세종전자사전을 이용한 한국어 부사격의 의미역 결정

신명철[O] 이용훈 김미영 정유진 이종혁
포항공과대학교
{mcshin[O], yhlee95, colorful, prizer, jhlee}@postech.ac.kr

Semantic Role Assignment for Korean Adverbial Case Using Sejong Electronic Dictionary

Myung-Chul Shin[O] Yong-Hun Lee Mi-Young Kim You-Jin Chung Jong-Hyeok Lee
Pohang University of Science and Technology

요 약

세종전자사전의 용언사전과 체언사전에 기재된 용언의 격틀과 명사의 의미부류는 문장의 의미분석을 위한 핵심적인 언어자원이다. 본 논문에서는 용언사전을 전산처리가 용이한 격틀사전으로 변형한 다음 이를 이용한 의미역 결정 시스템을 구축하였고 기계학습 방법에 기반한 의미역 결정 시스템과 혼합하여 한국어에 있어 '에, 로'를 격표지로 하는 부사격에 대한 의미역 결정 방법에 대해 다루고 있다.

1. 서 론

의미 분석의 목적은 자연언어 문장의 의미적 구조를 분석하여 문장 내의 단어 의미와 단어간 의미관계로 이루어진 의미 표현(Semantic representation)을 얻는 데 있다고 할 수 있다. 이러한 과정에서 문제가 되는 것은 단어의 다의성(Polysemy)과 구문관계의 심층격 사상(Mapping)에서의 애매성이다. 일반적으로 문장의 의미분석 과정은 앞의 두 가지 문제를 해결하는 단어 의미 중의성 해소(WSD : Word Sense Disambiguation) 단계와 논항의 의미역 결정(SRA : Semantic Role Assignment) 단계를 거쳐 이루어 진다.

본 연구에서는 이러한 의미 분석 과정 중 의미역 결정에 관해 다루고자 한다. 의미역 결정이란 일반적으로 서술어-논항 관계에 적합한 의미 관계(Semantic Relation)를 정해주는 과정이라 할 수 있으며 이러한 의미 관계는 전통적으로 심층격(Deep Case), 격 관계(Case Role), 의미역(Thematic Role) 등으로 불려져 왔다[1].

자연언어처리에 있어서 의미역 결정은 기계 번역(MT), 정보 추출(IE), 질의 응답 시스템(QA)의 성능 및 질 향상에 중요한 역할을 하기 때문에 최근 들어 정확하고 견고한 의미역 결정 방법론에 대한 필요성이 증가되고 있는 추세이다.

한국어의 경우 의미역 결정에 관한 다양한 연구[2,3,4,5,6]가 있었고 [2]은 부사격 조사에 대한 의미역 결정 문제를 중점적으로 다루었다. 한국어의 격조사는 구문 관계를 나타내는 격표지(Case Marker)라 할 수 있으며 하나의 격조사는 여러 가지 다양한 의미격을 표상할 수 있다. 특히 부사격 조사는 가장 많은 의미격을 나타낼 수 있어서 논항의 의미역 결정에 있어 심각한 문제를 드러내고 있다[2,5,6]. 따라서 본 연구는 한국어 부사격 조사 특히 애매성(Ambiguity)이 가장 큰 '에,로'를 격표지로 가진 부사격의 의미역을 결정하는 것을 목표로 한다.

이를 위해 본 연구에서는 다음과 같은 혼합 접근법(Hybrid approach)을 사용하였다. 첫째, 언어자원인 세종전자사전의 용언사전과 체언사전을 이용하여 전산처리가 용이한 격틀사전을 구축하고 이를 이용해 부사격의 의미역을 먼저 결정한다. 이는 일종의 언어학적 규칙에 의한 방법이라고 할 수 있다. 둘째, 앞에서 결정되지 못한 부사격의 의미역을 용언사전의 예문을 이용해 구축된 SVM(Support Vector Machine) 모델을 통한 통계적 방법으로 결정한다.

실험 결과 격틀사전과 SVM 모델을 이용한 혼합시스템의 성능이 어느 하나를 단독으로 사용하는 것보다 월등한 성능의 우위를 보이는 것을 보였다.

2. 관련 연구

의미역 결정의 방법론은 크게 격틀사전에 기반한 방법(Case frame-based)[7,8,9]과 말뭉치에 기반한 방법(Corpus-based)[2,3,4,1,10,11]으로 분류될 수 있다. 먼저 격틀사전에 기반한 방법에 대해 살펴보자. 이 방법에서는 구문 분석된 입력문장의 용언과 표층격(Surface case)에 가장 잘 부합하는 격틀을 선택하는 것으로 각 표층격에 대한 심층격 즉, 의미격을 얻을 수 있다. 여기에서 적합한 격틀을 선택하는 방법으로는 다음과 같이 두 가지

접근법이 있다.

