한국어 용언 위계구조 자동구축*

송 상 헌 · 최 재 웅*
(Univ. of Washington · 고려대학교)

Sanghoun Song · Jae-Woong Choe, 2012. Automatic Construction of Verbal Type Hierarchy for Korean. *Language and Linguistics* 54, 201-238. This paper discusses an automatic way to derive a type hierarchy for verbal items in Korean based on their subcategorization. There are three steps: First, all the dependent categories of the each verb are extracted from the Sejong Treebank. Second, based on the frequency of the dependent categories of each verb, the most stable subcategorization frames are selected, and two statistical measures are tested with some variations in their cutoff values. The resulting subcategorization frames are then compared with those from the Sejong Electronic dictionary for evaluation. The final step is to form a type hierarchy for Korean verbal items, based on the chosen subcategorization information.

Keywords: subcategorization, Korean, Sejong Treebank, dependency relations, statistical test, Jaccard coefficient, HPSG, clustering, type hierarchy, automatic construction

주 제 어: 하위 범주화, 세종 구문분석 말뭉치, 의존관계, 통계적 검증, Jaccard 계수, 군집화, 유형 위계구조

---

* 본 연구는 'Automatic Construction of Korean Verbal Type Hierarchy using Treebank'이라는 제목으로 The 15th International Conference on Head-Driven Phrase Structure Grammar(HPSG08, 2008년 7월, Keihanna, Japan)에서 발표된 것을 발전시킨 것으로, 전 과정을 재분석하였다. 특히 통계식의 적용 및 임계치의 설정과 관련된 부분은 새롭게 구성되었다. 연구 초기부터 관심을 가지고 조언을 아끼지 않은 김종복 선생님과 2008년 당시 발표장에서 귀중한 지적을 해준 Hans Uszkoreit, Dan Flickinger, Laurie Poulson, Bart Cramer 등 여러 선생님들, 그리고 심사의 과정에서 좋은 지적을 해준 심사위원들께 깊은 감사를 드린다. 이 논문은 2010년 정부(교육인적자원부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(NRF-2010-327-A00212).

** 교신저자

1. 서론

전산언어학의 주요 쟁점 중의 하나는 개별 언어 별로 방대한 양의 어휘 정보를 어떻게 획득하느냐 하는 문제다. 기존 사전의 정보를 적절하게 가공하여 사용하던 단계를 지나 현재는 대규모 말뭉치로부터 통사-의미 정보를 직접 자동으로 추출하려는 시도가 많이 이루어지고 있다(Brent, 1993; Utsuro et al., 1998; Sarkar and Zeman, 2000; Korhonen et al., 2000; Gamallo, 2001; Chesley and Salmon-Alt, 2006). 이와 같은 어휘 정보의 자동 추출은 해당 언어의 모국어 화자가 어휘를 습득하는 방식과 개념적으로 다르지 않을 것이라는 전제에서 출발한다.

문장의 구성을 이해하는 데는 해당 문장의 술어가 지니는 하위범주화 정보가 가장 중요하다는 점에는 이견이 없다. 언어 이론상으로도 그러하고, 또한 언어 처리의 관점에서도 마찬가지다. 예컨대 아래에서 보이는 바와 같이 영어에서 'want'와 'hope'는 서로 비슷한 의미적 속성을 지님에도 불구하고, 그들이 취할 수 있는 문형은 각기 다르다(Brent, 1993:243).

(1) a. John wants Mary to be happy.
b. John hopes that Mary is happy.
c. *John wants that Mary is happy.
d. *John hopes Mary to be happy.

따라서 이러한 하위범주화 정보를 어떻게 획득하느냐 하는 것은 전산언어학 내에서도 어휘 정보 습득과 관련한 핵심 주제 중 하나가 된다.

이러한 맥락에서 본 연구가 관심을 가지는 바는 한국어 용언의 논항 관계 및 위계 구조를 어떻게 하면 (i) 효율적이면서도 (ii) 포괄적으로 그리고 동시에 (iii) 객관적인 방법으로 구축하여, 그 (iv) 활용가능성을 극대화 할 수 있을 것인가이다. 전체 연구는 세 가지 하위 단계로 구성이 된다. 첫 번째 단계는 말뭉치에서 하위범주화의 후보군을 모두 뽑아내는 (i) 의존소 추출이다(3절). 두 번째 단계는 추출된 의존소 가운데 실제 주어진 용언의 논항으로 판단되는 집합만을 통계적으로 걸러내는 (ii) 하위범주화 선별에[1] 해당한다(4절). 끝으로 세

번째 단계는 (iii) 위계구조 군집화이며, 이는 각각의 용언에 대하여 확립된 논항 구조를 군집화하여 얻어진 전체 용언의 위계구조를 말한다(5절).

이어지는 2절에서는 본 연구의 배경에 대한 개괄적인 소개와 함께 하위범주와 추출 및 위계 구조 구성의 대상이 되는 어휘군의 선정 절차가 논의될 것이다. 3절은 수형도 기반 접근법을 제시하며, 관련된 주요 이슈들에 대해서 살펴본다. 여기에서는 현재 가용한 가장 큰 규모의 통사분석 말뭉치인 세종구문분석 말뭉치를 논항 추출의 주 대상으로 활용한다. 4절은 통계적 기법을 통해 논항 구조를 확정하는 과정에 대해 살펴볼 것이다. 구체적으로는 Jaccard 계수와 이항 가설 검증에 기초한 통계식을 활용하여 도출된 결과를 세종전자사전과 교차 비교를 통해 평가할 것이다. 5절에서는 구성된 논항 관계에 입각하여 Head-driven Phrase Structure Grammar(=HPSG; Sag et al., 2003) 기반의 용언 위계구조를 구축하는 과정이 제시될 것이다. 끝으로 6절은 본 연구가 지니는 의의를 짚어보고, 추후 과제를 떠올려 본다.

2. 연구의 토대

서론에서 언급하였듯이 본 연구는 기존 사전에 대한 의존도를 최소한으로 줄인 전산 모형을 큰 방향으로 설정하고 있다.[2] 그러한 방향 속에서 한국어 용언 위계구조를 도출하는 것이 본 연구의 목표다. 단계별로 그러한 목표가 어떻게 달성될 수 있는지 세밀하게 보이기 전에 우선 본 절에서는 연구의 방법, 구성, 연구의 자원 및 대상 등 본 연구의 토대와 관련한 보다 구체적인 사항들을 논하기로 한다.

1) 본고에서는 '하위범주화 틀'이라는 용어와 '논항 구조'라는 용어가 사실상 같은 것을 지칭하는 것으로 사용되었다. 언어 이론적인 측면에서 양자 사이의 보다 세밀한 구분이 가능하겠으나, 실제 시스템을 구현하여 전산적으로 표상을 하는 것을 목적으로 한 본 연구에서는 양자의 엄정한 구분이 요구되지 않는다.

2) 자연어처리의 모형을 크게 대별하자면, 하나의 축은 기계-가독형 사전과 같은 언어 지식에 크게 의존하는 모형(knowledge-rich)일 것이고 다른 하나는 언어 지식에 대한 의존도를 줄인 모형(knowledge-poor)이다(Gamallo et al., 2001). 본 연구는 구문분석 말뭉치를 활용한다는 점에서는 전자처럼 보이나, 사전처럼 정제된 자원을 사용하지 않는다는 점에서 후자에 가깝다.

2.1. 연구의 방법

한국어의 각 용언이 어떠한 논항 구조를 지니고 있는가를 포괄적인 차원에서 검토를 하고 이들 사이의 관계를 파악하는 일은 이론 언어학 자체에서도 중요한 부분이다. 동사성 어휘의 논항 구조 및 그 변이 현상을 바탕으로 하여 그 위계를 설정한 연구로는 Levin(1993)을 들 수 있다. 그 연구는 영어의 주요 동사를 대상으로 한 것으로 각 어휘에서 투사되는 통사 구조를 하나씩 분류하고 이를 유형화한 것이다. 마찬가지로 한국어에서 이에 해당하는 자원으로서는 1998년부터 2007년까지 10년간의 연구 결과를 통해 구축된 세종전자사전을 들 수 있다. 이러한 기존의 연구가 논항 구조에 입각한 동사성 어휘의 위계를 설정하는 데 중요한 성과를 거둔 것은 사실이나, 이를 실제 자연어 처리 시스템에 곧바로 적용하기에는 몇 가지 문제가 따른다. 아래와 같은 네 가지 면에서의 한계를 지적할 수 있다.

활용가능성: 최근의 자연어 처리는 빈도 등의 정보에 입각한 통계적 수치에 크고 작은 의존을 하고 있는 관계로, 실제의 각 어휘가 가지는 논항 구조의 분포에 대한 계량적 연구가 요구된다. 시스템 구축의 차원에서는 통사 및 의미 단계의 정리와 함께, 빈도 등과 같은 분포적 특질에 대한 통계 정보가 필요하기 때문이다. 예컨대, '통계기반 구문분석' 또는 '통사 정보에 기반한 통계적 기계번역' 등의 최근 자연어 처리 모형은 이와 같은 통사 구조에 계량적 연구에 입각하고 있다. 이는 실제의 대규모 언어 자원을 활용하지 않고서는 성취하기가 어려운 목표다.

포괄성: 실제 시스템은 각 매개 언어의 다양한 현상을 반영하여 처리 결과를 내어야 한다는 점에서, 제한된 수의 어휘를 대상으로 진행한 연구 결과는 필연적인 한계를 보인다. 즉, 매개 언어의 동사성 어휘 전반을 대상으로 하여 포괄적인 연구 결과가 밑받침되어야 실제 자연어 처리 시스템의 성능향상을 도모할 수 있다. 뿐만 아니라 자연어의 어휘는 폭넓은 변이 현상을 보이는 바, 각 어휘 사이의 관계성을 포착하기가 쉽지 않다는 점도 한계로 지적될 수 있다. 한국어에 사용

되는 모든 용언의 통사 및 의미 관계를 유형화하고 이를 위계 구조로 정리한 연구는 그 성과가 아직 뚜렷하지 않은 측면이 존재한다.

객관성: 기존 사전의 경우, 연구자의 직관의 차이에 따른 기술상의 불일치가 얼마든 존재할 수 있다. 예컨대, 연세 한국어 사전에서는 이른바 'tough' 동사군에 해당하는 '어렵다'의 논항 구조를 아래 (2)와 같이 〈NP(nom)〉, 〈NP(nom), NP(nom)〉, 그리고 〈S(nom)〉의 세 가지로 설정하고 있다.

(2) a. 언어학이 어렵다.
b. 내가 공부가 어렵다.
c. 언어학을 공부하기가 어렵다.

