절 경계와 트리 거리를 사용한 2단계 부분 의미 분석 시스템

(A Two-Phase Shallow Semantic Parsing System Using Clause Boundary Information and Tree Distance)

박 경 미 † 황 규 백 ††
(Kyung-Mi Park) (Kyu-Baek Hwang)

요 약 본 논문은 최대 엔트로피 모형에 기반한 두 단계 부분 의미 분석 방법을 제안한다. 먼저, 의미 논항의 경계를 인식하고, 그 다음 단계에서 확인된 논항에 적절한 의미역을 할당한다. 두 단계 부분 의미 분석에서는 두 번째 단계인 논항 분류가 논항 확인 단계의 결과에 기반하여 수행되기 때문에 논항 확인의 성능이 매우 중요하다. 본 논문은 논항 확인의 성능을 향상시키기 위하여 논항 확인의 전처리 단계에 구문 지식을 통합한다. 구체적으로, 절 인식 결과로부터 술어의 인접절 및 상위절들을 확인하고, 구문 분석 결과로부터 술어의 부모 노드로부터 구문 구성 요소의 부모 노드까지의 트리 거리를 추출하여 전처리 단계에서 활용한다. 실험을 통해, 구문 지식을 활용하는 것이 부분 의미 분석 성능에 기여함과 제안하는 두 단계 방법이 한 단계 방법보다 우수한 성능을 낼 수 있음을 보인다.

키워드 : 부분 의미 분석, 의미 논항 확인, 의미 논항 분류, 최대 엔트로피 모형, 절 경계 제약, 트리 거리 제약

Abstract In this paper, we present a two-phase shallow semantic parsing method based on a maximum entropy model. The first phase is to recognize semantic arguments, i.e., argument identification. The second phase is to assign appropriate semantic roles to the recognized arguments, i.e., argument classification. Here, the performance of the first phase is crucial for the success of the entire system, because the second phase is performed on the regions recognized at the identification stage. In order to improve performances of the argument identification, we incorporate syntactic knowledge into its pre-processing step. More precisely, boundaries of the immediate clause and the upper clauses of a predicate obtained from clause identification are utilized for reducing the search space. Further, the distance on parse trees from the parent node of a predicate to the parent node of a parse constituent is exploited. Experimental results show that incorporation of syntactic knowledge and the separation of argument identification from the entire procedure enhance performances of the shallow semantic parsing system.

Key words : Shallow semantic parsing, semantic argument identification, semantic argument classification, maximum entropy models, clause boundary restriction, tree distance restriction

· 이 연구는 2009년 정부(교육과학기술부) 재원으로 한국연구재단의 지원을 받아 수행되었음(NRF-2009-351-D00078)

† 정 회 원 : 숭실대학교 컴퓨터학부
parkkyungmi75@gmail.com
†† 정 회 원 : 숭실대학교 컴퓨터학부 교수
kbhwang@ssu.ac.kr
(Corresponding author임)

논문접수 : 2010년 1월 5일
심사완료 : 2010년 3월 19일

정보과학회논문지: 컴퓨팅의 실제 및 레터 제16권 제5호(2010.5)

1. 서 론

부분 의미 분석(shallow semantic parsing)은 두 가지 단계로 구분할 수 있다. 주어진 문장에서 의미 논항의 경계를 인식하는 의미 논항 확인(semantic argument identification) 단계와 인식된 논항의 의미적 역할을 결정하는 의미 논항 분류(semantic argument classification) 단계이다. 두 단계 방법을 적용하면 첫 단계에서 클래스의 수를 줄일 수 있기 때문에, 다중 클래스 분류 문제에서 발생하는 클래스 분포의 불균형 문제(unbalanced class distribution problem)를 완화할 수

있다. 즉, 논항 확인 단계에서는 대상 단어들에 세 개(B-ARG, I-ARG, O)의 클래스 중 하나를 할당하고, 논항 분류 단계에서는 인식된 논항들에 대해서만 적절한 의미역을 할당한다. B-ARG는 첫 번째 논항 구성 요소를 나타내고, I-ARG는 논항 구성 요소이면서 첫 번째에 위치하지 않는 요소를 나타내며, O는 논항을 구성하지 않는 요소임을 나타낸다. 또한 논항 확인을 위하여 중요한 자질들과 논항 분류를 위하여 중요한 자질들이 다르기 때문에, 두 단계로 구분하면 각 단계에 적절한 자질 집합을 구성할 수 있다.

본 논문에서는 최대 엔트로피 모형(maximum entropy model)에 기반한 두 단계 부분 의미 분석 방법을 제안한다. 두 단계 부분 의미 분석 방법에서는, 논항 확인의 오류들이 논항 분류 단계로 전파되기 때문에, 논항 확인의 성능이 전체 성능을 좌우한다. 본 논문은 논항 확인 단계의 성능을 향상시키기 위하여 구문 지식을 활용한 논항 확인의 전처리 방법을 제안한다.

대부분의 의미 논항들은 술어의 특정 범위 내에 존재하고, 술어의 원거리에서는 거의 나타나지 않는 특징이 있다. 따라서 의미 논항을 인식하기 위하여 문장의 모든 요소를 탐색할 필요가 없다. 본 논문은 절 인식 결과 및 구문 분석 결과로부터 추출한 구문 지식을 바탕으로 의미 논항을 찾는 범위를 한정한다. 구문 지식은 주어진 문장 중에서 의미 논항을 포함할 가능성이 높은 부분을 분별하는 데 유용한 정보를 제공할 수 있다.

절 인식 결과로부터 술어의 인접절 및 상위절들을 확인할 수 있다. 대부분의 의미 논항들은 술어의 특정절의 왼쪽 경계 및 오른쪽 경계에서 나타난다고 가정하여 논항 확인의 전처리에서 절 경계 제약(clause boundary restriction)을 수행한다. 후보 논항은 구문 분석기의 결과로 얻어진 트리의 구성 요소들이다. 트리 거리는 술어의 부모 노드로부터 구문 구성 요소인 후보 논항의 부모 노드까지의 거리이다. 대부분의 경우 특정 트리 거리 내에서 의미 논항이 발생한다고 가정하여 트리 거리 제약(tree distance restriction)을 적용한다. 논항 확인 단계의 전처리에서 의미 논항을 발생시킬 가능성이 높은 문장의 일부분을 절 경계 제약 또는 트리 거리 제약을 통하여 확인함으로써, 후보 논항의 수를 줄일 수 있고, O 클래스에 해당하는 논항 비구성 구문 요소들로부터 생성되는 부정적인 학습 예제의 수를 줄일 수 있다. 따라서 클래스 분포의 불균형 문제를 완화할 수 있고 학습 비용(training cost)을 낮출 수 있다.

본 논문의 구성은 다음과 같다. 2절에서는 부분 의미 분석과 관련한 기존의 연구들을 살펴본다. 3절에서는 최대 엔트로피 모형에 기반한 두 단계 부분 의미 분석 방법을 기술한다. 부분 의미 분석을 논항 확인과 논항 분류의 두 단계로 구분하고 전체 성능을 좌우하는 논항 확인의 성능을 향상시키기 위한 전처리 방법을 제안한다. 또한, 최대 엔트로피 모형 및 자연 언어 처리 기술을 적용하여 추출한 자질들에 대하여 설명한다. 4절에서는 여러 실험을 통해, 제안한 방법이 부분 의미 분석에 유용한지를 검증한다. 끝으로 5절에서는 본 논문을 통하여 얻은 결론과 연구의 기여에 대하여 기술한다.

