

전역 및 지역 정보를 이용한 SVM 기반 한국어 문장 구조 및 격 레이블 분석

임수종[○] 이창기 장명길 나동렬
한국전자통신연구원 음성/언어정보연구부, 연세대학교 전산학과
{isj, cklee, mgjang}@etri.re.kr, dyra@yonsei.ac.kr

Labeled Statistical Korean Dependency Parsing with Global and Local Information

Soojong Lim[○] 저Changki Lee, Myung-Gil Jang, DongRyul Ra
Speech/Language Information Research Center ETRI, Yonsei University

요 약

한국어 문장의 구조 및 격 레이블 분석을 위해서 SVM 모델을 이용하여 얻어진 전역 및 지역 정보 통계 모델에 기반한 방법을 제안한다. 제안하는 방법은 후방 beam search 알고리즘을 이용하여 부분 구문 분석을 하는 과정에서 지역 의존 정보를 사용하였고 이렇게 구성된 문장의 후보 구조에 대해서는 전역 정보 모델을 사용하여 최적의 문장 구조 및 격 레이블을 분석하였다. 제안하는 방법은 지역이나 전역 중 한 개의 모델만을 사용할 때 발생할 수 있는 오류를 최소화하였다. 지식 DB 사업의 한국어 의존 구문 분석 말뭉치를 이용하여 실험한 결과 전역 정보나 지역 정보만을 사용한 결과보다 각각 1.2%, 3.3% 높은 79.1%의 문장 구조 및 격 레이블 분석 정확률을 나타냈고 전역 정보만을 사용할 때보다 약 76배 이상의 빠른 속도 향상을 보였다. 향후 연구로는 지배소 단위, 구 묶음 단위 등으로 통계 정보를 세분화하여 좀더 높은 성능 향상을 기대한다.

주제어: 의존 구문 분석, 격 관계 인식, 통계 정보

1. 서 론

구문 분석은 자연언어처리 과정에서 문장의 기본 단위인 어절에 대한 형태소 분석 후 전체 문장의 구조를 분석하는 과정으로 정보 추출, 질의응답시스템, 기계 번역 등 자연언어처리 응용 분야의 성능을 확보하기 위해서는 필수적인 처리 과정이다

자연언어의 경우 인공적으로 만든 프로그래밍 언어와는 다르게, 사람들이 사용하는 문장이 정해진 문법을 따르지 않거나 혹은 시간이 흐름에 따라 문법 자체가 바뀌는 경우도 있으며 심지어는 문법 자체를 규정하기도 힘든 경우도 있다. 이러한 자연언어의 특성을 극복하기 위해 기존의 많은 연구들이 규칙을 이용하거나 기계 학습 기법과 같은 통계적인 방법은 이용하여 진행되어 왔다

통계적인 방법론은 결정트리(Decision tree), ME(Maximum Entropy), SVM(Support Vector Machine), CRF(Conditional Random Field)등 다양한 방법이 개발되어 널리 알려졌고 학습 데이터를 이용하여 도메인 이식이 쉬운 점 등의 장점 때문에 자연언어 모든 분야에 걸쳐 널리 사용되고 있다

한국어 문장을 구문 분석하기 위해서는 구구조 범주 문법, 자질연산 문법 등 여러 가지 문법을 이용한 다양한 방법론이 적용되어 왔지만 자유 어순인 한국어에 가장 적합한 문법으로는 의존 문법이 있으며 전 세계적인 로도 다양한 언어의 문장을 분석하기 위해서는 의존문법

을 채택하고 있다[1, 2].

이러한 의존 문법을 적용하여 문장의 구조를 분석했을 때 성능을 저하시키는 이유 중 하나는 너무 많은 의존 관계 후보가 생성된다는 점이다 이러한 문제점을 해결하기 위해서 기존의 연구들은 문법적으로 연관이 있는 단어들을 하나의 묶음으로 인식하여 복잡도를 줄이기도 하고[3], Beam search 알고리즘을 적용하여 후보를 줄이기도 한다[4].

2. 관련 연구

의존 문법을 이용한 문장 구조 분석에 대한 연구는 통계 정보를 이용하거나 규칙을 이용한 방법으로 분류할 수 있다.

