工學碩士學位論文

반지도 학습법을 이용한 한국어 개체명 인식

2013年　**12**月

昌原大學校　大學院

컴 퓨 터　工 學 科

金　　柱　　根

工學碩士學位論文

반지도 학습법을 이용한 한국어 개체명 인식

Korean **Named Entity Recognition Using Semi-Supervised Learning**

指導敎授 車揖遠

이 論文을 工學碩士學位論文으로 提出함.

2013年 **12**月

昌原大學校 大學院

컴 퓨 터 工 學 科

金 柱 根

金柱根의 碩士學位 論文을 認准함.

審 査 委 員 長 이 종 근 ㊞

審 査 委 員 김 한 경 ㊞

審 査 委 員 차 정 원 ㊞

2013年 **12**月 日

昌 原 大 學 校 大 學 院

목차

그림 목차

표 목차

제 Ⅰ 장

서론

빅데이터, 시멘틱웹 등에 대한 연구가 활발히 진행되고 있다. 인터넷에 있는 수많은 데이터를 이용하는 이러한 연구에서 출발점이 되는 것은 인터넷에 있는 데이터를 자동으로 인식하는 것이다.

영어권에서는 개체명 인식에 대한 연구가 활발히 진행되어[1,2,3,7] 온톨로지 구성, 의미분석 등에 이용되고 있다. 하지만 한국어의 경우는 오랜 연구에도 불구하고 응용에 사용할 수 있는 개체명 인식기가 없는 상황이다. 또한 지금까지 개발된 개체명 인식기[5,6]는 인명, 지명, 조직명과 같은 정해진 범주만을 인식하기 때문에 다양한 응용에 사용하는데 부족함이 많다. 예를 들어 정보추출과 같은 분야에서 '지진'에 대

한 정보를 추출할 경우, 진앙, 피해액, 진도 등을 추출하려고 하면 일단 기존의 개체명 인식기로 추출한 후에 사상(寫像, mapping)을 통해서 추출해야 한다. 그러나 이러한 과정에서 오류가 증가한다.

가) 1985년에 멕시코에서 진도 8.1의 지진이 발생했다.

문장 가)에서 ‘8.1’ 이라는 ‘진도’를 추출하려고 한다면 보통의 경우 숫자의 패턴의 이용할 것이다. 하지만 패턴을 이용하게 되면 '8.1' 이라는 개체는 추출할 수 있겠지만 이 개체가 의미하는 범주를 파악할 수 는 없다.

나) 윈도우 8.1의 미라캐스트 기능을 지원한다고 5일 밝혔다.

문장 나)에서 보면 '8.1'이라는 개체가 특정 소프트웨어의 버전을 나타내는 숫자로 쓰였다. 이처럼 '8.1'이라는 개체가 버전을 표시할 수 도 있고 온도일 수 도 있으며, 또는 8월 1일이라는 날짜를 나타낼 수 도 있다. 이 문제를 해결하기 위해서 본 연구에서는 다양한 자질(feature)를 도입하였다. 그 중에서 가장 중점을 두는 것은 구문 자질이다.

<표 Ⅰ.1> 문장 가)의 구문분석 결과

번호	구문	구문태그	수식번호
1	1985년에	NP_AJT	6
2	멕시코에서	NP_AJT	6
3	진앙	NP	4
4	8.1의	NP_MOD	5
5	지진이	NP_SBJ	6
6	발생했다.	VP	6

<표 Ⅰ.2> 문장 나)의 구문분석 결과

번호	구문	구문태그	수식번호
1	윈도우	NP	2
2	8.1의	NP_MOD	4
3	미라캐스트	NP	4
4	기능을	NP_OBJ	5
5	지원한다고	VP	7
6	5일	NP_AJT	7
7	밝혔다.	VP	7

구문분석이란 문장 안에서 문장성분들의 관계를 찾아주는 것이다. <표 Ⅰ.1> 은 가)의 문장이 구문분석 된 결과이다. <표 Ⅰ.1> 에서 '8.1'의 구문의 수식번호가 5번

이라는 것을 볼 수 있다. ‘8.1’이라는 구문이 ‘지진이’라는 구문을 수식하는 것을 볼 수 있으며 지배소 동사구가 '발생했다.' 라는 것을 알 수 있다.

<표 Ⅰ.2>에서는 '8.1'이라는 구문이 '기능을'이라는 구문을 수식하며 가장 가까운 지배소 동사구도 '지원한다고'라는 것을 알 수 있다. 위와 같이 구분분석을 이용하여 특정 범주에 의존적인 어절이나 동사구를 찾아낸다면 '8.1'이라는 개체가 우리가 원하는 범주인지 아닌지 분석 할 수 있다.

본 논문에서는 사용자가 지정하는 범주를 찾아주는 개체명 인식기를 제안한다. 즉, 진앙를 선택하고자 하는 경우 이를 학습하여 진앙만을 선택해주는 개체명 인식기를 제안한다. 여기서 또 다른 문제가 발생한다. 사용자가 지정하는 범주는 기존에 나와있는 개체명 인식기[1,2,3,5,6,7]의 기본적인 인명, 지명, 조직명 등이 아니기 때문에 코퍼스와 사전을 구축하는데 시간과 비용이 많이 발생한다. 그래서 본 논문에서는 초기 씨앗 예제를 이용하여 학습코퍼스를 확장시켜 나가는 반지도 학습법(Semi-supervised learning)을 이용하여 코퍼스 구축하는데 비용과 시간을 줄였고, 위키피디아를 이용한 사전구축방법을 제안하였다.

본 논문의 구성은 다음과 같다. Ⅱ장에는 영어, 한국어, 다국어로 개발된 개체명 인식 시스템에 대해서 시스템의 특징과 성능을 알아 본다. Ⅲ장에서는 제안 시스템의 구조도와 특징에 대해서 기술한다. Ⅳ장에서는 다양한 실험을 통해서 시스템을 평가하고 분석하며 끝으로 Ⅴ장에서는 결론과 향후 과제를 다룬다.

제 Ⅱ 장

관련 연구

본 장에서 개체명 인식시스템의 관련연구를 살펴본다.

영어권에서는 오래 전부터 많은 개체명 인식에 대한 연구가 진행되었다. 초기 연구는 변형된 HMM(Hidden Markov Model)을 이용하여 8개의 범주(사람, 단체, 지역, 시간, 날짜, 백분율, 금액, NOT-A-NAME)에 대하여 개체명을 부착 하였다[1]. 이 연구에서 사용된 자질은 문자의 특징을 이용한 자질(첫 번째 문자가 대문자 이거나 단어의 전체 문자가 대문자인 것)을 사용하였다. 이 방법은 문자의 특징을 이용한 자질을 사용하고 학습코퍼스가 있는 지도학습법을 이용하였고, 영어의 경우 성능은 93%, 스페인어의 경우 90%로 높은 성능을 보였다. 최근의 방법으로는 트위터 문서를

이용하여 개체명을 정규화 시켜서 개체명 인식을 하는 방법[2]도 나왔다. 개체명 정규화는 변형된 개체명의 복원시켜주는 작업이다. 예를 들면 “lady gaga”라는 개체명이 트위터에서 “lady gaaaaaaaaaaaaga”라고 되어 있을 때 본래의 “lady gaga”를 찾아주는 것이다. 개체명 인식에서는 철자 자질, 어휘 자질, 사전 자질을 이용하며 factor graph model을 이용하여 태깅한다. 성능은 개체명 정규화에 대해선 82.6%를 보였고 개체명 인식의 성능은 83.6%의 성능을 보였다.

최근 영어처럼 대문자 자질 등 특정 언어에 의존적인 자질을 다른 언어에서도 이용하기 위한 연구도 진행 중이다. 그중 하나가 위키피디아의 데이터와 병렬데이터를 이용한 개체명 인식 방법[3]이 제안되었다. 위키피디아의 분류 정보를 이용하여 분류와 개체명 간의 맵을 수동으로 만들어서 위키피디아 태거와 특정 언어에서만 이용할 수 있는 자질을 병렬 코퍼스를 사용하여 다른 언어에서도 사용할 수 있게 만든 프로젝션 모델을 이용한 태거, 이 두가지의 위키 태거와 프로젝션의 태거의 결과를 자질로 이용하여 CRFs를 이용한 태거를 제시하였다.

한국어에서도 개체명 인식에 대한 다양한 연구[5,6]가 있었다. 한국어 개체명 인식은 CRFs를 이용하여 개체명의 경계를 인식하고 그 인식 결과를 최대 엔트로피(Maximum Entropy)를 이용하여 개체명을 분류하였다[5]. 이 연구에서는 다양한 개체명을 인식하는데 CRFs의 분류 성능은 좋으나 태그가 많아 질 경우 계산량이 매우 많아지는 문제가 있기에 CRFs는 개체명 경계만의 인식하고 그 분류는 Maximum Entropy를 이용한다. 사용되는 자질은 어휘자질과 품사 자질, 사전 자질 정규표현식 자질을 사용하였고, 83.4%의 성능을 보였다. 다른 방법으로는 형태소분석 자질과 사전 자질을 이용하여 Structural SVM으로 개체명을 인식하는 연구[6]도 있었다. 이 방법은

기존의 CRFs 방법[5]보다 학습시간에 4%가량의 향상이 있었고, 1%이상의 성능 향상도 있었다. 사용한 자질은 기존 CRFs 방법[5]에서 추가적으로 접미사 자질과 형태소 어절내 위치 자질과 여러 조합자질을 추가하여서 사용하였다.