2. 관련 연구

부분 의미 분석에 다양한 기계 학습 방법을 적용하여 구문분석 결과로부터 논항을 추출하고 그것의 의미적 역할을 인식하는 다양한 연구들이 있었다[1-5]. 이 연구들의 초점은 새로운 어휘 정보로 인한 자료 부족 문제의 완화[2,5], 자질의 고안[3], 구문 분석기의 오류 전파를 완화하는 방법의 제안[4,5] 등에 맞추어져 있었다. 또한, 부분 의미 분석의 결과를 정보 추출 등의 응용 분야에 어떻게 적용할 것인가의 문제를 다룬 연구들도 있었다[1,2].

어휘 정보로 인한 자료 부족 문제를 완화하기 위해서 WordNet이나 대량의 신문 기사와 같은 외부 자원을 이용하거나 학습 말뭉치 내에서 자질로 빈번하게 사용되는 동사와 명사를 군집화하였다[2,5]. 동사를 군집화하기 위하여 이용한 직관은 비슷한 의미를 갖는 동사들은 비슷한 직접 목적어를 갖는다는 것이다[5]. 명사를 군집화하기 위해서는 같은 동사와 자주 공기하는 명사들은 유사한 의미를 갖는다는 직관을 사용하였다[2].

논항 확인 단계는 구문 분석기의 결과로부터 후보 논항을 추출하는데 구문 분석기의 오류로 정답 논항과 경계가 일치하는 후보 논항이 존재하지 않을 수 있다. 이러한 구문 분석기의 오류 전파 문제를 완화하기 위하여 여러 가지 구문 분석기로부터 후보 논항을 추출하는 것이 부분 의미 분석의 성능에 기여함을 보인 연구가 있었다[4,5].

3. 최대 엔트로피 기반 부분 의미 분석

최대 엔트로피 기반의 두 단계 부분 의미 분석 시스템의 구성도는 그림 1과 같다. 시스템은 자동 분석, 논항 확인, 논항 분류의 과정을 거치게 된다. 먼저, 자동 분석 과정은 기존의 자연언어처리 시스템을 적용하여 입력 문장으로부터 그림 4와 같은 구문 분석된 문장을 생성한다[6-8]. 다음으로, 논항 확인 단계에서는 절 경계 제약 또는 트리 거리 제약을 통한 전처리를 수행하고, 후보 논항 중 자질 추출 및 분류기 적용을 통하여 논항을 찾는다. 마지막으로, 논항 분류 단계에서는 술어-논항 인식 결과로부터 자질 추출 및 분류기 적용을 통하여 각 논항의 의미적 역할이 무엇인지 판단한다.

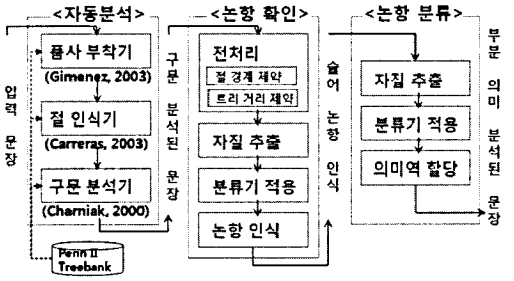

그림 1 시스템 구성도

본 시스템의 입력과 출력은 그림 2와 같다. 시스템의 입력은 문장과 술어 정보이다. 술어는 동사 원형 정보를 포함한다. 시스템의 출력은 각 동사의 논항과 그것의 의미역이다. 예를 들어, 'exist'는 문장에서 의미역이 A1인 'contact'만을 논항으로 갖는 반면에, 'deliver'는 5개의 논항을 갖고 그것들의 의미역은 각각 A0, A1, A2, AM-LOC, AM-TMP이다. 각 의미역에 대한 설명은 4.1절에서 주어진다.

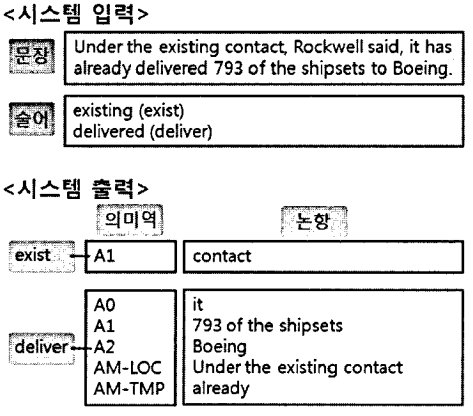

그림 2 시스템의 입력과 출력

3.1 최대 엔트로피 모형

최대 엔트로피 모형에서 history h가 주어졌을 때 outcome o를 예측하는 조건부 확률은 다음과 같이 정의된다.

$$P(o \mid h)=\frac{1}{Z_\lambda(h)}\exp\left(\sum_{i=1}^{k}\lambda_i f_i(h,o)\right)$$

여기서, $f_i(h,o)$는 0 또는 1의 값을 갖는 자질 함수이고, λ_i는 함수 $f_i(h,o)$의 가중치 파라미터이며, k는 자질의 수를 나타내고, $Z_\lambda(h)$는 $\Sigma_o P(o|h)=1$을 위한 정규화 인자이다[9]. 확률 $P(o|h)$는 $f_i(h,o)=1$을 만족하는 유효한(active) 자질들의 가중치 합에 의해서 계산된다.

본 논문은 부분 의미 분석을 논항 확인 단계와 논항 분류 단계로 구분하고 두 단계 모두에서 최대 엔트로피 모형에 기반한 분류를 수행한다. 첫 번째 단계인 논항 확인 단계에서 자질의 예는 다음과 같다.

$$f_i(h,o) = \begin{cases}1 & \text{if path}=\text{NP}\uparrow\text{S}\downarrow\text{VP},\ o=\text{B}-\text{ARG}\\ 0 & \text{otherwise}\end{cases}$$

위의 예제가 의미하는 것은 구문 구성 요소와 술어 사이의구문 경로가 NP↑S↓VP인 경우 구문 구성 요소가 술어의 논항을 구성하는 첫 번째 요소일 가능성이 높다는 것이다. 두 번째 단계인 논항 분류 단계에서 자질의 예는 다음과 같다.

$$f_i(h,o) = \begin{cases}1 & \text{if head}=\text{window},\ o=\text{A0}\\ 0 & \text{otherwise}\end{cases}$$

위의 예제가 의미하는 것은 구문 구성 요소의 중심어가 window인 경우 인식된 논항의 의미역이 A0일 가능성이 높다는 것이다.

3.2 전처리

본 시스템의 논항 확인 모듈은 최대 엔트로피 모형에 기반한 술어-논항 인식기로 구문 분석기의 결과로 얻어진 구문 구성 요소들 중에서 각 동사의 논항을 식별한다. 이것의 특징은 후보 논항의 수 및 부정적인 학습 예제들을 줄이기 위하여 절 경계 또는 트리 거리 제약 조건을 적용한 전처리를 수행한다는 것이다.