반면, 세종전자사전에서는 동일한 어휘 '어렵다'에 대해 총 여섯 개의 논항 구조를 설정하고 있다. 이 양자의 입장 가운데 어느 한쪽이 더 타당성을 지니는가를 밝히는 것은 이론 언어학 차원에서 중요하게 논의될만한 것이나, 보다 객관화된 방법론의 도입 역시 매우 중요한 연구 방향이라 할 수 있다.

효율성: 기존의 연구 방식은 그 구축에 따르는 시간과 비용 그리고 노력이 지나치게 많이 요구된다는 점에서 효율성에 문제가 많다. 최소한 수년 이상, 적지 않은 인력의 집중적인 투자가 있기 전에는 그 소기의 성과를 기대하기 어렵다. 실제로 앞서 언급한 Levin(1993) 및 세종전자사전의 경우만 보아도 용언 위계가 상당한 고비용 자원임을 알 수 있다.

본 연구는 앞서 언급된 한계점을 극복하기 위한 방향으로의 연구로, 기존 연구와의 차별성을 보이기 위하여 다음과 같은 방법론에 입각하고자 한다. 첫째, 구축에 소요되는 시간과 비용을 최소화하는 차원에서 전체 연구 과정을 최대한 (i) 자동화하는 것을 기본 골자로 한다. 이는 한편으로 'annotate automatically, correct manually'라는 입증된 방법론과 맥락을 함께 한다 (Marcus et al., 1993). 둘째로 실제 (ii) 언어자원을 활용하는 것을 중요한

목표로 상정한다. 따라서 본 연구의 결과가 되는 위계 구조에는 그 통사적 및 의미적 분포 특질에 대한 계량적 정보가 함께 부착될 것이다. 셋째로 단순히 용언 구조를 도출하는 것에 끝나지 않고 이를 (iii) 군집화하여 전체 용언의 위계 구조를 밝힐 것이다. 위계 구조는 HPSG의 이론적 틀에 따라 구축한다.

2.2. 연구의 구성

말뭉치에서 유의미한 언어 정보를 자동 습득해내는 연구는 일반적으로 Gamallo et al.(2001)에서 제시된 아래의 구성을 따른다.

(3) a. 분석(parsing): 해당 언어 자원을 처리하여 연구의 목적에 맞도록 태그(품사표지, 구문표지, 의미표지 등)를 부착하는 과정
b. 추출(extracting): 분석된 언어 자원에서 관심의 대상이 되는 언어 정보를 일관된 방식으로 뽑아내는 과정
c. 선별(filtering): 추출된 자료를 대상으로 하여 이 가운데 통계적으로 유의미한 것만을 선택하여 자료를 정제하는 과정
d. 군집화(clustering): 도출된 자료를 추상화 또는 유형화하여, 이를 세부 단위로 나누는 과정

본 연구의 경우 이미 분석된 자료에서 연구를 시작할 계획이므로, 위의 네 단계 가운데 첫 번째인 분석은 논외가 될 것이다. 따라서 본고의 전체 연구는 세 개의 하위 단계로 구성된다. 1단계는 추출에 해당하는 것으로 실제 언어자원에서 논항으로 파악될 수 있는 모든 후보군을 도출한다. 2단계는 선별에 해당하며, 이 도출된 후보를 대상으로 하여 통계적 모형을 활용하여 유의미한 집합을 걸러낸다. 끝으로 3단계는 확립된 논항 구조를 바탕으로 이를 군집화하여 그 관계를 위계화하는 것이다.

2.3. 연구의 자원

본 연구에 활용되는 자원은[3] 두 가지 차원에서 나누어 살필 수 있다. 첫 번째는 활용되는 목적에 따라 구분되는 것으로, (i-a) 구축을 목적으로 한 자원인가 아니면 (i-b) 평가를 위해 쓰이는 자원인가 하는 점이다. 두 번째는 자원의 성격에 따른 것으로, (ii-a) 자료적 성격을 띠는 것인가 아니면 (ii-b) 도구에 해당하는가의 구분이다. 이에 따라 본 연구에 활용되는 자원을 정리하면 아래 표와 같다.

	구축(i-a)	평가(i-b)
자료(ii-a)	세종 구문분석 말뭉치	세종전자사전
도구(ii-b)	Xavier ver. 2.0	

<표 1> 연구의 자원

먼저 본 연구의 모든 결과는 세종 구문분석 말뭉치에서 추출된 언어적 정보를 기반으로 한다. 어휘 습득 연구에서 사용되는 정보 추출의 대상이 되는 '개발용 말뭉치(development corpus)'가 필수적인데, 당연히 정밀하게 주석처리가 된 말뭉치를 활용하는 것이 보다 나은 결과를 산출할 것이다.[4] 현재 이용 가능한 한국어 구문분석 말뭉치에는 두 가지 종류가 있다. 하나는 펜실베니아 대학에서 구축한 Penn Korean Treebank(약 30만 어절 규모)이며, 다른 하나는 21세기 세종계획의 일환으로 구축된 세종 구문분석 말뭉치(약 80만 어절 규

3) 세종 구문분석 말뭉치와 세종전자사전에 대한 정보 및 자료의 입수는 아래 홈페이지 참조. http://www.sejong.or.kr

4) 물론 주석처리가 되지 않은 원시 말뭉치(raw text)를 사용하여도 원하는 결과를 일정 정도 얻을 수 있다(Manning, 1993). 그러나 이 경우에도 대체로 주어진 원시 말뭉치를 바로 활용하기 보다는 1차 분석된 결과를 바탕으로 연구를 수행하는 것이 일반적이다. 예컨대, Gamallo et al. (2001)은 품사 태깅과 부분 구문분석을 거쳐 파악된 의존 관계에서 정보 수렴을 시작하며, Erk(2007)는 BNC를 대상으로 하여, 구문분석기를 통해 처리된 결과를 바탕으로 어휘 정보를 추출하였다. 즉, 정보 추출을 위해서는 원시 말뭉치를 어떠한 형식으로든 선처리하는 과정이 요구된다. 그러나 품사부착기나 구문분석기의 성능이 완벽한 것이 아니며, 세종 구문분석 말뭉치와 같이 충분한 양의 심층 분석 자료가 존재하는 경우 굳이 원시 말뭉치를 분석하는 과정을 거칠 필요가 없다.

모)이다. 양자는 크게 세 가지 점에서 차이는 보이는데, 우선 세종 구문분석 말뭉치는 균형 말뭉치로서의 성격을 보여 다양한 장르의 텍스트를 대상으로 하였다. 반면, Penn Korean Treebank는 군사 교본과 뉴스기사로 그 대상이 한정되어 있다. 두 번째로 Penn Korean Treebank는 공범주를 그 기술의 과정에 포함시킨 반면, 세종 구문분석 말뭉치에서는 공범주 표지가 존재하지 않는다. 세 번째로 사격(oblique) 명사구[5]를 논항으로 인정할 것인가에 있어서 뚜렷한 차이가 있다. 세종 구문분석 말뭉치는 논항의 인정 범위에 대해 비교적 엄격한 반면, Penn Korean Treebank는 다양한 사격 논항을 포함하고 있다. 본 연구에서 세종 구문분석 말뭉치를 기본 자원으로 선택한 이유는 우선 규모에서 세종 구문분석 말뭉치가 더 크다는 점이다. 한국어 용언의 논항 구조를 종합적 차원에서 논의하고자 하는 본 연구의 특성상, 양의 차이는 무시할 수 없는 요소이기 때문이다. 또한 선택적인 장르 특성 역시 결과의 왜곡을 초래할 수 있는 바, 포괄적 성격의 연구에는 세종 구문분석 말뭉치가 보다 적합하다. 실제로 Roland and Jurafsky(1998)는 어떠한 특성의 자료를 일반화 말뭉치로 선택하느냐가 검출된 하위범주화의 출현 빈도에 적지 않은 영향을 준다는 사실을 입증하였는데, 한국어를 보다 더 대표할 수 있도록 균형 말뭉치를 활용하는 것이 바람직하다 하겠다.[6]

다음으로 이 세종 구문분석 말뭉치에 접근하여 원하는 정보를 취합하기 위한 도구로는 Xavier 모듈이 사용된다(Song and Jeon, 2008). Xavier 모듈은 세종 구문분석 말뭉치에서 사용자가 원하는 정보를 빠르고 단순하게 추출할 수 있도록 구성된 프로그램 패키지로서 용례 검색, 빈도 추출, 문맥자유문법 추론, 의존소 추출, 하위범주화 구성 등의 기능을 지니고 있다.

끝으로 평가를 위한 비교의 대상이 필요하다. 4절에서는 하위범주화가 얼마나 설명력 있게 구성되었는지를 판단하기 위한 내부평가의 용도로 세종전자사전이 사용될 것이다. 세종전자사전의 각 정보는 세종 구문분석 말뭉치와는 별도로 구축된 것이다. 즉, 세종 구문분석 말뭉치가 실제 텍스트에서 기초 자료를

5) 본고에서 사격 명사구는 주격 표지(NP_SBJ) 또는 목적격 표지(NP_OBJ)를 달지 않은 모든 명사구를 말한다. 세종 구문분석 말뭉치에서 사격 명사구는 통상 'NP_AJT'로 표지되어 있다.

6) 그러나 이러한 판단이 특정 말뭉치가 더 우월하다는 주장과는 무관하다. 어떠한 말뭉치도 나름의 장점과 단점을 가지기 마련이고, 일반화 말뭉치의 선택은 철저히 연구의 목적에 따라 결정되는 것이기 때문이다.

취한 반면, 세종전자사전에 망라된 정보는 말뭉치를 참조하되 기본적으로 연구자의 직관에 따라 구축된 것이다. 특히 세종전자사전은 각 용언의 격틀정보를 연구자의 수작업을 통해 망라하고 있는데, 이 정보를 통계적인 처리과정을 통해 얻어진 결과와 교차비교를 하면 전산적으로 자동 추출된 결과와 연구자의 직관에 근거 확립된 결과가 상호 어느 정도의 합치점을 보이는가를 살필 수 있다. 하향식(top-down)으로 구성된 세종전자사전의 격틀정보와 상향식(bottom-up) 방법에 따른 본고의 결과는 흥미로운 비교거리가 될 것이다.