규칙을 사용한 연구로는 의존 문법을 사용하여 규칙을 이용한 차트 파싱을 이용한 연구[5], 지역적 어순 제약을 이용한 인접 의존관계 규칙과 지배가능경로를 이용한 2단계 구문 분석을 한 연구[6], 초기 트리를 인접으로 가정하여 규칙을 사용하여 올바른 트리 생성을 시도한 연구[7], 하위범주 정보와 개념 유사도를 규칙으로 사용한 연구[8], 구문 분석된 말뭉치를 사용하여 격들을 구축하고 구축된 격들에서 격명사, 조사 쌍으로 구성된 격 확률을 이용한 연구[9], 모호성을 줄이기 위해 구간 분할 기법을 사용하고 규칙 기반 트리를 생성한 후에 통계 정보를 이용하여 모호성을 해소한 연구[10]등이 있다.

규칙 기반의 방법의 경우 기술 개발 초기에 쉽게 적용할 수 있다는 장점이 있으나 규칙이 복잡해질 경우 규칙을 구축 및 관리 비용이 많이 든다는 단점이 있다

J.Eisner 연구[11] 이후로 규칙보다는 통계에 기반한 연구가 활발하게 진행되었다.

통계 모델에 기반한 연구 중 외국의 사례로는 문장 전체를 하나의 단위로 인식하여 문장 단위로 통계 모듈을 적용한 방법[12, 13]과 문장 전체를 한 단위로 학습하기 보다는 의존 관계로 순차적으로 접근한 방법[14]이 있으며 두 가지 방법을 통합한 연구[15]도 있다. 이 밖에 SVM 모델을 이용하여 어절 간의 관계를 LEFT, RIGHT, SHIFT 세 가지로 분류한 방법[16, 17]이 있으며 문장 구조와 관계 없이 일반적으로 단어 간에 공기할 확률과 지역 의존 확률을 분리하여 적용한 연구[18, 19]가 있다.

본 논문에서는 전역 정보와 지역 정보를 이용하여 각각 통계 모델을 학습하고 지역 정보 모델을 이용하여 beam search 알고리즘의 효율성을 높이고 이 과정을 반복하여 생성된 문장 구조 후보에 대해 전역 모델을 이용하여 주어진 문장에 대해 최종적으로 문장의 구조 및 격 레이블을 분석한다.

3. 제안 시스템

본 논문에서는 입력된 문장에 대해서 SVM에 기반하여 어절 간의 지배소-피지배소 관계에 대해 학습된 지역 모델과 한 문장에 있는 어절 간의 지배소-피지배소를 이용하여 전체 문장에 대한 구조에 대해 학습된 전역 모델을 이용한다.

기존 한국어 의존 관계 모델과는 다르게 제안하는 시스템은 의존 관계 뿐만 아니라 의존 관계에 대한 레이블 분석까지 연구 범위를 확장하고 의존 관계를 유무 여부를 파악하는 이진 분류보다는 관계 없음, 주어, 목적어, 부사어, 보어, 관형격 관계, 명사구 관계, 절 관계로 세분하여 실제 응용 시스템에서 활용도가 높도록 하였다. 문장의 의존 관계를 분석한다는 것은 문장에서 가능한 모든 의존 관계를 생성한 후에 그 중에서 가장 최적의 결과를 찾아내는 과정으로 볼 수 있는데 가장 최적의 결

법과 SVM 모델을 이용하여 후보를 제거해 나가는 방법을 사용한다.

본 논문에서 제안하는 시스템의 처리 과정은[그림1]과 같다.

3.1 한국어 의존 관계 및 격 관계 규칙

한국어 의존 관계 규칙은 먼저 각 어절의 지배소가 유일해야 하고 각각의 의존 관계는 서로 교차하지 않는다는(projective)는 일반적인 의존 규칙과 대다수의 한국어 문장이 지배소 후위 제약을 따르기 때문에 최상의 의존 관계 트리를 찾는 과정에서 이러한 한국어 의존 관계 규칙에 어긋나는 트리를 [그림2]와 같이 배제하도록 하였다.