절 경계 제약은 절 인식 결과로부터 해당 동사의 인접절(immediate clause) 및 상위절들(upper clauses)을 확인하고 실험 결과에 따라 특정절 내에서 논항을 찾는 것이다. 대부분의 논항들은 특정절의 왼쪽 경계 및 오른쪽 경계 내에서 발생한다고 가정하여 후보 논항들이 이 제약 조건을 만족하는지를 검사한다.

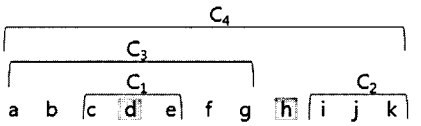

그림 3 인접절 및 상위절들의 예

그림 3은 절 인식 결과에서 술어의 인접절(immediate clause) 및 상위절들(upper clauses)의 예를 나타낸다. 예를 들어, d와 h가 술어일 때, d의 인접절은 d를 포함한 가장 짧은 길이의 절인 C_1이고 d를 포함하는 나머지 절들은 상위절이 된다. 이 중 첫 번째 상위절은 C_3이고 두 번째 상위절은 C_4이다. 술어 h를 포함하는 절은 C_4 하나이기 때문에 주어진 문장에서 h의 상위절은 존재하지 않고 인접절 C_4만 존재한다. 만약 술어 d의 절 경계 제약 조건이 첫 번째 상위절의 왼쪽 경계와 인접절의 오른쪽 경계 내에서 논항을 찾는 것이라면 단어 a부터 e까지 내에서 논항 확인 작업을 수행한다. 이 범위를 벗어난 후보 논항들은 고려 대상에서 제외된다.

구문 트리상의 거리는 술어의 부모 노드로부터 구문 구성요소의 부모 노드까지 같은 방향으로만 이동한 횟수를 의미한다. 구문 구조를 갖는 문장에서 논항의 부모 노드는 술어의 부모 노드로부터 일정한 트리 거리 내에 주로 존재한다고 가정하여 특정 트리 거리 내에서 논항을 찾는다. 이와 같은 트리 거리 제약 조건을 적용한 전처리는 논항을 구성하지 않는 대량의 구문 구성 요소들로부터 발생하는 클래스 분포의 불균형 문제를 완화할 수 있고, 학습 비용을 낮출 수 있다.

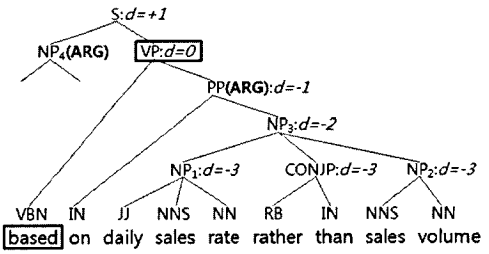

그림 4 트리 거리의 예

그림 4는 구문 트리에서 직사각형으로 표시된 술어 *based*의 부모 노드 *VP*를 기준으로 각 구문 구성 요소의 부모 노드까지의 거리를 나타낸다. 술어의 부모 노드인 *VP*는 트리 거리 값으로 *d=0*을 갖는다. *S* 노드는 *VP*에서 한 번 위쪽으로의 이동이기 때문에 *d=+1*값을 갖고, *PP* 노드는 *VP*에서 한 번 아래쪽으로의 이동이기 때문에 *d=-1*값을 갖는다. 또한 *VP*를 기준으로 아래쪽으로 두 번 이동하면 만나는 노드인 NP_3는 *d=-2*의 값을 갖고, NP_1, *CONJP*, NP_2는 아래쪽으로 세 번 이동하면 만나는 노드들로 *d=-3*의 값을 갖는다. 거리값에서 +, - 기호는 이동 방향을 나타내는 것으로, +는 위쪽으로의 이동을, -는 구문 트리 상에서 아래쪽으로의 이동을 나타낸다.

트리 거리를 측정하기 위해서는 비말단 노드(non-terminal)이면서, 술어의 부모 노드로부터 위쪽 또는 아래쪽으로의 이동으로 도착 가능한 노드들이어야 한다. 그림 4에서 비말단 노드는 NP_1, *CONJP*, NP_2, NP_3, *PP*, *VP*, *S*, NP_4이고 이 중 술어의 부모 노드 *VP*를 기준으로 위쪽 또는 아래쪽으로 한 방향으로만 이동하였을 때 NP_4를 만날 수 없으므로 NP_4를 제외한 7개의 비말단 노드들을 대상으로 트리 제약 조건을 적용하게 된다. 예를 들어, 논항을 찾기 위한 트리 거리 기준이 $-1 \leq d \leq +1$라면 노드 *PP*, *VP*, *S*가 이 조건을 만족하기 때문에 이 노드들의 자식 노드들인 *IN*, NP_3, *PP*, NP_4가 후보 논항들이 된다. 그림 4에서 (ARG)로 표시된 NP_4와 *PP*는 실제로 술어 *based*가 갖는 정답 논항들을 나타낸다.

표 1 제약 조건 $-3 \leq d \leq +1$을 만족하는 후보 논항

거리	노드	후보 논항
$d = +1$	S	NP_4
$d = 0$	VP	PP
$d = -1$	PP	IN, NP_3
$d = -2$	NP_3	NP_1, CONJP, NP_2
$d = -3$	NP_1 CONJP NP_2	JJ, NNS, NN RB, IN NNS, NN

표 1은 그림 4의 노드들을 대상으로 제약 조건 $-3 \leq d \leq +1$을 만족하는 후보 논항들을 보여준다. 술어의 부모 노드인 *VP*로부터의 트리 거리가 제약 조건을 만족하면 그 노드의 자식 노드들을 후보 논항으로 사용한다.

3.3 자질 집합

최대 엔트로피 모형에 기반한 분류기를 학습하기 위해서 다양한 자질들이 사용될 수 있다. 자질들은 관련된 정보에 따라 표 2에 기술한 '술어 관련 자질', 표 3에 기술한 '후보 논항 관련 자질', 표 4에 기술한 '술어 및 후보 논항 관련 자질'로 구분할 수 있다.

표 2의 '술어 관련 자질'의 예는 다음과 같다. 그림 4의 'based'에 대하여 '단어' 자질값은 'based', '품사' 자질값은 'VBN', '구범주' 자질값은 'VP', '유형' 자질값은 'VBN', '태' 자질값은 수동태를 나타내는 '1'(능동태는 0으로 나타냄), '하위 범주화' 자질값은 'VP->VBN+PP', '유형+단어' 자질값은 'VBN+based'이다.

