 문제점

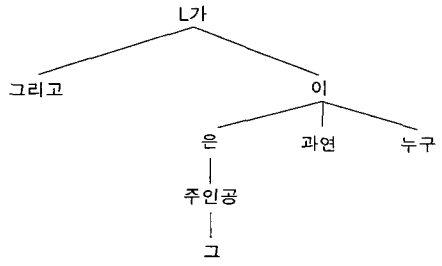
한국어 의존문법 학습은 한국어의 특징을 잘 반영한 학습이여야 한다. 한국어의 특징 중 특히 주목할 사항은 어절 내 관계를 명시하는 기능어 즉, 어절 내에서 성분의 변화를 유발시키는 형식 형태소는 의존관계 또한 달리한다는 사실이다. 다음 문장을 보면,

"그리고, 그 주인공은 과연 누구인가."

'누구인가' 라는 어절 내에는 어절내의 관계를 명시하는

형식 형태소인 '이/jp'와 'ㄴ가/ef'가 있다. 위 문장의 구문구조[1]는 다음과 같이 정의 할 수 있다.

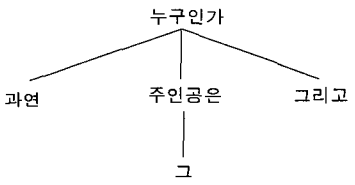
```
( S ( IP 그리고/maj )
    ( VP
        ( NP ( MODP 그/mmd ) 주인공/ncn ) + 은/jxt
        ( VP ( ADVP 과연/mag )
            ( VP (NP 누구/npp ) + 이/jp ))) + ㄴ가/ef + ./sf )
```



(그림 1) 의존 구문구조

(그림 1)에서 보이듯이 어절내의 관계를 명시하는 형식 형태소 '이/jp'와 'ㄴ가/ef'가 서로 각기 다른 의존관계를 갖는다. 즉, 'ㄴ가'는 '그리고'와 '이'를 지배하고 다시, '이'는 '은', '과연', '누구'를 지배하게 된다.

이승미[21]의 연구는 한국어의 경우 지배소가 피지배소의 오른쪽에 항상 위치한다는 제약을 이용하여 문장을 어절 단위별로 구분하여 어절의 마지막 Tag를 그 어절의 대표 지배소로 간주하고 어절 단위의 대표 지배소들 간의 의존관계만을 학습하였다. 이러한 학습 방법은 앞에서 말한, 한국어 한 어절 내에 존재할 수 있는 의존관계를 학습할 수 없다. 즉, "누구인가" 라는 어절 내부의 '이'가 '은','과연' 그리고 '누구'를 지배하는 의존관계는 학습하지 않게 된다. 이는 정확한 의존문법 학습이라고 말할 수 없으며, 이렇게 학습된 의존문법으로 올바른 의존관계를 나타낼 수 없다. (그림 2)는 어절 단위의 의존관계만을 고려했을 때 보여지는 의존구문구조다.



(그림 2) 어절단위 의존 구문구조

한국어는 실질어 부분과 기능어 부분으로 나뉘어질 수 있다. 각 실질어 부분과 기능어 부분은 하나 이상의 형태소로 구성된다. 이중 특히 기능어 부분은 그 어절의 구문적인 정보를 담고 있어서 한국어 의존문법을 정의 할 때, 단순히 어절간 의존관계로 정의하기 보다는 실질어와 기능어가 분리된 형태소간 의존관계로 정의하는 것이 더 의미가 있다. 또한 실질어-기능어를 분리하는 것이 보다 더 일반적이고 넓은 적용범위의 어휘 문법을 학습할 수 있는 방식일 것이다.

따라서, 본 논문에서는 한국어 의존문법 학습의 경우 최소 학습의 단위를 어절이 아닌, 형태소 단위로 학습하고 그 결과 얼마나 정확한 의존문법이 생성되었는지를 보인다.

4. 확률 의존문법 학습 알고리즘

확률 의존문법 재추정 알고리즘은 인사이드-아웃사이드 알고리즘[6]을 의존문법에 적합하게 변형시킨 것[21]이다. 본 장에서는 완결-링크와 완결-링크열을 정의하고, 이에 대한 인사이드-아웃사이드 확률을 기반으로 확률 의존문법을 재추정 알고리즘을 기술한다.