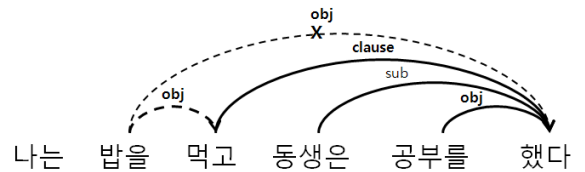


그림 2 격관계 규칙을 이용한 관계 모호성 해소

앞에서 언급한 의존 관계 규칙은 이미 널리 알려진 규칙이며 본 논문에서는 의존 관계 유무 뿐 아니라 격 관계까지 파악하기 때문에 이중 주어/목적어를 금지하여 1개의 동사가 2개 이상의 주어나 목적어를 가지는 문장 구조의 경우 [그림3]과 같이 배제하도록 하였다.

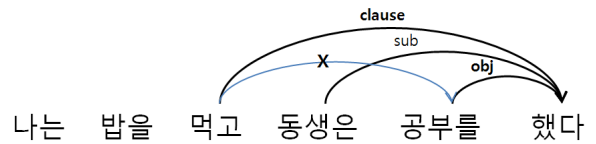


그림 3 교차 금지를 이용한 관계 모호성 해소

3.2 문맥 자질 집합

전역 정보를 이용한 모델과 지역 의존 정보를 이용한 학습 모델을 각각 구성하기 위해서는 아래와 같이 문맥 자질 집합을 따로 적용하여 학습 모델을 구축한다 중복되어 사용되는 문맥 자질도 있으나 전역 정보에서 적합한 자질과 지역 의존 정보에 적합한 자질을 분리하여 각각 최적의 학습 모델이 구축될 수 있도록 하였다 사용된 문맥 자질은 [표1]과 같다.

[표1]에서 설명한 문맥 자질을 기본으로 하여 문맥 자질 간의 조합을 사용하였으며 전역 정보는 지역 의존 정보 학습 모델에는 포함시키지 않았다

3.3 통계 모델을 이용한 구문 분석

일반적으로 통계적인 방법을 사용하는 의존 구문 분석에서는 학습을 하는 단위로 문장 단위 구조를 이용하는 전역 정보를 이용하는 방법과 지배소-피지배소 단위의

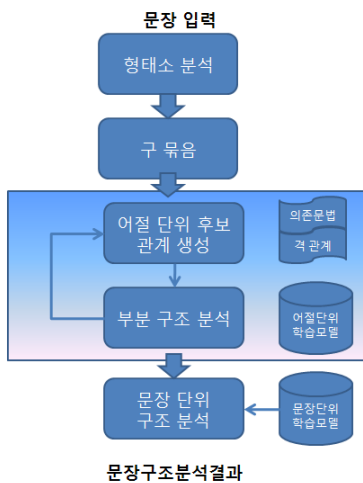


그림 1 제안 시스템

과를 찾아내기 위해서 가능한 모든 의존 관계 후보 및 문장 구조 후보를 규칙을 이용하여 후보를 제거하는 방

표 1 학습에 사용된 문맥 자질

	문맥 자질	설명
어절 단 위 정보	어절 어휘	‘학교를’ 와 같은 형태
	형태소 어휘	학교, 를
	형태소 품사	명사, 격조사
	어절이 속한 청크 타입	명사구, 동사구, 부사구, 독립어구, 복합 명사/동사구 등
	어절의 문장 내 위치	첫 어절을 1번으로 하여 숫자로 표현
	어절이 속한 청크 위치	첫 청크를 1번으로 하여 숫자로 표현
	관형형 어미 포함 여부	용언구의 경우 관형절에 해당하는지 0, 1로 표현
어절 간 정보	지정사 포함 여부	지정사를 포함하는지를 0, 1로 표현
	어절간 거리	어절간의 거리를 숫자로 표현
문장 단 위 정보	청크간 거리	청크간의 거리를 숫자로 표현. 같은 청크일 경우 0.
	문장 길이	문장의 길이를 숫자로 표현
	청크 개수	청크의 개수를 숫자로 표현
	지배-피지배 어절간 평균 거리	거리를 숫자로 표현
	지배 어절 개수	개수를 숫자로 표현