첫째, 격틀사전에 수록된 선택제약 정보 중 의미소(Semantic primitive)를 사용하는 방법이다. 이 방법에서 입력문장과 격틀의 유사도는 입력문장의 표층격과 그것에 대응되는 격틀의 요소 사이의 유사도 총합으로 결정되는데, 유사도의 계산에 의미소간 유사도를 이용한다. 의미소의 입도(Granularity)가 높지 않은 의미소 체계를 사용한다면 큰 효과를 보기는 어렵다.

둘째, 선택제약 정보로 각 심층격 요소가 될 수 있는 예제(Examples)을 사용하는 방법이다. 이 방법도 앞에서 언급한 의미소에 기반한 방법과 비슷하지만 차이점이 있다면 유사도의 계산시 시소러스를 사용한다는 점과 격틀에 예제를 추가하여 쉽게 시스템을 확장할 수 있다는 점이다.

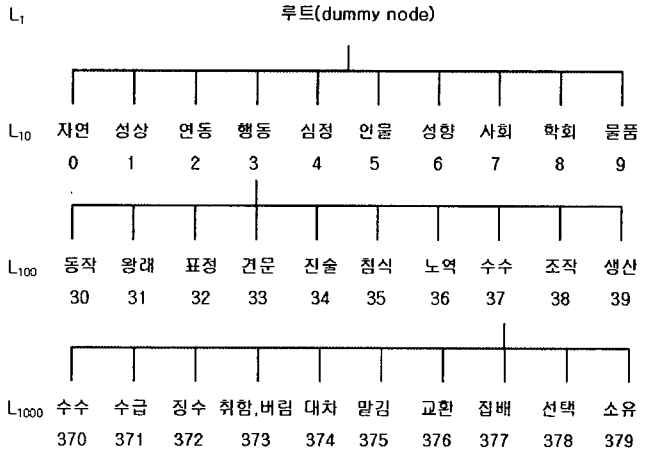
두 방법 모두 선택제약에 기반한 방법으로서 의미의 변별 능력에 있어서는 거의 유사하다고 본다. 격틀사전에 기반한 방법의 장점은 빠른 처리 속도에 있지만 단점으로는 대규모의 격틀사전과 의미소 체계와 같은 고비용의 언어자원을 필요로 한다는 점과 낮은 적용률(Cove-rage)라고 볼 수 있다.

다음으로 말뭉치에 기반한 방법은 말뭉치의 각 문장 마다 논항에 적합한 의미역을 부착하여 의미역 태깅된 말뭉치를 구축하고 통계적 혹은 기계학습 방법을 사용하여 의미역을 결정하는 방법론이라 할 수 있다. 통계적 혹은 기계학습의 관점에서 의미역 결정의 문제는 의미역이 부착된 말뭉치를 의미역 결정에 도움이 되는 자질로 표현된 학습 데이터로 변환한 후 모델을 학습하고 학습된 모델을 이용하여 새로운 데이터의 의미역을 결정하는 문제 즉 다중 분류 문제(Multi-class classification)로 변환될 수 있다. 이 방법의 장점은 격틀사전을 이용하는 방법에서 사용되지 않는 여러 가지 자질 정보를 사용할 수 있다는 점과 높은 적용률 그리고 견고성(Robustness)에 있다. 단점이라면 의미역 태깅된 말뭉치를 구축하는 데 많은 시간과 노력이 필요하고 구축된 모델을 통해 결정된 결과에 대한 해석이 불명확하다는 점이다. 하지만 최근 들어 영어권에서는 FrameNet이나 PropBank와 같이 의미역이 부착된 대규모의 말뭉치를 이용한 통계적 방법의 연구가 다수 진행되고 있다[10,11].

3. 연구 범위

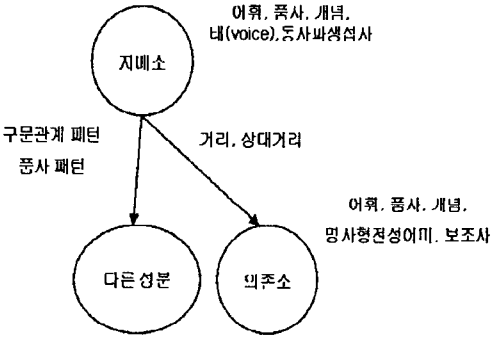
본 연구에서는 문장의 구분분석과 단어 의미 중의성 해소 결과를 입력으로 받아서 문장 내에 있는 부사격의 의미역을 결정하는 시스템의 구축을 목적으로 한다. 즉, 원시 문장은 형태소분석기, 구문분석기, 단어의미중의성 해소 모듈을 통해 단어의미중의성이 해소된 구문 트리 형태로 바뀌고 이것이 의미역 결정 시스템의 입력으로 사용된다. 형태소분석기, 구문분석기, 단어의미중의성 해소 모듈은 포항공대 지식 및 언어공학 연구실에서 자체 개발한 시스템을 이용하였다.