앞서 언급된 개체명 인식에 대한 연구들은 모두다 지도학습법에 의한 방법들로 본 논문에서 제시하는 사용자가 정한 범주를 찾아주는 개체명 인식기에는 부적합하다. 사용자가 정하는 범주가 인명, 지명, 조직명 등의 특정 범주가 아니기 때문에 코퍼스와 사전을 구축하는데 시간이 오래 걸린다. 그렇기 때문에 본 논문에서는 반지도 학습법을 이용한다.

이전에 연구된 반지도 학습법으로는 2006년도에 제안된 부트스트랩핑을 이용한 개체명 인식방법[7]이 있다. 해당 방법은 의사결정트리와 메모리 기반 방법, 두 가지의 분류기를 번갈아가며 학습하는 방법을 제시하였다. 이 방법은 그래프 기반 방법을 이용하여 인명과 지역명 사전을 수집한다. 사전 자질은 개체명 인식에서 좋은 자질 많이 사용되기 때문에 특히 지역명과 인명을 인식하는데 좋은 방법이다. 하지만 해당 방법은 영어권의 언어에 특화된 자질(단어의 첫글자가 대문자이거나, Mr, Mis, Sir 등)을 사용하기 때문에 한국어 적용에 어렵고, 특정 선택된 영역(인명, 지명, 조직명, 기타)만을 학습하기 때문에 여러 범주에 적용하기는 어렵다. 이 방법은 지도학습으로 수집된 사전 자질을 적용하였을 시, 인명 84.32%. 지명 75.06%, 조직명의 경우 77.83%. 기타 범주명의 경우 53.98%가 나왔고, 부트스트래핑 방법으로는 인명 62.59%, 지명은 51.19%, 조직명은 50.18%, 기타 범주명은 33.04%로 나왔다. 지도학습 방법과 부트스트랩핑 방법과 평균적으로 20%가량의 성능차이를 보였다.

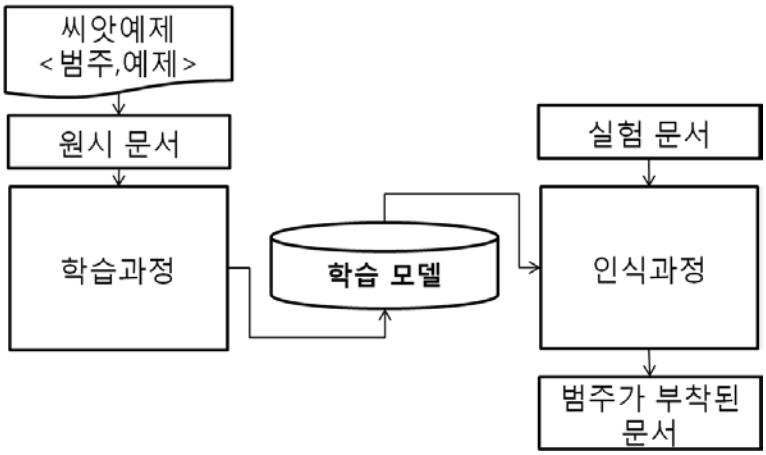
본 연구에서는 씨앗 예제를 이용하여 raw text를 초벌 태깅한 후 CRFs를 이용하

여 서로 다른 템플릿을 반복 학습하여 다양한 범주에 대한 학습 코퍼스가 없어도 학습이 가능한 반지도 학습법을 이용한 개체명 인식기를 제안한다.

제 Ⅲ 장

반지도 학습법을 이용한 개체명 인식

개체명 인식 시스템은 입력된 문장에서 사용자가 원하는 개체명을 인식/분류해주는 시스템이다. 우리는 모든 개체명 인식 시스템이 '왜 고정된 개체명만을 인식할까?'라는 의문에서 연구를 시작했다. Nymble 시스템[1]은 ‘사람’, ‘단체’, ‘지역’, ‘시간’, ‘날짜’, ‘백분율’, ‘금액’, ‘NOT-A-NAME’를 인식한다. 그런데 만약 '지진 발생 연도, 지진 발생 지역, 진도, 피해액, 사상자, 여진 여부'를 인식하기 원한다면 기존의 시스템을 이용할 수가 없다. 본 시스템 사용자가 선택한 임의의 범주를 인식하는 것을 목적으로 하고 있다. 제안 시스템의 전체 구조도는 <그림 III-1> 과 같다.



<그림 III-1> 제안 시스템의 전체 구조도

제안 시스템은 크게 학습과정과 인식과정으로 구성된다. 학습과정은 원시문서에 입력받은 씨앗 예제를 이용하여 학습모델과 학습코퍼스를 생성하는 과정이고, 인식과정은 학습된 모델을 이용하여 개체명에 범주를 할당하는 과정이다.

본 장에서는 논문에서 해결하려는 문제를 정의하고 제약사항을 설명한다. 또한 제안 시스템의 학습과정과 인식과정의 각 부분에서 대해서 설명한다.

1. 문제정의

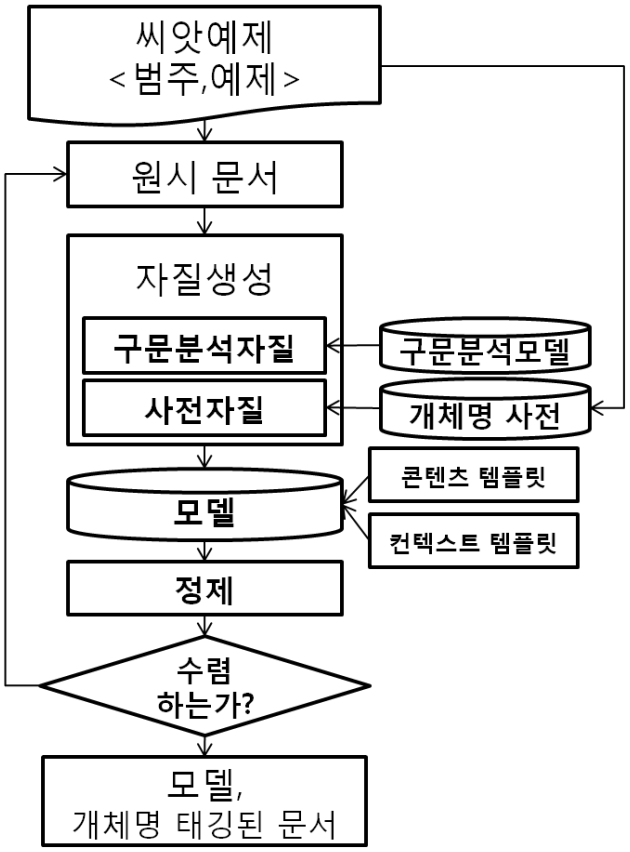
최근 제안된 방법은 통계적 기계학습 방법을 이용한다. 대게는 CRFs와 SVM을 사용하는 방법들이 제시되어 왔는데, 이 두 가지 방법 모두, 학습말뭉치가 필요하다.

하지만 학습말뭉치를 구축하기 많은 시간과 비용이 필요하다. 본 논문에서는 이와 같은 문제를 해결하기 위해서 CRFs를 이용한 반지도 학습법을 제안한다. 초기 씨앗 예제만을 이용하여 개체명 인식 방법을 제안하고 개체명 인식에서 좋은 자질인

사전 자질을 이용하기 위해서 위키피디아를 이용해서 자동으로 구축하는 방법을 제안한다.

본 논문에서 제안한 개체명 인식기는 어떤 범주든 상관없이 사용자가 원하는 범주에 대한 개체명을 구축된 학습말뭉치가 없어도 인식해주는 시스템이다. <그림 III-1>는 제안 시스템의 구조도이다. 제안 시스템은 크게 학습과정과 개체명 인식과정으로 구성된다. 학습과정은 초기 씨앗 데이터를 이용하여 초기 raw text에 개체명 태그를 부착하고, 부착된 문장에 대하여 구문분석 자질과 사전 자질을 생성한다 그리고 필터링 작업을 거친 후 CRFs모델을 생성한다. 이때 Co-traning을 이용한다[11].

콘텐츠에 해당하는 템플릿과 컨텍스트에 해당하는 템플릿를 번갈아 가며 학습하여 모델을 생성한다. 해당 모델을 이용해서 다시 raw text에 개체명을 부착하고 정제하는 작업을 반복해서 일정 횟수가 종료하면 최종 CRFs모델을 생성하는 과정이다. 개체명 인식과정은 학습된 CRFs모델을 이용하여 개체명 범주를 부착하는 과정이다.