지역 정보를 이용하는 방법으로 나뉘어져 있는데 CONLL 06, 07[1, 2]에서는 전역 정보를 이용하는 방법이 조금 나은 성능을 보이는 것으로 알려졌다. 그러나 문장 단위 구조 정보를 이용하는 방법도 후보가 너무 많이 생성되어 결과적으로는 처리 시간이 많이 걸리는 단점이 있다. 지역 정보를 보는 경우에도 최종 목표가 문장에 대한 전체 문장의 구조를 구성하는 것이기 때문에 자칫 지역적인 특성만을 반영하여 부분 문장 구조 분석 초기에 잘못된 의존 관계를 선택할 경우 국지적 오류가 발생하여 전체 문장 단위에서는 최적의 트리를 찾지 못하는 경우가 많이 발생한다.

본 논문에서는 통계 모델로는 FSMO(fixed-threshold sequential minimal optimization)을 적용한 structural SVM 모델[20]을 사용하였다. Structural SVM은 기존의 SVM을 확장하여 multi-classification 뿐만 아니라 sequence labeling 등에 적용할 수 있다[21]. Structural SVM은 다음과 같은 linear discriminant 함수를 학습하게 된다.

$$h(x) = \arg \max_{y \in Y} \mathbf{w}^T \cdot \Psi(x, y)$$

본 논문에서는 다음 수식과 같이 margin-rescaling [20] 방법을 사용하였다.

$$\min_{\mathbf{w}, \xi} \frac{1}{2} \mathbf{w}^T \mathbf{w} + \frac{C}{n} \sum_i \xi_i, \text{ s.t. } \forall i, \xi_i \geq 0$$

$$\forall i, \forall y \in Y \setminus y_i : \mathbf{w}^T \cdot \Psi(x_i, y_i) - \mathbf{w}^T \cdot \Psi(x_i, y) \geq \Delta(y_i, y) - \xi_i$$

위 식에서 (x_i, y_i) 는 학습데이터이고, $\Delta(y_i, y)$ 는 loss 함수이다.

이러한 모델을 이용하여 지역 정보를 학습하기 위해서

$$x_i = (f_1, f_2, \dots, f_n) \quad y: \text{label}_k$$

$$f_j : \text{문맥 자질}, \quad y: \text{어절의 label}$$

를 사용하였고

전역 정보 학습을 위해서는 길이가 m인 문장은

$$X = (x_1, x_2, \dots, x_m), \quad Y = (y_1, y_2, \dots, y_m)$$

$$x_i : i\text{번째 어절의 문맥 자질}$$

$$y_i : i\text{번째 어절의 레이블}$$

을 사용하였다.

3.4 의존 관계 분석 알고리즘

앞에서 언급한 의존 관계과 SVM 모델을 이용하여 전역 및 지역 정보를 이용하여 최적의 문장 구조 및 레이블을 분석하는 알고리즘은 [표2]와 같다.

표 2 격 관계 및 문장 구조 분석 알고리즘

```

Mod_l : Model of Word Relation Level
Mod_g : Model of Sentence Level
CandRel : set of Relation between Word
CandSubTree : set of candidated Sub Tree

begin
for i = n ... 1
  clear CandRel
  for j = i-1 ... 1
    generate possible (i, j) pair;
    add (i, j) to CandRel;
  if( count(CandRel) > 1 )
    score  $\forall (i, j) \in \text{CandRel}$ , with Mod_l;
     $\forall (i, j) \in \text{CandRel}$ , delete if (i, j) lower than Mean
    Score of CandRel;
  else continue;

  generate possible subTik including  $\forall (i, j) \in \text{CandRel}$ ;
   $\forall \text{subT}^{\text{ik}} \in \text{CandSubTree}$ , delete subTik
  if subTik violate Dependency Rule;
  score  $\forall \text{subT}^{\text{ik}} \in \text{CandSubTree}$  with Mod_g;
   $\forall \text{subT}^{\text{ik}} \in \text{CandSubTree}$ , delete subTik
  if subTik violate Case Rule;

  delete subTik  $\in \text{CandSubTree}$  if Rank of subTik lower than N

score Tree with Mod_g and select best score Tree
end