단어의미중의성 모듈은 입력된 단어를 가도카와 시소러스 상의 개념으로 사상한다. 가도카와 시소러스는 [그림 1]과 같이 총 1,110 개의 개념과 4 단계의 계층 구조를 가지고 있으며 L_1, L_{10}, L_{100} 레벨에 속해 있는 개념들은 각각 10개의 하위 개념들로 나뉜다.



[그림 1] 가도카와 시소러스의 계층 구조

이와 같은 시스템을 통해서 얻어진 구문 트리로부터 얻을 수 있는 자질 정보는 아래 [그림 2]와 같다.



[그림 2] 구문 트리와 자질 정보

의미역 결정 시스템은 [그림 2]에서 보는 바와 같이 부사격 관계인 지배소와 의존소 관계를 적합한 의미격에 사상한다.

4. 격틀사전을 이용한 의미역 결정

세종전자사전은 대규모의 한국어 어휘를 수집하고 이를 체제적으로 분석하여 한국어 관련 언어 정보의 자동 처리에 필수적이면서도, 보편적으로 활용될 수 있는 기반 전자사전의 개발을 목표로 구축된 전자사전이다[15]. 현재 계속적으로 개발 중에 있으며 핵심 전자사전인 체언사전은 24,137개, 용언사전은 18,195개의 표제어가 수록되어 있다.

4.1 용언사전으로부터 격틀사전 구축

세종 용언사전에는 표제어에 대한 다양한 통사적, 의미적 정보가 XML 형태로 수록되어 있지만 이를 활용하기

위해서는 목적에 맞는 정보 항목을 선별하여 전산적 처리가 용이하게 재구축해야 할 필요가 있다. 연구의 목적상 다음과 같이 필요한 정보 항목을 선정하여 격틀사전을 구축하였다.

```
<orth>가다</orth>
  <sense n="01">
    <frame>X=N0-이 Y=N1-로 V</frame>
      <sel_rst arg="X" tht="THM">교통기관(버스|기차|비행기)</sel_rst>
      <sel_rst arg="Y" tht="GOL">장소</sel_rst>
      <eg>철수는 무작정 부산으로 가는 버스를 탔다.</eg>
  </sense>
  <sense n="04">
    <frame>X=N0-이 Y=N1-로 V</frame>
      <sel_rst arg="X" tht="THM">교통기관(자동차|전차)</sel_rst>
      <sel_rst arg="Y" tht="INS">연료(가스)|에너지원(전기)</sel_rst>
      <eg>이 차는 기름이 아니라 전기로 간다.</eg>
  </sense>
```

[그림 3] 표제어 '가다' XML 의 일부분

위의 [그림 3]에서 프레임인 'X=N0-이 Y=N1-로 V'의 변수인 X, Y와 선택 제약 정보인 <sel_rst>의 X, Y 간의 연결 정보를 조사를 기준으로 변형하면 아래와 같은 구조를 얻을 수 있다.

[표 1] 구성된 격틀사전의 엔트리 '가다'의 일부분

표제어		가다	
격조사	의미부류	예제	의미역
Sense 1			
이	교통기관	버스, 기차, ...	대상
로	장소		장소
Sense 2			
이	교통기관	자동차, 전차, ...	대상
로	연료, 에너지원	가스, 전기, ...	도구

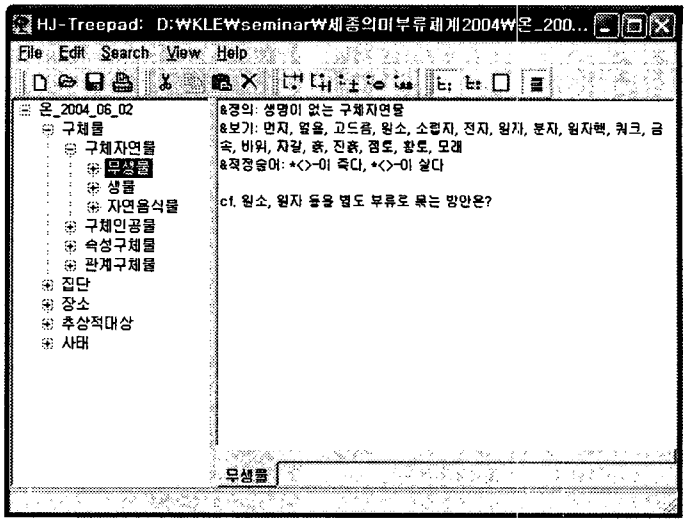
완성된 격틀사전은 총 18,195개의 표제어와 32,845개의 격틀로 구성되어 있다. 따라서 표제어 하나당 평균적으로 1.8개의 격틀이 있다고 할 수 있겠다.