2.4. 연구의 대상

어휘군 선정과 관련하여, 본 연구에서는 세종 구문분석 말뭉치에서 1회 이상 출현하는 용언 5,370개를 분석의 대상으로 하며, 그것은 동사, 형용사는 물론 서술성 명사까지 포함하는 것이다. 우선 일반적인 동사와 형용사는 각기 형태표지 'VV'와 'VA'를 달고 있는 어휘가 그 대상이 될 것이다. 반면에 서술성 명사의 경우에는 목록 선정이 비교적 단순하지가 않은데,[7] 본 연구에서는 아래의 두 가지 원칙에 의거하여 그에 해당되는 어휘군만을 대상 집합에 포함시켰다. 첫째로 서술성 명사에 부착되는 경동사는 '하다'만을 인정하였다. 즉, '되다', '받다', '당하다' 류의 피동형 경동사와 '스럽다' 등의 경동사가 제외된 것인데, 이들 경동사는 논항교체와 관련되기 때문이다. 교체된 논항관계를 주어진 구문분석 말뭉치에서 거꾸로 복원하는 일은 현재의 시스템에서 상당히 어렵거나 혹은 불가능하기 때문에 '하다' 이외 경동사는 논의에 포함시키지 않았다. 두 번째로 〔명사군 + 을/를 하다〕의 형태는 제외하고, 일반명사(NNG) 또는 어근명사(XR)가 한 단어 안에서 경동사 '하다'(XSV)와 결합하는 경우만 서술성 명사로 인정하였다. 이러한 판단의 근거는 실제 자료의 분포를 보면 〔명사군 + 을/를 하다〕의 구조가 〔명사군+하다〕와 완전히 동일하다고 보기 어려운 반례들이 상당수 존재하기 때문이다.[8] 실제 말뭉치에서 취한 아래 예시들을 살펴보자.

7) 기본적으로 이론적 차원에서 서술성 명사의 경계를 인정하는 기준에 아직 충분한 합의가 이루어지지 않았다고 보기 때문이다.

8) 채희락(1996)에서는 이론적 차원에서 이와 유사한 입장을 제시하고 있다. 즉, 서술성 명사가 '하다'와 한 단어로 구현된 것과 〔명사군 +을/를 하다〕가 통사적으로 결합된 것의 언어적 구조가 완전히 동일하지는 않다는 것이다.

(4) a. 모든 토끼는 이 월토와 사랑의 작업을 하고 …
b. 적 진지에 대한 공격을 빨리 하도록 …

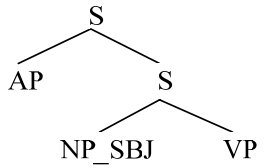
위 (4a)에서 '하다'는 명사 '작업'과 결합을 하고 있으나, '작업'은 동시에 '사랑'과 결합하여 하나의 명사구를 이루고 있다. 이때 '월토'가 '사랑'과 관련되는 항목이라면, '-와 사랑하다'의 측면에서 논항으로 간주될 수 있지만, '작업'과 관련되는 항목이라면 그것은 부가어로 간주하는 것이 더 타당하다. 즉, 여기에서 〔〔사랑의 작업〕을 하고〕 또는 〔사랑의 〔작업을 하고〕〕와 같은 괄호묶기 문제가 발생하는 것이다. 더군다나, 이 때 '작업을 하고…'가 '작업하다'와 동일한 구조를 지닌다고 가정하면, "사랑의 작업하고…"의 형태도 가능해야 할 것이나 우리의 직관이 비추어 타당치 않다. 한편, (4b)와 같이 〔명사군 + 을/를〕과 '하다' 사이에 부사어 등의 다른 단어가 삽입된 예시도 종종 관찰되는데 이 역시 '공격하다'등의 구조와 동등하다고 볼 수 없는 부분이다. 결론적으로 〔명사군+하다〕의 형태만을 경동사가 결합된 서술성 명사구로 인정하며, 〔명사군 + 을/를 하다〕에서 '하다'는 중동사의 일종으로 간주한다. 이하에서 서술성 명사는 '작업하/VV', '공격하/VV' 와 같은 〔명사+하다〕의 단어 형태로 'VV' 또는 'VA'에 포함되는 것으로 가정한다.

3. 의존소 추출

본 절에서는 우선 세종구문분석 말뭉치로부터 의존소를 추출하는 절차를 핵심 알고리즘 중심으로 소개하고, 이어서 그러는 과정에서 제기되는 문제점 및 한계를 논한 뒤에, 마지막으로 '어렵/VA-'과 '놓/VV-'을 예시로 하여 실제 추출된 자료의 한 단면을 소개하기로 한다.

3.1. 구현[9]

최초의 단계는 구문분석 말뭉치를 처리하기 위한 파스트리(Parse-Tree) 알고리즘을 구축하는 것이다.[10][11] 파스트리 자료구조는 부모 절점(Mother Node)과 좌측 및 우측 딸 절점(Left Daughter Node / Right Daughter Node)으로 구성된다. 아래 그림은 전형적인 파스트리 구조를 보여주고 있다.



<그림 1> 파스트리의 예시

최상위 절점 'S'는 좌측 딸 절점 'AP'와 우측 딸 절점 'S'를 지니며, 다시 두 번째 'S' 절점은 좌측 딸 절점 'NP_SBJ'와 우측 딸 절점 'VP'를 지닌다. 다시 말해, 모든 절점이 최상위 절점에 계층적으로 연결되는 이분지 구조가 파스트리 자료구조이며, 그 알고리즘은 아래 (5)와 같다.[12]

```
1: parse_tree(n):
2: n→left = n→right = n→parent = NIL
3: if n is not a terminal node:
4: n→right = pop()
```

9) 본 절에서 논하는 알고리즘 및 관련 문제점에 대한 상세한 논의는 Song and Choe(2008) 참고.

10) 세종 구문분석 말뭉치가 X-bar 이론에 근거, 완전한 이분지 구조로 이루어져 있음을 반영하였다.

11) 본고에서는 자료 구조상의 용어 선택과 관련하여 가급적 한국어 대역어를 사용하였다. 예컨대, 노드(Node)는 절점으로, 이진 트리(Binary Tree)는 이분지 구조로, 루트 노드(Root Node)는 최상위 절점으로 기술하였다. 이는 전산적 배경 지식이 많지 않는 독자들도 고려한 차원에서 선택한 것이다. 따라서 한 심사위원의 지적대로 전산학 분야에서 일반적으로 사용하는 용어와는 약간의 차이가 있음을 밝힌다.

12) 전체 알고리즘과 처리 모형은 Song and Jeon(2008) 및 Song and Choe(2008) 참조.

```
5: n→left = pop()
6: if n→left is NIL:
7: n→left = n→right
8: n→right = NIL
9:  n→left→parent = n→right→parent = n
10: push(n)
```

어떠한 새로운 절점 n이 출현한 경우 (1행), 우선 그 절점의 좌측, 우측, 및 부모 절점은 null 값을 할당받는다 (2행). 다음으로 그 절점이 단말 절점(어휘항)이 아니면(3행), 우측 및 좌측 절점은 스택에 저장된 값을 차례로 할당받게 된다(4,5행). 최종단 절점의 경우 좌측 딸 절점만을 가지기 때문에 이 경우(6행), 우측 딸 절점에 다시 null 값이 할당된다(7,8행). 현재의 절점은 좌측 딸 절점과 우측 딸 절점의 부모 절점으로 명시되고(9행), 이제까지 처리된 절점은 다른 절점과의 추가적인 결합을 위해 스택에 저장된다(10행).

위 알고리즘을 토대로 하여 본 연구의 1단계에서는 대상이 되는 5,370개 용언이 세종 구문분석 말뭉치에서 가지는 의존소를 모두 추출하였으며, 추출된 의존 관계의 수는 총 104,442개다. 이것은 곧 총 토큰의 수이다. 반면 전체 의존 관계 유형의 가짓수, 다시 말해 타입의 수는 103개다. 결과적으로 토큰/타입 비율은 약 1,014가 된다.

3.2. 문제점

본 연구에서는 표층에서의 실현을 최대한 존중하여 각 동사성 어휘와 의존관계를 지닌 모든 범주를 추출하는 것을 기본적인 출발점으로 삼는다. 그러나 이러한 말뭉치 기반 구축 방법론 역시 문제점이 없는 것은 아니다. 대표적으로 세종 구문분석 말뭉치 역시 (i) 논항의 인정 범위가 제한적이고 (ii) 공범주가 없다는 점에서 그 자체로 논항 관계를 완전히 보여주고 있다고는 할 수 없다.[13]

13) 이 이외에도 언어 자료의 특성상 전산적인 일괄 처리를 완전히 보장할 수 없는 현상들이 존재한다. 대표적인 것으로 장거리 의존 문제와 동음이의어의 구별을 들 수 있다. 이러한 예외 항목들에 대한 고려는 추후 연구로 미루고자 한다.

먼저 논항의 인정 범위 문제를 생각해보자. 한국어에서 논항과 부가어를 판별하는 여러 검증 도구가 제시되어 있기는 하지만(Chae, 2000; 김영희, 2004), 양자의 경계가 항상 명확한 것은 아니다. 본 연구에서는 통계적 검증 모형을 도입하여 논항과 부가어의 모호한 경계면을 해결하기 위한 방안으로 사용한다. 본고는 논항과 부가어의 구분이 범주적일 수 없다는 Choi(2010)과 기본적인 입장을 함께한다. 즉, 양자를 이분법적으로 명확히 구분하는 것은 자연언어의 특성상 불가능하며, 다만 이들의 관계를 정도의 문제로 파악하는 것이 더 타당할 수 있다는 관점을 취한다. 이러한 견지에서, 논항의 선택은 가부의 문제가 아니라 통계적으로 유의미성을 추론할 수 있는 대상으로 규정된다.

다른 한편으로 공범주의 문제가 있다. 예컨대 관계절과 피동구문은 하위범주화를 말뭉치에서 추출하는 절차에 다소 난제로 작용한다. 관계절의 경우 논항 가운데 하나가 그 절의 밖에서 실현될 수 있으며, 피동구문의 경우 논항이 교체되어 실현될 뿐만 아니라 논항의 개수가 하나 줄어드는 경우가 빈번하다. 세종 구문분석 말뭉치에서는 이들의 원형 정보에 대한 주석처리가 별도로 존재하지 않는다. 결국 현재로서는 처리의 중간 과정에서 이들을 복원할 수 있는 기제가 불분명한 까닭에 전처리를 통해 이들 구문을 제외하였다. 관계절의 경우 그 최대 절점이 VP_MOD 또는 S_MOD로 태깅되어 있기 때문에 해당 절점 이하의 구조를 무시하는 방법을 택하였다. 피동구문의 경우, 어떠한 용언이 피동 보조용언 '지다'와 결합하거나 서술성 명사가 피동형 경동사 '되다', '받다', '당하다'와 결합을 하는 경우 이들의 의존소를 추출하지 않도록 하였다. 단, 세종 말뭉치는 '이', '히', '리', '기'와 같은 피동 접사를 파생접사로 간주하기 때문에, 접미 피동사는 애초에 '먹히/VV', '뚫리/VV', '잘리/VV'와 같은 형태로 주석처리 되어 있다. 따라서 이들은 일반적인 자동사와 마찬가지로 처리하였다.

3.3. 예시: 어렵(VA)-, 놓(VV)-

추출된 의존소 목록을 살펴보면 형용사 '어렵다'의 경우 19개의 논항 관계 유형이 검출되며, 그 전체 토큰 수는 195개에 달한다. 동사 '놓다'는 마찬가지로 14개의 논항 관계 유형을 보이며, 전체 출현 빈도는 170회이다. 각각의 대표적인 실례를 살펴보면 아래와 같다. (6-7)에서 각 논항 구조 우측의 숫자는 해당

빈도 및 비율을 말한다.

