의존 구문분석을 위한 한국어 의존관계 가이드라인 및 엑소브레인 언어분석 말뭉치

임준호, 배용진, 김현기, 김윤정[1], 이규철[O]
한국전자통신연구원 지식마이닝연구실, 울산대학교 국어국문학과[1], 충남대학교 컴퓨터공학과[O]
{joonho.lim, yongjin, hkk}@etri.re.kr, jungi0006@mail.ulsan.ac.kr[1], kclee@cnu.ac.kr[O]

Korean Dependency Guidelines for Dependency Parsing and Exo-Brain Language Analysis Corpus

Joon-Ho Lim, Yongjin Bae, Hyunki Kim, Yunjeong Kim[1], Kyu-Chul Lee[O]
Electronics and Telecommunications Research Institute, Ulsan University[1], Chung-nam University[O]

요 약

2000년대 중반 세종 구구조 구문분석 말뭉치가 배포된 이후 의존 구문분석이 구문분석 연구의 주요 흐름으로 자리 잡으면서 많은 연구자들이 구구조 구문분석 말뭉치를 개별적으로 의존구조로 변환하여 구문분석 연구를 수행하였다. 하지만 한국어 문장의 의존구조 표현에 대한 논의가 부족하여 서로 다른 의존구조로 변환 후 구문분석을 연구함으로써 연구 효율성이 저하되는 문제가 발생하였다 본 연구에서는 이와 같은 문제에 접근하기 위하여 한국어 문장에 대한 의존관계 가이드라인을 제안한다. 그리고 제안하는 가이드라인을 기반으로 구축한 엑소브레인 언어분석 말뭉치(725 문장)에 대해 소개한다.[1)]

주제어: 한국어 의존구문분석, 의존관계 가이드라인, 엑소브레인 언어분석 말뭉치

1. 서론

2000년대 중반 21세기 세종계획 연구 결과물인 세종 구구조 구문분석 말뭉치가 구축 및 배포되어 많은 구구조 기반 한국어 구문분석 연구가 가능하게 되었다 [1]. 그리고 2000년대 중반 이후 영어권을 중심으로 의존구조 기반 구문분석이 연구의 주요 흐름으로 자리 잡으면서, 많은 구문분석 연구자들이 개별적으로 세종계획 구구조 말뭉치를 의존구조로 변환하여 구문분석을 연구하게 되었다 [2-9].

구구조 말뭉치의 의존구조 변환과 관련하여 영어권 연구를 살펴보면, 1994년 Magerman이 지배소 퍼컬레이션 테이블(head percolation table)을 처음 제안하였고, 이후, Collins (1999) 지배소 규칙, Stanford (2006) 변환 방법, Yamada and Matsumoto (2003)에 기반을 둔 Penn2Malt 변환 방법 등이 제안되었다 [10-13]. 그리고 최근 LTH에서 새로운 변환 방식을 제안하였고, 이를 이용할 경우 구문분석 자체의 성능은 약 3~4% 정도 하락하지만, 의미역 인식에 있어서는 더 높은 성능을 보임을 보였다 [14].

하지만 한국어 의존 구문분석 연구는 아직 의존구조 표현에 대한 논의가 부족하여, 각 연구자들마다 서로 다른 의존구조로 변환하여 구문분석을 연구하게 되었고, 이로 인해 연구의 효율성이 떨어지는 문제가 발생하게 되었다.

본 연구에서는 이와 같은 문제에 접근하기 위하여 한국어 문장에 대한 의존구조 표현 방안을 제안하고자 한다. 제안하는 의존관계 가이드라인은 해외 의존 구문분석 연구의 결과물을 한국어에서도 활용할 수 있도록 영어권 연구에서 일반적으로 적용되는 단일 지배소 원칙 (Single head constraint)과 투사성 원칙 (Projective Constraint)을 적용하였다. 그리고 기구축된 대용량의 세종 구구조 구문분석 말뭉치를 변환하여 활용할 수 있도록 세종 구구조 구문분석 말뭉치와 동일한 태그세트를 사용하였다. 마지막으로 한국어의 고유 특성을 반영하기 위하여 지배소 후위 원칙을 적용하였다.