4.1 완결-링크, 완결-링크열

의존관계로 표현되는 문장구조는 기본적으로 두 단어 사이의 의존관계 정의에서부터 시작해 일련의 부분 단어열에 대하여 구문구조의 조건을 만족하는 의존관계 집합을 찾아 확장해 나가는 과정에서 찾게 된다. 즉, 부분 단어열에 대한 의존관계 집합을 표현하는 단위요소에 대한 정의를 필요로 한다. 본 논문에서는 완결-링크와 완결-링크열을 이용하여 단어간 의존관계의 확률값 학습과 문장의 구문구조 파싱을 한다.

완결-링크와 완결-링크열은 다음과 같이 정의된다. 단어열 $W_{i,j}$[2]에 대해서 형성된 하나의 의존관계 집합은 다음의 조건들을 만족할 때 완결-링크로 정의된다.

- 배타적으로 $w_i \rightarrow w_j$ 혹은 $w_i \leftarrow w_j$가 존재
- $j-i$개 의존관계로 구성
- 내부 단어들은 그 단어열 안에 각각 지배소 단어를 갖음.
- 의존관계의 교차, 순환이 없다.

완결-링크는 방향성을 가지는데 이는 가장 바깥 의존관계의 방향에 의해 결정된다. 즉, $W_{i,j}$에 대한 완결-링크의 가장 바깥 의존관계가 $w_i \rightarrow w_j$이면 우향이고 $w_i \leftarrow w_j$이면 좌향이다.

완결-링크열은 같은 방향성을 가진 0개 혹은 그 이상의 일련의 인접한 완결-링크들로 구성된다. 즉, 최소단위 완결링크열은 한 단어로 구성된 부분 단어열에 정의되는 0개의 연속된 완결-링크이다. 완결-링크열 역시 방향성을 가지며, 구성요소인 완결-링크들의 방향성에 의해 결정된다. 만일 좌향 완결-링크들로 구성된 완결-링크열이라면 그것은 좌향 완결-링크열이다.

1) KAIST 트리부착코퍼스 3만여 문장 중 일부

2) w_i, w_j는 각 문장내 i번째, j번째 단어를 가리키고, $W_{i,j}$는 문장내 i에서 j까지의 단어열을 가리킨다.

앞으로 완결-링크와 완결-링크열을 위해 다음의 표기법이 쓰인다. L은 완결-링크를 의미하고, S는 완결-링크열을 의미한다. 아랫첨자 r과 l은 방향성을 의미하여, r은 우향을, l은 좌향을 의미한다. 즉, L_r은 우향 완결-링크, L_l은 좌향 완결-링크, S_r은 우향 완결-링크열, 그리고 S_l은 좌향 완결-링크열을 나타낸다.

완결-링크 및 완결-링크열이라 함은 그것이 형성되는 부분 단어열에 대해서는 의존관계의 설정이 모두 완결되어 있어서 그 안의 단어들에 대해서는 더 이상 의존관계의 형성을 위한 고려가 필요치 않다는 의미이다.

완결-링크는 특별한 구조적 특성을 갖는다. 즉, 어떤 완결-링크가 임의의 단어열 $W_{i,j}$에 정의되었다고 할 때 그 구성은 다음과 같은 세 가지 요소로 이루어진다. 임의의 $m(i \le m \le j)$에 대하여

- w_i와 w_j사이의 가장 바깥쪽 의존관계,
- $W_{i,m}$단어열에 대한 우향 완결-링크열,
- $W_{m+1,j}$단어열에 대한 좌향 완결-링크열

의 조합으로 표현된다. 그렇지 않으면, 이 완결-링크를 표현하는 링크집합은 링크 교차나 다중 지배소, 혹은 링크-순환의 조건을 만족하지 못하게 되고, 따라서 더 이상 완결-링크가 아니게 된다.