표 3의 '후보 논항 관련 자질'의 예는 다음과 같다. 그림 4의 'based'에 대하여 후보 논항이 'sales volume'일 때 '중심어' 자질값은 'sales', '품사' 자질값은 'NNS', '구범주' 자질값은 'NP', '의미어' 자질값은 'volume', '의미

표 2 술어 관련 자질

자질	설명
단어	술어의 어휘
품사	술어의 품사
구범주	술어의 부모 노드의 구문 범주
유형	술어의 부모 노드에서 첫 단어의 품사를 사용하는 것으로 술어의 부모 노드의 구문 중심어의 품사를 나타냄. 자질값은 *'MD, TO, VB, VBD, VBG, VBN, VBP, VBZ'*이고 동사구의 특성을 표현함. 예) 자질값이 *'TO'*인 경우 동사가 *to-부정사* 형태로 발생함을 나타냄
태	술어가 능동태인지 수동태인지를 나타내는 이진값
하위범주화	구문 트리에서 술어의 부모 노드가 갖는 자식 노드들을 표현하는 구 구조 규칙
유형+단어	술어의 '유형' 자질과 술어의 '단어' 자질을 결합한 것으로 최대 엔트로피 모형이 자질들을 독립적으로 고려하기 때문에 이와 같은 결합 자질을 추가하는 것이 필요함

표 3 후보 논항 관련 자질

자질	설명
중심어	후보 논항인 구문 구성 요소의 중심어로 Collins의 중심어 규칙을 이용하여 추출함[10]
품사	중심어의 품사
구범주	후보 논항인 구문 구성 요소의 구문 범주
의미어	후보 논항인 구문 구성 요소의 의미적 중심어를 의미하는 것으로 Buchholz의 중심어 규칙을 이용하여 추출함[11]
의미품사	의미적 중심어의 품사
지배범주	논항 후보인 구문 구성 요소의 부모 노드

품사' 자질값은 'NN', '지배범주' 자질값은 'NP'이다.

표 4의 '술어 및 후보 논항 관련 자질'의 예는 다음과 같다. 그림 4의 'based'에 대하여 후보 논항이 'sales volume'일 때 '경로' 자질값은 'NP↑NP↑PP↑VP↓VBN', '위치' 자질값은 오른쪽 발생을 나타내는 '1' (왼쪽 발생은 0으로 나타냄), '위치+절' 자질값은 술어의 인접절을 나타내는 '1+0' (첫 번째 상위절은 1로써, 두 번째 상위절은 2로써 나타냄)이다. 후보 논항과 술어 사이의 구문 경로 중 NP, PP의 구문 범주가 존재하기 때문에, '위치+VP' 자질값은 '1+0', '위치+NP' 자질값은 '1+1', '위치+SBAR' 자질값은 '1+0'이다. '위치+CC', '위치+쉼표', '위치+콜론', '위치+따옴표' 자질값은 품사 CC, 쉼표, 콜론, 따옴표가 후보 논항과 술어 사이에 존재하지 않기 때문에 모두 '1+0'이다. 두 개의 자질들을 결합한 형태로써, '단어+중심어' 자질값은 'based+sales', '단어+의미어' 자질값은 'based+volume', '태+지배범주' 자질값은 '1+NP'이다.

표 4 술어 및 후보 논항 관련 자질

자질	논항
경로	구문 트리상에서 구문 구성 요소로부터 술어까지의 이동 경로
위치	대상 구문 구성 요소가 술어의 왼쪽에서 발생하는지 오른쪽에서 발생하는지를 나타냄
위치+절	위치 자질과 절 인식 결과를 결합한 것으로 대상 구문 구성 요소가 술어의 인접절 또는 첫 번째 상위절, 두 번째 상위절 등에 위치하는지를 나타냄
위치+VP 위치+NP 위치+SBAR	대상 구문 구성요소와 술어 사이에 존재하는 동사구(Verb Phrase, VP), 명사구(Noun Phrase, NP), 종속 접속사(SBAR)의 수를 나타냄. 두 기본구 사이에서 발생하는 동사구 등의 수가 증가할수록 의존 관계가 성립할 가능성은 낮아짐
위치+CC 위치+쉼표 위치+콜론 위치+따옴표	대상 구문 구성 요소와 술어 사이에 존재하는 품사인 등위 접속사(Coordinating Conjunction, CC), 쉼표, 콜론, 큰따옴표의 수를 나타냄. 이전 자질과 마찬가지로 쉼표 등의 수가 증가할수록 두 기본구 사이에 의존 관계가 존재할 확률이 낮아짐.
단어+중심어 단어+의미어 태+지배범주	기존의 자질들을 결합한 정보

4. 실 험

4.1 실험 말뭉치

본 논문에서 사용하는 실험 말뭉치는 PropBank[1]에 기반한 CoNLL-2005 shared task[2] 데이터 집합이다 [12]. 이 실험 말뭉치는 문장을 구성하는 요소가 주어, 목적어 또는 보어로 발생하였는지에 상관없이 문장에서 술어와 갖는 의미적 관계만을 고려하여 술어의 논항 및 그것의 의미적 역할을 할당한 것이다. 의미역의 종류는 4.2.5절의 표 10과 같고 세가지 집합으로 나눌 수 있다. 먼저, A0부터 A5까지는 주로 행위의 주체, 객체, 도구 등을 나타내는 것으로 동사의 의미에 따라 각 태그가 의미하는 바가 다르다. 이 말뭉치는 동사의 의미마다 각 태그의 의미를 정의한 하위범주화(sub-categorization) 프레임을 가지고 있다. 예를 들어, 동사 'pass'가 '법안을 가결하다'란 의미일 때는 A0가 'legislative body'로, '추월하다'란 의미일 때는 A0가 'entity moving ahead'로 정의되어 있다. 다음으로, AM으로 시작하는 태그들은 동사와 독립적이고, 논항이 시간, 장소, 방법 등을 기술함을 나타낸다. 마지막으로, R로 시작하는 태그들은 논항이 관계대명사와 같은 형태로 발생함을 나타낸다. 논항의 의미적 역할은 R 다음에 나오는 태그들이 표현한다.

최대 엔트로피 모형을 적용하기 위하여 Zhang Le의 MaxEnt[3] 툴킷을 사용하였고 Gaussian prior smoothing과 함께 L-BFGS 파라미터 추정 알고리즘을 적용하였다[13]. 실험 말뭉치는 여러 개의 section으로 구성되어 있는데 관례에 따라, section 02부터 21까지를 학습 집합(training set)으로, section 23을 테스트 집합(test set)으로, section 24를 개발 집합(development set)으로 사용하였다. 개발 집합을 이용하여 전처리 효과(4.2.1절) 및 자질 기여도 평가(4.2.2절)를 수행하였다. 4.2.3절의 각 단계의 성능 및 4.2.4절의 한 단계 부분 의미 분석 방법과 두 단계 부분 의미 분석 방법의 비교 실험도 개발 집합에서 측정하였다. 테스트 집합은 부분 의미 분석 시스템 성능을 측정하기 위해서만 사용하였다.