```

제안하는 알고리즘에 따라 ‘나는 밥을 먹고 동생은 학교에 갔다’ 라는 문장을 후방 탐색하며 어절에 대해 모든

가능한 의존 관계 쌍을 생성하고 생성된 관계 쌍을 어절 단위 학습 모델을 이용하여 레이블과 점수(score)를 얻는다. 그림은 어절 ‘먹고’의 지배소를 찾는 과정을 표시한 것으로 그림의 상단 부분의 점선으로 표시된 것과 같이 (먹고, 동생은), (먹고, 공부를), (먹고, 했다) 모든 관계 후보를 생성한다 (먹고, 공부를) 관계는 앞에서 언급한 ‘교차하지 않는다’는 의존 문법에 어긋나기 때문에 후보에서 제외되며 남은 후보인 (먹고, 동생은), (먹고, 했다)를 대상으로 지역 정보 통계를 통해 레이블과 점수를 그림4와 같이 얻게 되고 일정한 기준에 의해 관계 후보를 제거하게 된 부분 문장 분석 및 격 관계 인식은 그림4의 하단과 같다.

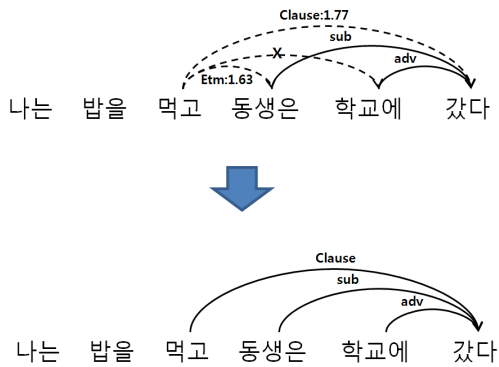


그림 4 지역 정보 모델을 이용한 후보 선택

다음 어절 ‘밥을’에 대해서는 의존 문법에 어긋나는 후보를 제외하고 (밥을, 먹고), (밥을, 갔다)의 두 후보만 남게되고 지역 모델에서는 두 후보의 점수가 변별력이 없다고 판단할 경우 두 후보 모두 남게 되고 부분 문장 구조는 그림5와 같이 2개로 늘어나며 다음 어절인 ‘나는’으로 넘어간다.

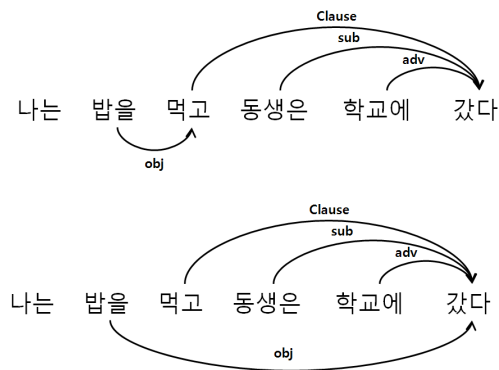


그림 5 부분 문장 구조 후보의 예

마지막 어절인 ‘나는’은 의존 문법을 적용하여 그림 상단의 부분 문장 구조에서는 (나는, 밥을), (나는, 먹고), (나는, 갔다)가 후보가 되고 그림5 하단의 부분 문장 구조에서는 (나는, 밥을), (나는, 갔다)가 후보가 되어 그림6과 같이 5개의 문장 구조가 가능하게 되며 이 중에서 최적의 문장 구조와 레이블을 분석하기 위해서 학습된 전역 통계 모델을 이용하여 다음과 같은 결과를