4.2 체언사전의 의미부류 정보 이용

격틀사전의 선택 제약 정보는 세종전자사전의 명사의미부류를 이용해 기록되었다. 이 명사의미부류는 [그림 4]에서 보는 바와 같이 최상위 의미 노드를 제외하면 총 581 개의 부류로 구성된다. 최상위 부류로는 <구체물>, <집단>, <장소>, <추상적대상>, <사태> 등 5 개의 부류가 설정되었는데 이중 처음 4개는 논항명사의 의미부류이고, 나머지 <사태> 부류는 술어명사의 의미부류이다.

마찬가지로 체언사전의 각 명사 의미도 이러한 명사의미부류를 이용해서 구분된다. 본 연구에서는 체언사전의 표제어와 해당 명사의미부류 정보를 다음과 같은 용도로 사용하였다.

첫째, [그림 2]에서와 같이 의미역 결정 시스템의 입력



[그림 4] 세종 명사의미부류 체계

으로 사용되는 구문 트리 노드의 '개념'은 가도카와 시소러스 상의 개념을 사용한다. 명사의 의미를 세종 명사의미부류로 사상하는 명사의미중의성 해소 모듈이 아직 개발되지 않았기 때문에, 격틀사전에서 선택 제약 정보를 사용하여 적합한 격틀을 선택하고자 할 때 가도카와 시소러스 개념 부류에서 세종 명사의미부류로의 연결 정보가 필요하다. 즉, 각각의 가도카와 개념 코드에 대응하는 세종 명사의미부류의 코드를 얻어야 한다. 이를 위해서 우선 가도카와 개념 코드가 지배하는 명사 목록을 얻고 해당하는 각각의 명사에 대응하는 세종 명사의미부류의 코드를 체언사전을 이용해서 얻는다. 이렇게 만들어진 세종 명사의미부류의 목록 중에서 가장 공유도가 큰 의미부류를 가도카와 개념 코드의 대응하는 의미부류로 선택한다. 즉, 가도카와 개념 코드가 지배하는 명사 목록과 세종 명사의미부류의 코드가 지배하는 명사 목록의 교집합의 크기가 최대가 되는 세종 명사의미부류의 코드를 선택하는 것이다. 물론 명사의 의미중의성 때문에 이 과정에서 발생하는 연결 오류가 있을 수 있지만 명사 목록의 수가 크면 그러한 오류는 쉽게 해결되어 진다고 본다.

둘째, [표 1]에서 보는 바와 같이 각 격틀에는 선택 제약 정보인 의미부류와 예제가 기록되어 있다. 하지만 격틀사전의 구축 과정 중 예제만 있고 해당하는 의미부류가 없는 경우를 상당 수 발견하였다. 이 문제도 앞에서와 마찬가지 방법으로 해당 의미부류의 코드를 얻어 해결할 수 있다.

4.3 적합한 격틀의 선택

격틀사전을 이용한 의미역 결정 시스템은 입력된 문장의 용언과 각 격 요소의 명사의미부류를 이용해서 해당 문장에 가장 잘 부합하는 격틀을 선택한다. 이러한 선택 시 기준이 되는 것은 바로 격틀에 기재되어 있는 선택 제약 정보이다. 본 연구에서는 [9]의 연구를 참조해서 아래와 같이 적합한 격틀을 선택하는 방법을 사용하였다.

4.3.1 입력문의 격 요소와 격틀의 격 요소 사이의 유사도

두 격 요소 사이의 유사도는 세종 명사의미부류의 개념 유사도로 정의한다. 기본적으로 두 개념 노드의 공통 부모 노드들 중 가장 구체적인 개념 노드의 레벨을 두 개념간 유사도로 사용한다.

$$sim(T,P_i)=\frac{2\times level(MSCA(T,P_i))}{level(T)+level(P_i)}\times weight$$

※ MSCA : most specific common ancestor
※ sim(T, P_i) = 0 if MSCA(T, P_i) is level 1
※ weight = 1 (direct descendent), 0.5 otherwise

4.3.2 입력문과 격틀의 유사도

입력문과 격틀의 유사도는 입력문을 기준으로, 입력문의 격 요소와 매칭되는 격틀의 격 요소 사이의 유사도의 합(S)과 몇 가지 매개 변수를 이용해 정의한다.

$$SimScore=\begin{cases}0 & \text{if } L>N\\ S\times\sqrt{(\frac{1}{N})}\sqrt{(\frac{N}{M})} & \text{otherwise}\end{cases}$$

S : 매칭되는 두 격 요소 간 유사도의 총합
M : 격틀에 있는 필수(Obligatory) 격 요소의 수
L : 입력문에 있는 필수 격 요소의 수
N : 입력문과 격틀간 매칭되는 격 요소의 수

[9]의 연구에서도 소개되었지만 'L > M' 인 경우 유사도가 0 인 이유는 입력문에 나타난 필수 격은 반드시 격틀에 있는 필수격과 매칭되어야 하기 때문이다. 그렇지 않다면 격틀의 기술에 잘못되었거나 입력된 문장이 비문일 수 있다. 또한 입력문의 필수격 요소와 격틀의 필수격 요소 간 매칭되는 비율이 클 수록 유사도가 높게 설정되어야 하기 때문에 'N/M' 을 추가하였으며 매칭되는 격 요소의 수가 많을 수록 전체 유사도가 커 질 수 있기 때문에 정규화를 위해 '1/N' 을 추가하였다.