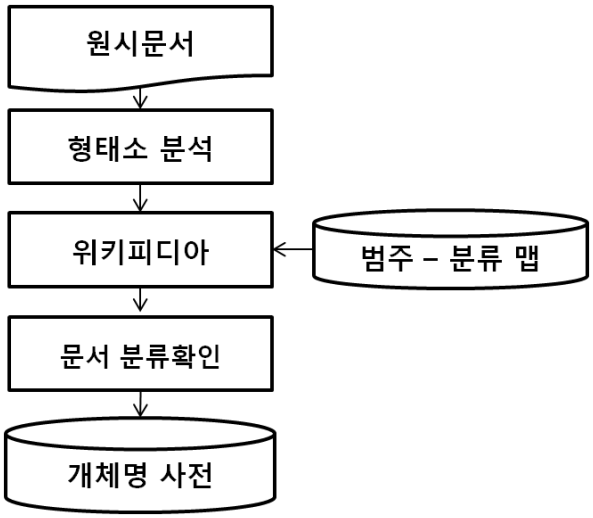
<그림 III-2> 제안 시스템의 학습과정 구조도

2. 학습과정

<그림 III-2>는 제안 시스템의 학습과정 구조도이다. 씨앗 예제를 이용하여 원시 문서에 범주를 부탁하여, 범주가 부착된 문장으로 자질을 생성한다. 생성된 자질을 이용하여 모델을 생성한 후 해당 모델의 결과가 일정 값에 수렴하지 않으면 생성된 모델로 다시 원시문서를 학습하는 과정을 반복하고, 수렴할 시 모델과, 개체명 태깅된 문서를 생성하고 종료한다.

2.1 씨앗 예제

초기 씨앗 예제는 10~20정도의 개체명 단어와 인식하려는 범주이다. 해당 씨앗 예제를 이용하여 raw text에 개체명을 부착한다. 이때 개체명을 초기 부착 시 단순 글자 매칭을 이용하기 때문에 다른 클래스와 중복된 의미를 가진 단어 (예:청와대는 지역명도 되고 조직명도 가능함)를 사용하는 것은 좋지 않다.



<그림 III-3> 위키피디아를 이용한 사전 생성 구조도

2.2 위키피디아를 이용한 사전 생성

위키피디아를 이용한 사전 생성은 사전 자질 생성을 위해 필요하다. 사전 자질은 한국어든 영어든 상관없이 모든 언어에서 개체명 범주를 인식하기 위해 좋은 자질이다. <그림 III-3>는 위키피디아를 이용한 사전 생성 구조도이다. 본 논문에서는 사전을 자동으로 수집하기 위해서 raw text을 형태소 분석을 하여 명사 어휘로 위키피디아에 검색한다.

<표 III.1> 범주와 분류명 테이블

인명	지명	조직명	직업명	건물명	식물명	기술명
살아있는사람	도시	법인	직업	건축물	작물	네트워크
축구선수	수도	행정기구	스포츠인	경기장	식물	컴퓨터 그래픽스
수상자	공항	대학	교사	시설	버섯	프로토콜
	행정 구역	정당	계급	건물	과일	컴퓨터
	읍.면	기관			채소	
	국가	학교			나무	
	지역				나무과	

<표 III.1>은 범주와 분류명 테이블이다. 범주와 분류명 테이블은 위키피디아에서 분류하고 있는 분류명이다. 예를 들어 인명일 경우 위키피디아에서는 대개 살아있는 사람, 축구 선수, 수상자 등을 분류 목록으로 가지고 있다. 해당 범주와 분류명 테이블은 사용자가 수집하려는 범주를 위키피디아에서 분류하고 있는 분류목록 중에 찾아서 입력해야 한다. 인명의 경우에는 3개의 분류명만을 사용해서도 많은 인명을 수집할 수 있지만, 식물이나 지역, 단체명 경우에는 그 범위가 광범위 하기 때문에 좀 더 많은 분류명이 필요하다. 향후 수집된 개체명을 이용하여 분류명을 자동으로 수집할 수 있다면 더 많은 사전을 수집할 수 있을 것이다.

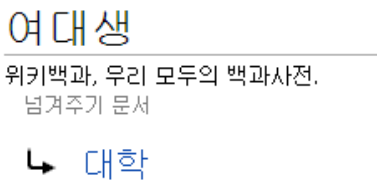
<표 III.2> 위키피디아를 이용하여 수집한 사전 결과

	인명	지명	조직명	직업명	건물명	식물명	기술명	평균
수집된 수	771	789	123	66	204	230	27	315.7
정확도	0.992	0.999	0.927	0.970	0.985	1.000	1.000	0.981

<표 III.2>는 위키피디아를 이용하여 수집한 사전 결과이다. 사전의 정확도는 전체 수집된 사전 개체중에 정확하게 수집된 개체의 비율이다. 수집된 사전은 평균적으로 0.981의 정확도을 보이고 있어 사전으로써 사용은 가능하지만 기술명과 같은 경우 수집된 사전의 양이 극히 작았다. 그 이유는 기술명의 경우 정의하기에 따라서 위키피디아 내에서 분류가 광범위 하며, 실제 위키피디아의 분류 목록이 없거나 하나뿐인 경우가 대부분이라서 분류명을 정하기가 어려웠다.

<표 III.3> 잘 못 수집된 사전 예제

인명	지명	조직명	직업명	건물명
스정경배	인천국제공항공사	털	숙박	속
영국		여대생	인프라	신분
수원삼성		피부	창문	



<그림 III-4> 위키피디아의 넘겨주기 문서의 예

<표 III.3>은 잘 못 수집된 데이터의 예제이다. 잘 못 수집된 이유는 대체로 분류가 중복 되는 경우이다. 단체명을 예로 들면 '털', '피부'의 경우는 분류에서 기관을 분류로 주었는데, 사람 몸의 '감각 기관'등이 기관으로 분류되었기 때문이며, '여대생' 같은 경우 <그림III-4>을 보면 넘겨주기 문서로 해당 단어가 '대학'이라는 제목의 문서로 넘겨주기 때문이다. 이는 위키피디아의 해당 문서에 대해 문서가 없고 넘겨주는 문서로 되어 있으면 자동으로 그 문서로 넘어가게 되는 특징 때문이다.

위키피디아에서 '호날두'의 경우 '크리스티아누 호날두'의 문서로 넘어가는 것처럼, 해당 '여대생'도 '대학'으로 넘어가서 대학이라는 단체로 넘어가서 잘 못 수집된 경우가 있다.

<표 III.4> 생성된 자질 예제. 자질 1은 형태소의 품사, 2는 해당 형태소가 포함된 어절의 구문태그, 3은 이전 어절의 구문태그, 4는 이전 어절, 5는 다음 어절의 구문태그, 6은 다음 어절, 7은 해당 형태소의 지배소 동사, 8은 해당 형태소의 길이, 9는 는 형태소의 첫 음절, 10은 형태소의 마지막 음절, 11은 형태소의 어절 내 위치, 12는 사전 내에 존재 여부이다.

	1	2	3	4	5	6	7	8	9	10	11	12
영표	NNP	NP_CNJ	NULL	NULL	NP_MOD	기현이의	-	2	영	표	0	0
나	JC	NP_CNJ	NULL	NULL	NP_MOD	기현이의	-	1	나	나	1	0
기현이	NNP	NP_MOD	NP_CNJ	영표나	NP_MOD	경우	-	3	기	이	0	0
의	JKG	NP_MOD	NP_CNJ	영표나	NP_MOD	경우	-	1	의	의	1	0
경	NNP	NP_MOD	NP_MOD	기현이의	NP_SBJ	감독들이	-	1	경	경	0	0
우	JKG	NP_MOD	NP_MOD	기현이의	NP_SBJ	감독들이	-	1	우	우	1	0
감독	NNG	NP_SBJ	NP_MOD	경우	VP	바뀌었다	바뀌/VV	2	감	독	0	0
들	XSN	NP_SBJ	NP_MOD	경우	VP	바뀌었다	바뀌/VV	1	들	들	1	0
이	JKS	NP_SBJ	NP_MOD	경우	VP	바뀌었다	바뀌/VV	1	이	이	2	0
바뀌	VV	VP	NP_SBJ	감독들이	R	.	-	2	바	뀌	0	0
었	EP	VP	NP_SBJ	감독들이	R	.	-	1	었	었	1	0
다	EF	VP	NP_SBJ	감독들이	R	.	-	1	다	다	2	0
.	SF	R	VP	바뀌었다	NULL	NULL	-	1	.	.	0	0

2.3 자질 생성

자질의 기본은 구문분석기의 결과로서, 문장을 이루는 형태소 단위로 구성하고 사전 및 어휘를 이용하여 자질을 추가 하였다.

<표 III.4>은 생성된 자질의 예제이다. 첫번째 열은 형태소 어휘의 품사태그이다. 품사태그는 수집하려는 범주에 대부분 연관성 높기 때문에 좋은 자질이라고 할 수

있다. 두번째 열은 형태소 어휘를 포함한 어절의 구문태그이다. 세번째, 네번째 열은 현재 형태소 어휘가 가지는 어절의 이전 어절의 태그와 어휘이다. 다섯번째와 여섯번째의 열은 현재 형태소 어휘가 가지는 어절의 이후 어절과 이후 어절의 태그이다. 세번째부터 여섯번째 자질은 어휘에 대한 컨텍스트 자질을 주기 위하여 생성된 자질이다. 일곱번째 열은 해당 형태소 어휘가 수식하는 어절이 동사일 경우 해당 동사 원형과 태그를 사용한다. 여덟번째 열은 어휘의 길이 자질이며 아홉번째 열번째 자질은 어휘의 첫번자 글자와 마지막 글자 자질이다. 열한번째 자질은 형태소의 어절 내 번호이다. 마지막 열 두번째 자질은 사전에 있는지 없는지에 대한 자질이다.