완결-링크열은 0개 혹은 그 이상의 연속된 완결-링크들로 구성되는데, 모두 같은 방향성을 가져야 한다. $W_{i,j}$에 정의되는 모든 유일한 우향 완결-링크열들은 모든 $m(i \le m \le j)$에 대하여 $W_{i,m}$의 우향 완결-링크열과 $W_{m,j}$의 우향 완결-링크 쌍의 모든 조합을 나열함으로써 구할 수 있다. 마찬가지로 같은 조합이 $W_{i,m}$의 우향 완결-링크와 $W_{m,j}$의 우향 완결-링크열 쌍의 모든 조합을 나열함으로써 구해질 수도 있다. 그러나 이 두가지 조합을 모두 허용하면 중복적으로 우향 완결-링크열이 만들어진다. 이 같은 중복을 막기위해서 다음의 제약을 준다. 즉, $W_{i,j}$의 우향 완결-링크열은 항상 $i \le m \le j$인 임의의 m에 대하여 $W_{i,m}$의 우향 완결-링크열과 그것과 인접한 $W_{m,j}$에 대한 우향 완결-링크로 구성된다. 마찬가지로, $W_{i,j}$의 좌향 완결-링크열은 항상 $i \le m \le j$인 임의의 m에 대하여 $W_{i,m}$에 대한 좌향 완결-링크열과 그것과 인접한 $W_{m,j}$에 대한 좌향 완결-링크로 구성되는 것으로 제한한다.

임의의 단어열 $W_{i,j}$의 완결-링크와 완결-링크열은 $i \le m \le j$인 임의의 m에 대해 다음과 같이 표현될 수 있다.

- $L_r(i,j):\{w_i \rightarrow w_j, S_r(i,m), S_l(m+1,j)\}$
- $L_l(i,j):\{w_i \leftarrow w_j, S_r(i,m), S_l(m+1,j)\}$
- $S_r(i,j):\{S_r(i,m), L_r(m,j)\}$
- $S_l(i,j):\{S_l(i,m), L_l(m,j)\}$

4.2 의존문법 확률모델

한 문장의 확률은 그 문장이 갖는 모든 의존 구조, D의 확률의 합이다. 또, 한 의존 구문구조, D의 확률은 그 안에 포함된 모든 의존관계의 확률의 곱으로 근사화 될 수 있다.

$$p(W_{1,n}) = \sum_D p(D, W_{1,n}) \cong \sum_D \prod_{w_i \in D} p(w_i \rightarrow w_j)$$

여기에서 $1 \le i \le n+1$(EOS : *End Of String*)이고 $1 \le j \le n$이다. 임의의 $p(x \rightarrow y)$는 다음과 같이 추정된다.

$$p(x \rightarrow y) = p(y \mid x) = \frac{C(x \rightarrow y)}{\sum_z C(x \rightarrow z)}$$

따라서

$$\sum_y p(x \rightarrow y) = 1$$

이 된다. 여기에서 V가 어휘집합을 표현한다고 할 때, $x \in V \cup \{EOS\}$이고 $y \in V$이다. 그러면, 임의의 문장 $W_{i,j}$의 확률은 완결-링크와 완결-링크열의 관점에서 표현하면 다음과 같다.

$$p(W_{1,n}) \sum_D p(D, W_{1,n}) \cong \sum_{S_l(1,EOS)} p(S_l(1,EOS))$$

이고,

$$p(L_r(i,j)) = p(w_i \rightarrow w_j) p(S_r(i,m)) p(S_l(m+1,j)),$$
$$p(L_l(i,j)) = p(w_i \leftarrow w_j) p(S_r(i,m)) p(S_l(m+1,j)),$$
$$p(S_r(i,j)) = p(S_r(j,m)) p(L_r(m,j)),$$
$$p(S_l(i,j)) = p(S_l(j,m)) p(L_l(m,j)),$$

이다.

4.3 학습 알고리즘

인사이드 확률들은 CYK 챠트의 관점에서 상향식(bottom-up)으로, 좌에서 우로 계산된다. 아웃사이드 확률들은 하향식으로, 우에서 좌로 계산된다. 이때, 미리 계산된 인사이드 확률값을 이용한다. 학습은 다음과 같이 진행된다.

① 초기 의존문법을 설정한다 : 학습 코퍼스에서 모든 가능한 단어쌍을 나열하고 의존관계의 확률값을 초기화 한다.
② 학습 코퍼스의 초기 엔트로피를 계산한다.
③ 학습 코퍼스를 분석하여 각각의 의존관계의 발생빈도수를 재계산한다.
④ 재 계산된 발생빈도수에 의거하여 의존관계의 확률값을 새로 계산한다.

$$p_{new}(w_x \rightarrow w_y) = \frac{C(W_x \rightarrow W_y)}{\sum_z C(W_x \rightarrow W_z)}$$

⑤ 수정된 의존문법을 이용하여 학습 코퍼스의 엔트로피를 새로 계산한다.