본 논문의 구성은 다음과 같다. 2장에서는 제안하는 한국어 의존관계 가이드라인에 대해 소개하고, 3장에서는 제안 가이드라인을 이용하여 구축한 엑소브레인 언어분석 말뭉치에 대해 소개한다. 이후 4장에서는 제안 가이드라인과 다른 한국어 의존 구문분석 말뭉치와의 비교를 소개하고, 마지막 5장에서 결론을 맺도록 한다.

2. 한국어 의존관계 가이드라인

본 절에서는 한국어 의존 구문분석을 위한 의존관계 가이드라인을 소개한다. 2.1절에서는 기본원칙을, 2.2절에서는 의존관계 태그세트를, 2.3절에서는 문장 유형 별 의존관계 가이드라인을, 마지막으로 2.4절에서는 구 유형 별 의존관계 가이드라인을 설명한다.

2.1 기본 원칙

[O]: 교신저자

1) 엑소브레인 언어분석 말뭉치는 https://astc.etri.re.kr/ 사이트를 통해 배포할 예정이다.

의존관계 가이드라인의 기본 원칙은 아래와 같다.[2)]

(1) 자연언어처리를 위한 일관성 유지와 효율성 제고에 초점을 두되, 일반 언어학적 관점에서도 크게 벗어나지 않도록 한다.
(2) 문장의 표층 구조를 중시하여 분석한다.
(3) 의존관계 분석의 기본 단위로 어절을 사용한다.
(4) 지배소 후위 원칙에 따라 각 어절의 지배소는 자신보다 뒤에 위치하도록 분석한다.
(5) 각 어절은 1개의 지배소를 가진다. (Single-Head Constraint)
(6) 각 어절 및 지배소 쌍은 서로 교차하지 않는다. (Projective Constraint)
(7) 보어와 부가어를 구분하되 보어의 범위를 엄격히 제한한다.
(8) 원칙적으로 접속과 내포를 구별하지 않으며, 접속절은 모두 부사절로 분석한다. (다만, 명사구 접속은 인정한다.)
(9) 하나의 주어가 모문과 내포문 모두에 관련되어 있으면 모문과 내포문의 관계에 따라 해당 주어의 지배소를 결정한다.

기본원칙 (5) 단일 지배소 제약과 (6) 비교차 제약은 한국어에 적합하지 않으나, 영어권 주요 연구와의 호환을 위하여 설정하였다. 한국어의 다중 지배소 현상 및 교차 발생 현상의 예는 아래와 같다.

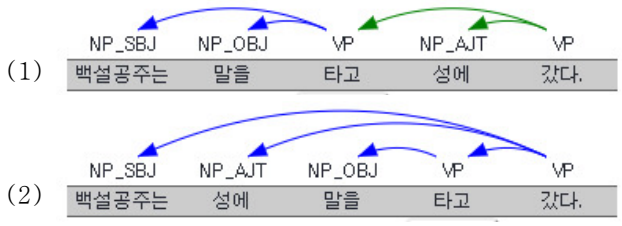

그림 1. 한국어 다중 지배소 현상(1- '백설공주는' 의 지배소는 '타고' 와 '갔다.' 이다.)과 교차 발생 현상 (2- '성에' 가 '갔다' 에 의존하여 '백설공주는' 이 '타고' 에 의존하지 못한다.)

2.2 의존관계 태그세트

문장 내의 각 어절은 자신 어절과 지배소 어절 사이의 관계를 표현하는 의존관계 태그를 가진다. 의존관계 태그는 구문 태그와 기능 태그를 결합하여 사용하고, 세종 구구조 말뭉치를 활용할 수 있도록 세종 구구조 말뭉치와 동일한 태그 세트를 사용한다.

2.3 문장 유형 별 의존관계 가이드라인

의존관계 설정을 위한 문장 유형 구분은 일반 언어학 통사론의 기준을 따라 '주어-서술어' 관계가 한 개인 홑문장과 '주어-서술어' 관계가 2개 이상인 겹문장으로 분류하여 분석한다 [15-16].