<표 III.4>에서 세번째 형태소의 "기현이"의 경우는 자신의 품사가 'NNP'[1]이고, 자신이 포함된 어절인 '기현이의' 구문태그가 'NP_MOD'[2]이고, 이전 어절의 구문태그는 'NP_CNJ'[3], 이전 어절의 문자열은 '영표나', 다음 어절인 '경우'의 구문태그인 'NP_MOD', 그리고 해당 어절의 지배소 동사가 없으므로 '-', 해당 형태소의 길이가 3이고, 해당 형태소의 첫 음절인 '기', 해당 형태소의 마지막 음절인 '이', 형태소가 어절 내에서 처음 나오므로 '0', 사전에 해당 형태소 어휘가 존재하지 않기 때문에 '0'이 자질로써 생성된다.

생성된 기본 자질들은 다양한 조합을 통해서 새로운 자질로 만들 수 있다. 이 경우, 대부분의 연구에서는 연구자의 사전지식이 중요한 요소가 된다. 본 논문에서는 실험적으로 가장 좋은 조합자질을 찾아서 사용하였다.

---

[1] NNP는 고유명사를 뜻한다.

[2] NP_MOD는 관형격 체언구이다.

[3] NP_CNJ는 접속격 체언구이다.

<표 III.5> 콘텐츠와 컨텍스트 템플릿 사용 자질 표. 괄호 안의 표시는 거리 표시

Uni-gram	콘텐츠 템플릿	컨텍스트 템플릿
1	1	1(+1)
2	2	0(-1) + 0(+1)
3	1 + 2	2(-1)
4	8	2 + 2 (-1)
5	9	0(+1) + 2(+1)
6	10	2(+1) + 2(+2)
7	11	1(-2) + 2(-2)
8	12	1 + 2(+1)
9	9 + 10	1 + 0(+1) + 1(+1)
10	1 + 7	1 + 1(+1) + 1(+2)
11	1 + 8	1 + 0(+1) + 1(+2)
12	1 + 2 + 1(-1)	1 + 0(-1) + 1(-1)
13	1,9	1 + 0(-2) + 1(-2)
14	-	1 + 2(-1)
15	-	1 + 2(-1) + 2(+1)
bi-gram	**콘텐츠 템플릿**	**컨텍스트 템플릿**
16	1,2	1 + 1 (-1)
17	9,10	1 + 1 (+1)
18	-	6 + 1(+1)
19	-	1(+1) + 1(+2)

2.4 모델 생성

모델을 생성할 때는 Co-training을 이용한다. Co-training은 두 가지의 다른 특성을 가진 방법을 이용하여 번갈아 학습하는 방법이다[11]. Co-training은 두 가지의 다른 특성을 가진 방법을 이용하여 번갈아 학습하는 방법이다. 본 논문에서는 콘텐츠 템플릿과 컨텍스트 템플릿 두 가지를 이용하여 반복 될 때마다 다른 모델을 생

성한다.

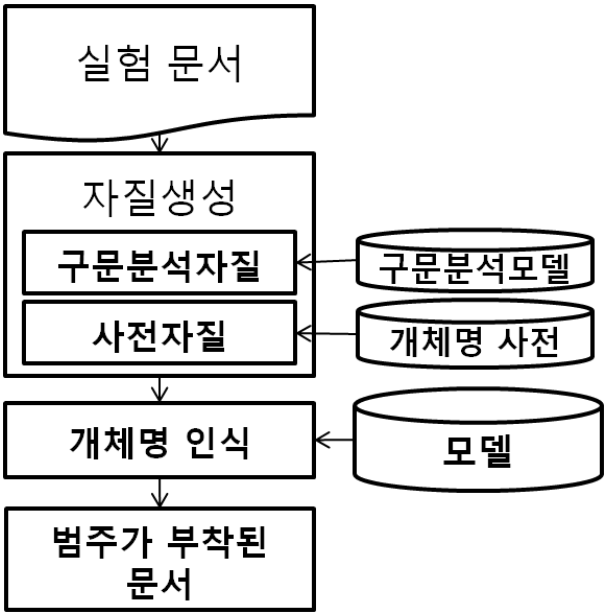
<표 III.5>는 콘텐츠와 컨텍스트 템플릿 사용 자질 표이다. uni-gram 자질은 현재 자신의 개체명 태그만을 확인하는 자질이고, bi-gram 자질은 자신과 자신 이전의 개체명 태그를 확인하는 자질이다.

콘텐츠 템플릿은 하나의 기준어휘를 중심으로 그 형태소 어휘가 가지는 자질만을 사용하는 템플릿이다. <표 III.4>에서 “기현이”를 예로 들었을 경우 품사태그로 “NNP”를 가지고 어절태그로 “NP”를 가지며, 어휘길이는 3, 첫번째 글자, 마지막 글자로 “기”와 “이”를 가지는 등 현재 어휘를 기준의 자질을 사용한다.

컨텍스트 템플릿은 그와 반대로 기준 어휘의 주변의 자질을 보게 된다. 예를 들면 “기현이”를 기준으로 이전 어휘 “나”, 이전 품사 태그 “JC”, 이후 어휘 “의”과 이후 품사태그 “JKG” 등 기준어휘 주변의 자질을 사용하는 템플릿이다.

2.5 정제

정제부분은 생성된 모델을 이용하여 원시문서에 개체명을 부착하여 개체명이 부착된 결과를 정제한다. 초기 5번 반복까지는 사용자가 직접 개입하여 잘못된 100개 가량의 문장을 수정해주고 해당 문장만을 다음 학습에 이용한다. 반복횟수가 5번 초과일 경우는 태깅된 어휘의 빈도수를 계산하여 일정 빈도 이상 나타난 어휘만을 정제하여 다음 학습에 이용한다.



<그림 III-5> 제안 시스템의 개체명 인식 과정 구조도

3. 개체명 인식 과정

박지성은 최근 부상을 회복하여 리그 복귀를 노리고 있다.

<그림 III-6> 개체명 인식 결과 문장

<그림 III-5>은 개체명 인식과정 구조도이다. 개체명 인식과정은 학습과정에서 생성된 모델을 이용하여 입력문장이 들어왔을 시 입력문장에 대하여 품사태깅과 구문분석을 수행한 후 입력문장에 대한 자질을 생성한다. 생성된 자질을 이용하여 개체명을 인식 후 결과문장을 내보낸다. <그림 III-6>는 입력문장에 대한 인명 범주의 개체명 인식 결과의 예제이다.

4. CRFs 학습 및 평가

CRFs는 조건부 확률을 최대로 하는 방향성이 없는 그래프 모델이다[8]. 입력열 $X = x_1 x_2 \ldots x_n$, 상태열 $T = t_1 t_2 \ldots t_n$이 주어지고 가중치 $\Lambda = \{\lambda \ldots\}$가 주어졌을 때, CRFs에서는 조건 확률로 식 (1)과 같이 정의된다.

$$P(T|X) = \frac{1}{Z(X)} exp\left(\sum_{i=1}^{n}\sum_{k} \lambda_k f_k(t_{i-1}, t_i, x, i)\right) \qquad (1)$$

여기서 $Z(X)$는 확률 값으로 만들어 주는 정규화 값이고 $f_k(t_{i-1}, t_i, x, i)$는 자질 함수이다. 또한 λ_k는 각 자질에 대한 가중치를 나타낸다. k는 k번째 자질이며, 자질 함수는 현재 시간에 대해 관측열 x_i, 상태변이 $t_{i-1} \rightarrow t_i$에 대해서 전이의 양상을 측정할 수 있다. 매개변수들은 주어진 입력열과 이에 대응하는 상태열에 대한 조건부 확률이 최대화하는 최대 우도(maximum likelihood)에 의해서 추정된다. 훈련 집합에 대해서 다음과 같은 로그 유사도(log-likelihood)를 계산한다.

$$L(\Lambda) = \sum_{l} \log P_{\Lambda}(t_l|x_l) = \sum_{l}\left(\sum_{i=1}^{n}\sum_{k} \lambda_k f_k(t_i, x, i) - \log Z_{x_i}\right) \qquad (2)$$

식 (2)를 최대로 하도록 학습한다. 일반적으로 CRFs는 IIS(Improved Iterative Scaling)나 GIS(Generalized Iterative Scaling)[4]를 사용하여 학습한다.

또한 학습 데이터의 과적합(overfitting) 문제를 해결하기 위해서 가우스 사전 평활

[10]을 적용한다. 본 연구에서 CRF++을 사용하였다.