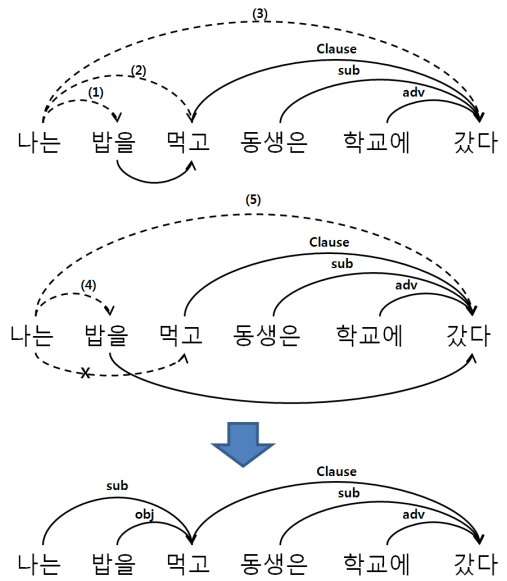


그림 6 최종 결과 선택 과정

아래의 결과는 1,4번과 2, 3, 5번은 레이블 열린 값지만 관계가 다르기 때문에 다른 문장 구조이다

- (1) (adv, sub, clause, obj, np) : 6.01
- (2) (adv, sub, clause, obj, sub) : 6.99
- (3) (adv, sub, clause, obj, sub) : 6.38
- (4) (adv, sub, clause, obj, np) : 6.32
- (5) (adv, sub, clause, obj, sub) : 6.05

5개의 후보 중에서 가장 점수가 높은 후보를 그림6 하단 부분과 같이 최종적으로 분석된 결과를 얻게 된다

4. 실험 및 평가

본 논문의 내용을 평가 하기 위해서 지식 DB 사업에서 구축한 의존 관계 태깅된 10만여 문장 중에서 1만 문장을 학습 문장으로 사용하고 258문장을 실험 집합으로 사용하였으며 지식 DB 사업의 결과물이 문장 구조 분석이라는 관점에서 난이도가 높기 때문에 일반 뉴스 기사 103 문장에 대해서도 실험을 하였다. 실험 문장의 평균 어절 수는 각각 21.72개, 14.34개이다.

입력 문장은 자동으로 형태소 분석 킷을 거쳤고 이 과정에서 오류는 수정하지 않아 결과에 포함되었다.

4.1 실험

제안한 의존 구문 분석 방법의 평가를 위해 문장의 의존 관계 및 레이블 분석 정확도를 평가 척도로 사용하였다. 제안하는 방법의 타당성을 확인하기 위해서 실험은 지역 정보 모델을 이용한 실험 전역 정보 모델을 이용한 실험, 제안하는 의존 관계 및 문장 단위 학습 모델을 같이 사용하는 실험으로 나누어 진행하였고 각각 뉴스 문장과 지식 DB 사업의 장문을 사용했다

표 4 성능 비교

	지역모델		전역모델		제안 방법	
	뉴스	장문	뉴스	장문	뉴스	장문
정확률(%)	75.8	69.3	78.0	70.9	79.1	71.9
최종후보수 (평균)	4.5	6.1	463.7	3588.3	13.4	26.0
처리시간 (초)	6.4	10.7	783.0	976분	10.3	55.9

4.2 평가

실험 결과 제안한 방법은 4와 같다. 뉴스 문장에서는 전역 정보나 지역 정보만 사용한 방법보다 각각 1.2%, 3.3% 성능 향상의 효과가 있었으며 상대적으로 장문인 지식 DB 사업의 문장에서는 각각 1%, 2.6%의 성능 향상 효과가 있었다. 지역 정보를 쓰지 않고 의존 문법에 어긋나지 않는 문장 구조의 후보를 대상으로 한 전역 모델만을 이용한 방법의 경우에는 성능은 같으면서 처리 시간이 76배나 빨랐으며 장문인 경우에 전역 모델은 처리 속도가 심각한 문제를 일으켜서 1문장 평균 3분 40초 정도의 시간이 걸려서 20어절 이상의 장문에는 적용하기 힘든 정도의 속도를 보였다.

성능의 경우 이론적으로는 전역 모델의 성능보다 제안하는 모델의 성능이 우수할 수 없으나 모델에서 가장 좋은 점수를 얻은 결과가 항상 정답이 아니기 때문에 이러한 모델 자체의 오답이 제안한 시스템에서는 지역 모델을 적용하는 과정에서 제거되어 제안한 시스템이 전역 모델에 비해 더 좋은 성능을 보이게 됐다.

지역 모델을 이용한 실험은 최상의 점수를 갖는 1개의 관계만을 선택할 경우 국지적 오류의 위험성이 있어서 상황에 따라 점수의 평균값을 이용하거나 n개의 후보를 선택하는 방법을 사용하였고 문장에 대한 점수는 각 관계간의 점수를 합한 형태로 정하였다

오류는 주로 후방 탐색 초기에 관계 후보를 제거하여 정답 구조를 최종 후보로 생성할 수 없는 경우와 거리가 먼 어절 간의 관계를 인식하지 못 해서 생긴 것들과 전 단계에서 형태소 분석 및 구 묶음 오류로 발생한 것들이 많았다.