표 1. 구문 태그

구문 태그	의미
NP	체언 (명사, 대명사, 수사)
VP	용언 (동사, 형용사, 보조용언)
AP	부사구
VNP	긍정 지정사구 (명사+이다)
DP	관형사구
IP	감탄사구 (호칭 및 대답 등의 표현)
X	의사 구 (pseudo phrase, 조사 단독 어절 또는 기호 등)
L	부호 (왼쪽 괄호 및 따옴표)
R	부호 (오른쪽 괄호 및 따옴표)

표 2. 기능 태그

기능 태그	의미
SBJ	주어
OBJ	목적어
MOD	관형어 (체언 수식어)
AJT	부사어 (용언 수식어)
CMP	보어
CNJ	접속어 (~와)

홑문장을 이루는 문장 구성 성분은 크게 주어, 목적어, 관형어, 부사어, 보어, 서술어로 구분한다. 관형어는 의미적으로 수식하는 명사구에 의존하도록 분석하고, 서술어는 문장 전체를 지배하는 가상 ROOT 노드에 의존하도록 분석한다. 그 이외 문장 성분은 서술어에 의존하도록 분석한다. 홑문장의 의존관계 설정 예는 그림3과 같다.

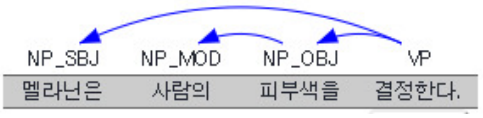

그림 2. 홑문장 의존관계 설정 예

겹문장은 다시 접속문과 내포문으로 구분되나, 기본원칙 (8)에 따라 접속문은 부사절 내포문의 유형으로 분석한다. 내포문은 크게 명사절 내포문, 관형절 내포문, 부사절 내포문, 서술절 내포문, 직접인용절 내포문, 간접인용절 내포문의 6가지 유형으로 구분된다. 각 내포문 유형에 대한 예는 아래와 같다.

2) 본 의존관계 가이드라인 설정에 21세기 세종계획의 결과물인 "구문분석 말뭉치 분석 지침(Ver. 2003-1)"을 참고하였다.

- 명사절 내포문: 우리는 시민의 관전태도도 그만큼 성숙했음을 잊지 말아야 한다.
- 관형절 내포문: 멜라닌은 사람의 피부색을 결정하는 주요 요소이다. (관계관형절) / 내가 사진을 좋아하는 사실을 친구들은 다 안다. (동격관형절)
- 부사절 내포문: 멜라닌은 자외선을 차단해서 자외선으로부터 피부를 보호해 준다. / 비가 와서 땅이 미끄럽다.
- 인용절 내포문: 그는 그녀가 그 일을 해냈다고 말했다. (간접인용절) / 그는 "그녀가 그 일을 해냈어."라고 말했다. (직접인용절)
- 서술절 내포문: 할머니께서 치아가 아프시다. / 사과가 색깔이 예뻤다.

내포문 분석은 모문과 내포문의 주어 및 서술어가 각각 존재하는 경우와, 모문과 내포문의 주어가 동일하여 하나가 생략된 경우로 나누어 볼 수 있다.

모문 및 내포문의 주어 및 서술어가 각각 존재하는 경우, 모문 및 내포문 각각을 홑문장과 같이 분석한다.

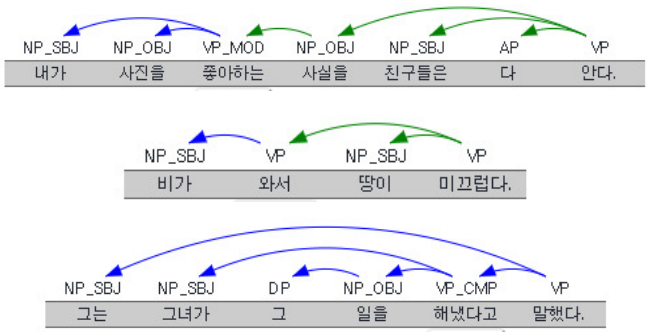

그림 3. 관형절, 부사절, 인용절 내포문의 의존관계 설정 예

모문과 내포문의 주어가 동일하여 하나가 생략된 경우, 기본원칙 (9)에 따라 모문과 내포문의 관계에 따라 해당 주어의 지배소를 결정한다. 명사절, 부사절, 간접인용절의 경우 동일 주어를 내포문의 서술어에 의존하도록 분석한다.[3] 관형절의 경우 동일 주어를 모문의 서술어에 의존하도록 분석한다.