(6) 어렵/VA
a. 〈VP(nom)〉 86 (44.10%)
b. 〈NP(nom)〉 51 (26.15%)
c. 〈S(nom)〉 11 (5.64%)
d. 〈VP(nom), NP(dat)〉 10 (5.13%)
e. 〈VP(nom), NP(dir)〉 7 (3.59%)
f. 〈NP(nom), NP(dat)〉 5 (2.56%)
…

(7) 놓/VV
a. 〈NP(nom), NP(acc)〉 94 (55.29%)
b. 〈NP(nom), NP(acc), NP(dat)〉 38 (22.35%)
c. 〈NP(nom), NP(acc), NP(loc)〉 9 (5.29%)
d. 〈NP(nom), NP(loc)〉 8 (4.70%)
e. 〈NP(nom), NP(acc), NP(dir)〉 7 (4.11%)
f. 〈NP(nom)〉 3 (1.76%)
…

먼저 (6)의 '어렵다'의 경우 (6a-d)의 구조는 차례로 아래와 같은 예문을 상정해 볼 수 있다.

(8) a. 언어학을 공부하기가 어렵다.
b. 언어학이 어렵다.
c. 내가 언어학을 공부하기가 어렵다.
d. 언어학이 나에게 어렵다.

직관에 비추어 이들 각각은 논항이라고 판별을 하여도 크게 무리가 없을 듯하지만, 다른 한편으로 (6e)에 해당하는 구문, 다시 말해 NP(dir)를 필수 요소

로 취하는 구문은 쉽게 찾을 수 없다는 문제점이 발생한다. 이때의 NP(dir)는 언어자원에서 부가어로 사용된 것으로 볼 수 있다.[14]

(7)의 '놓다' 예에서는 예상하는 바와 같이 NP(loc)가 동시에 출현하는 구문이 어느 정도 검색되었다. 한편으로 유의하여 볼 것은 '놓다'가 타동사임에도 불구하고 (7d)와 (7f)에 NP(acc)에 해당하는 논항이 결여되어 있다는 점이다. 이러한 점은 앞서 설명한 공범주의 문제에 해당한다.

4. 하위범주화 선별

하위범주화 정보가 구문분석의 성능 향상에 크게 도움이 된다는 점은 실제 그 동안 여러 실험을 통해서도 입증되어 왔다(Briscoe and Carroll, 1997; Carroll et al., 1998). 또한 실제의 말뭉치에서 추출한 정보로 편성된 하위범주화 정보를 활용하는 것이 단순히 기계-가독형 사전에 의존하는 구문분석 보다 좋은 성능을 보인다는 점 역시 실험을 통해 입증된 사실이다(Manning 1993). 한편으로 하위범주화 정보는 구문분석기의 종류에 구애받지 않고 거의 모든 시스템에 긍정적인 기여를 한다는 것 역시 실험으로 확인되었다. 예컨대, 통계기반 구문분석기는 물론, HPSG기반 구문분석기와 같이 특정 언어이론에 입각한 규칙기반 구문분석에서도 그 실효성은 이미 증명된 바 있다(Carroll and Fang, 2005). 다른 한편으로 언어의 유형을 막론하고 하위범주화는 실제 구문분석 시스템에 상당한 기여를 한다는 점이 증명되었는데, 대표적으로 영어(Brent, 1993; Manning, 1993; Korhonen et al., 2000), 프랑스어(Chesley and Salmon-Alt, 2006), 체코어(Sarkar and Zeman, 2000), 일본어(Utsuro et al., 1998) 등이 있다. 정리하자면 하위범주화 정보를 말뭉치에서 자동 습득하여 활용하는 것은 단기간 내에 구문분석 기의 성능 향상을 도모할 수 있는 가장 안정적인 방법론이라 할 수 있다.

하위범주화 자동 구성에 관련하여, 위에서 열거된 모든 선행 연구는 크게 두 가지 측면에 주안점을 두고 있다. 하나는 통계적 선별(statistical filtering)을 위하여 어떠한 통계식을 사용할 것인가의 문제이다. 다른 하나는 각 통계식을

14) 실제 세종 자료에 등장하는 예로는 "그것은 현실적으로 어려웠다" 등이 있다.

이용할 때 확정 범위를 결정하는 임계치(cutoff-value) 또는 유의 수준(confidence level)을 어떻게 설정할 것이냐의 문제이다.

기존 연구에서 사용된 통계식은 크게 Log Likelihood Ratio, T-score, 이항 가설 검증, 상대 빈도, Jaccard 계수 등이다. 이 가운데 거의 공통적으로 좋은 성적을 보인다고 평가 받는 것은 이항 가설 검증(Binominal Hypothesis Testing)인데, 연구에 따라 크고 작은 차이는 있으나 대략 80% 내외의 정확도(precision)을 보이는 것으로 보고되고 있다.[15] 그러나 다른 언어에서 이 이항 가설 검증을 적용하여 좋은 성능을 보였다고 해서 한국어에서도 마찬가지로 최적의 성능을 보인다는 보장은 할 수 없다. 앞서 설명한 바와 같이 한국어는 여타 언어와 구별되는 그 나름의 형태-통사적 특성을 지니기 때문이다. 실제로 이항 가설 검증을 적용하여 성공적인 결과를 거둔 사례 연구는 대개 인구어에 속한다. 반면 Tsunakawa and Kaji(2010)은 일본어를 대상으로 Jaccard 계수가 여타의 통계식 보다 좋은 결과를 낸다는 점을 실험으로 입증하였으며 이때의 정확도는 약 40%의 선으로 나왔다. Tsunakawa and Kaji(2010)은 하위범주화 자체를 대상으로 한다기보다는 기계번역의 대역어를 찾는 차원의 연구에 가깝기는 하지만, 그 기본 적용 모형이 유사하다는 점에서 시사하는 바가 있다. 특히 한국어와 형태-통사적 특성이 유사한 일본어에 적용된 결과라는 점에서 적극 고려해볼 가치가 있다.

한편 임계치에 대한 결정은 절대적인 정답이 존재하지 않으며 통상 대개의 경우 실제 실험 및 평가를 통하여 어떠한 임계치를 사용하는 것이 가장 결과를 잘 내는가를 비교한 뒤 그에 따라 선정을 하는 것이 일반적인 방법이다. 즉, 다양한 처리를 통해 경험적으로 최적의 수치가 설정된다.

이러한 점을 반영하여 본고에서는 통계식으로 이항 가설 검증과 Jaccard 계수를 사용하여 그 결과를 비교할 것이다. 임계치 설정은 각 통계식에 대하여 선행 연구에서 주로 사용한 값을 차용하여 역시 그 결과를 비교할 것이다. 비교평가는 다시 두 가지 차원에서 진행되는데, 하나는 '얼마나 정확한가'(precision)의 문제이며 다른 하나는 '얼마나 빠짐없이 도출되는가'(recall)의 문제이다.

15) 기존 연구의 성능 비교는 Sarkar and Zeman(2000), Chesley and Salmon-Alt(2006) 등을 참조.

4.1. 통계적 검증 모형

먼저 이항 가설 검증은 아래와 같은 통계식에 의거하여 계산된다(Sarkar and Zeman, 2000).

$$\sum_{i=m}^{n} \frac{n!}{i!(n-i)!} P_{\neg f}{}^{i}(1-P_{\neg f})^{n-i} \leq cutoff-value$$

위 식에서, $P_{\neg f}$는 어떠한 하위 범주화 틀이 그 동사에 사용되긴 하였지만 그것이 해당 동사의 하위범주화로 보기 어려운 경우의 확률값을 말한다. n은 어떤 동사가 말뭉치에 출현한 총 횟수이며, m은 그 동사가 해당 논항 관계로 실현된 횟수를 의미한다. 이렇게 하면 어떤 동사가 어떤 하위범주화 틀 f에 대하여 취하는 값을 구할 수 있는데, 이 값이 임계치 보다 작으면 그 하위범주화 틀 f는 해당 동사와 유의미한 상관 관계를 가지는 것으로 파악한다.

다음으로 Jaccard 계수는 보다 단순하여 다음 수식으로 연산된다(Smadja et al., 1996).

$$Jaccard(v,f) = \frac{m}{n_v + n_f - m} \geq cutoff-value$$

n_v, n_f는 각각 용언 v와 틀 f의 해당 출현 빈도를 나타내며, m은 용언 v가 틀 f가 공기하는 회수를 의미한다. 이때 계산된 값이 임계치보다 크면, 틀 f는 해당 용언 v의 유의미한 하위범주화로 간주된다.

위에서 주목해야 할 점은 임계치(cutoff-value)를 대하는 두 통계식의 관점이 서로 반대라는 것이다. 이러한 점은 위 두 식에서 부등호의 방향이 서로 반대라는 점에서 드러난다. 즉, 이항 가설 검증은 임계치보다 값이 작아야 하고, Jaccard 계수는 거꾸로 값이 커야 한다. 따라서 이항 가설 검증에서는 작은 임계치를 사용하는 것이 보다 엄격한 검증이 되는 반면, 역으로 Jaccard 계수에서는 큰 임계치를 사용하는 것이 보다 엄격한 검증이 된다.

본 연구에서는 5,370개의 용언을 대상으로 추출한 104,442개의 전체 논항

관계 및 103개의 논항 유형에 위 두 수식을 적용하여 전체 결과치를 도출하였다. 이때의 임계치는 이항 가설 검증의 경우 선행 연구에서 흔히 사용된 〔0.05, 0.025, 0.01, 0.005, 0.001〕의 다섯 개의 값을 사용하였다. 반면, Jaccard 계수는 Tsunakawa and Kaji(2010)을 참고하여 〔0.01, 0.001, 0.0001〕을 사용하였다. 결과를 검증하는 기준은 이항 가설 검증의 경우 임계치 0.05가 가장 느슨하며 임계치 0.001이 가장 엄격하다. 반대로 Jaccard 계수에서는 임계치 0.01이 가장 엄격하고 임계치 0.0001이 가장 느슨하다.

4.2. 평가

앞 소절에서 계산된 각 값에 임계치를 적용하여 104,442개의 전체 논항 관계에서 상대적으로 중요성이 떨어지는 항목을 걸러내고 나면, 다음 단계는 이들의 평가이다. 총 2개의 통계 모형에 대하여 각 5개, 3개씩의 임계치가 설정되어 있기 때문에 총 비교의 대상이 되는 집합은 8개다.

이들 8개의 집합은 세종전자사전의 격틀정보와 교차 비교의 대상이 되는데, 세종전자사전의 격틀정보 구성은 아래와 같은 형식으로 되어 있다.