4.3.3 격틀 선택 시 발생하는 문제점

앞에서도 언급하였듯이 세종 용언사전에는 약 1,8000 개의 표제어가 수록되어 있다. 하지만 아직도 계속 개발 중에 있기 때문에 실제 적용 시 표제어가 없거나 해당하는 격틀이 없는 경우가 있을 수 있다. 또한 격틀사전의 격틀은 용언이 실제 활용될 때 필요한 필수적인 격에 대해서만 기술되어 있어서 문장 분석 결과로 나온 구문 트리의 부사격이 임의격(Optional case)인 경우, 격틀 검색 시 해당되는 격틀이 없는 경우가 있다. 이런 문제 이외에 보다 근본적인 문제는 격틀 검색 결과 같은 점수를 갖는 격틀이 둘 이상 발견되는 경우이다. 이러한 문제가 발생되는 근본적인 원인은 격틀의 선택 제약 정보를 기술할 때 사전 기술에 상당히 익숙한 기술자라 할 지라도 정밀한 명사의미부류를 선택해서 기술하는 것은 현실적으로 어렵기 때문에 적정 수준의 상위 명사의미부류로 선택 제약 정보를 기술했기 때문이다. 마지막으로 격틀 선택 시 문제가 되는 것으로는 같은 점수를 얻는 격틀이 존재하지 않지만 실제 잘못 선택된 격틀인 경우이다. 여러 가지 요인이 있을 수 있겠으나 한 가지 예를 들면, 선택된 격틀의 점수가 낮지만 하나 뿐이어서 어쩔 수 없이 선택되는 경우가 있다. 이러한 문제를 예방하기 위해서 선택된 격틀의 점수가 어떤 임계값 이하이면 차후 처리를 위해 결정을 보류하는 방법이 있을 수 있다.

이와 같은 문제점들을 해결하는 방법은 여러 가지 있을 수 있겠으나 본 연구에서는 통계적인 방법론을 이용하였다. 즉, 입력된 문장 내 부사격의 의미역 결정을 먼저 격틀사전을 이용해서 처리하고 앞에서 언급한 문제가 발생한 경우 확률을 이용해 결정하겠다는 것이다. 이 부분에 대해서는 다음절에 자세히 언급하겠다.

5. SVM을 이용한 의미역 결정

기존 연구 부분에서도 언급하였지만 통계적인 방법으로 의미역을 결정하기 위해서는 우선 의미역이 태깅된 코퍼스의 구축이 필수적이다. 하지만 현실적으로 코퍼스의 구축에는 상당한 시간과 노력의 비용이 소모되기 마련이다. 따라서 본 연구에서는 [그림 2]에서 보는 바와 같이 세종 용언사전의 각 격틀 정보에 추가적으로 기술되어 있는 예문을 코퍼스로 활용하였다. 이러한 예문의 해당 부사격에 대한 의미역 태깅은 쉽고 빠르게 이루어 질 수 있다.

다음으로 구축된 코퍼스의 각 문장에서 의미역 결정에 유용한 자질을 추출해야 한다. 이를 위해서 2 절에서 언급한 형태소 분석기, 구문 분석기, 단어의미중의성 해소 모듈을 사용하여 [그림 2]와 같은 형태의 단어의미중의성이 해소된 구문 트리를 얻고 각 단계에서 나타나는 오류는 수작업을 통해 수정하여 아래 표와 같은 자질 집합을 얻었다.

[표 2] 학습을 위해 선택된 자질 집합
예문, [그녀는 불길한 **예감-에** 몸을 벌떡 **일으켰다**]

자 질		약 어	예
의존소	어휘	LD	예감
	개념	CD	느낌-400
	품사	PD	CMCPA(서술성 명사)*
	명사형 전성어미	ED	-
	보조사	AD	-
지배소	어휘	LG	일으키
	개념	CG	일어남-353
	품사	PG	YBDO(일반동사)
	선어말어미	EG	-
	동사파생접사	SG	-
주어	품사	PS	CTP3(3인칭 대명사)
	개념	CS	타칭-503
목적어	품사	PO	CMCN(일반 명사)
	개념	CO	육체-600
기타	조사 패턴	CP	는-에-을
	품사 패턴	PP	CTP3-CMCPA-CMCN-YBDO

* KLE tag set

기계학습을 위한 모델로는 SVM(Support Vector Machine)을 사용하였다. [10]에서 밝힌 바와 같이 SVM은 이진 분류기(Binary classifier)이기 때문에 다중 분류 문제(Multi-class classification problem)를 위해서는 여러 개의 이진 분류 문제로 나누고 각 결과를 통합하는 방법을 사용해야 한다. 이 문제에 대한 일반적인 접근 방법은 PAIRWISE 접근법과 ONE VS ALL 접근법이 있는데 본 연구에서는 PAIRWISE 접근법을 구현한 LIBSVM[12]을 사용하였다.