서술절 내포문은 이중주어 구문과 동일하게 각 구성성분 별로 동일한 서술어에 의존하도록 분석한다. 직접인용절 내포문은 모문과 내포문을 인용부호에 의해서 구분할 수 있으므로, 모문은 모문 내의 주어와 서술어 간의 의존관계를 분석하고, 내포문은 내포문 내의 주어와 서술어 간의 의존관계를 분석한다.

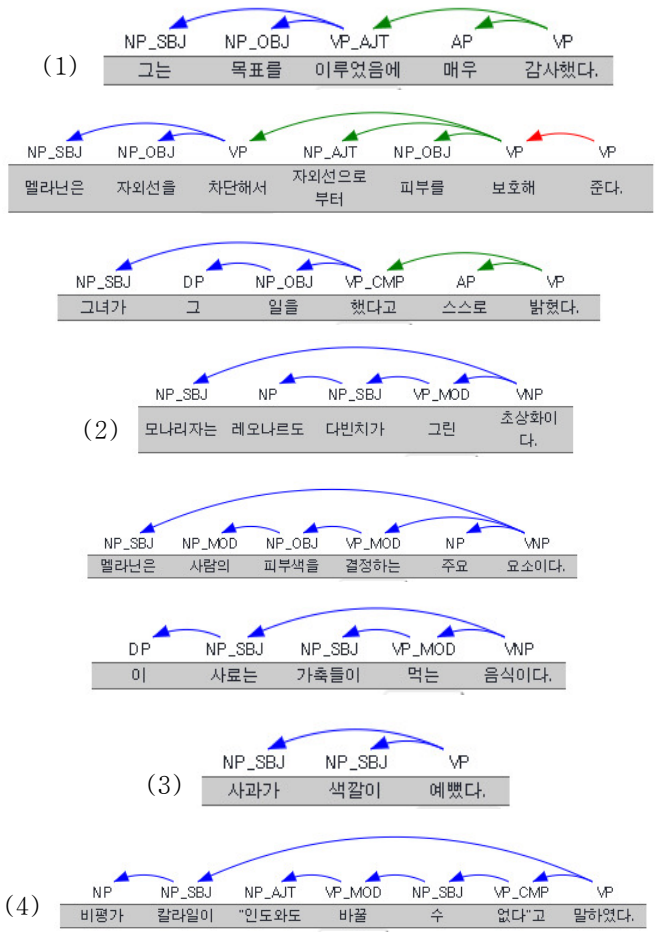

그림 4. (1) 명사절, 부사절, 간접인용절 내포문 분석 예 (주어생략 경우), (2) 관형절 내포문 분석 예 (주어생략 경우), (3) 서술절 내포문 분석 예, (4) 직접인용절 내포문 분석 예

2.4 구 유형 별 의존관계 가이드라인

구 유형 별 의존관계 가이드라인에서는 [관형어 + 명사 + 명사] 유형, 명사구 접속 유형, [용언 + 용언] 유형에 따른 분석 가이드라인을 소개한다.

첫 번째로, [관형어 + 명사 + 명사] 구의 경우, 관형어는 의미적으로 수식하는 명사에 의존하도록 분석한다. 예를 들어, "무분별한 포획 문제"의 경우 "무분별한"의 지배소는 "포획"이고, "심각한 사회 문제"의 경우 "심각한"의 지배소는 "문제"이다.

두 번째로, 복수 개의 명사구가 접속 또는 나열된 경우, 나열된 명사구는 가장 마지막 명사구에 의존하도록 분석한다.

세 번째로, [용언 + 용언] 유형의 경우, 다시 세부적으로 [본용언 + 본용언] 구성, [본용언 + 보조용언] 구성, 의사보조용언 구성으로 구분하여 분석한다.

[본용언 + 본용언]과 같이 본용언이 연속적으로 나타날 경우, 주어는 앞에 위치한 본용언에 연결한다.

[본용언 + 보조용언]과 같이 본용언과 보조용언이 연속하여 두 개 이상 나올 때는 주어는 본용언에 연결하고

3) 하나의 주어가 여러 서술어를 지배소로 가질 수 있는 경우는 다중 지배소 가이드라인을 통해서만 올바르게 해결이 가능한 문제이다. 다만, 내포문과 모문 둘 다 지배소가 맞는 경우에 한 해, 본 연구에서는 관형절이 아닐 때 내포문에 의존하도록 하였다. 예를 들어, "결국 그는 목표를 이루었음에 감사했다."와 같이 문두에 첫 번째 서술어를 수식하는 부사가 나타날 경우, 비교차 제약에 의해 주어를 모문의 서술어에 연결하기 어려운 만큼, 내포문의 서술어에 연결하는 것이 지역적 안정성이 높다 판단된다. 또한, 본 연구에서는 모문의 생략된 주어는 별도의 생략된 필수격 인식 모듈을 통하여 모문의 서술어의 생략된 주어를 인식하도록 하였다.