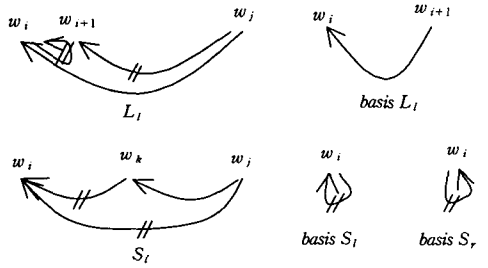
⑥ 이전 엔트로피 - 새 엔트로피 $> \varepsilon$ 이면, 3에서 5까지의 과정을 반복한다.

학습 코퍼스가 주어지면 초기 문법은 학습 코퍼스에서 발생한 모든 유일한 단어들의 쌍으로 초기화 될 수 있다. 이 쌍들은 잠정적인 지배-피지배 관계를 표현한다. 초기 확률값은 무작위적으로 주어질 수 있다. 학습은 이 초기문법으로부터 시작한다. 그러나 이 알고리즘은 이미 어떤 의존문법이 존재한다면 그것을 초기 문법으로 가정하고 학습을 실행할 수도 있다. 이 경우 1단계는 생략되고 기존 의존문법이 초기 문법으로 간주된다. 3단계에서 5단계까지의 반복은 모든 의존관계의 확률값이 안정되거나, 혹은 학습 코퍼스의 엔트로피 값이 최소값으로 수렴할 때까지 계속된다.

5. 한국어 확률 의존문법 학습 실험

본 장에서는 확률 의존문법 학습 실험 결과로서 의존문법 학습 알고리즘의 수렴과 학습된 문법을 보인다. 또한 학습된 문법과 초기 문법의 크기 비교를 통해 학습 정도를 보이고 문법의 파싱 정확도를 실험한다. 이 알고리즘은 C언어로 구현되었으며 실험은 Window2000에서 시행되었다. 알고리즘은 KAIST 코퍼스[3]의 트리 부착 코퍼스에서 추출한 한국어 문장 집합에 대해서 학습되고 실험되었다. 실험은 어절 내부 의존관계까지 다루었으며, 품사 단위에 대해 이루어졌다.

한국어는 부분 자유 어순 언어로서 지배소인 어휘가 항상 피 지배소인 어휘의 오른쪽에 위치하는 제약을 가진다. 이 같은 제약 때문에 한국어 의존구조에서는 $S_r(i,j)$이나 $L_l(j,j)$은 의미가 없고 따라서 고려할 필요가 없다. 오직 S_l, L_l 그리고 null S_r만이 고려 대상이 된다. $L_l(j,j)$은 항상 null $S_r(i,i)$과 $S_l(i+1,j)$의 결합으로 구성된다. 한국어에 대한 추상적 의존구조를 (그림 3)에서 보인다.



(그림 3) 한국어의 추상적 완결-링크와 완결-링크열

3) KAIST(Korean Advanced Institute of Science and Technology : 한국과학기술원) 코퍼스는 1994년부터 문체부의 지원 하에 구축되어 오고 있으며, 현재 약 4,5000만 어절의 원시 텍스트 코퍼스와 675만 어절의 품사 부착 코퍼스, 그리고 3만 문장의 트리 부착 코퍼스로 구성되어 있다. 본 실험을 위해서 트리 부착 코퍼스를 사용하였다.

이승미[21] 연구는 각 어절의 기능적부분의 마지막 품사를 그 어절의 대표 품사로 보고 어절간 의존관계를 대표 품사간의 의존관계로 가정하였다. 본 논문에서는 어절 내부의 의존관계까지 고려하여 학습한다. 사용한 품사 집합, T는 55개의 품사로 구성되어 있다. 따라서 초기 문법은 T의 모든 품사쌍으로 구성되므로 3,080개의 의존관계로 구성되었다. 초기 확률값은 치우침 없도록 모두 같은 값을 갖도록 설정되었다.

재추정 알고리즘의 실험은 KAIST의 트리부착코퍼스 31,086문장 중 25,000문장의 태그 부착 문장을 추출하여 학습문장으로 구성하였고, 350 문장의 태그 부착 문장을 추출하여 실험문장으로 사용하였다.