4.2 실험 결과

본 논문은 부분 의미 분석 시스템의 성능을 확인하기 위하여 *srl-eval.pl*[4] 프로그램을 사용한다. 이 프로그램은 시스템의 결과가 말뭉치의 정답과 일치한 수를 확인

1) Penn Treebank II에 술어의 논항 및 의미역 정보를 추가한 것으로 논항의 구문적 위치(syntactic position)에 상관없이 의미역을 할당함
2) http://www.lsi.upc.edu/~srlconll/
3) http://homepages.inf.ed.ac.uk/lzhang10/maxent.html
4) http://www.lsi.upc.es/~srlconll/srl-eval.pl

하여 부분 의미 분석의 성능을 정확률(precision), 재현율(recall), $F_{\beta=1}$로 표기한다. 시스템이 찾은 하나의 논항에 대하여 말뭉치의 정답과 시작/끝 경계 및 의미역이 일치하면 시스템은 올바른 결과를 찾은 것이다. 각 측정치가 의미하는 바는 다음과 같다. 정확률은 시스템의 결과가 얼마나 정확한지를 나타내는 것으로 시스템과 말뭉치가 일치한 수를 시스템이 찾은 논항 수로 나눈 것이다. 재현율은 말뭉치의 정답을 얼마나 발견했는지를 나타내는 것으로 시스템과 말뭉치가 일치한 수를 말뭉치의 논항 수로 나눈 것이다. $F_{\beta=1}$은 동일한 가중치를 적용하여 반비례 관계인 정확률과 재현율을 통합한 것이다.

$$\text{정확률(P)} = \frac{\text{시스템과 말뭉치가 일치한 수}}{\text{시스템이 찾은 논항 수}}$$

$$\text{재현률(R)} = \frac{\text{시스템과 말뭉치가 일치한 수}}{\text{말뭉치의 논항 수}}$$

$$F_{\beta=1} = \frac{2PR}{P+R}$$

부분 의미 분석의 성능뿐만 아니라 그것을 구성하는 각 단계의 성능을 나타내기 위하여 다음과 같은 측정치를 사용한다. 먼저, 논항 확인 단계의 성능은 전체 성능처럼 정확률(precision), 재현율(recall), $F_{\beta=1}$로 표기한다. 시스템으로부터 얻은 논항의 시작/끝 경계가 말뭉치 정답과 일치하는지를 비교한다. 다음으로 논항 분류 단계의 성능은 정확도(accuracy)로 표기한다. 논항은 이미 주어지고 의미역 분류 결과가 말뭉치 정답과 일치하는지를 확인한다. 정확도는 시스템과 말뭉치가 일치한 수를 말뭉치의 논항 수로 나눈 것으로 전체 논항 중 적절한 의미역을 할당 받은 논항의 비율을 나타낸다.

4.2.1 전처리 효과

구문 분석기가 생성한 모든 구문 구성 요소에 대해서 논항인지 여부를 확인하는 것이 아니라 논항일 가능성이 높은 구문 구성 요소만을 선별하여 분류기를 학습하고 테스트하기 위하여 절 경계 및 트리 거리를 제약 조건으로 사용하여 논항을 구성할 가능성이 희박한 구문 구성 요소들을 제거하는 전처리 방법을 3.2절에서 제안하였다. 이 전처리의 효과를 보이기 위하여 실험 말뭉치의 개발 집합에서 표 5와 같은 실험을 수행하였다. 이것은 전처리 과정을 통하여 후보 논항들이 감소된 비율과 그에 따른 성능 변화를 나타낸다.

표 5의 각 열이 의미하는 바는 다음과 같다. 첫 번째 열 '제약'은 절 경계 및 트리 거리 제약 조건을 의미한다. 두 번째 열부터 다섯 번째 열까지는 학습 말뭉치에서 측정한 값이다. 두 번째 열 '#후보'는 각 제약 조건을 만족하는 후보 논항의 수를 나타내고, 세 번째 열 '%후보'는 총 후보 논항의 수가 '기준선 (다)'의 1,598,726개일 때 각 조건을 만족하는 후보 논항의 비율이다. 네 번째 열 '#논항'은 각 제약 조건을 만족하는 말뭉치 논항의 수를 나타내고, 다섯 번째 열 '%논항'은 학습 말뭉치에 존재하는 논항의 총 수를 기준으로 해당 제약 조건을 만족하는 논항의 비율을 나타낸다. 마지막 열 '$F_{\beta=1}$'은 구문 트리 경로만을 자질로 사용하여 논항 확인 단계를 수행한 결과이다.

표 5 전처리에 따른 후보 논항의 감소 정도 및 성능 변화

제약	#후보	%후보	#논항	%논항	$F_{\beta=1}$
기준선					
(가)	3,709,080		233,394	96.06	79.37
(나)	2,579,278		233,004	95.90	79.52
(다)	1,598,726	100.00	231,120	95.13	79.92
절 경계 제약					
1/0	1,303,596	81.54	222,238	91.47	78.97
1/1	1,370,760	85.74	223,571	92.02	79.14
2/0	1,403,630	87.80	228,891	94.21	79.66
2/1	1,470,794	92.00	230,224	94.76	79.89
3/0	1,439,755	90.06	229,548	94.48	79.63
3/1	1,506,919	94.26	230,881	95.03	79.79
트리 거리 제약					
6/1	804,413	50.32	226,875	93.38	80.17
6/2	936,021	58.55	227,637	93.69	79.94
7/1	**842,453**	**52.70**	**228,129**	**93.90**	**80.44**
7/2	974,061	60.93	228,891	94.21	80.03
8/1	871,541	54.51	228,795	94.17	80.24
8/2	1,003,149	62.75	229,557	94.48	80.04
절 경계 제약 및 트리 거리 제약					
2/1,7/1	786,951	49.22	227,523	93.65	80.12
2/1,8/1	803,040	50.23	228,081	93.88	80.11
3/1,7/1	800,740	50.09	227,947	93.82	80.28
3/1,8/1	822,225	51.43	228,599	94.09	80.06

표 5는 '기준선', '절 경계 제약', '트리 거리 제약', '절 경계 제약 및 트리 거리 제약'의 네 부분으로 나누어져 있다. 이 중 '기준선'은 '(가)', '(나)', '(다)'의 세 가지 실험의 결과이다. '(가)'의 실험은 기존의 품사 부착기[6] 및 절 인식기[7]뿐만 아니라 Charniak 구문 분석기[8]의 결과로 얻어진 모든 구문 노드들을 후보 논항으로 사용하였다. 이 때 '%논항' 값이 96.06%라는 것은 말뭉치에 존재하는 논항들 중 약 4%가 구문 분석기의 결과로부터 추출한 후보 논항들 중에서 시작/끝 경계가 일치하는 것이 없다는 것을 나타낸다. 이것은 구문 분석기의 오류에 기인한다. '(가)'의 실험을 통하여 후보 논항의 수 3,709,080개 중 논항과 경계가 일치하는 긍정 학습 예제 233,394개를 제외한 나머지 대부분의 부정 학습 예제가 압도적인 분포를 차지함을 알 수 있다.

'(나)' 실험은 특정 노드의 자식 노드들이 품사로만 이루어진 경우 그 노드를 배제하는 작업을 수행하였다. 구문 트리에서 노드가 품사만을 자식 노드로 갖는 경우 각 자식 노드가 논항일 가능성이 희박하다고 가정하여 이와 같은 전처리를 수행하였다. 그 결과 약 1,130,000

개의 후보 논항이 제거되었으나 이 중 390개의 논항도 포함되었다. '(나)' 실험의 제약 조건은 '(다)' 실험과 나머지 세 부분에도 적용되었다.