표 3. 의사 보조용언 구성 예

의사 보조용언 구성	예문
-ㄹ 수/리(가) 있다/없다	- 그는 일어날 수 없었다.
-ㄴ/ㄹ + 의존(일반)명사 + 이다 (※의존명사: 것/터/뿐/따름/모양/지경/참/중 ※일반명사: 노릇/예정/길)	- 나는 곧 밥을 먹을 것이다. - 그가 고향이 그리운 모양이다. - 나는 외갓집에 심부름을 갔다 오는 길이다.
-ㄹ {만/법/듯}하다	- 그 영화는 아이들과 같이 볼 만하다.
-는 말이다	- 정녕 일을 그르쳤단 말이냐?
-ㄴ/ㄹ 듯(도) 하다	- 따스한 손길로 머리를 어루만져 주시는 듯 했다.
-ㄹ 것 같다	- 곧 비가 올 것 같다.
-ㄹ 것을(걸) 그랬다	- 거기 나가서 동창들한테 협찬금 받아낼 걸 그랬잖아.
-어서는 안되다	- 우리는 경계를 늦추어서는 안된다.
-고 해서	- 시간도 없고 해서 그는 친척집에 들르지 않았다.
-든지 하다	- 모처럼 왔으면 따뜻한 밥 한 그릇 먹고 가든지 해야지.

보조용언은 본용언에 연결한다.

마지막으로 의사 보조용언 구성은 언어학에서는 양태 표현에 포함되는 개념으로, 추측, 바람, 판단, 행동지시, 의도, 의지 표현 등의 표현이 있다. 의사보조용언 구성은 주 서술어 다음에 보조용언은 아니지만 서법을 나타내는 언어 단위들이 오는 경우로, 구체적인 예는 표 3과 같다. 이 경우 주어는 주 서술어와 의존관계를 연결하고, 뒤따르는 언어 단위들은 주서술어의 뒤에 차례대로 의존관계를 연결한다. 의사 보조용언 구성의 의존 구문분석 예는 아래와 같다.

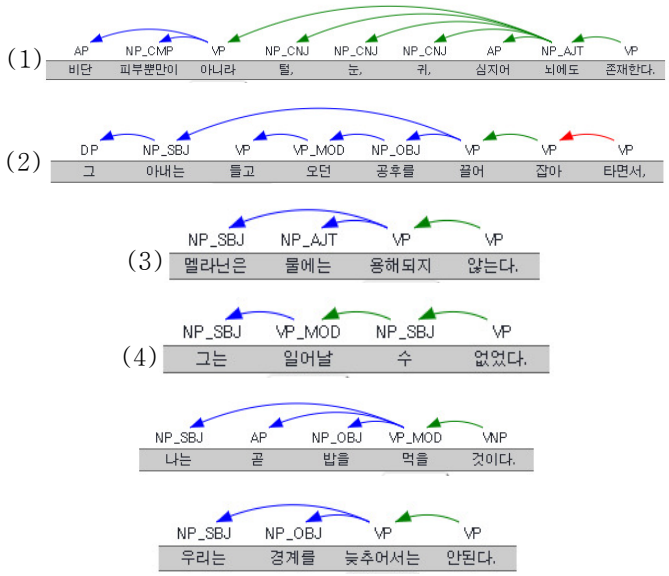

그림 5. 구 유형 별 의존관계 분석 예 (1- 명사구 접속 예, 2- 본용언+본용언 예, 3-본용언+보조용언 예, 4-의사보조용언 구성 예)

3. 엑소브레인 언어분석 말뭉치

엑소브레인 언어분석 말뭉치는 질의응답(Question Answering) 시스템 개발을 위하여 구축된 말뭉치로, 퀴즈 질문 문서 및 그 정답 단락 문서의 쌍으로 구성되어 있다. (각 질문 및 정답 단락은 복수의 문장으로 구성될 수 있으며, 각 질문 및 정답 단락 별로 별도의 문서로 구성된다.) 엑소브레인 언어분석 말뭉치의 세부적인 통계정보는 아래와 같다.