(9) 어렵다
  a. X=N0-이 A
  b. Y=N1-에게는|이 X=N0-이 A

(10)놓다
  a. X=N0-이 W=N3-에게 Z=N2-을 Y=N1-을 V
  b. X=N0-이 Y=N1-을 V
  …

(6-7)의 형태로 되어 있는 도출 의존 관계를 (9-10)의 형태로 되어 있는 격틀정보에 대응시켜 보면, 두 개의 자료가 얼마나 일치를 하고 있는가를 판단할 수 있다.[16] 두 자료의 비교를 통한 평가는 precision, recall, 및 F-measure

16) 실제의 이 작업은 세종 격틀 정보를 (6-7)과 같은 형태로 변형하는 프로그램을 개발하여

의 계산을 통해 이루어진다. 아래에서 tp는 어떠한 논항 관계 f가 4.1절 에서 도출된 결과와 세종전자사전 모두에서 하위범주화로 인정되는 경우의 수를 말한다. fp는 상단에서 도출된 결과에 포함된 논항 관계가 세종전자사전에는 나타나지 않는 경우의 수이며, fn은 거꾸로 세종전자사전에 포함된 논항 관계가 위 결과에서 인정되지 않는 경우를 말한다. 끝으로 F-measure는 precision과 recall을 조합한 값으로 양자의 일치 정도를 종합적으로 살피게끔 한다.

$$precision = \frac{tp}{tp + fp}$$

$$recall = \frac{tp}{tp + fn}$$

$$F-measure = \frac{2 \times precision \times recall}{precision + recall}$$

이러한 방식에 의거 8개의 후보집합의 각 평가값을 계산하면 아래 표와 같다. 각 항목에서 가장 높은 점수를 취한 셀은 굵은 글씨로 표시하였다.

통계 모형	임계치	precision	recall	F-measure
이항 가설 검증	0.05	34.05%	63.97%	44.45%
	0.025	34.17%	57.86%	42.97%
	0.01	34.32%	49.61%	40.57%
	0.005	**34.43%**	39.22%	36.67%
	0.001	34.38%	23.41%	27.85%
Jaccard 계수	0.01	31.24%	**87.02%**	45.98%
	0.001	31.44%	86.95%	46.18%
	0.0001	32.36%	86.34%	**47.07%**

<표 2> 통계 검증 모형의 비교

우선 precision의 경우에는 이항 가설 검증의 값이 Jaccard 계수의 값보다 약간 높은 편이나 최대차이가 약 3%로서(이항 가설 검증의 임계치 0.005와

이루어 졌으며, 하단의 평가 결과 역시 컴퓨터 프로그램을 통해 자동으로 연산하였다.

Jaccard 계수의 임계치 0.01), 별다른 차이가 없다. 반면 recall의 경우에는 차이가 크게 나는데 대체로 Jaccard 계수의 값은 86% 이상의 양호한 일치도를 보이나, 이항 가설 검증의 경우에는 그렇지 못하다. 결과적으로, F-measure에서는 Jaccard 계수의 값이 이항 가설 검증의 값을 항상 상회하는 것을 확인할 수 있다. 한편으로 Jaccard 계수 안에서는 F-measure의 최대 편차가 1% 정도에 지나지 않아 큰 유의미성을 지닌다고는 볼 수 없다. 결론적으로 2단계 하위범주화 구성에서는 Jaccard 계수를 통해 도출된 결과치를 활용하고자 한다.

한편으로 위의 평가는 세 가지 측면에서 보완되어야 한다. 첫째는, 세종전자사전 역시 절대적인 기준(golden standard)가 아니기 때문에 위에서 높은 수치를 보였다고 하여 반드시 최선의 결과라고 장담할 수는 없다. 다만, 위 표는 이항 가설 검증을 통한 결과보다 Jaccard 계수를 통한 결과가 연구자의 직관에 기초하여 구축된 자원과 더 합치하는 경향성이 있다는 것을 나타낼 뿐이다. 두 번째로 Jaccard 계수 안에서 우월성이 결정되지 않았기 때문에 각 임계치에 따른 결과는 다른 방식으로 재평가가 뒤따라야 할 것이다. 이러한 점을 보완하기 위해 5절에서는 각 임계치에 준거하여 구축된 최종 결과를 놓고 그 분포적 양상에 대한 질적 평가를 시도할 것이다.

4.3. 예시: 어렵(VA)-, 놓(VV)-

3.3절에서 언급된 바와 같이, 최초 세종 구문분석 말뭉치에서 추출된 의존관계는 '어렵다'의 경우 28개, '놓다'의 경우에는 23개의 유형을 지닌다. 이들이 Jaccard 계수의 의거 어떻게 걸러지는가를 보면 아래 표3과 같다.

임계치 / 어휘	0	0.0001	0.001	0.01
어렵다	19	18	16	4
놓다	14	14	8	1

<표 3> 임계치에 따른 하위범주화 틀 개수

주목할 점은 가장 엄격한 임계치를 사용하는 0.01의 경우 논항 인정 범위가 대폭 축소된다는 것이다. 구체적으로 이들 각각은 아래와 같은 논항 구조로 정리된다. 각 논항 구조 우측의 수치는 해당 Jaccard 계수이다. (11d)의 경우가 약간 의아스러운 부분이 있으나 그 값이 임계치에서 크게 벗어나지 않음을 감안하면 아래의 결과는 비교적 타당하다고 판단된다. 특히, '놓다'의 논항으로 NP(loc)가 명시된 (12a)가 선택된 점은 흥미로운 결과라 하겠다.

(11)어렵/VA

a. 〈VP(nom)〉 0.146258503401
b. 〈VP(nom), NP(dat)〉 0.0381679389313
c. 〈S(nom)〉 0.0345911949686
d. 〈VP(nom), NP(src)〉 0.0181818181818

(12)놓/VV

a. 〈NP(nom), NP(acc), NP(loc)〉 0.0286624203822

그러나 한편으로 위의 예시는 Jaccard 계수의 단점 역시 드러내고 있다. 대다수의 통계식은 대상이 지나치게 자주 출현하거나 혹은 지나치게 드물게 출현하는 경우를 어떻게 처리하느냐에 따른 장단점을 지니게 마련인데, Jaccard 계수는 이 관점에서 약간 취약하다.[17] 그 이유는 Jaccard 계수가 해당 검증 대상의 가장 특징적인 분포를 대변해주는 기능을 수행하기 때문이다. 예를 들어, '놓다'의 경우 실제 자료에서는 〈NP(nom), NP(acc)〉 구조가 고빈도로 출현하였으나 위 (12)에서는 제외된 것을 볼 수 있는데 그 이유는 〈NP(nom), NP(acc)〉의 하위범주화 틀이 모든 동사에 걸쳐 가장 고빈도 형태이기 때문이다. 즉, 고빈도로 출현하는 하위범주화 틀에는 지나친 불이익을 주는 경향성을 Jaccard 계수는 내포하고 있다. 이러한 점을 실증적으로 보완하기 위하여, 다

17) 모든 통계적 검증은 나름의 취약점을 일정 정도 내포하기 마련이다. 예컨대, 하위범주화 틀의 자동 추출 연구에 종종 사용되어온 T-score의 경우 Jaccard 계수와 대립되는 특성을 보이는데, 저빈도 분포를 지나치게 무시하는 경향성을 나타낸다. 또한 이항 가설 검증의 경우 표2에서 드러난 바와 같이 대개 recall이 precision에 비해 상대적으로 떨어지는 경향이 있다.

음 5절은 도출된 하위범주화 틀에 후처리를 한 결과를 사용하여 위계구조를 도출하였다. 어떠한 용언의 하위범주화 틀 가운데 가장 고빈도로 출현한 것은 Jaccard 계수에 의해 임의로 걸러지지 않게끔 하여 전체 결과가 편중되지 않도록 추가적인 조정을 하였다.[18)]

5. 위계구조

지금까지의 과정에서 우리는 구문분석 말뭉치에서 의존소를 일관된 기준에 의거 추출하고, 이를 다시 통계적 모형에 의거 정제하여 유의미한 논항 구조까지 자동 구성하였다. 다음으로 5절에서는 앞서 구성된 5,370개 용언의 논항 구조를 토대로 이들 사이의 관계를 체계화하고 계층화하여 용언 위계구조를 설정한다. 여기에서 이론적 배경이 되는 문법적 틀은 HPSG이며, 구체적으로는 김종복(2004)의 방식을 기준점으로 한다. 다만, 기존 연구가 연구자의 직관을 통해 수작업으로 위계 구조를 만들고, 개별 어휘를 하나하나 분석하여 이들에 대입하는 방식이었다면, 본 연구의 방식은 대량의 자료에서 추출된 언어 정보를 바탕으로 이들을 군집화하는 방식에 속한다.

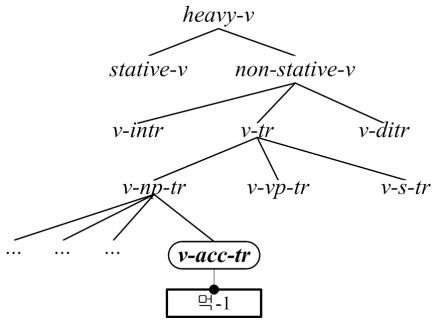
5.1. 위계구조 확립

HPSG에서는 이론적 틀 안에서는 유형자질구조(Type Feature Structure)가 중요한 기제로 상정되고 있다. 이 유형자질구조는 다시 언어적 각 유형의 종적 및 횡적 잉여성을 경감하고 언어 구조가 상호 작동하는 방식의 일반화를 포착하기 위해 계층화되어 표상되는데, 이를 유형 위계(Type Hierarchy)라 칭한다. 이 유형위계가 성립되는 중요한 원리 가운데 하나가 바로 다항상속인데, 이것은 위계상의 하나의 절점, 다시 말해 하나의 유형은 복수

18) 이러한 추가 조정은 다소 실용적인 목적에서 선택된 방법이다. 먼저 몇 가지 통계적 검증 모형을 통해 얻어진 결과에서 실제 시스템에 탑재되었을 때 가장 우수한 성능을 낼 것으로 기대되는 것을 선택한다. 다음으로 그 단점을 어떻게 보완할 수 있는가를 판단하여 결과를 후보정한다. 즉, 시스템의 성능향상을 위해 가장 실질적인 도움이 되는 방식을 적극적으로 사용하는 것이다.

의 상위 유형의 속성값을 상호 모순이 존재하지 않는 한 모두 할당 받을 수 있다는 의미다. 본 연구의 용언 위계 구조는 이러한 점에 착안하여, 기본적으로 두 가지 층위의 상위 유형의 다중상속으로 이루어진다. 하나의 층위는 (i) 범주적 계층 정보이며, 다른 하나는 (ii) 어휘적 자질 정보이다.