제 IV 장

실험 및 토의

1. 실험환경

본 논문에서 사용한 코퍼스는 다음과 같다. 코퍼스는 네이버 뉴스에서 추출한 데이터를 사용한다. 반지도 학습을 위한 20,000문장의 학습 코퍼스(Training Set)와, 학습시 평가를 위한 2,000문장의 개발 코퍼스(Development Set), 최종 평가를 위한 1,000문장의 평가 코퍼스(Test Set)을 구축하였다.

개발 코퍼스와 평가 코퍼스는 학습하고자 하는 개체명 범주마다 각 2,000문장 1,000문장을 구축하였다. 즉 한 개체명 범주를 위한 코퍼스로 23,000문장을 사용한

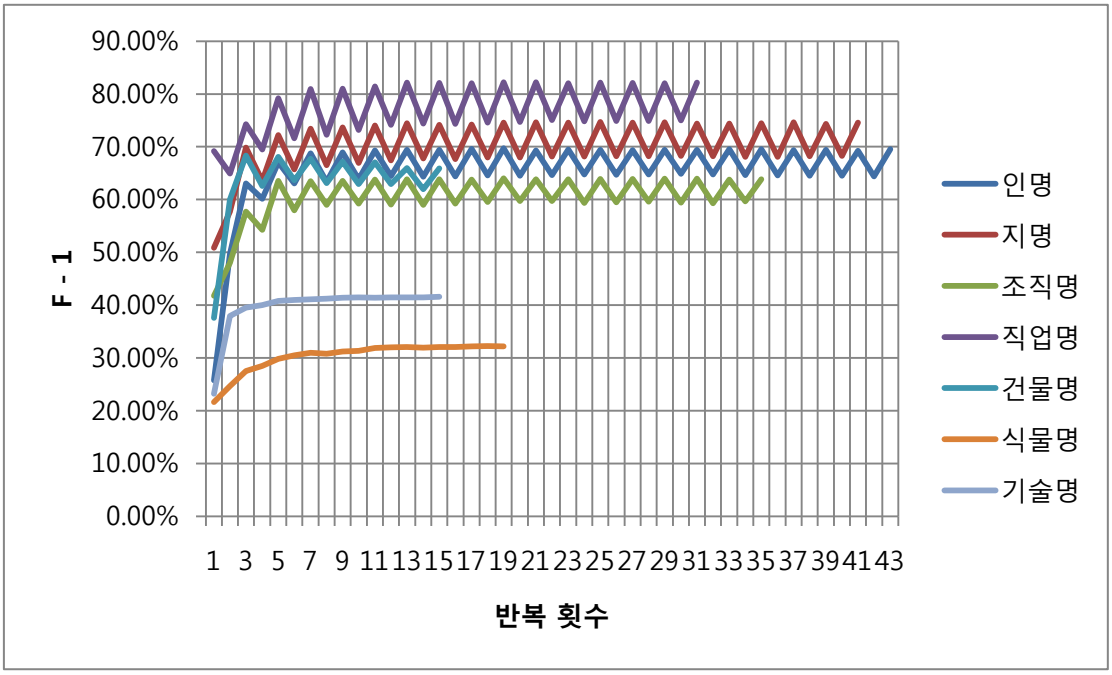
다.

제안한 시스템의 성능 평가를 위해 정확도와 재현율을 결합한 $F_1 - measure$ 를 사용하였다. 평가 척도는 식 (3)과 같다.

$$\text{정확도}(Precision, P) = \frac{\text{실제 정답의 수}}{\text{시스템의 모든 정답의 수}},$$
$$\text{재현율}(Recall, R) = \frac{\text{실제 정답의 수}}{\text{정답문서의 모든 정답의 수}},$$
$$F_1 - measure = \frac{2 \cdot P \cdot R}{P + R} \quad (3)$$

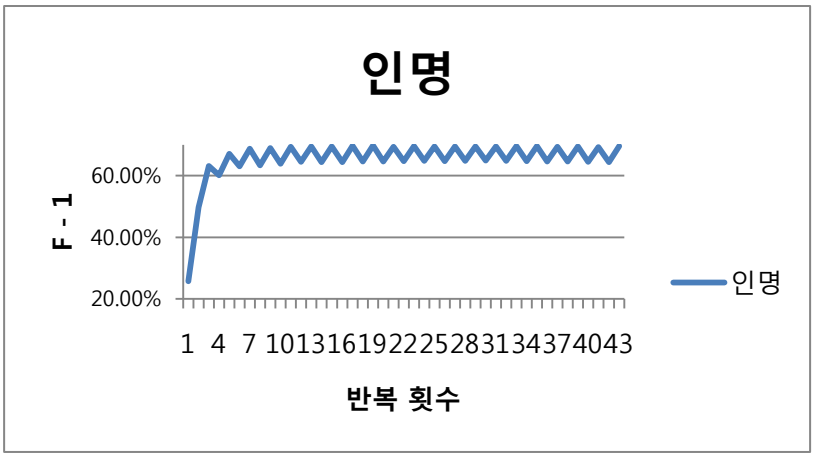
2. 실험 결과

본 절에서는 제안한 방법에 따른 실험결과에 대해 설명한다. 본 논문에서 분류하고자 하는 개체명은 인명, 지명, 조직명, 직업명, 건물명, 식물명, 기술명으로 총 7개의 범주에 대해서 실험하였다.



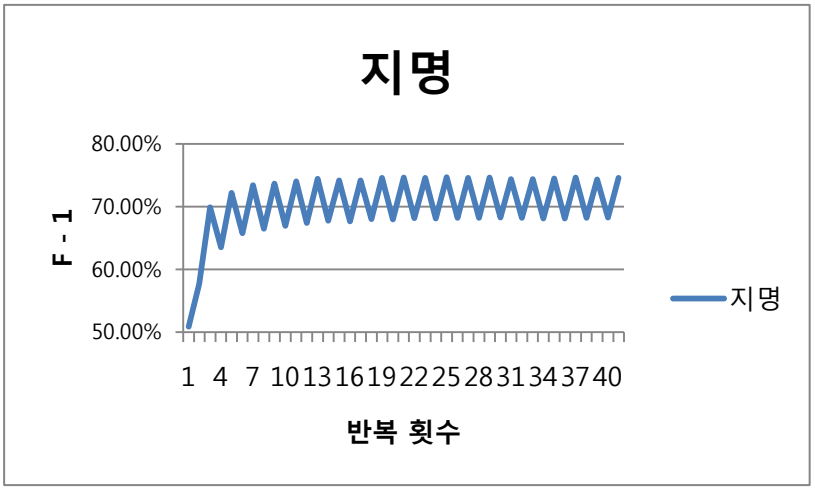
<그림 IV-1> 7개의 범주에 대한 학습 결과 그래프

<그림 IV-1>는 7개의 범주에 대한 학습 결과 그래프이다. 반복횟수가 증가함에 따라 콘텐츠템플릿과 컨텍스트 템플릿에 따라서 성능은 들쭉날쭉 했지만 전체적인 성능은 증가하였으며 일정 횟수 이상 반복했을 시, 더 이상 성능은 증가하지 않고 수렴하였다. 성능이 들쭉거리는 이유는, 콘텐츠 템플릿이 컨텍스트 템플릿보다 상대적으로 성능이 더 잘 나왔기 때문이다. 상대적으로 식물명과 기술명의 성능이 떨어지는 이유는 학습 코퍼스에서 인명, 지명, 조직명 등은 상당수 나타나 학습에 도움이 되지만 식물명과 기술명은 학습코퍼스에서 나타나는 횟수가 상대적으로 많이 적기 때문에 학습되는 문장의 양도 줄어들게 되어 성능이 다른 범주들에 비해 많이 낮게 나왔다.



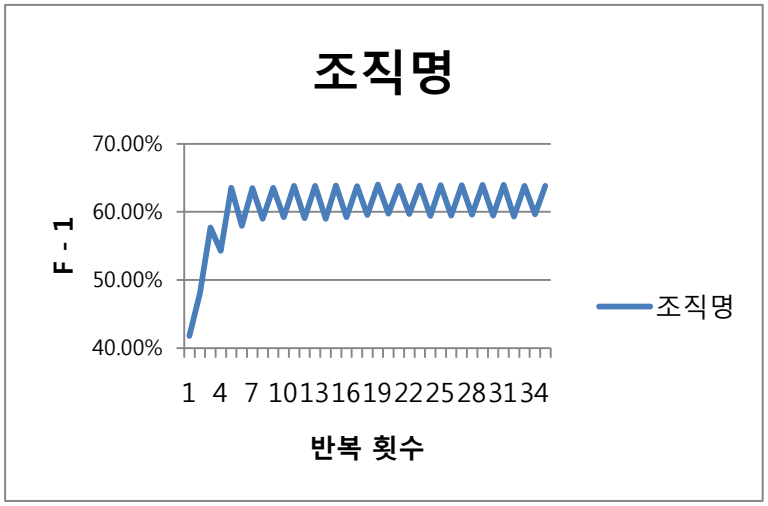
<그림 IV-2> 인명 범주에 대한 학습 결과 그래프

<그림 IV-2>는 인명 범주에 대한 학습 결과 그래프이다. 인명의 경우 초기 씨앗 예제로 학습한 결과가 25.76%였고 다섯번째 반복 횟수까지 성능이 증가하다가, 여섯번째 컨텍스트 템플릿으로 학습을 하였을 시 처음으로 성능이 줄어든다. 이후 다시 콘텐츠 템플릿으로 학습을 하면서 성능이 증가하고, 컨텍스트 템플릿에서 학습하면서 줄어들고를 반복하며 최종 69.52%까지 성능이 증가하였다. 컨텍스트 템플릿에서 성능이 줄어드는 이유는 콘텐츠 템플릿에 비해서 컨텍스트 템플릿의 학습성능이 상대적으로 떨어지기 때문이다. 인명 범주의 경우 총 43회까지 반복 횟수가 증가하였고, 총 20000문장 중 최종적으로 17177문장이 최종 학습 코퍼스로 생성되었다.