표 4. 엑소브레인 언어분석 말뭉치 통계

	질문	정답단락	총계
문서 수	117	322	439
문장 수	183	542	725
어절 수	1,998	6,522	8,520

엑소브레인 언어분석 말뭉치는 형태소분석, 어휘의미분석, 개체명인식, 구문분석, 의미역인식에 대한 언어분석 정답을 제공한다 [17]. 포맷은 JSON 포맷으로 제공되고, 언어분석 문장의 예는 표5와 같다.

엑소브레인 언어분석 말뭉치는 언어분석 기술 개발을 위한 학습용으로는 그 양이 많지는 않으나, 동일 문장에 대해서 형태소분석부터 어휘 의미 분석, 개체명인식, 구문분석, 의미역인식까지의 언어분석 정답을 포함하고 있기 때문에, 세부 언어분석 기술뿐 아니라 전체 언어분석 파이프라인을 평가하기 위한 용도로 활용이 가능할 것이다.

4. 논의

본 절에서는 본 논문에서 제안한 한국어 의존관계 가이드라인과 다른 한국어 의존 구문분석 말뭉치와의 비교를 논의하고자 한다. 비교 대상 한국어 의존 구문분석 말뭉치로는 최진호(2011)의 세종 구구조 코퍼스의 의존구조 변환 규칙, UCorpus 의존 구문분석 말뭉치, Universal Treebank 이다.

최진호(2011) 연구에서는 세종 구구조 말뭉치를 의존구조로 변환하기 위한 의존관계 변환 규칙과 의존태그를 PennTreeBank와 유사하게 변환하기 위한 의존태그 변환 규칙을 제시하였다 [7]. 하지만, 세종 구구조 말뭉치의 의존구조 변환에 있어서, S, NP, VP 만을 이용하여 변환하면 본용언, 보조용언, 의사보조용언 등에 대한 구분이 어렵다는 단점이 있다. 예를 들어, “걱정이 아니 될 수 없었다.” 와 같은 문장의 경우 단순 변환 규칙을 적용하면 “걱정이” 어절이 “될”에 의존하지 않고 “없었다.” 를 지배소로 가지게 된다. 그리고, 세종 구구조 말뭉치는 <‘색의 컬렉션’ 이라는>과 같이 기호가 포함된 어절의 경우 <‘ / 색의 / 컬렉션 / ’ / 이라는> 과 같이 기호를 별도로 분리하여 태깅하여서, 이를 그대로 변환할 경우 어절 간의 태깅 단위가 다른 문제가 발생하게

된다[4]. (최진호(2011)의 의존 레이블 분포를 보면, 단일 기호 어절이 전체 어절의 2.42%에 이른다.)

UCorpus는 울산대학교에서 구축하여 배포 중인 말뭉치로, 세종 구구조 말뭉치를 전문가의 검토를 거쳐 의존 구조로 변환하였다 [18]. 문장 수는 약 36,000여 문장으로 구문분석기 개발 및 평가가 가능한 수준이다. UCorpus는 본 연구와 동일하게 세종 구구조 말뭉치의 기호 구분 어절을 공백 단위의 어절 기준으로 통합하여 구축하였다. 그리고 본 연구의 의사 보조용언 처리와 같이 주어의 의미적 서술어에 해당하는 어절로 의존관계를 설정하였다는 공통점이 있다. 세부적인 차이점으로는 본 연구에서는 명사구 나열형 구성에 대해서 가장 마지막 어절을 지배소로 본 반면, 명사구들의 순차적 연결로 보았다. 그리고, 하나의 주어의 지배 어절이 2개 이상일 경우, 본 연구에서는 문장 유형에 따라 모문의 서술어 및 내포문의 서술어를 선택하여 구축하였고, UCorpus는 주로 모문의 서술어를 선택하여 구축한 것으로 판단된다.