범주적 계층 정보는 하나의 유형이 문법범주적으로 판단하였을 때 어느 유형의 상위 유형이 되고 또 어느 유형의 하위 유형이 되는가를 말한다. 이 범주적 계층 정보는 5단계로 구성이 된다. 우선 동사(VV)와 형용사(VA)의 경우를 살펴보면 다음과 같다. (i) 용언 위계의 제일 상단에는 heavy-v가 놓이며 전체 위계 안에서는 경동사(light-v)와 자매 관계를 이루어 주동사(main-verb-lex)의 딸 절점에 놓인다. (ii) 다음 단계는 동사/형용사를 구분하기 위한 것으로 각기 non-stative-v 및 stative-v으로 명명되어 heavy-v의 딸 절점이 된다. (iii) 이들 각각은 다시 타동성의 여부에 따라 구분이 되어, 각자 {v-intr, v-tr, v-ditr} 및 {a-intr, a-tr}을 자신의 딸 절점으로 취한다. (iv) 다음 단계는 논항의 범주에 따른 구분이다. 즉, 논항이 명사구인가, 동사구인가, 문장인가에 따른 것으로, 예컨대 v-np-tr, v-vp-tr, v-s-tr 등과 같은 세부 유형이 여기에 속한다. (v) 마지막 단계로 논항의 범주가 명사구일 경우, 그 격정보를 취하여 세분화된다. 예컨대, 논항이 '밥을 먹다'와 같이 목적격으로 구성될 경우에는 v-np-tr의 하위 유형인 v-acc-tr으로 판별되고, '연필을 책상에다 놓다'와 같이 논항이 2개면서 각기 목적격과 처소격을 지니는 경우에는 v-np-np-ditr의 하위 유형인 v-acc-loc-ditr으로 처리된다. 아래의 그림은 전체 계층을 단순화하여 제시한 것으로 타동사 '먹다'의 계층상의 위치를 예시하고 있다.



<그림 2> 범주적 계층 정보

다음으로 어휘 자질 정보는 위의 각 절점이 가지는 어휘적 속성을 말하며, 각 절점의 특성을 규정해야 하기 때문에 마찬가지로 5개의 단위로 구성되어 있다. (i) 첫 번째는 lex-heavy로서 이것은 중동사의 속성값을 기술한다. (ii) 두 번째는 상태성을 명시하기 위한 lex-stative / lex-non-stative이며, (iii) 세 번째 항목은 타동성을 명시한 lex-intransitive, lex-intransitive 및 lex-ditransitive 등이다. (iv) 네 번째 항목은 논항의 범주 정보를 명시한 lex-argst-np-tr, lex-argst-np-s-ditr 등이며, (v) 마지막 항목은 각 lex-argst-acc-tr와 같이 논항의 격정보를 구분해 준다. 각각의 예시는 아래와 같다. 즉, (13)은 타동사 '먹다'의 어휘적 속성을 순차적으로 반영한다.[19]

(13)a. lex-heavy → 〔 SYNSEM.LOCAL.CAT.HC-LIGHT — 〕
b. lex-non-stative → 〔 SYNSEM.LOCAL.CAT.STATIVE — 〕
c. lex-transitive → 〔 SYNSEM.LOCAL.CAT.VAL.COMPS 〈1〉,
ARG-ST 〈 〔〕, 1〉 〕
d. lex-argst-np-tr → 〔 ARG-ST 〈 〔〕,
〔 LOCAL.CAT.HEAD.NOMINAL + 〕 〉 〕
e. lex-argst-acc-tr → 〔 ARG-ST 〈 〔〕,
〔 LOCAL.CAT.HEAD.CASE.SCASE no_scase 〕 〉 〕

19) 각 자질 구조는 김종복(2004)에서 제시된 바에 일부 기초하였다.

다시 말해, 범주적 계층 정보는 전체 위계구조의 틀을 구성하고 위계내의 각 유형의 실질적인 속성값은 어휘 자질 정보에서 부여 받는다. 예컨대, v-acc-tr은 범주 계층 정보인 v-np-tr과 어휘 계층 정보인 lex-argst-acc-tr으로부터 다중상속된 유형이다.

5.2. 평가

지금까지 도출된 전체 결과를 평가를 하여 가장 우수한 결과를 보인다고 판단되는 대상을 선택하여 보기로 하자.

우선 5,370개의 어휘소에서 표면형 어휘는 총 5,223개로 조사되었다. 어휘소의 개수와 표면형의 개수가 147개가 차이가 나는 것인데, 이러한 차이에는 두 가지 이유가 있다. 하나는 동음이의어의 처리가 이루어지지 않았기 때문이다. 예를 들어, 형용사로서의 '쓰다'(bitter)와 동사로서의 '쓰다'(write)가 실제 사전부에는 모두 '쓰-'로 등재되기 때문이다. 두 번째 이유는 세종 구문분석 말뭉치에 일반명사를 지칭하는 NNG와 어근명사를 지칭하는 XR이 하나의 표면형에 혼재되어 사용되는 경우가 있기 때문이다. 예를 들어 '가능하다'의 '가능'이 경우에 따라 일반명사로 주석되는 경우가 있는 반면 때로는 어근명사로 처리되는 경우도 존재한다. 이러한 문제점은 실제 말뭉치에 표기된 표면형을 존중하는 차원에서 별다른 전처리를 하지 않았다. 이러한 결과 어휘항 개수에 있어서 101개의 차이가 발생하였다.

두 번째로 도출된 하위범주화 틀의 계량적 분포는 Jaccard 계수식에서 사용한 임계치에 따라 아래 표와 같이 조사되었다.

임계치	0.01	0.001	0.0001
전체 하위범주화 개수	5,833	9,467	14,804
한 어휘당 평균 하위범주화 개수	1.12	1.82	2.83
표준편차	0.46	2.33	3.54

<표 4> 하위범주화 틀의 계량적 분포

위에서 나타난 바와 같이 가장 엄격한 임계치 0.01을 사용하였을 때에는 하나의 표면형 어휘가 평균적으로 하나씩의 하위범주화 틀을 가지는 것으로 보이는 반면, 느슨한 임계치를 사용한 경우에는 그 평균 개수 및 편차값이 상당히 커진다는 점을 알 수 있다. 특히, 임계치 0.0001의 경우, 표준편차값 3.54는 다소 의아스럽다. 큰 표준편차는 대부분의 용언의 하위범주화가 예측 가능하지 않다는 것을 의미할 수 있기 때문이다. 다음으로 각 임계치에 따라 선별된 용언별 하위범주화 틀의 개수를 살펴보자. 편의상 여기에서는 개수 순으로 상위 10개의 용언을 비교하기로 한다.

임계치: 0.01		임계치: 0.001		임계치: 0.0001	
어휘	개수	어휘	개수	어휘	개수
되다	7	느끼다	30	하다	58
하다	7	보이다	29	되다	54
말다	6	생각하다	28	있다	44
보다	6	알다	28	말다	39
보이다	6	있다	26	보이다	38
없다	6	되다	25	없다	36
있다	6	시작하다	25	가다	33
적다	6	말다	24	알다	32
중요하다	6	하다	24	보다	31
낮다	5	나오다	23	느끼다	30

<표 5> 용언별 하위범주화 틀의 개수

위 〈표 5〉를 통해 우리는 임계치 0.001과 0.0001을 사용하여 도출된 결과는 하위범주화의 개수가 지나치게 편중되어 있음을 짐작할 수 있다. 대표적으로 형용사 '느끼다'가 30개나 되는 하위범주화 틀을 가진다는 점은 납득하기 어려운 부분이다. 실제 추출된 '느끼다'의 하위범주화 틀 가운데 전체 누적 비율의 75% 이상을 차지하는 주요 항목은 아래와 같다. 각 괄호 안의 수치는 차례로 출현 빈도, 비율, Jaccard 계수를 나타낸다.

(14) 느끼/VV

a. **〈NP(nom), NP(acc)〉** **(114, 30.32%, 0.003196859)**
b. 〈NP(nom)〉 (58, 15.43%, 0.002733915)
c. 〈NP(nom), NP(acc), NP(dat)〉 (35, 9.31%, 0.006542056)
d. **〈NP(nom), NP(equ)〉** **(24, 6.38%, 0.023369036)**
e. 〈NP(nom), VP(acc)〉 (16, 4.26%, 0.009484292)
f. 〈NP(nom), NP(dat)〉 (15, 3.99%, 0.001062248)
h. 〈NP(nom), NP(acc), NP(loc)〉 (15, 3.99%, 0.005820722)
i. **〈NP(nom), S(acc)〉** **(14, 3.72%, 0.013220019)**

…

이 가운데, 임계치를 0.01로 설정하였을 경우 도출되는 하위범주화 틀은 굵은 글씨로 표시된 3개뿐이다. (14)를 세종전자사전에 추출한 아래의 각 유형 및 예문과 비교해 보자.

(15) a. X=N0-이 Y=N1-을 V (**=14a**)
우리 마누라는 자유를 느끼고 싶단다.
b. X=N0-이 Y=S1-고 V (**=14i**)
철호는 민서가 정상이 아니라고 느꼈다.
c. X=N0-이 Y=N2-에|에서|에게|에게서|에대해 Npr1-을 V (**=14c**)
그는 친구에게 심한 모욕감을 느꼈다.
d. X=N0-이 Y=N2-에 Npr1-을 V (**=14h**)
나는 갑자기 옆구리에 통증을 느꼈다.
e. X=N0-이 Y=N1-을 ADV V (**≈14i**)
철수는 애인을 가깝게 느끼면서도 …

첫 번째와 두 번째 격틀 (15a-b)는 임계치 0.01을 사용하여 추출된 하위범주화 틀과 완전한 일치를 보이고 있다. 이러한 일치는 평가에서 precision 값에 긍정적인 영향을 주는 요소이다. 그러나 (15c-d)의 격틀은 임계치 0,001 및

0.0001을 사용한 결과에는 포함되나 임계치 0.01을 사용한 결과에서는 배제되었다. 즉, 평가의 recall 값에 부정적인 영향을 주게 되는 것이다. 끝으로 (15e)는 완벽히 일치하는 것은 아니나, (14i)의 하위범주화와 부분 일치를 보이는 항목이다. (15e)같은 예는 precision 및 recall 각각의 하락을 가져오는 요소이지만, 완전히 잘못된 도출이라고 판단할 수는 없는 것이다. 여기서 precision과 recall 가운데 우선시되어야할 항목이 무엇이냐의 문제가 제기되는데, Sarkar and Zeman(2000), Chesley and Salmon-Alt(2006), Tsunakawa and Kaji(2010) 등의 선행연구는 공통적으로 precision에 더 비중을 두고 있다. 이는 경험적인 이유에 근거하는데, 실제 결과가 자연어처리에 활용될 때 발생할 수 있는 문제점을 최소화하도록 하는 장치이다. 이러한 측면에서 표5 및 (14)의 수치를 다시 고려하면, 임계치 0.01을 사용하는 것이 과잉 일반화의 오류를 피할 수 있는 선택이다.