'(다)' 실험에서는 술어의 부모 노드를 기준으로 위쪽으로만 이동하였을 때 만나는 노드들, 아래쪽으로만 이동하였을 때 만나는 노드들을 수집하여 그것들의 자식 노드들만을 후보 논항으로 고려하였다. 즉 술어의 부모 노드와 트리 거리를 측정할 수 있는 노드들의 자식 노드들만을 후보 논항으로 간주하였다. 나머지 세 부분의 '%후보' 값은 '(다)' 실험의 후보 논항의 총 수 1,598,726개를 분모로 하고, 각 조건을 만족하는 후보 논항의 수를 분자로 했을 때의 결과이다. 이것을 통하여 학습 예제가 감소한 정도를 알 수 있다.

절 경계 제약은 절 인식 결과를 바탕으로 해당 술어의 인접절 및 상위절들을 확인한 후 특정 범위 내에서 논항 확인을 수행하는 것이다. 제약 조건은 두 개의 숫자로 표기하는 데 첫 번째 숫자는 논항 확인의 왼쪽 경계를 두 번째 숫자는 오른쪽 경계를 나타낸다. 숫자 0은 술어의 인접절을, 숫자 1은 술어의 첫 번째 상위절을, 숫자 2는 술어의 두 번째 상위절을 나타낸다. 예를 들어, 제약 조건 '2/1'은 술어의 두 번째 상위절의 왼쪽 경계부터 술어의 첫 번째 상위절의 오른쪽 경계 안에서 논항 확인을 수행하였음을 나타낸다. 절 인식 결과로부터 술어가 영향을 미치는 범위를 제한하고자 하였으나 트리 거리 제약에 비하여 전체적으로 '#후보' 값 및 '$F_{\beta=1}$' 값에서 좋지 않은 결과를 보였다. 가장 좋은 성능을 보인 제약 조건 '2/1'의 경우 $F_{\beta=1}$값 기준 79.89%의 논항 확인 성능을 보였고, 후보 논항의 비율이 8% 정도 감소하였다.

트리 거리 제약은 술어의 논항은 대부분 특정 트리 거리 내에 존재한다고 가정하여 논항 확인 작업을 수행하는 것이다. 절 경계 제약과 마찬가지로 두 개의 숫자에 의하여 제약 조건이 표시되는데 첫 번째 숫자는 술어의 부모 노드를 기준으로 위쪽 방향으로의 이동 거리를, 두 번째 숫자는 아래쪽 방향으로의 이동 거리를 나타낸다. 예를 들어, 제약 조건 '7/1'은 술어의 부모 노드를 기준으로 위쪽으로 7번 아래쪽으로 1번 이동하면서 만나는 노드들의 자식 노드들을 후보 논항으로 간주하는 것이다. 트리 거리 제약을 사용하지 않은 기준선 '(다)'에 비하여 제약을 적용한 경우 후보 논항의 비율 및 논항 확인의 성능 면에서 전체적으로 좋은 결과를 보였다. 가장 좋은 성능을 보인 제약 조건 '7/1'은 '%논항' 값이 약 2% 정도 떨어졌으나, 총 후보 논항의 약 47.3%를 제거하였고, $F_{\beta=1}$값 기준 80.44%의 논항 확인 성능을 보였다. 4.2.2절부터는 실험을 통하여 얻은 트리 거리 제약 조건 '7/1'을 적용하여 성능을 측정하였다. 이 조건은 말뭉치에 존재하는 논항 중 93.9%를 포함하기 때문에 이 값은 시스템이 얻을 수 있는 재현율의 상계(upper bound)이다.

표 5의 마지막 부분은 절 경계 제약 및 트리 거리 제약을 결합하여 전처리를 수행한 것이다. 예를 들어, 제약 조건 '3/1,7/1'은 트리 거리 제약 조건 '7/1'을 만족하는 후보 논항들 중 술어의 세 번째 상위절의 왼쪽 경계와 첫 번째 상위절의 오른쪽 경계 안에 존재하는 것만을 고려하는 방법이다. 실험 결과, 두 가지 제약 조건을 결합하여 사용한 경우 트리 거리 제약을 능가하는 성능을 보이지 못했다.

4.2.2 자질 기여도 평가

3.3절에서 기술한 각 자질들이 논항 확인 단계 또는 논항 분류 단계의 성능 향상에 기여하는지를 확인하기 위하여 표 6과 같은 실험을 개발 집합에서 수행하였다. 여기서 기준선 '전체'는 제안한 모든 자질을 사용하였을 때의 성능이고 나머지 '술어 관련 자질', '후보 논항 관련 자질' 및 '술어 및 후보 논항 관련 자질' 부분은 자질

표 6 개발 집합에서 각 자질의 효과

자질	논항 확인			논항 분류
	정확률	재현율	$F_{\beta=1}$	정확도
기준선				
전체	82.57	78.41	80.44	86.00
술어 관련 자질				
전체 - 단어	82.80	77.78	80.21	84.93
전체 - 품사	83.40	76.72	79.92	85.95
전체 - 구범주	83.11	77.57	80.24	85.87
전체 - 유형	82.76	77.91	80.26	**85.99**
전체 - 태	82.87	77.88	80.30	85.88
전체 - 하위범주화	82.48	77.68	80.00	84.88
전체 - 유형+단어	83.20	77.40	80.20	85.62
후보 논항 관련 자질				
전체 - 중심어	82.58	77.87	80.16	85.61
전체 - 품사	82.66	77.88	80.20	85.89
전체 - 구범주	83.52	76.82	80.03	85.81
전체 - 의미어	82.57	77.87	80.15	85.64
전체 - 의미품사	82.65	77.92	80.22	**86.09**
전체 - 지배범주	82.69	78.34	**80.46**	85.91
술어 및 후보 논항 관련 자질				
전체 - 경로	78.39	67.96	72.80	85.69
전체 - 위치	82.70	77.74	80.14	85.85
전체 - 위치+절	82.94	78.34	**80.57**	**86.19**
전체 - 위치+VP	82.69	77.87	80.20	85.87
전체 - 위치+NP	82.78	77.69	80.15	85.77
전체 - 위치+SBAR	82.51	78.00	80.19	85.83
전체 - 위치+CC	82.84	78.10	80.40	85.70
전체 - 위치+쉼표	82.78	77.69	80.15	85.70
전체 - 위치+콜론	82.67	77.96	80.25	85.72
전체 - 위치+따옴표	82.63	77.98	80.24	85.66
전체 - 단어+중심어	82.62	77.71	80.09	84.98
전체 - 단어+의미어	82.72	77.79	80.18	85.24
전체 - 태+지배범주	82.93	77.81	80.29	85.85

집합에서 하나의 자질을 제거하였을 때 성능이 감소한 정도를 나타낸다. 자질을 제거하였을 때 성능 하락의 폭이 클수록 중요한 자질이다. 표 6의 첫 번째 행에 제시한 것처럼 전체 자질 집합에서 특정 자질을 제거하고 실험하였을 때 논항 확인의 성능은 정확률, 재현율, $F_{\beta=1}$ 값으로 나타내었고, 논항 분류의 성능은 정확도로 나타내었다. 각 단계는 다른 일을 수행하기 때문에 각 단계에 유용한 자질 집합은 다를 수 있다. 본 논문은 직관에 의해서가 아니라 표 6의 실험 결과를 바탕으로 각 단계에 적절한 자질 집합을 구성하였다.