SVM 모델의 구축을 위해서 앞의 [표 2]와 같은 자질 집합으로 구성된 자질 벡터 집합(학습 데이터)을 이용해서 10-fold cross validation과 같은 방법으로 SVM에 적합한 매개변수를 결정해야 한다. 즉, 최적의 성능을 보이는 모델의 매개변수를 결정한 후 이를 이용해 SVM 모델을 구축한다. 또한 고려해야 할 사항은 [표 2]에 있는 자질 집합 중에서 의미역 결정에 유효한 자질들을 골라내는 자질 선택 작업도 필요하다.

6. 제안된 의미역 결정 시스템의 구조

5절과 6절에서 언급한 데로 본 연구에서는 의미역 결정을 위해서 먼저 격틀사전을 이용하고 여기서 문제가 발생하는 경우 SVM 모델에 의해 그 문제를 해결하는 혼합 시스템을 제안한다. 여기서 짚고 넘어가야 할 것은 격틀 사전의 검색으로 해결 가능한 문제의 범위가 정확히 어디까지 인가를 명확히 정해야 한다는 것이다. 만약 그렇지 않으면 뒤에서 SVM 모델로 해결 가능한 문제를 격틀사전을 이용한 의미역 결정 시스템이 잘못 결정하여 오류의 비율을 증가시킬 수 있기 때문이다. 따라서 격틀사전 검색 시스템은 최대한 자신이 확실히 처리할 수 있는 부분만을 처리하고 나머지는 미해결 상태로 남겨두어야 할 것이다. 이를 위해서 5절에서 보았듯이 격틀사전의 검색 시 발생할 수 있는 문제의 부류를 다음과 같이 구분했다.

[표 3] 격틀사전 검색 시 발생 가능한 문제의 종류

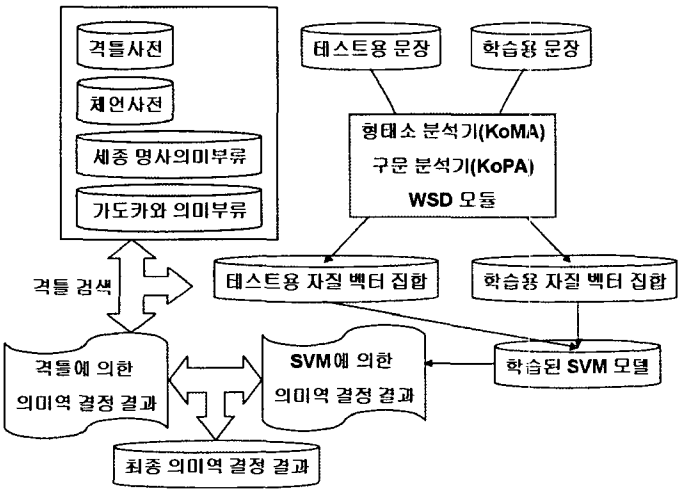
문제의 종류	내 용
NO ENTRY	해당 표제어가 격틀 사전에 없음
NO FRAME	표제어는 있지만 해당 부사격을 가지고 있는 격틀이 없음
CONFLICT	해당 부사격을 가지고 있는 격틀이 여러 개 있고 입력 문장과 유사도 계산 결과 같은 점수를 얻었음
ERROR	선택된 격틀의 유사도가 낮지만 어쩔 수 없이 선택 된 경우나 기타 사전 오류

위의 표에서 마지막에 있는 'ERROR'는 격틀사전 검색 시에는 알 수 없는 문제이며 SVM 모델을 통해 의미역을 결정한 후에도 여전히 오류로 남아 있는 문제이다. 왜냐하면 격틀사전 검색기는 이미 찾아낸 격틀을 통해 의미역을 결정해 버렸기 때문이다. 따라서 이 부분에 해당되는 오류를 최대한 줄이는 방향으로 격틀사전 검색 시스템을 수정해 나가는 것이 필요하다.

격틀사전 검색 시 위의 표와 같은 문제들('ERROR'는 제외)이 발생되는 경우에 대해서는 격틀사전 검색사전 시스템은 책임지지 않고 다음 처리로 넘긴다.

격틀사전 검색 시스템에서 해결되지 못한 데이터는 5절에서 언급한 데로 학습된 SVM 모델을 이용해서 그 의미역이 결정된다.