5. 결론 및 향후 연구

제안한 방법은 한국어 문장의 구조와 격 레이블을 분석하기 위한 것이다. 전역 정보와 지역 정보를 적절히 사용하여 전역 정보에서 발생하는 속도 지연 문제와 지역 정보에서 발생할 수 있는 국지적 오류를 극복하기 위해서 전역 및 지역 정보 단위로 학습 모델을 분리하여 최상의 레이블된 문장 구조를 탐색하는 방법을 제안하였다.

향후 연구로는 학습 모델을 전체 문장 단위와 두 어절 간의 의존 관계만을 구축하였으나 중간 단계로 지배소를 중심으로 학습 모델을 구축하여 추가가 필요하다. 그리고, 구성된 트리의 경우 이중 주어목적어, 문장 성분에 대해 인식을 하지 못 하기 때문에 본 논문을 바탕으로 하여 이러한 문장 성분에 대해서도 인식할 수 있는 연구

가 필요하다.

참고문헌

[1] CoNLL-x, <http://www.cnts.ua.ac.be/conll2006/>

[2] CoNLL-2007 <http://www.cnts.ua.ac.be/conll2007/>

[3] Soojong Lim, "Dependency Relation Analysis Using Caseframe for Encyclopedia Question-Answering System", IECON, 2004

[4] Sekine, S., Uchimoto.k. Isahara, H., "Backward beam search algorithm for dependency analysis of Japanese", Proceedings of the 18th conference on computational linguistics vol 2 pp.754-760, 2000

[5] 김창현, "한국어 구문 분석을 위한 오른쪽 우선 차트 파서", KAIST 석사논문, 1993.

[6] 류범모, "한국어 파서에서의 지역 의존관계의 이용", 제8회 한글 및 한국어 정보처리, 1996

[7] 이성욱, "변환 규칙 학습기를 이용한 한국어 의존구조 분석기", 제9회 한글 및 한국어 정보처리, 1997.

[8] 장명길, "통계/의미 정보를 이용한 한국어 의존 파싱", 제9회 한글 및 한국어 정보처리, 1997.

[9] 전은희, "한국어 동사의 격틀 정보를 이용한 구문분석 후처리기", 제13회 한글 및 한국어 정보 처리, 2001.

[10] 김광백, "구간 분할 기반 한국어 구문 분석", 연세대 석사논문, 2003.

[11] Jason M. Eisner, "Three New Probabilistic Models for Dependency Parsing: An Exploration", COLING, 1996

[12] Ryan McDonald, Koby Crammer, Fernando Pereira, "Online Large-Margin Training of Dependency Parsers", ACL, pp 91-98, 2005

[13] Tetsuji Nakagawa, "Multilingual Dependency Parsing using Global Features", EMNLP-CoNLL, pp.952-956, 2007

[14] Joakim Nivre, Gulsen Eryigit, Svetoslav Marinov, "Labeled Pseudo-Projective Dependency Parsing with Support Vector Machines", CoNLL, pp.221-225, 2007

[15] Joakim Nivre, Ryan McDonald, "Integrating Graph-Based and Transition-Based Dependency Parsers", HLT pp950-958, 2008

[16] Hiroyasu Yamada, Yuji Matsumoto, "Statistical Dependency Analysis with Support Vector Machines", IWPT, 2003

[17] 이용훈, 이종혁, "기계학습 기법을 이용한 한국어

구문 분석", 한국컴퓨터 종합 학술대회 Vol.35
pp.285-288, 2008

- [18] Hoojung Chung, "Statistical Korean Dependency Parsing Model based on the Surface Contextual Information", PH.D. dissertation, 2004
- [19] Yoshihide Kato et al, "Efficient Incremental Dependency Parsing", NLPRS 2001
- [20] Changki Lee, "Structured SVM Using Fixed-Threshold Sequential Minimal Optimization" , ETRI Journal Vol 31 No 2, 2009
- [21] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun, "Support Vector Machine Learning for Interdependent and Structured Output Spaces," Proc. ICML 2004, 2004.