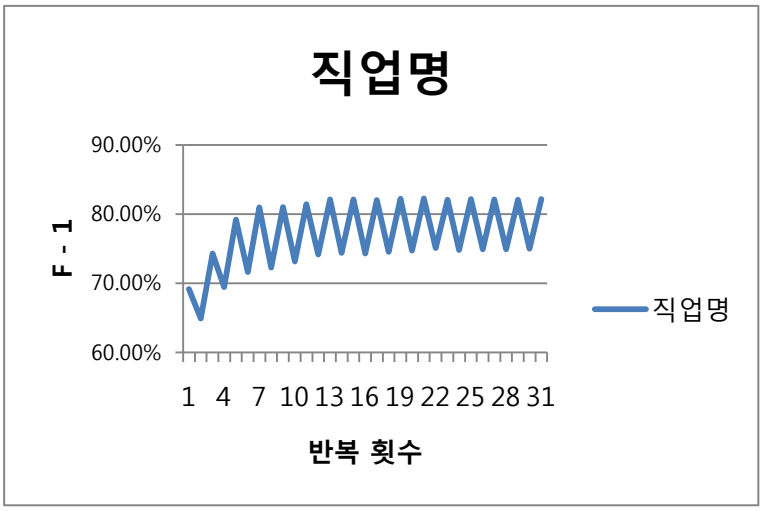
<그림 IV -3> 지명 범주에 대한 학습 결과 그래프

<그림 IV -3>은 지명 범주에 대한 학습결과 그래프이다. 지명 범주는 초기 씨앗 예제로 학습한 결과는 50.88% 였다. 다른 범주들에 비해 초기 학습결과의 성능이 높은데 이것의 이유는 씨앗 예제로 학습되는 개체명의 수가 다른 것들에 비해 많아서 초기 학습코퍼스의 양이 다른 범주에 비해 많았기 때문이다. 지명 범주는 반복 횟수 3회까지 증가하다가 4회째 컨텍스트 템플릿으로 학습 하였을 시 처음으로 줄어들었다. 그 이후 다시 콘텐츠 템플릿으로 학습하면서 성능이 증가하였고, 컨텍스트 템플릿과 콘텐츠 템플릿을 반복 학습하면서 성능이 74.56%까지 증가하였다. 지명 범주는 총 40회까지 반복 횟수가 증가하였고, 20000문장 중 16635문장이 최종 학습 코퍼스로 생성되었다.



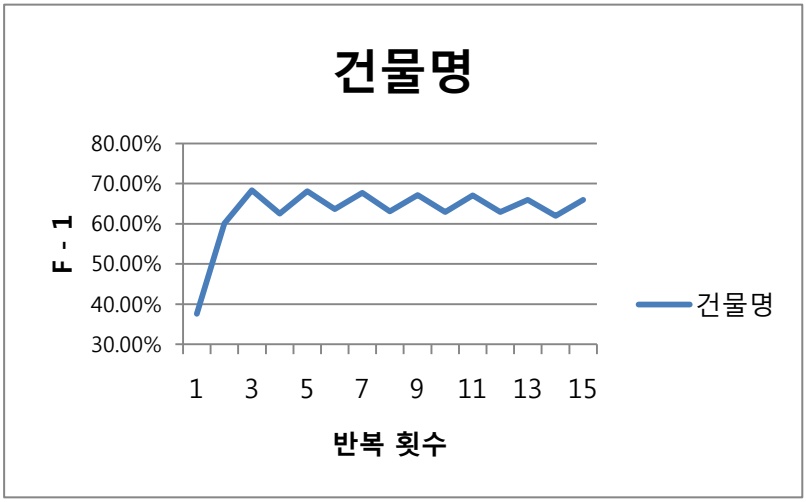
<그림 IV-4> 조직명 범주에 대한 학습 결과 그래프

<그림 IV -4>는 조직명 범주에 대한 학습 결과 그래프이다. 조직명 범주는 초기 씨앗 예제로 학습한 결과는 40.76%였다. 이후 3회까지 57.73%까지 성능이 증가하다가 지역명과 마찬가지로 4회째에 54.27%로 성능이 조금 줄어들었다가, 다시 5회째에 성능이 증가하고 그 뒤로 성능이 줄어들고, 늘어나고를 반복하면서 최종적으로 63.82%에 수렴하였다. 조직명은 총 34회까지 반복 횟수가 증가하였고, 20000문장 중 15760문장이 최종 학습 코퍼스로 생성되었다.



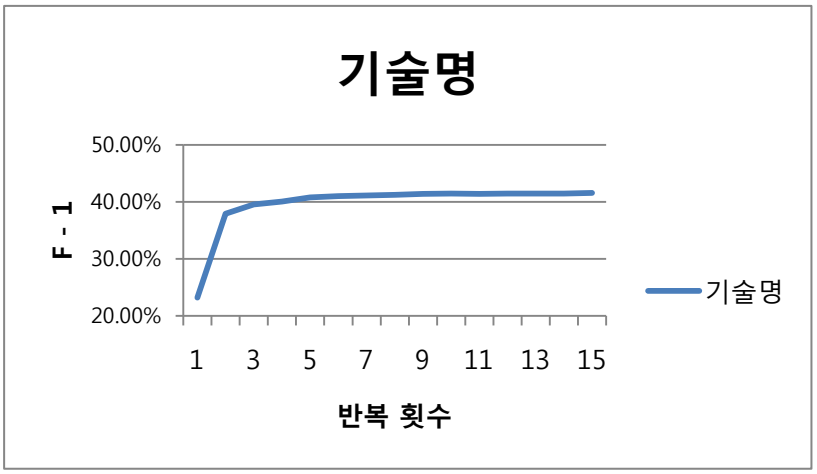
<그림 IV-5> 직업명 범주에 대한 학습 결과 그래프

<그림 IV -5>는 직업명 범주에 대한 학습 결과 그래프이다. 직업명 범주의 경우에는 초기 씨앗 예제로 학습한 결과가 69.17%로 매우 높다. 2회째 컨텍스트 템플릿을 학습할 때 성능이 64.92%로 떨어졌다. 하지만 다시 콘텐츠 템플릿을 학습할 경우 74.29로 성능이 증가하였다. 콘텐츠 템플릿과 컨텍스트 템플릿을 번갈아 학습할 때마다, 성능이 오르락 내리락 하였지만, 반복횟수가 증가할수록 성능은 점차 증가하였고, 최종적으로 82.14%의 성능에서 수렴하였다. 직업명 범주는 총 31회 반복하였고, 20000문장 중 15953문장이 최종 학습 코퍼스로 생성이 되었다.



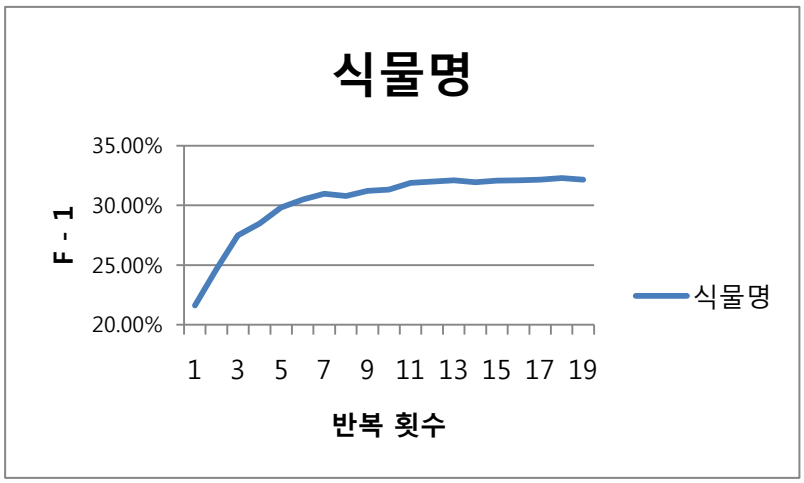
<그림 IV-6> 건물명 범주에 대한 학습 결과 그래프

<그림 IV -6>은 건물명 범주에 대한 학습 결과 그래프이다. 건물명 범주의 경우 초기 씨앗 예제로 학습했을 경우 37.58%의 성능을 보였다. 3회째까지 68.33%로 증가하다가 그 이후로 조금씩 하락하는 추세를 보였다. 건물명의 경우 '~경기장', '~구장' 같은 형태의 경우가 많았기 때문에 컨텍스트 템플릿은 성능을 계속 증가시켰지만 현재 어휘에 대해서만 자질을 생성하는 콘텐츠 템플릿의 경우 성능을 떨어뜨렸다. 최종적으로 65.92%의 성능을 보였고, 총 15번의 반복 하였고, 20000문장 중 9874문장이 최종 학습 코퍼스로 생성되었다.