지금까지 설명한 내용을 종합하여 도식하면 아래 그림과 같다.



[그림 5] 의미역 결정 시스템의 구조

7. 실험 결과

7.1 학습 데이터와 테스트 데이터

5절에서도 언급했다시피 세종 용언사전의 격틀 정보에 추가되어 있는 예문 4,355 문장을 SVM 모델의 학습을 위한 기본 코퍼스로 사용하였다. 또한 한국어 형태소 분석기 및 품사 태거 평가 워크숍을 위해 제공 받은 말뭉치(MATEC' 99)에서 임의로 추출한 1,320 문장을 테스트 데이터로 사용하여 시스템의 성능을 평가하였다.

7.2 SVM 모델의 구축 결과

[표 2]의 자질 집합 중에서 최적의 성능을 보이는 자질 조합을 찾고 SVM 모델의 매개변수를 결정하기 위해서 10-fold cross validation 실험을 한 결과는 아래와 같다.

[표 4] 기본 모델에 추가된 자질 별 효과(10-fold CV)

자 질	정확률(%)	
	에	로
CD+CG (baseline)	69.44	73.80
+ PD + PG	69.65	74.88
+ ED	69.45	73.50
+ AD	69.75	73.50
+ EG + SG	70.67	74.10
+ PS + PO	69.44	73.80
+ CS + CO	69.44	73.80
+ CP	69.75	74.04
+ PP	69.51	73.26
+ LD + LG	73.46	74.70

자 질	정확률(%)	
	에	로
CD+CG (baseline)	69.44	73.80
+ LG	79.10	78.52
+ LG + CP	79.48	79.17
+ LG + CP + EG + SG	80.57	80.87
MFC	43.86	33.95

* MFC : Most Frequent Class
* SVM Kernel function : RBF, C = 2**5, G=2*(-11)

위와 같이 실험한 결과 최적의 성능을 보이는 자질 조합은 의존소의 경우는 '개념', 지배소의 경우는 '어휘, 개념, 선어말어미, 파생접사', 그리고 기타 자질로 문장의 조사 패턴이다. 또한 최적의 성능을 보이는 SVM 모델의 매개변수는 위의 [표 4]의 아래 부분에 보는 바와 같다.

7.3 테스트 데이터를 통한 실험 결과

테스트 데이터를 격틀사전 검색 시스템과 SVM 모델을 통해 그 의미역을 결정한 결과를 아래의 표와 같이 순차적으로 제시하였다.

[표 5] 격틀사전에 의한 성능(오류 비율)

오류 타입	에(%)	로(%)	총 계(%)
NOENTRY	13.67	11.84	12.75
NOFRAME	11.65	34.24	22.94
CONFLICT	32.08	20.8	26.44
ERROR	1.87	4.64	3.25
TOTAL	59.27	71.52	65.39

위의 [표 5]를 살펴보면 부사격 조사 '에'와 '로'에서 오류 타입의 비율은 '에'의 경우 'CONFLICT'가 '로'의 경우는 'NOFRAME'이 상대적으로 많은 것을 알 수 있다. 'CONFLICT'가 많다는 것은 그만큼 관련된 격틀의 수도 많다고 해석할 수 있고 'NOFRAME'이 많다는 것은 '로'의 경우와 같이 자주 도구격과 같은 임의격으로 사용되는 조사이기 때문에 부합되지 않는 격틀이 많다고 해석이 가능하다.

격틀사전에 의한 시스템의 성능은 시스템이 결정한 의미역 중에서 올바른 의미역 수의 비율이다. 이에 따라 계산하면 격틀사전에 의한 시스템의 성능은 90.79% 이다. 이 비율은 'ERROR' 타입의 오류가 발생한 경우를 포함하여 시스템이 결정한 의미역의 수와 올바르게 결정된 의미역의 수의 비율이다.

[표 6] SVM 모델에 의한 성능(오류 비율)

오류 타입	에(%)	로(%)	총 계(%)
NOENTRY	5.9	4.96	5.43
NOFRAME	4.46	17.44	10.95
CONFLICT	9.78	8.32	9.05
ERROR	1.87	4.64	3.25
TOTAL	22.01	35.36	28.68

다음으로 위의 [표 6]을 분석해 보면 격틀사전에 기반한 시스템이 결정을 미루고 넘겨준 62.13% 의 데이터 중에서 59.34%의 데이터를 정확하게 처리하였다.

[표 7] SVM 모델만을 사용한 경우와 혼합 시스템의 성능 비교

구분	SVM 모델 (%)	혼합 (%)	성능향상(%)
에	69.06	77.98	8.92
로	58.56	64.64	6.08
소계	63.81	71.31	7.5

SVM 모델을 단독으로 사용하는 경우와 비교해 보면 위의 [표 7]과 같이 평균 7.5 % 의 성능 향상 효과를 보이는 것을 알 수 있다.