논항 확인 단계에 제안된 모든 자질을 사용하였을 때 80.44%의 $F_{\beta=1}$ 값을 보였다. 실험 결과 전체 자질 집합에서 '경로' 자질을 제거하였을 때 $F_{\beta=1}$ 값이 가장 크게 떨어졌고 이 자질이 논항 확인의 성능에 가장 큰 영향을 미침을 알 수 있었다. 성능 하락의 폭으로 확인한 중요한 5개의 자질을 나열하면 다음과 같다. '경로', 술어의 '품사', '하위범주화', 후보 논항의 '구범주', '단어+중심어' 등이다. 반면에, 모든 자질을 사용했을 때와 비교하여 성능 변화가 없거나 오히려 성능이 올라간 경우가 있었다. 논항 확인 단계에 '위치+절'과 '지배범주' 자질을 제거하였을 때 그러한 결과를 보였다. 이 자질들은 자질 집합을 구성할 때 유용할 것이라는 처음의 가정과 다르게 이 자질들을 제거하였을 때 논항 확인 성능이 증가하였다. 이후의 실험에서는 이 자질들을 제외하고 논항 확인 단계의 최종 자질 집합을 구성하였고 그 결과는 표 7에 제시된 것처럼 80.59%의 $F_{\beta=1}$ 값을 보였다.

논항 분류 단계에 모든 자질을 적용하였을 때 86.00%의 정확도를 보였다. 자질 집합에서 하위 범주화 정보를 제외시켰을 때 정확도가 86.00%에서 84.88%로 가장 크게 하락하였다. 성능 하락의 정도로 논항 분류 단계에 유용한 5가지 자질을 나열하면 다음과 같다. '하위범주화', 술어의 '단어', '단어+중심어', '단어+의미어', 후보 논항의 '중심어' 등이다. 앞선 논항 확인 단계에서 가장 중요한 자질이었던 '경로' 자질은 이 안에 포함되지 않았다. 즉 논항 확인 단계에서 보였던 만큼의 중요도를 나타내지 않았다. 성능 하락의 정도로 각 단계마다 선정한 5가지 중요한 자질들을 보면 각 단계에 유용한 자질들이 서로 상이함을 알 수 있다. 두 단계에서 중요한 자질들 중 겹치는 것은 '하위범주화'와 '단어+중심어' 자질뿐이었다. 논항 확인 단계에서는 '경로', 술어의 '품사', 후보 논항의 '구범주'처럼 구문 자질들이 많이 선정된 반면에 논항 분류 단계에서는 술어의 '단어', '단어+의미어', 후보 논항의 '중심어'처럼 어휘 자질들이 주로 추출되었다. 논항 확인 단계와 마찬가지로 논항 분류 단계에서도 성능에 좋지 않은 영향을 미치는 자질들이 있었다. '위치+절', '의미품사', '유형' 자질이 그러하였다. 논항 분류를 위한 최종 자질 집합은 이 세 개의 자질을 제거하고 구성하였고 정확도는 표 7에 제시된 것처럼 87.16%를 보였다.

표 7 개발 집합에서 각 단계의 성능

단계	정확률	재현율	$F_{\beta=1}$	정확도
논항 확인	82.56	78.72	80.59	
논항 분류				87.16

4.2.3 각 단계의 성능

표 7은 개발 집합에서 '논항 확인' 단계와 '논항 분류' 단계의 성능을 나타낸다. 이것은 각 단계의 독립적인 성능을 제시하는 것으로 '논항 확인' 단계에서는 논항의 시작/끝 경계를 찾는 일만을 수행한 결과이고, 논항 분류 단계에서는 논항의 시작/끝 경계가 입력으로 주어졌을 때 의미역 할당의 정확도만을 측정한 것이다. 실험 결과, 논항 확인 단계는 80.59%의 $F_{\beta=1}$ 값을 보였고 논항 분류 단계는 87.16%의 정확도를 나타내었다.

4.2.4 두 단계 부분 의미 분석 방법의 효과

표 8 한 단계 방법과 두 단계 방법의 성능 비교

방법	정확률	재현율	$F_{\beta=1}$
한 단계	71.94	68.70	70.29
두 단계	72.68	69.16	70.87

본 논문은 클래스 분포의 불균형 문제를 완화하기 위하여 두 단계 부분 의미 분석 방법을 제안하였다. 그런데 항상 순차적으로 수행하는 두 단계 부분 의미 분석 방법이 논항 확인 단계와 논항 분류 단계를 통합하여 수행하는 한 단계 부분 의미 분석 방법보다 좋은 것은 아니다. 두 단계 방법의 경우 첫 단계의 오류가 다음 단계로 전파되는 문제가 있다. 따라서 실험을 통하여 두 단계 부분 의미 분석 방법의 효과를 확인하는 것이 필요하다. 표 8은 한 단계 부분 의미 분석 방법과 두 단계 부분 의미 분석 방법을 실험적으로 비교한 것이다. 실험을 위하여 두 단계 방법에 적용한 모든 자질을 사용하여 한 단계 방법을 수행하였다. 최종적인 실험 결과, 두 단계로 부분 의미 분석을 수행하는 것이 논항 확인 단계와 논항 분류 단계를 통합하여 한 단계로 부분 의미 분석을 수행하는 것보다 좋은 성능을 보였다.

4.2.5 부분 의미 분석 시스템 성능 평가

표 9는 부분 의미 분석의 성능을 측정하기 위하여 4가지 실험 집합을 사용한 결과이다. '개발 집합'과 '테스트 집합'은 본 논문의 실험 말뭉치인 PropBank의 일부인 반면에, 'Brown 말뭉치[5]'는 학습 말뭉치와는 다른

5) http://icame.uib.no/brown/bcm.html

표 9 여러 실험 말뭉치에서 부분 의미 분석의 성능

실험 말뭉치	정확률	재현율	$F_{\beta=1}$
개발 집합	72.68	69.16	70.87
테스트 집합	74.69	70.78	72.68
Brown 말뭉치	64.58	60.31	62.38
테스트+Brown	73.35	69.37	71.31

종류의 말뭉치로써 시스템이 이질적인 말뭉치에서도 견고성을 보이는지를 측정하기 위하여 사용하였다. '테스트+Brown'은 '테스트 집합'과 'Brown 말뭉치'를 함께 사용한 것으로 전자가 후자에 비해 대량의 말뭉치이기 때문에 '테스트 집합' 성능이 전체 성능을 지배하였다. 본 시스템은 Wall Street Journal 기사들로 이루어진 PropBank의 테스트 집합에서 정확률 74.69%, 재현율 70.78%, $F_{\beta=1}$ 값 기준 72.68%의 부분 의미 분석 성능을 보였다. 'Brown 말뭉치'에서는 '테스트 집합'보다 $F_{\beta=1}$값 기준 약 10% 정도 낮은 부분 의미 분석 성능을 보였다.