다음으로 각각의 임계치에 따른 결과를 바탕으로 자동 구성된 위계구조의 분포를 살펴보자.

임계치	0.01	0.001	0.0001
전체 용언 유형 개수	84	96	98
한 유형당 평균 어휘수	69.44	98.61	151.06
표준편차	256.12	250.22	336.58

<표 6> 위계구조의 분포

더 느슨한 임계치를 사용할수록 용언의 유형의 개수가 늘어남을 알 수 있는데 그 이유는 그만큼 다양한 종류의 하위범주화 틀을 포괄하고 있기 때문이다.

끝으로 각 임계치에 따른 결과의 주요 유형을 살펴보도록 한다.

0.01			0.001			0.0001		
유형	비율	누적	유형	비율	누적	유형	비율	누적
v-acc-tr	34.06%	34.06%	v-acc-tr	21.17%	21.17%	v-acc-tr	14.21%	14.21%
v-intr	17.32%	51.38%	v-intr	11.23%	32.40%	v-intr	9.56%	23.76%
a-intr	11.14%	62.52%	a-intr	6.90%	39.29%	v-dir-tr	7.56%	31.32%
v-dat-tr	9.41%	71.94%	v-dat-tr	6.41%	45.71%	v-acc-dir-ditr	7.06%	38.38%
v-dir-tr	4.87%	76.80%	v-dir-tr	4.25%	49.95%	v-dat-tr	6.61%	44.99%
v-acc-dat-ditr	2.31%	79.12%	v-v-tr	3.75%	53.70%	v-acc-dat-ditr	6.56%	51.55%
v-acc-dir-ditr	2.19%	81.31%	v-acc-dir-ditr	2.98%	56.68%	v-src-tr	4.71%	56.26%
v-src-tr	1.92%	83.23%	v-s-tr	2.90%	59.59%	v-acc-src-ditr	4.54%	60.80%
a-dat-tr	1.03%	84.26%	v-equ-tr	2.86%	62.45%	a-intr	4.49%	65.29%
v-com-tr	1.01%	85.27%	v-acc-dir-ditr	2.84%	65.29%	v-v-tr	4.19%	69.48%

<표 7> 임계치별 주요 유형 비교

위 표에서 상위 2개의 비율을 점하는 유형은 모두 v-acc-tr 및 v-intr로서 동일하다. 다만 그 비율의 크기가 서로 상이한데, 가장 엄격한 임계치를 사용하는 좌측 0.01 항목에서는 그 누적 비율이 50%를 상회하는 반면, 가장 느슨한 임계치를 사용하는 우측 0.0001에서는 25%에도 채 미치지 못한다. 기본형인 자동사 혹은 목적격을 취하는 기본형 타동사가 전체 용언의 절반 가까이를 차지한다는 것이 화자의 직관에도 부합된다고 본다. 또한 일반적인 형용사인 a-intr을 포함시킬 경우 임계치 0.01은 전체 용언의 60%이상을 포괄하고 있어 자연스러워 보인다. 반면, 임계치 0.001은 상위 3개의 누적비율이 40%에 미치지 못하며, 임계치 0.0001에서는 a-intr이 전체의 5%에도 미치지 못하는 점을 관찰할 수 있다. 따라서, 위에서 결국 임계치 0.01을 사용하는 결과가 가장 좋은 결과를 보인다고 판단해 볼 수 있다. 이러한 여러 측면을 종합적으로 고려하여 본 연구에서는 0.01을 잠정적인 임계치로 설정하였다.

결과적으로 도출된 전체 위계구조는 부록 1과 같다. 부록 1에서 각 유형 우측의 숫자는 해당 유형에 속한 어휘의 수를 말한다.

5.3. 예시: 어렵(VA)-, 놓(VV)-

임계치 0.01을 적용한 용언 위계구조를 결과로 택하였으므로, 두 가지 예시 '어렵다'와 '놓다'가 각기 어떻게 최종 구현되어 있는지 살펴보도록 하자.

먼저 '어렵다'의 경우 (11)에서 논의된 논항 구조와 동일한 세부 유형을 지닌다. 이때, (16b)와 (16d)의 경우 주어의 범주 정보 vp가 명시되지 않았으나, 주어의 경우에는 해당 용언이 타동성을 지니는 경우 범주제약을 미명세 상태로 남겨두므로 문제가 되지 않는다.

(16)어렵/VA

a. 어렵-1a-vp-intr 〈VP(nom)〉
b. 어렵-2a-dat-tr 〈VP(nom), NP(dat)〉
c. 어렵-3a-s-intr 〈S(nom)〉
d. 어렵-4a-src-tr 〈VP(nom), NP(src)〉

다음으로 '놓다'의 경우에는 (12)에서 살핀 하위범주화 틀 이외에 가장 잦은 빈도로 출현한 하위범주화가 처리에 포함되어 아래와 같은 유형을 취한다.

(17)놓/VV

a. 놓-1 v-acc-tr 〈NP(nom), NP(acc)〉
b. 놓-2 v-acc-loc-ditr 〈NP(nom),NP(acc), NP(loc)〉

6. 결론

지금까지 세종 구문분석 말뭉치를 기반으로 하여 한국어 용언의 하위범주화 틀과 그 위계구조를 자동 구축하는 과정을 제시하였다. 첫 번째 단계는 의존소 추출로서 말뭉치에서 의존소를 추출하는 구체적인 알고리즘을 제시하고, 관련된 몇 가지 이슈에 대해 검토하였다. 하위범주화 틀을 기술하는 데 논항/부가어의 구분과 공범주 출현 가능성을 반영하기 위해 두 번째 단계에서는 통계적인

검증 모형을 도입하였다. 이항 가설 검증과 Jaccard 계수가 활용되었으며, 그 결과 Jaccard 계수를 통한 결과값이 주어진 자료에 더 잘 맞는다는 사실을 확인하였다. 그러나 각 임계치에 따른 하부 결과 가운데 어느 것이 더 우월한지는 판단하지 못하였으며, 그 점은 마지막 단계에서 재검토되었다. 최종 단계에서는 HPSG의 이론적 틀에 준거하여 주어진 하위범주화의 군집화를 통해 위계구조를 도출하였고 그 구체적 결과를 평가하였다. 각각의 분포적 성향 가운데 타당성이 높다고 판단되는 것은 가장 엄밀한 임계치인 0.01을 사용한 결과였다.

6.1. 연구의 의의

이상의 연구의 가장 큰 의의는 무엇보다 한국어의 언어자원을 구축하는 방법론을 설정하는 데 있어서 기존의 연구와 차별성을 보인다는 점이다. 구체적으로는 아래 표와 같다.

	기존 연구	**본 연구**
기술방식	수동	자동
배경	언어이론	자료기반
판단의 근거	언어직관	언어자료(구문분석 말뭉치)
처리의 모형	심리언어적 기술	전산적/통계적 처리
도출 방식	분류(classification)	군집화(clustering)
	하향식(top-down)	상향식(bottom-up)

<표 8> 구축 방법론 비교

물론 본 연구의 방식이 기존 연구의 방식에 비해 절대적으로 우수하다는 입장은 아니다. 다만, 기존 연구의 방식을 보완할 수 있는 다른 각도의 접근법을 제시하였다는 점에서 의의가 있을 것이다.

두 번째로는 현재까지 구축되어온 한국어 언어 자원을 적극 활용하였다는 점을 본 연구의 의의로 들 수 있다. 특히, 10여년간 정부 주도로 구축되어 일반에

게 공개된 세종 말뭉치와 전자사전을 연구의 중심적인 자료로 활용하였다는 점에서 기존 대부분 연구와 차별화된다.

세 번째로 말뭉치를 활용한 위계구조 설정은 최근의 연구 추세에 발맞춘 방법론이다. 어휘의 자동 습득에 관한 최근의 연구 경향은 단순히 언어 자원에서 특정 언어 정보를 뽑아내는 것에 그치지 않고 이를 재가공하여 보다 넓은 범위의 활용가능성을 지니는 자원을 생성하는 것이다(Dorr and Jones, 1996; Gamallo et al., 2001; Korhonen et al., 2003). 아울러 HPSG기반의 시스템 구현 연구의 측면에서도 언어 자원의 활용은 단기간에 우수한 성과를 낼 수 있는 방법론으로 여러 차례 검토된 바 있으며, 따라서 위 둘째 이유와 마찬가지로 최근의 추세에 맞물린 연구 흐름이라 할 수 있다. 대표적으로 Cramer and Zhang(2010)에서는 독일어 구문분석 말뭉치인 Tiger Treebank에서 문법을 자동 도출하여 HPSG/MRS기반 독일어 문법인 Cheetah를 제시하였다. Miyao and Tsujii(2008)은 Penn English Treebank에서 HPSG 문법을 반자동 도출한 확률기반 HPSG 분석기 Enju를 제시하였으며, 마찬가지의 방법론으로 Yu et al.(2010)에서는 중국어 HPSG문법을 구성하였다. 즉, HPSG기반의 시스템을 구현하는 데 있어서 실증적인 언어 자원을 활용하는 것은 이미 검증된 방법론에 속한다.

네 번째로는 언어학적 연구를 수행하는 데, 통계를 활용한 계량적 모형을 적극적으로 도입하였다는데 의의가 있다. 본 연구에서 사용된 통계적 검증 모형들은 비단 하위범주화 틀을 도출하는 데에만 국한된 것이 아니기 때문에 다른 언어 현상의 분포적 특성을 살피는 데 있어서도 충분히 활용될 여지가 있다. 계량적 검토를 통해 언어의 분포적 특질을 밝히는 일은 이론언어학적 접근에 실증적 증거를 제시하는 일이 될 것이다.

끝으로 실제 시스템의 성능을 향상시킬 수 있는 구체적인 방안에 대한 조사와 모형 개발이 수반되었다는 점을 들 수 있다. 즉, 단순히 이론적 차원의 결과 제시에 그치지 않고, 이 결과가 실제 시스템에 어떻게 탑재될 수 있는가를 연구의 핵심 방향으로 설정하였다. 이러한 연구는 한국어자원문법을 비롯한 실제 시스템의 개발에 중요한 토대가 될 것이다.