<그림 IV -7> 기술명 범주에 대한 학습 결과 그래프

<그림 IV -7>은 기술명 범주에 대한 학습 결과 그래프이다. 기술명 범주의 경우 초기 씨앗 예제로 학습했을 경우 23.22%의 낮은 성능을 보였다. 14회째까지 41.58%로 증가하였다. 기술명의 경우 성능은 천천히 증가하였지만, 성능이 크게 증가하지는 못했다. 그 이유는 학습 코퍼스에서 해당 범주의 개체명이 나타나는 횟수가 상대적으로 많이 적었기 때문에 학습되는 문장의 양도 줄어들게 되어 성능이 다른 범주에 비해 많이 낮게 나왔다. 기술명 범주는 최종적으로 41.58%의 성능을 보였고, 총 15회 반복하였고, 20000문장 중 3251문장이 최종 학습코퍼스로 생성되었다.



<그림 IV-8> 식물명 범주에 대한 학습 결과 그래프

<그림 IV -8>은 식물명 범주에 대한 학습 결과 그래프이다. 식물명 범주는 초기 씨앗 예제로 학습하였을 시 21.63%의 낮은 성능을 보였다. 19회째 반복까지 천천히 성능은 증가하였고, 최종적으로 32.16%의 성능을 보였다. 식물명 범주의 경우에도 높은 성능을 보이지 못했는데, 그 이유는 기술명 범주와 마찬가지로 학습코퍼스에서 해당 범주의 개체명이 나타나는 횟수가 적어서 학습되는 문서의 양이 적었기 때문이다. 최종적으로 32.16%의 성능을 보였고, 총 19번의 반복 하였으며, 20000문장 중 4528문장이 학습 코퍼스로 생성되었다.

<표IV .1> 7개의 범주에 대한 실험 결과

	반복횟수	학습문장의 수	정확도	재현율	F1
인명	43	17177/20000	85.39%	58.18%	69.21%
지명	41	16635/20000	79.78%	67.31%	73.02%
조직명	35	15760/20000	82.29%	55.00%	65.93%
건물명	15	9874/20000	74.08%	39.87%	51.84%
직업명	31	15953/20000	91.18%	73.13%	81.16%
기술명	19	3251/20000	63.76%	29.14%	40.00%
식물명	15	4528/20000	53.01%	23.12%	32.20%

<표IV .1>은 반지도 학습법을 이용하여 개체명 각 범주별로 실험 한 결과이다. 인명, 지명, 조직명의 경우에 평균적으로 69%정도의 성능이 나왔고 나머지 기타 범주는 예외적으로 직업명을 제외하고 50%에 근접하거나 그 이하로 나왔다. 직업명의 성능이 높은 이유는 다른 범주에 비해 자질로서 표현할 수 있는 것이 많았다. 직업명은 예를 들어 '~사장', '~교수'와 같은 것인데 이런 것은 대부분 인명과 연관되어 문장에서 표현을 한다. 이것을 자질로서 표현하는 것이 다른 범주에 비해 좀더 세부적인 자질 선정을 할 수 있다. 그리고 직업명은 다른 범주에 비해 새로 생겨나는 어휘가 적을 뿐더러 코퍼스에 나타나는 종류 또한 적었다.

<표 IV .2> 부트스트랩핑 방법[7]과의 비교

	인명	지명	조직명	기타
부트스트랩핑	62.90%	51.19%	50.18%	33.04%
제안방법	69.21%	73.02%	65.93%	52.04%

<표 IV .2>는 부트스트랩핑 방법[7]과의 성능비교이다. 영어 개체명 인식 방법인 부트스트랩핑 방법과의 절대적인 성능 비교는 어렵다. 하지만, 한국어가 영어보다 사용할 수 있는 자질이 적은 점(예를 들어, 대문자로 표시되는 자질이거나, 인명일 경우 Mr, Mis, Sir 등)과 한국어 개체명 인식기의 평균적인 성능이 영어 개체명 인식기보다 낮다는 것을 감안했을 경우(한국어의 경우 평균적으로 85% 근처인 반면, 영어의 경우 90% 이상 성능을 보이는 것도 있다), 제안 방법을 부트스트래핑 방법과 비교 했을 시 인명의 경우 6.31%, 지명에서는 21.83%, 조직명에서는 15.75%, 기타 범주 부분에서는 19% 이상 성능이 향상된 것을 볼 수 있다.

<표 IV .3> 한국어 개체명 인식기와 비교

기계학습 알고리즘	F1(%)
CRFs (지도학습)	84.99
Structural SVM(지도학습)	85.14
Modified Pegasos(지도학습)	85.43
제안방법(반지도 학습)	65.05

<표 IV .3> 는 기존 한국어 개체명 인식기와의 성능 비교표이다. 본 논문에서 제안한 방법은 반지도 학습법을 이용하여 학습말뭉치를 생성하여 학습하는 방법이다.

이전에 제시된 지도 학습법을 이용한 개체명 인식기의 성능보다 20%가량 성능이 떨어지지만, <표IV .1>에서 볼 수 있듯이 학습코퍼스에서 포함된 개체명이 적은 식물명과 이론명을 제외한 인명, 지명, 조직명, 직업명, 건물명의 정확도를 보았을 때 평균적으로 82.54%로 지도 학습법과 비슷한 성능을 보인다.

3. 오류 분석

오류는 크게 두 가지 유형으로 나눌 수 있다. 첫번째가 정답으로 나와야 하는데 정답으로 나오지 않은 경우, 다른 하나가 정답이 아닌데 정답으로 나온 경우가 있다.

먼저 정답으로 나와야 하는데 나오지 않는 경우를 살펴보면 크게 두 가지로, 형태소 분석단계에서의 오류이거나, “세르히오 가르시아” 또는 “타이거 우즈” 같은 2어절 이상의 개체명의 경우 앞의 부분인 “세르히오”, “타이거” 등이 잡히지 않는 오류이다.

야	NNG
이	VCP
아	EC
투레	NNP

<그림 IV-9> 개체명 '야야 투레'의 형태소 분석 오류

먼저 형태소 분석 오류부터 살펴보면, '야야 투레'라는 축구선수의 인명이지만 이것이 <그림 IV -9>과 같이 분석이 잘 못 되는 경우가 많았다. 정상적인 경우라면, '야야/NNP', '투레/NNP'와 같이 분석되어야 하지만 <그림 IV -9>처럼 형태소 분석 단계에서 잘 못 분석되는 경우가 많았다. 대게 외국어 개체명이거나, 한국 개체명에 조사가 붙는 경우(예를 들면 '이수만' 같은 경우 'NNP'로 분류되어야 하지만, '이수/NNG+만/JX'으로 분석)에 형태소 분석 오류가 많았다. 그리고 2어절 이상의 개체명일 경우 못 잡는 경우가 많았다. 이는 초기 씨앗 예제에 포함된 2어절 이상의 개체명들이 실제 학습코퍼스에서 나타나는 수가 다른 씨앗 예제들에 비해 적어서, 여러 번 학습과정을 반복과정을 거치면서 못 잡아내게 되는 경우가 있었다.

다음으로 정답이 아니지만 정답으로 나온 경우가 있다. 크게 두 가지로 나뉘는데 하나는 콘텐츠 템플릿에서 발생하는 오류이며 다른 하나는 컨텍스트 템플릿에서 발생하는 오류이다. 콘텐츠 템플릿에서 발생하는 오류는 같은 단어에 대한 오류이다. 예를 들면 '밤' 같은 경우에는 먹는 '밤'과 저녁을 표현하는 '밤'이 있다. 식물명 범주에서는 먹는 '밤'만을 찾아내야 하지만 두 경우 모두다, 'NNG'라는 품사와, 어휘의 길이가 1, 첫글자, 마지막글자가 모두 '밤'이고 둘 다 어절내 첫번째 형태소라는 자질을 가지기 때문에 저녁을 표현하는 '밤'도 식품명 범주로 잘 못 분석되었다. 컨텍

스트 템플릿에서의 오류는 단체명이 인명으로 잡히는 오류가 있었다. 예를 들면 ‘아스널 앙리가 말했다’라는 문장에서 인명 범주를 인식할 경우 ‘앙리’만을 인식해야 하지만, 씨앗 예제에서 ‘티에리 앙리’라는 예제가 학습되었을 경우 ‘앙리’ 앞에 오는 어휘의 품사가 ‘NNP’ 또는 ‘NNG’일 경우 앞에 오는 어휘를 해당 범주로 인식하는 경우가 있었다. 이 오류는 특히 인명과 조직명의 경우 많이 나타났다. 인명과 조직명의 경우 유사한 컨텍스트나 콘텐츠를 가지는 경우가 많기 때문으로 분석되었다.