마지막으로, Universal Treebank는 전 세계 언어를 하나의 의존관계 기준으로 동일하게 구문분석하자는 목표로 구축 중인 코퍼스이다 [19]. 이 코퍼스의 가이드라인은 영어, 독일어, 프랑스어 등 다양한 언어의 특징을 반영하여 설계되었다. Universal Treebank 중 한국어 태깅 결과를 비교하면, 공백 기반 어절 단위는 본 연구와 비슷하나, 여러 기호 중 콤마만 별도의 어절로 분리하였다는 차이가 있다. 그리고, Universal Treebank에서는 compmod, dep, det, dobj, iobj와 같은 별도로 정의한 레이블을 사용하였다. 명사구 접속은 본 연구와 같이 마지막 어절에 의존하도록 분석하였다.

5. 결론

자연어처리에 있어서 문장의 구조적 중의성 해소는 필수적인 만큼, 한국어 의존 구문분석과 관련한 많은 연구가 있었지만, 아직 의미처리에 있어서 효율적인 의존관계 설정 방법에 대한 논의는 부족했던 것으로 판단된다.

본 논문은 한국어 의미처리에 있어서 효율적인 구문분석 가이드라인을 제안하였고, 이를 이용하여 실질적인 자연어 질의응답 시스템을 개발 중이다. 향후, 본 연구에서 제안한 가이드라인에 더불어 한국어에 적합한 다중 지배소 가이드라인, 의존관계 교차 가이드라인 등으로 가이드라인을 확장하고, 이에 기반한 말뭉치 및 구문분석 기법의 확장이 필요할 것이다.

감사의 글

이 논문은 2015년도 정부(미래창조과학부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임. (No.R0101-15-0062, 휴먼 지식증강 서비스를 위한 지능진화형 WiseQA 플랫폼 기술 개발)

4) 기호를 분리한 말뭉치를 기반으로 구문분석기를 학습한다면, 이 구문분석기를 실질적인 시스템에 적용하기 위해서는 입력 문장에 대해서 학습 말뭉치와 동일한 기호 분리 전처리가 필요하다는 단점이 따른다.

참고문헌

[1] 국립국어원, “21세기세종계획”, 2012
[2] S. Bucholz, E. Marsi, "CoNLL-X shared task on Multilingual Dependency Parsing," Proc. of CoNLL, pp.149-164, 2006.
[3] R. McDonald, K. Crammar, F. Pereira, "Online Large-margin Training of Dependency Parsers," Proc. of ACL, pp.91-98, 2005.
[4] J. Nivre, "An Efficient Algorithm for Projective Dependency Parsing," Proc. of IWPT, pp.149-160, 2003.
[5] 임수종, 김영태, 나동열, "자질 가중치의 기계학습에 기반한 한국어 의존파싱", 정보과학회논문지, 소프트웨어 및 응용 제38권 제4호, 2011.4, 214-223
[6] 안광모, 서영훈, "지배소 후보 집합을 이용한 한국어 의존 구문 분석 알고리즘", 정보과학회논문지, 소프트웨어 및 응용 제 41 권 제 1 호(2014.1)
[7] J.D. Choi, Martha Palmer, "Statistical Dependency Parsing in Korean: From Corpus Generation To Automatic Parsing", Proceedings of the 2nd Workshop on Statistical Parsing of Morphologically-Rich Languages (SPMRL 2011), pages 1-11, Dublin, Ireland, October 6, 2011.
[8] 오진영, 차정원, "고성능 비어휘정보 한국어 구문분석", 2010 한국컴퓨터종합학술대회 논문집, Vol. 37, No. 1(C), 295-298
[9] 오진영, 차정원, "키어절을 이용한 새로운 한국어 구문분석", 정보과학회논문지: 소프트웨어 및 응용 제 40 권 제 10 호(2013.10), pp. 600-608
[10] David M. Magerman. 1994. Natural language parsing as statistical pattern recognition. Ph.D. thesis, Stanford University.
[11] Michael J. Collins. 1999. Head-driven statistical models for natural language parsing. Ph.D. thesis, University of Pennsylvania, Philadelphia.
[12] M. de Marneffe, B. MacCartney and C. D. Manning. 2006. Generating Typed Dependency Parses from Phrase Structure Parses. In LREC 2006.
[13] Hiroyasu Yamada, Yuji Matsumoto, “Statistical Dependency Analysis With Support Vector Machines”, In proceedings of 8th International Workshop on Parsing Technologies, 2003,
[14] Richard Johansson and Pierre Nugues. Extended Constituent-to-dependency Conversion for English. In Proceedings of NODALIDA 2007. Tartu, Estonia, 2007
[15] 남기심, 고영근. “표준 국어 문법론”, 박이정
[16] 이재성, “글쓰기를 위한 4천만의 국어책”, 들녘
[17] 최미란, "형태소 태깅 말뭉치 작성용 품사 태그 세트", TTAK.KO-11.0010/R, 2015
[18] 김홍순,옥철영, “동형이의어 분별에 의한 한국어