8. 결론 및 향후 연구

본 연구에서는 한국어 부사격의 의미역을 결정하기 위해서 먼저 세종전자사전의 용언사전과 체언사전을 이용하여 전산처리가 용이한 격틀사전을 구축하고 이를 이용해 문장에 나타난 부사격 '에'와 '로'에 대한 의미역을 결정하였다. 격틀사전의 검색 시 앞에서 정의한 오류 타입인 경우는 의미역을 결정하지 않고 차후에 SVM 모델을 이용하여 처리하였다. 실험 결과 격틀사전과 SVM 모델을 이용한 혼합 시스템의 성능은 어느 하나를 단독으로 사용하는 것 보다 월등한 우위를 보이는 것으로 나타났다.

또한 SVM 모델의 구축 시 사용되는 학습데이터인 말뭉치의 획득 및 의미역 태깅의 용이성을 위해서 세종전자사전 용언사전의 격틀 정보에 추가되어 있는 예문을 활용하였다. 이는 의미역 태깅에 드는 비용을 줄일 수 있는 효과적인 방법이라 할 수 있다.

마지막으로, 이번 연구에 이어서 향후에 연구되어야 할 내용은 다음과 같은 것이 있을 수 있겠다.

첫째로, 격틀사전 검색 시스템의 적용률을 높여야 할 것이다. 즉, 'NOENTRY' 타입의 오류와 같은 경우 비슷한 용언 부류의 격틀을 이용하는 방법에 대한 연구와 'CONFLICT' 타입의 오류가 발생하는 비율을 줄이기 위해 유사도 계산 공식에 대한 고찰 등등이 필요하다.

둘째로, 구축된 SVM 모델 자체의 성능 향상을 위해 효과적인 자질 등에 대한 연구가 필요하다.

9. 참고 문헌

[1]Daniel Gildea and D. Jurafsky. *Automatic Labeling of Semantic Roles*, Computational Linguistics, 28(3):245-288, 2002

[2]S.B. Park. *Decision Tree Based Disambiguation of Semantic Roles for Korean Adverbial Postpositions*, IEICE Transaction Information and System, Vol.E86-D, No. 8, 2003

[3]양단희, 송만석. *기계학습에 의한 단어의 격 원형성 자동 획득*, 정보과학회지, 제 25권, 제 7 호, pp. 1116-1127, 1998

[4]Kang, W.S., et al. *A Neural Network Method for the*

Semantic Analysis of Prepositional Phrases in English-to-Korean Machine Translation, An International Journal of the Chinese Language Computer Society, vol.8, no.2, pp. 143-162, 1994
[5]Jung-Hye Park. *Determination of Thematic Roles according to Syntactic Relations Using Rules and Statistical Models*, MS Thesis, Pohang University of Science and Technology, 2002
[6]강신재, 박정혜. *대규모 말뭉치와 전산 언어 사전을 이용한 의미역 결정 규칙의 구축*, 정보처리학회 논문지 B 제 10-B권 제 2호, 2003
[7]Hui-Feng Li. *Conceptual Graph Generation from Syntactic Dependency Structures for an Interlingua-Based MT System*, Phd Thesis, Pohang University of Science and Technology, 1998
[8]Beale, S., S. Nirenburg and K. Mahesh. *Semantic Analysis in The Mikrokosmos Machine Translation*, In Proc. Of Symposium on NLP, Kaset Sart Univerity, Bangkok, Thailand, 1995
[9] Kurohashi, S., and Nagao, M. *A Method of Case Structure Analysis for Japanese Based on Examples in Case Frame Dictionary*. IEICE Transactions on Information and System, vol.E77-D, no.2, pp. 227-239. 1994
[10]Kadri Hacioglu, et al. *Shallow Semantic Parsing Using Support Vector Machines*. CSLR Tech. Report, CSLR-TR-2003-1, 2003
[11]Kadri Hacioglu, et al. *Semantic Role Labeling Using Dependency Trees*, In COLING 2004, 2004
[12]C.-C. Chang and C.-J. Lin. LIBSVM: a library for support vector machine, Software available at http://www.csie.ntu.edu-.tw/~cjlin/libsvm , 2001
[13]이성헌. *전자사전에서의 기능동사구문 처리 문제-세종 체언사전의 경우-*, 한국사전학회 제5회 학술대회 발표자료집, 2004
[14]You-Jin Chung, et al. *Word Sense Disambiguation Using Neural Networks with Concept Co-occurrence Information*, NLPRS 2001. pp. 715 - 722, 2001
[15]홍재성 외. *21세기 세종계획 전자사전개발 연구보고서*, 문화관광부, pp. 110 ~ 116, 2004
[16]Matoko Nagao et al. 최기선 외. 번역. (1999). *자연언어이해*, 홍릉과학출판사, pp. 13-42