표 10은 테스트 집합에서 각 의미역별 부분 의미 분석의 성능을 나타낸다. 학습 말뭉치에서 고빈도로 발생하는 행위의 주체를 나타내는 A0와 행위의 객체를 나타

표 10 테스트 집합에서 각 의미역의 부분 의미 분석 성능

의미역	정확률	재현율	$F_{\beta=1}$
A0	85.02	81.53	83.24
A1	73.98	72.25	73.11
A2	63.20	57.57	60.25
A3	62.96	49.13	55.19
A4	73.40	67.65	70.41
A5	100.00	40.00	57.14
AM-ADV	56.73	50.00	53.15
AM-CAU	70.21	45.21	55.00
AM-DIR	46.48	38.82	42.31
AM-DIS	70.95	65.62	68.18
AM-EXT	87.50	43.75	58.33
AM-LOC	44.09	46.28	45.16
AM-MNR	55.56	52.33	53.89
AM-MOD	97.59	95.64	96.61
AM-NEG	96.05	95.22	95.63
AM-PNC	40.68	41.74	41.20
AM-PRD	50.00	20.00	28.57
AM-REC	0.00	0.00	0.00
AM-TMP	70.11	61.73	65.66
R-A0	84.68	83.93	84.30
R-A1	73.33	70.51	71.90
R-A2	50.00	31.25	38.46
R-A3	0.00	0.00	0.00
R-A4	0.00	0.00	0.00
R-AM-ADV	0.00	0.00	0.00
R-AM-CAU	100.00	25.00	40.00
R-AM-EXT	0.00	0.00	0.00
R-AM-LOC	85.71	57.14	68.57
R-AM-MNR	16.67	16.67	16.67
R-AM-TMP	72.50	55.77	63.04

5. 결 론

본 논문은 부분 의미 분석 문제를 풀기 위하여 최대 엔트로피 분류기를 활용한 두 단계 방법을 제안하였다. 부분 의미 분석을 논항 확인 단계와 논항 분류 단계로 구분하고 전체 성능에 결정적인 영향을 미치는 논항 확인 단계의 성능 및 학습 효율을 개선하기 위하여 전처리 방법을 제안하였다. 절 경계 제약 및 트리 거리 제약을 통하여, 후보 논항들 중 제약 조건을 만족하는 후보 논항들만을 대상으로 논항의 경계를 찾는 작업을 수행하였다. 실험 결과 트리 거리를 이용한 전처리가 효과적이었다. 각 제약 조건의 효과 및 두 가지 제약 조건을 결합하였을 때의 효과를 측정하였는데 트리 거리를 제약 조건으로 사용하였을 때 성능 및 후보 논항의 제거 면에서 가장 효과적이었다. 전처리 과정 없이 모든 구문 분석기의 결과를 사용하였을 때보다 약 1% 이상의 성능 향상이 있었다. 본 논문은 전처리의 효과뿐만 아니라 다양한 실험 결과를 제시하였다. 제안한 자질들과 분류기를 적용하였을 때, 부분 의미 분석을 위한 한 단계의 통합 방법보다 두 단계의 순차 방법을 사용하는 것이 더 좋은 성능을 보임을 실험적으로 증명하였다. 또한 각 단계에 최적화된 자질 집합을 구성하기 위하여 각 자질이 각 단계에 미치는 영향을 평가하고 각 단계의 최종 성능을 제시하였다. 마지막으로, 실험 말뭉치의 일부뿐만 아니라 이질적인 말뭉치에 대한 부분 의미 분석의 성능을 제시하였다. 향후 연구 과제로 일반 문서뿐만 아니라 바이오 문서들에 부분 의미 분석 방법을 적용하여, 자연어 처리로부터 추출한 정보들이 바이오 문서의 정보 추출에 기여하는 정도를 보일 것이다.

참 고 문 헌

[1] M. Surdeanu, S. Harabagiu, J. Williams, and P. Aarseth, "Using Predicate Arguments Structures for Information Extraction," *In Proceedings of the Association for Computational Linguistics*, 2003.

[2] D. Gildea and D. Jurafsky, "Automatic Labeling of Semantic Roles," *Computational Linguistics*, vol.28, no.3, pp.1-45, 2002.

[3] N. Xue and M. Palmer, "Calibrating Features for Semantic Role Labeling," *In Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2004.

[4] N. Kwon, M. Fleischman, and E. Hovy, "FrameNet-based Semantic Parsing Using Maximum Entropy Models," *In Proceedings of the International Conference on Computational Linguistics*, 2004.

[5] S. Pradhan, W. Ward, K. Hacioglu, J. Martin, and D. Jurafsky, "Semantic Role Labeling Using Different Syntactic Views," *In Proceedings of the*

Association for Computational Linguistics, 2005.
[6] J. Gimenez and L. Marquez, "Fast and Accurate Part-of-Speech Tagging: The SVM Approach Revisited," *In Proceedings of the Recent Advances in Natural Language Processing*, 2003.
[7] X. Carreras and L. Marquez, "Phrase Recognition by Filtering and Ranking with Perceptrons," *In Proceedings of the Recent Advances in Natural Language Processing*, 2003.
[8] E. Charniak, "A Maximum-Entropy-Inspired Parser," *In Proceedings of the North American Chapter of the Association for Computational Linguistics*, 2000.
[9] A. Berger, S. Pietra, and V. Pietra, "A Maximum Entropy Approach to Natural Language Processing," Computational Linguistics, vol.22, no.1, pp.39-71, 1996.
[10] M. Collins, "Head-Driven Statistical Models for Natural Language Parsing," *PhD Dissertation*, University of Pennsylvania, 1999.
[11] S. Buchholz, "Memory-Based Grammatical Relation Finding," *PhD Thesis*, Tilburg University, 2002.
[12] X. Carreras and L. Marquez, "Introduction to the CoNLL-2005 Shared Task: Semantic Role Labeling," *In Proceedings of the Eighth Conference on Natural Language Learning*, 2005.
[13] S. Chen and R. Rosenfeld, "A Gaussian Prior for Smoothing Maximum Entropy Models," *Technical Report CMUCS-99-108*, Carnegie Mellon University, 1999.

박 경 미

1998년 연세대학교 식품영양학과 졸업(학사). 2000년 연세대학교 정보산업공학 졸업(학사). 2002년 연세대학교 컴퓨터산업시스템공학과 졸업(석사). 2008년 고려대학교 컴퓨터학과 졸업(박사). 2009년~현재 숭실대 컴퓨터학부 전임연구원. 관심분야는 Natural Language Processing, Text Mining

황 규 백

1997년 서울대학교 컴퓨터공학과 졸업(학사). 1999년 서울대학교 컴퓨터공학과 졸업(석사). 2005년 서울대학교 컴퓨터공학부 졸업(박사). 2003년 12월~2004년 6월 Harvard Medical School Children's Hospital Informatics Program 객원연구원. 2005년 8월~2006년 2월 서울대학교 컴퓨터연구소 박사후연구원. 2006년 3월~현재 숭실대학교 전임강사/조교수. 관심분야는 Machine Learning, Bioinformatics, Natural Language Processing