표 5 엑소브레인 언어분석 말뭉치 예


```
{"sentence" : [{
        "id" : 0,
        "reserve_str" : "",
        "text" : "수사자와 암호랑이 사이에서 태어난 새끼",
        "morp" : [
                {"id" : 0, "lemma" : "수", "type" : "XPN", "position" : 0, "weight" : 1 },
                {"id" : 1, "lemma" : "사자", "type" : "NNG", "position" : 0, "weight" : 1 },
                {"id" : 2, "lemma" : "와", "type" : "JC", "position" : 9, "weight" : 0.0390333 },
                {"id" : 3, "lemma" : "암", "type" : "XPN", "position" : 13, "weight" : 1 },
                {"id" : 4, "lemma" : "호랑이", "type" : "NNG", "position" : 16, "weight" : 1 },
                {"id" : 5, "lemma" : "사이", "type" : "NNG", "position" : 26, "weight" : 1 },
                {"id" : 6, "lemma" : "에서", "type" : "JKB", "position" : 32, "weight" : 0.519517 },
                {"id" : 7, "lemma" : "태어나", "type" : "VV", "position" : 39, "weight" : 1 },
                {"id" : 8, "lemma" : "ㄴ", "type" : "ETM", "position" : 45, "weight" : 0.0390333 },
                {"id" : 9, "lemma" : "새끼", "type" : "NNG", "position" : 49, "weight" : 1 }
        ],
        "WSD" : [],
        "word" : [
                {"id" : 0, "text" : "수사자와", "type" : "", "begin" : 0, "end" : 2},
                {"id" : 1, "text" : "암호랑이", "type" : "", "begin" : 3, "end" : 4},
                {"id" : 2, "text" : "사이에서", "type" : "", "begin" : 5, "end" : 6},
                {"id" : 3, "text" : "태어난", "type" : "", "begin" : 7, "end" : 8},
                {"id" : 4, "text" : "새끼", "type" : "", "begin" : 9, "end" : 9}
        ],
        "NE" : [
                {"id":0, "text":"사자", "type":"AM_MAMMALIA", "begin" : 1, "end" : 1, "weight" : 0.133554, "common_noun" : 0},
                {"id":1, "text":"호랑이", "type":"AM_MAMMALIA", "begin" : 4, "end" : 4, "weight" : 0.133554, "common_noun" : 0}
        ],
        "chunk" : [],
        "dependency" : [
                {"id" : 0, "text" : "수사자와", "head" : 1, "label" : "NP_CNJ", "mod" : [], "weight" : 0.775765 },
                {"id" : 1, "text" : "암호랑이", "head" : 2, "label" : "NP", "mod" : [0], "weight" : 0.591125 },
                {"id" : 2, "text" : "사이에서", "head" : 3, "label" : "NP_AJT", "mod" : [1], "weight" : 0.724939 },
                {"id" : 3, "text" : "태어난", "head" : 4, "label" : "VP_MOD", "mod" : [2], "weight" : 0.556066 },
                {"id" : 4, "text" : "새끼", "head" : -1, "label" : "NP", "mod" : [3], "weight" : 0.109298 }
        ],
        "SRL" : [{
                "verb": "태어나", "sense": 1, "word_id": 3, "weight": 0.983073,
                "argument": [
                        {"type": "ARG2","word_id": 2,"text": "사이에서","weight" : 0.528739},
                        {"type": "ARG1","word_id": 4, "text": "새끼", "weight" : 0.79628}]
        }],
        "relation" : [],
        "SA" : [],
        "ZA" : []
 }],
 "entity" : []
}
```


의존관계 분석," 정보처리학회논문지/소프트웨어 및 데이터 공학 제3권 제6호 (2014. 6)

[19] Joakim Nivre. "Towards a Universal Grammar for Natural Language Processing," Computational Linguistics and Intelligent Text Processing, 2015.