제 V 장

결론 및 향후 연구

본 논문에서는 고정된 범주가 아니라 사용자가 원하는 범주를 구하는 시스템을 제안하였다. 이것은 기존의 정해진 범주만 태깅하는 개체명 인식기에 비해 활용성이 높다는 장점을 가지고 있다. 그러나 정해진 범주가 아니라 사용자가 설정한 범주를 태깅하는 시스템을 구현하기 위해서는 미리 범주가 할당된 코퍼스를 만들기 힘들기 때문에 지도학습을 할 수가 없다. 따라서 본 논문에서는 반지도 학습법을 이용하였다. 반 지도 학습에서는 Co-training 을 사용하였다. Co-training 에서는 반복 중에 한번은 문맥 정보를 사용하고 한번은 태깅되는 자신의 정보를 사용하게 된다. 따라서 문맥 정보를 사용하여 재현율을 올리게 되고 자신의 정보를 사용하여 정확도을 올리게 된다.

반지도 학습에서 다양한 자질 스키마를 정하였다. 그 중에서 해당 범주의 초기 사전을 초기 수집하려는 범주에 대한 소량의 분류명을 이용해서 위키피디아를 이용하여 자동으로 수집하는 방법을 제시하였다. 위키피디아의 다양한 관련 범주 이름을 활용하여 요구하는 범주에 대한 사전을 확보할 수 있다.

실험 결과로 보았을 때 대량으로 구축된 학습코퍼스만큼의 성능이 나오진 않았어도 그에 가까운 성능이 나왔다. 그리고 영어권의 부트스트래핑 방법[7]과 비교하였을 때는 오히려 6%~21%가량 높은 성능을 보였다.

또한 위키피디아를 이용해 사전을 자동으로 구축이 가능해서 사용자가 정한 임의의 범주를 인식하기 위하여 소량의 씨앗 데이터와, 수집하려는 범주를 분류할 수 있는 분류명만 필요로 하기 때문에 대량의 학습코퍼스와 사전을 구축하기 위하여 많은 시간과 비용을 필요로 하지 않는다. 하지만 지진의 강도 같은 위키피디아에서 사전을 구축하기 어려운 데이터에 대한 범주에 대한 추가적인 자질에 대한 필요성과 위키피디아에서 수집하려는 범주에 대한 분류명을 사용자가 직접 넣어주기 힘든 경우도 있기 때문에 분류명을 자동으로 확장하는 방법도 필요하다.

개체명 인식기의 성능을 향상하기 위해 추가적인 자질이 필요하다. 이 부분에 대해서는 추가 적인 연구로 남겨둔다.

참고문헌

1. Daniel M. Bikel, Scott Miller, Richard Schwartz, Ralph Weischedel, *"Nymble: a High-Performance Learning Name-finder"*, Proceedings of the Fifth Conference on Applied Natural Language Processing, 1997
2. Xiaohua Liu, Ming Zhou, Furu Wei, Zhongyang Fu, Xiangyang Zhou, *"Joint Inference of Named Entity Recognition and Normalization for Tweets"*, Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, pp. 526–535, 2012
3. Sungchul Kim, Kristina Toutanova, Hwanjo Yu, *"Multilingual Named Entity Recognition using Parallel Data and Metadata from Wikipedia"*, Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, pp. 694–702, 2012.
4. S. Della Pietra, V. Della Pietra, J. Lafferty, *"Inducing features of random fields"*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.19, No.4, pp.380-393, 1997(IIS)
5. 이창기, 황이규, 오효정, 임수종, 허정, 이충희, 김현지, 왕지현, 장명길, *"Conditional Ranfom Fields 를 이용한 세부 분류 개체명 인식"*, 제 18 회 한글 및 한국어 정보처리 학술대회, pp. 268-272, 2006
6. 이창기, 장명길, *"Structural SVMs 및 Pegasos 알고리즘을 이용한 한국어 개체명 인식"*, 인지과학 제 21 권 제 4 호, pp.655-667, 2010
7. Zornitsa Kozareva, *"Bootstrapping Named Entity Recognition with Automatically Generated Gazetteer Lists"*, In Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics, 2006
8. J. Lafferty, A. McCallum, F. Pereira, *"Conditional random fields: Probabilistic models for segmenting and labeling sequence data"*, In Proceedings. 18th International Conference on Machine Learning, pp. 282-289, 2001

9. 홍진표, 차정원, “어절패턴 사전을 이용한 새로운 한국어 형태소 분석기”, 한국 컴퓨터 종합 학술대회 논문집 제 35 권, pp. 279-284, 2008

10. A. L. Berger, V. J. Della Pietra, S. A. Della Pietra, “*A maximum entropy approach to natural language processing*”, Computational Linguistics, Vol.22, No.1, pp.39-71, 1996

11. Avrim Blum, Tom Mitchell, “*Combining labeled and unlabeled data with co-training*” In Proceeding COLT' 98 Proceedings of the eleventh annual conference on Computational learning theory, Pages 92-100.

ABSTRACT

Korean Named Entity Recognition Using Semi-Supervised Learning

by JooGeun Kim

Dept. of Computer Engineering
Graduate School, Changwon National University
Changwon, Korea

We describe a new semi supervised learning method in conditional random fields (CRFs) framework for named entity recognition. We recognize that any category methods. We don't use learning corpus by using semi-supervised learning to generate a learning corpus. In addition, we automatically generate a gazetteer method. our method achieves 65.05% Without the need for learning corpus. From the results of experiment, we can see that the proposed method shows new method over the previous methods. Additionally, the proposed method can be applied to other applications easily since its implementation is independent on corpus.

부록 A. 품사 집합

TAG	POS	TAG	POS
NNG	일반명사	IC	감탄사
NNB	의존명사	VCP	긍정지정사
NNP	고유명사	VCN	부정지정사
NP	대명사	VV	동사
NR	수사	VA	형용사
JKS	주격조사	VX	보조용언
JKC	보격조사	EF	종결어미
JKO	목적격조사	EC	연결어미
JKG	관형격조사	ETN	명사형 전성어미
JKB	부사격조사	ETM	관형형전성어미
JKV	호격조사	EP	선어말어미
JKQ	인용격조사	SF	마침표, 물음표, 느낌표
JC	접속조사	SP	쉼표, 가운뎃점, 콜론, 빗금
JX	보조사	SS	따옴표, 괄호표, 줄표
XPN	명사접두사	SE	줄임표
XSN	명사파생접미사	SO	붙임표(물결, 숨김, 빠짐)
XSB	부사파생접미사	SL	외국어
XSV	동사파생접미사	SH	한자
XSA	형용사파생접미사	SN	숫자
XR	어근	NF	명사추정범주
MM	관형사	NV	용언추정범주
MAG	일반부사	SW	기타기호
MAJ	접속부사	NA	분석불능범주

부록 B. 구문태그 집합

- 구문표지

	범주	사례
S	문장	
Q	인용절	인용부호(“”) 안에 들어 있는 두 개 이상의 문장
NP	체언구	체언(명사, 대명사, 수사)
VP	용언구	용언(동사, 형용사, 보조용언)
VNP	긍정 지정사구	긍정 지정사 ‘이다’
AP	부사구	부사
DP	관형사구	관형사
IP	감탄사구	감탄사

- 기능표지

	범주	사례
SBJ	주어	주격 체언구, 명사 전성 용언구, 명사절 (NP_SBJ, VP_SBJ, S_SBJ)
OBJ	목적어	목적격 체언구, 명사 전성 용언구, 명사절 (NP_OBJ, VP_OBJ, S_OBJ)
CMP	보어	보격 체언구, 명사 전성 용언구, 인용절 (NP_CMP, VP_CMP, S_CMP)
MOD	체언 수식어	관형격 체언구, 관형형 용언구, 관형절 (NP_MOD, VP_MOD, S_MOD)
AJT	용언 수식어	부사격 체언구, 문말어미+부사격조사 (NP_AJT, VP_AJT, S_AJT)
CNJ	접속어	접속격 체언(NP_CNJ)
INT	독립어	체언(NP_INT)

- 기타표지

	범주	사례
RPN	삽입어구	삽입된 성분의 기능표지 위치에 표시(예: NP_PRN)
X	의사구 (pseudo phrase)	인용부호와 괄호를 제외한 나머지의 부호나, 조사, 어미가 단독으로 어절을 이룰 때 그 구문표지 위치 표시(예: X_CMP)
L, R	부호	인용부호나 괄호의 구문표지 위치에 표시. 왼쪽 부호에는 L 을, 오른쪽 부호에는 R 을 표시
Q, U, W, Y, Z	인용절	인용부호(“”)에 이끌려 나온 두 개 이상의 인용절을 대신하여 표기되는 부호

감사의 글

이 력 서

성　　명: 김 주 근
생년월일: 1986년 04월 15일
출 생 지: 부산광역시 사상구
주　　소: 경상남도 창원시 의창구 봉곡동 154-8번지

학 력

2005-20012: 창원대학교 공과대학 컴퓨터공학과(B.S.)
20012-2014: 창원대학교 대학원 컴퓨터공학과(M.S.)

발표논문

1. 김주근, 배원식, 차정원, BEOLTONG: 트위터 기반 정서분석 시스템, 제 22 회 한글 및 한국어 정보처리 학술대회(HCLT2010), pp. 107-111. 2010.10.