〈표 2〉 학습 코퍼스 엔트로피

반 복	엔트로피
1	4.158206e+000
2	1.921697e+000
3	1.799868e+000
...	...
104	1.618935e+000
105	1.618834e+000
106	1.618735e+000

<표 2>에서 학습과정이 반복됨에 따라 학습 코퍼스의 엔트로피(bits/word)가 점차로 감소함을 보이고 있다.

〈표 3〉 문법의 크기

	문법크기
초기문법	3,080
학습 후	2,742 (-10.97%)
Cut-off(freq. < 1.0)	2,349 (-23.73%)

<표 3>에서 학습에 의한 문법의 크기 변화를 보인다. 초기 문법은 3,080개의 의존관계로 구성되는데 학습한 결과 10.97%의 문법 크기 감소를 가져왔다. 확률값이 0에 가까운 의존관계를 제거하여 걸렀을 경우 문법 크기를 보기 위해서 학습 후 빈도수가 1보다 작은 의존관계는 모두 제거하였다. <표 3>에서 보이듯이 23.73%의 감소를 가져왔다.

학습된 문법의 파싱 정확도를 실험하기 위하여 실험대상으로 350문장 트리부착코퍼스[4]를 사용하였다. 실험 코퍼스 문장에 대해서, 학습된 문법을 이용하여 n-최적해 파서로 가장 좋은 점수의 파스만을 추출한 뒤 이를 트리뱅크 파스와 비교하였다. 파싱 정확도를 위한 비교 기준으로는 의존관계 정확도를 채택하였다. 의존관계 정확도는 분석결과에 있는 모든 의존관계에 대한 정확한 의존관계의 비율로 정의된다.

$$\frac{\textit{트리뱅크의존관계와일치하는의존관계수}}{\textit{분석결과의의존관계수}} \times 100(\%)$$

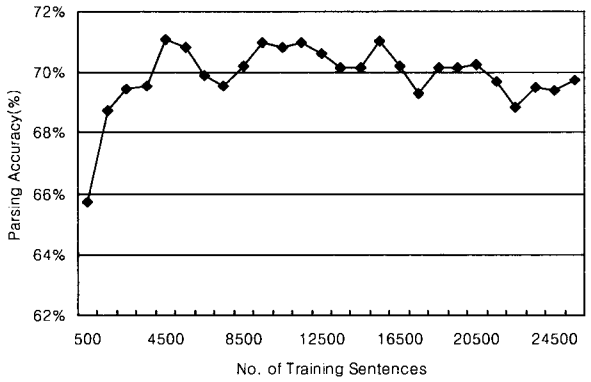
4) KAIST코퍼스 중 3만여 문장의 트리부착코퍼스로부터 추출한 350문장을 실험 코퍼스로 사용하였다. 이승미[21] 역시 실험 코퍼스로 KAIST 코퍼스를 사용하고 있지만 본 논문의 실험문장과는 다른 문장이다.

실험 대상인 자동 학습된 문법은 의존관계 정확도는 69.77%로 기존 연구보다 높게 나타남을 알 수 있다. 이 결과는 <표 4>에 표시하였다.

〈표 4〉 실험 집합 평가

	본 논문	이승미[21]
문장 개수	350	409
평균 문장 길이(단어)	13.2	11.4
문장 길이 범위	2-25	3-21
의존관계 정확도	69.77%(+6.95%)	62.82%

학습 코퍼스의 크기가 문법의 학습과 학습된 문법의 정확도에 미치는 영향을 보기 위하여 25,000의 학습 코퍼스를 25개의 부분적 학습 집합으로 나누었다. 각 부분적 학습 집합은 전체 학습 코퍼스의 첫 c개의 문장을 포함하도록 하였고, c는 500에서 25,000까지 500씩 증가시키며 달리 하였다. 모든 부분적 학습 집합에 대하여, 초기 의존문법은 마찬가지로 3,080개의 모든 가능한 품사쌍으로 구성되게 하였으며, 초기 확률값은 동일하게 하였다. 그리고 그 초기문법을 각 부분적 학습 집합에 대하여 재추정 알고리즘을 이용하여, 실험 코퍼스 엔트로피와 파싱 정확도, 학습된 문법의 크기, 즉 학습된 의존문법 규칙의 수를 조사하였다.