6.2. 향후 과제

향후 과제는 크게 세 가지 차원에서 고려될 수 있다. 먼저 본 연구에서는 용언의 통사적 특성만을 살피었으나 향후 연구에서는 그에 더하여 어휘 의미적 특성까지 함께 파악되어야 할 것이다. 이러한 판단은 비슷한 통사적 속성을 지니는 범주는 비슷한 의미적 성향을 보인다는 관점에 따른 것이다(송상헌 외, 2008; 송상헌·최재웅, 2010). 용언 자체의 어휘 의미적 속성이 통사적 환경에 어떠한 영향을 주고 받는지에 대한 연구와 함께(Dorr and Jones, 1996; Korhonen et al., 2003), 하위범주화의 각 논항들이 해당 용언과 가지는 의미적 선택 관계가 어떠한 양상으로 존재하는가에 대해서도 살필 수 있을 것이다(Gamallo et al., 2001). 두 번째로는 한국어 단일언어자원을 뛰어 넘어서, 병렬 구문분석 말뭉치를 대상으로 한 연구 역시 흥미로운 연구가 될 것이다. 즉, 한국어 용언의 하위범주화 틀이 영어 또는 일본어와 같은 언어에서는 어떠한 구조로 구현되는가에 대한 계량적 연구는 기계번역을 비롯한 실제 시스템 개발에 중요하게 쓰일 수 있다(Haugereid and Bond, 2011). 끝으로, 현재까지 제시된 각 결과를 실제 한국어자원문법을 비롯한 HPSG 전산 문법에 적용하여 그 성능향상의 정도를 실험하여야 한다(Song et al. 2010). 또한 언어이론에 독립적으로 통계기반 구문분석기의 성능에도 긍정적인 기여를 할 수 있는지의 여부 역시 추후 검토의 대상이 될 것이다.

부록 1: 용언 위계 구조 및 빈도

heavy-v 5833
- stative-v 976
  - a-intr 650
  - a-tr 254
    - a-np-tr 223
      - a-abl-tr 1
      - a-acc-tr 30
      - a-as-tr 5
      - a-comp-tr 28
      - a-com-tr 16
      - a-con-tr 1
      - a-dat-tr 60
      - a-dir-tr 29
      - a-equ-tr 25
      - a-loc-tr 1
      - a-nom-tr 17
      - a-src-tr 10
    - a-s-tr 12
    - a-v-tr 19
  - a-ditr 72
    - a-np-np-ditr 36
      - a-acc-dat-ditr 1
      - a-acc-dir-ditr 1
      - a-acc-src-ditr 2
      - a-nom-as-ditr 2
      - a-nom-com-ditr 6
      - a-nom-comp-ditr 15
      - a-nom-dat-ditr 1
      - a-nom-dir-ditr 2
      - a-nom-equ-ditr 6
    - a-s-np-ditr 13
      - a-s-abl-ditr 1
      - a-s-as-ditr 1
      - a-s-com-ditr 2
      - a-s-comp-ditr 1
      - a-s-dat-ditr 5
      - a-s-equ-ditr 2
      - a-s-src-ditr 1
    - a-v-np-ditr 23
      - a-v-as-ditr 1
      - a-v-com-ditr 2
      - a-v-comp-ditr 2
      - a-v-dat-ditr 9
      - a-v-dir-ditr 6
      - a-v-src-ditr 3

- non-stative-v 4857
  - v-intr 1010
  - v-tr 3231
    - v-np-tr 3124
      - v-abl-tr 13
      - v-acc-tr 1987
      - v-as-tr 12
      - v-comp-tr 12
      - v-com-tr 59
      - v-con-tr 2
      - v-dat-tr 549
      - v-dir-tr 284
      - v-equ-tr 31
      - v-inst-tr 8
      - v-loc-tr 22
      - v-nom-tr 33
      - v-src-tr 112
    - v-s-tr 59
    - v-v-tr 48
  - v-ditr 616
    - v-np-np-ditr 405
      - v-acc-abl-ditr 7
      - v-acc-as-ditr 8
      - v-acc-com-ditr 7
      - v-acc-comp-ditr 7
      - v-acc-dat-ditr 135
      - v-acc-dir-ditr 128
      - v-acc-equ-ditr 8
      - v-acc-inst-ditr 10
      - v-acc-loc-ditr 16
      - v-acc-src-ditr 48
      - v-nom-abl-ditr 4
      - v-nom-com-ditr 6
      - v-nom-comp-ditr 4
      - v-nom-dat-ditr 6
      - v-nom-dir-ditr 4
      - v-nom-equ-ditr 7
    - v-s-np-ditr 91
      - v-s-as-ditr 3
      - v-s-com-ditr 2
      - v-s-comp-ditr 1
      - v-s-dat-ditr 26
      - v-s-dir-ditr 26
      - v-s-equ-ditr 2
      - v-s-src-ditr 31
    - v-v-np-ditr 120
      - v-v-abl-ditr 1
      - v-v-com-ditr 4
      - v-v-comp-ditr 3
      - v-v-dat-ditr 39
      - v-v-dir-ditr 33
      - v-v-equ-ditr 10
      - v-v-loc-ditr 1
      - v-v-src-ditr 29

참고문헌

김영희 (2004) 논항의 판별 기준. 『한글』 266: 139-167.

김종복 (2004) 『한국어 구구조 문법』 한국문화사.

송상헌·전지은·최재웅 (2008) 영어 형용사+전치사구 구문의 의미적 제약 - ICE-GB와 WordNet을 활용한 통계적 검증 -. 『언어와 언어학』 41: 75-103.

송상헌·최재웅 (2010) 영어 동사의 의미적 유사도와 논항 선택 사이의 연관성: ICE-GB와 WordNet을 이용한 통계적 검증. 『언어와 정보』 14(1): 113-144.

채희락 (1996) "하-"의 특성과 경술어 구문. 『어학연구』 32: 409-476.

Brent, M. R. (1993) "From Grammar to Lexicon: Unsupervised Learning of Lexical Syntax". *Computational Linguistics* 19: 243-262.

Briscoe, T. & J. Carroll (1997) "Automatic Extraction of Subcategorization from Corpora". *Proceedings of the 5th Conference on Applied Natural Language*. Washington, DC.

Carroll, J., M. Guido, & T. Briscoe (1998) "Can Subcategorisation Probabilities Help a Statistical Parser?" *Proceedings of the 6th ACL/SIGDAT Workshop on Very Large Corpora*. Montreal, Canada.

Carroll, J. & A. C. Fang (2005) "The Automatic Acquisition of Verb Subcategorisations and Their Impact on the Performance of an HPSG Parser". *Lecture Notes in Computer Science* 3248: 646-654.

Chae, H. (2000) "Complements vs. Adjuncts (in Korean)". *Studies in Modern Grammar* 19: 69-85.

Chesley, P. & S. Salmon-Alt (2006) "Automatic Extraction of Subcategorization Frames for French". *Proceedings of the Language Resources and Evaluation Conference (LREC)*.

Genua, Italy.

Choi, H. (2010) "The Distinction of Argument and Adjunct as a Gradient Notion". *Language and Information* 14: 25-48.

Cramer, B. & Z. Yi (2010) "Constraining Robust Constructions for Broad-Coverage Parsing with Precision Grammars". *Proceedings of the 23rd International Conference on Computational Linguistics*. Beijing, China.

Dorr, B. J. & J. Doug (1996) "Role of Word Sense Disambiguation in Lexical Acquisition: Predicting Semantics from Syntactic Cues". *Proceedings of the 16th conference on Computational Linguistics.* Copenhagen, Denmark.

Erk, K. (2007) "A Simple, Similarity-based Model for Selectional Preferences". *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics.* Prague, Czech Republic.

Gamallo, P., A. Agustini, & G. P. Lopes (2001) "Selection Restrictions Acquisition from Corpora". *Lecture Notes in Computer Science* 2258: 67-75.

Haugereid, P. & F. Bond (2011) "Extracting Transfer Rules for Multiword Expressions from Parallel Corpora". *Proceedings of the Workshop on Multiword Expressions: from Parsing and Generation to the Real World (MWE 2011).* Portland, Oregon.

Korhonen, A., G. Gorrell & D. McCarthy (2000) "Statistical Filtering and Subcategorization Frame Acquisition". *Proceedings of the 2000 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*. Hong Kong.

Korhonen, A., Y. Krymolowski & Z. Marx (2003) "Clustering Polysemic Subcategorization Frame Distributions

Semantically" *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*. Sapporo, Japan.

Levin, B. (1993) *English Verb Classes and Alternations: a Preliminary Investigation*. University Of Chicago Press.

Manning, C. D. (1993) "Automatic Acquisition of a Large Subcategorization Dictionary from Corpora". *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*. Columbus, Ohio.

Marcus, M. P., M. A. Marcinkiewicz, & B. Santorini (1993) "Building a Large Annotated Corpus of English: the Penn Treebank". *Journal of Computational Linguistics* 19: 313-330.

Miyao, Y. & J. Tsujii (2008) "Feature Forest Models for Probabilistic HPSG Parsing". *Computational Linguistics* 34(1): 35-80.

Roland, D. & D. Jurafsky (1998) "How Verb Subcategorization Frequencies are Affected by Corpus Choice". *Proceedings of the 17th International Conference on Computational Linguistics*. Morristown, NJ, USA.

Sag, I. A., T. Wasow, & E. M. Bender (2003) *Syntactic Theory: A Formal Introduction*. CSLI Publications.

Sarkar, A. & D. Zeman (2000) "Automatic Extraction of Subcategorization Frames for Czech" *Proceedings of the 18th Conference on Computational Linguistics*. Saarbrücken, Germany.

Song, S. & J. Choe (2008) "Automatic Construction of Korean Verbal Type Hierarchy using Treebank" *Proceedings of the 15th International Conference on Head-Driven Phrase Structure Grammar*. Keihanna, Japan.

Song, S. & J. Jeon (2008). "The Xavier Module - Information

Processing of Treebanks" *Proceedings of the International Conference of Cognitive Science 2008*. Seoul, Korea.

Song, S., J. Kim, F. Bond, and J. Yang (2010) "Development of the Korean Resource Grammar: Towards Grammar Customization" *Proceedings of the 8th Workshop on Asian Language Resources*. Beijing, China.

Smadja, F., K. R. McKeown, & V. Hatzivassiloglou (1996) "Translating Collocations for Bilingual Lexicons: a Statistical Approach". *Computational Linguistics* 22: 3-38.

Tsunakawa, T. & H. Kaji (2010) "Augmenting a Bilingual Lexicon with Information for Word Translation Disambiguation". *Proceedings of the 8th Workshop on Asian Language Resources*. Beijing, China.

Utsuro, T., T. Miyata, & Y. Matsumoto (1998) "General-to-Specific Model Selection for Subcategorization Preference". *Proceedings of the 17th International Conference on Computational Linguistics*. Morristown, NJ, USA.

Yu, K., Y. Miyao, X. Wang, T. Matsuzaki, & J. Tsujii (2010) "Semi-Automatically Developing Chinese HPSG Grammar from the Penn Chinese Treebank for Deep Parsing". *Proceedings of the 23rd International Conference on Computational Linguistics*, Beijing, China.

[136-701] 서울특별시 성북구 안암동 고려대학교 문과대학 언어학과
E-mail: sanghoun@gmail.com / jchoe@korea.ac.kr

논문접수: 2011년 12월 30일
수정완료: 2012년 2월 16일
게재확정: 2012년 2월 17일