(그림 4) 평균 파싱 정확도 변화

(그림 4)는 각 부분적 학습 집합에 대한 의존관계 정확도를 보인다. (그림 4)에서 보이듯이 학습 코퍼스의 크기가 증가함에 따라 학습된 문법의 정확도도 같이 증가하였다. 즉, 학습 코퍼스가 4,500문장까지 증가할 때는 정확도도 같이 증가하다가 9,500문장부터는 그 정확도가 꾸준히 증가하지 않았고 점차 감소하는 경향을 보였다. 따라서 정확한 문법을 학습하기 위해서는 학습 코퍼스의 크기는 4,500문장에서 9,500문장은 학습해야 함을 알 수 있다. 또한, 생각해 볼 수 있는 것은 품사 기반 의존문법은 어휘 기반 의존문법보다 상대적으로 단순하여서 학습 코퍼스가 커질수록 좋은 문법을 학습하기보다는 어느 적정량의 크기가 있다는 점이다.

이 경우 이승미[21]의 연구에서는 약 1,000문장의 학습 코퍼스 만으로도 적절한 문법을 학습할 수 있다고 밝혔다. 하지만, (그림 4)에서 보이듯이 4,500문장까지 정확도가 꾸준히 증가하므로 최소한 4,500문장은 학습하여야 함을 알 수 있다. 즉, 55개의 품사로 이루어진 품사 의존문법의 경우에는 실험결과에서 보이듯이 약 4,500문장의 학습 코퍼스 만으로도 적절한 문법을 학습할 수 있다는 점이다.

6. 결 론

본 논문에서는 (부분) 자유어순 언어의 문법 자동학습을 위해 확률 의존문법에 적용할 수 있는 변형된 확률 파라미터 재추정 알고리즘인 인사이드-아웃사이드 알고리즘을 사용하였다. 한국어의 중심어 후위원칙하에 어절의 마지막 품사를 그 어절의 대표 지배소로 간주하고 그 들간의 의존관계만을 고려[21]했던 학습방법은 한국어 어절 내부에 존재하는 의존관계를 학습하지 못함으로써 올바른 한국어 의존문법을 생성하지 못한다. 또한, 일반적이고 보다 넓은 적용범위의 어휘 의존문법을 학습하기가 어렵다.

본 논문에서는 어절 내부에 존재하는 의존관계를 학습하기 위하여 학습의 단위를 어절이 아닌 형태소 단위로 하고 학습한 결과, 총 2,349개의 정확한 문법을 학습할 수 있었다. 학습 과정에서 보여지지 않은 실험문장에 대해서도 평균 69.77%의 의존관계 정확도를 보였다. 이는 어절 단위 의존관계만을 학습한 방법 보다 6.95%가 높은 정확률이다. 즉, 어절 내부 의존관계를 고려한 확률 의존문법 학습이 더 효과적임을 알 수 있다.

확률 의존문법 학습은 여러 방향으로 확장될 수 있다. 먼저, 어휘 의존문법 학습을 실험해 볼 수 있다. 본 논문에서는 품사 간 의존문법의 학습 실험에 그쳤지만, 사실 의존문법은 의존관계가 품사 단위 사이가 아니라 어휘 단위 사이에 정의될 때, 더 효과적이고 의미 있는 의존문법이 될 수 있다. 품사는 어휘의 특성에 관한 많은 정보를 일반화 과정을 통해 잃어버리기 때문이다.

두 번째로 재추정 알고리즘은 초기 문법에 다소 민감하게 작용해서 국부적 최적해를 찾게 된다. 즉, 초기 확률 값에 따라 학습 결과가 달라지므로 초기 확률값이 학습 결과에 미치는 영향에 대한 고찰이 필요하다. 따라서 알고리즘의 변화가 없다면, 될 수 있는 한 초기문법이 효과적이면 학습 결과도 나아질 것이다. 어휘 기반의 문법이건, 품사 기반의 문법이건 문법을 학습하고자 할 때, 미리 알고 있는 품사정보를 이용해서 수작업으로 약간의 품사 간 지배/피지배 관계에 대한 선호도를 주고, 그 정보를 품사가 부착된 학습 코퍼스에 적용하면 단순하게 학습 코퍼스의 모든 단어간 의존관계를 가정하는 것보다 더 나은 초기 의존문법을 구성할 수 있을 것이다.

참 고 문 헌

[1] De. Marcken, "Lexical heads, phase structure and the induction of grammar," *In Third Workshop on Very Large Corpora*, 1995.

[2] M. Magerman, "Natural Language Parsingas Statistical pattern Recognition," *PhD thesis*, Standford University, 1994.

[3] Black, Lafferty and S. Roukos, "Development and evaluation of a road-coverage probabilistic grammar of English-language computer manuals," *In 30th Annual Meeting of the Assocation for Computational Linguistics*, pp.185-192, 1992.

[4] E. Brill and M. Marcus, "Tagging an unfamiliar text with minimal human supersision," *In Fall Symposium on Probabilistic Approaches to Natural Language*, 1992.

[5] E. Charniak, "Statistical Language Learning," *The MIT Press*, 1993.

[6] F. Jelinek, J. D. Lafferty and R. L. Mercer, "Basic methods of Probabilistic Context Free Grammars," Technical Report, IBM-T. J. Watson Research Center, 1990.

[7] F. Pereira and Y. Schabes, "Inside-outside reestimation from partially bracketed corpora," *In 30th Annual Meeting of the Association for Computational Linguistics*, pp.128-135, 1992.

[8] G. Carroll and E. Charniak, "Learning probabilistic dependency grammars from labeled text," *In Working Notes Fall Symposium Series AAAI*, pp.25-31, 1992.

[9] G. Carroll and E. Charniak, "Two Experiments on Learning Probabilistic Dependency Grammars for Corpora," Technical Report CS-92-16, Brown University, 1992.

[10] H. Gaifman, "Dependency systems and phrase-structure system," *Information and Control*, 8, pp.304-337, 1965.

[11] J. K. Baker, "Trainable grammars for speech recognition," *In 97th Meeting of the Acoustical Society of America*, pp.547-550, 1979.

[12] K. Lari and S. J. Young, "The estimation of stochastic context-free grammars using the inside-outside algorithm," *Computer Speech and Language*, 4, pp.35-56, 1990.

[13] M. A. Covington, "A Dependency Parser for Variable-Word-Order Languages," Technical Report AI-1990-01, The University of Georgia, 1990.

[14] M. J. Collins, "A New Statistical Parser Based on Bi-gram Lexical Dependencies," *In COLING-96*, 1996.

[15] P. F. Brown, V. J. Della Pietra, P. V. deSouza, J. C. Lai and R. L. Mercer, "Class-Based n-gram Models of Natural Language," *Computational Linguistics*, 18(4) : pp.467-480, 1992.

[16] S. F. Chen, "Bayesian grammar induction for language modeling," *In 33rd Annual Meeting of the Association for Computational Linguistics*, pp.228-235, 1995.

[17] 김형근, "확률적 의존문법과 한국어 구문 분석", 석사논문, 한국과학기술원, 1994.

[18] 나동렬, "한국어 파싱에 대한 고찰", 한국정보과학회지, 12(8), pp.33-46, 1994.

[19] 이공주, "언어적 특성에 기반한 한국어의 확률적 구문분석", 박사논문, 한국과학기술원, 1998.

[20] 이공주, 김재훈, 장병규, 최기선, 김길창, "한국어 구문트리 태깅 코퍼스 작성을 위한 한국어 구문 태그", 한국과학기술원 전산학과 기술보고서, CS/TR-96-102, http://hanul.kaist.ac.kr/~kjlee/paper.html, 1996.

[21] 이승미, "확률 의존문법 학습", 박사논문, 한국과학기술원, 1998.

[22] 홍영국, 이종혁, 이근배, "의존문법에 기반을 둔 한국어 구문분석기", 한국정보과학회 봄 학술발표논문집, pp.781-784, 1993.

최 선 화

e-mail : shchoi@dal.chonnam.ac.kr

1991년 광주대학교 전자계산학과(학사)

2002년 전남대학교 전산학과(석사)

1995년~1998년 송원백화점 정보시스템부

2002년~현재 전남대학교 전산학과 박사과정

관심분야 : 정보검색, 자연어처리, 음성인식

박 혁 로

e-mail : hyukro@chonnam.ac.kr

1987년 서울대학교 전산학과(학사)

1989년 한국과학기술원 전산학과(석사)

1997년 한국과학기술원 전산학과(박사)

1994년~1996년 연구개발정보센터 연구원

1997년~1998년 연구개발정보센터선임연구원

1999년~현재 전남대학교 전산학과 조교수

관심분야 : 정보검색, 자연어처리, 데이터베이스