인지과학, 제27권 제1호
Korean Journal of Cognitive Science
2016, Vol. 27, No. 1, 27～41.

개체명 인식 코퍼스 생성을 위한 지식베이스 활용 기법*

박 영 민　　김 예 진　　강 상 우†　　서 정 연

서강대학교 컴퓨터공학과

개체명 인식은 미리 정의된 개체 범주로 텍스트의 요소를 분류하는 과정을 의미하며 최근 주목 받고 있는 음성 비서 서비스 등 다양한 응용 분야에 널리 활용되고 있다. 본 논문에서는 지식베이스를 사용하여 개체명 인식 코퍼스를 자동으로 생성하는 방법을 제안한다. 지식베이스의 종류에 따라 두 가지 방법을 적용하며 그 중 첫 번째 방법은 위키피디아를 기반으로 위키피디아 본문의 문장에 개체명 표지를 부착하여 학습 코퍼스를 생성하는 방법이다. 두 번째 방법은 인터넷으로부터 다양한 형태의 문장을 수집하고 다양한 개체들 간의 관계를 데이터베이스에 보유 중인 프리베이스를 이용하여 개체명 표지를 부착하는 방법으로 학습 코퍼스를 생성한다. 자동 생성된 학습 코퍼스의 질과 본 논문에서 제안하는 학습 코퍼스 자동 생성 기법을 평가하기 위해 두 가지로 실험했다. 첫 번째, 다른 형태의 지식베이스인 위키피디아와 프리베이스(Freebase)를 기반으로 생성된 학습 코퍼스의 표지 부착 성능을 수동으로 측정하여 코퍼스의 질을 평가하였다. 두 번째, 각 코퍼스로 학습된 개체명 인식 모델의 성능을 통해 제안하는 학습 코퍼스 자동 생성 기법의 실용성을 평가하였다. 실험을 통해 본 방법이 타당함을 증명하였으며 특히 실제 응용에서 많이 사용되는 웹 데이터 환경에서 의미 있는 성능 향상을 보여주었다.

주제어 : 개체명 인식, 준지도 학습, 지식베이스, 코퍼스 생성

---

* 본 연구는 미래창조과학부 및 정보통신기술진흥센터의 SW중심대학지원사업의 연구결과로 수행되었음(2016-SW-01).

† 교신저자: 강상우, 서강대학교 컴퓨터공학과, 서울특별시 마포구 백범로 35 R904
연구분야: 자연어처리
Tel: 02-706-8954, E-mail: gahng.sw@gmail.com

서 론

개체명(Named Entity)은 인명, 기관명, 지명 등과 같이 고유명사나 일반적인 사전에 등록되지 않은 단어를 의미한다. 이는 정보 추출(Information Extraction)의 응용 분야에서 사용되고 있으며 1990년대 정보추출 연구 학술대회인 MUC-6(Sixth Message Understanding Conference)에서 유래되었다. MUC의 주목적은 비정형화된 텍스트에서 보안 목적의 정보를 추출하는 것이었으나 이 과정을 위해 개체명 인식(Named Entity Recognition)이 필수적임을 인지하게 되었고 현재 정보 추출 분야의 중요 연구 과제 중 하나로 활발히 연구가 진행 중이다. 정보 추출은 비정형적인 문장으로부터 유용한 정보를 추출하는 자연어 처리(Natural Language Processing) 및 텍스트 마이닝(Text Mining) 분야의 주요 연구 분야 중 하나이고, 개체명 인식은 미리 정의된 개체 종류 별로 텍스트의 요소를 분류하는 과정을 의미한다. 이는 문서 내에 존재하는 다양한 개체명의 부류를 인식하는 작업 자체만으로도 의미가 있으며 관계 추출 및 대용어 참조 해소와 같은 상위 작업을 위해서 중요한 역할을 한다(Godbole et al., 2015; Mintz et al., 2009).

기존의 개체명 인식에 대한 연구는 다양한 기계학습 기법을 이용하여 진행 되어 왔다. 지도 학습법(Supervised Learning)은 특징 기반의 기법, 커널 기반 기법 등으로 사람이 태깅(Tagging) 작업을 직접 수행하는 과정이 필요하다. 최근에는 지도 학습을 위해 소비되는 비용을 최소화하기 위한 준지도 학습(Semi-Supervised Learning) 기법에 대한 연구가 진행되고 있다. 일반적인 준지도 학습 기법은 태깅된 소량의 초기 데이터를 사용하여 태깅 되지 않은 다량의 학습 코퍼스를 추출해내는 방법이다. 하지만 이러한 준지도 학습 방법은 여전히 초기 데이터에 표지를 수작업으로 부착해야 하는 번거로움이 따르며, 양질의 초기 데이터를 선택하기 위한 추가적인 작업이 불가피하다. 이로 인해 준지도 학습 방법 중 지식베이스(Knowledge Base)를 이용하여 자동 학습을 하는 준지도 학습 기법인 거리 통제(Distant Supervision)가 주목 받고 있다(Blum et al., 2015). 거리 통제는 관계 추출에 대표적으로 이용되는 학습 기법으로 지도 학습과 준지도 학습 방법의 몇 가지 이점들을 결합한 방법이다. 본 연구에서는 거리 통제 학습 기법을 개체명 인식에 적용하여 개체명 표지를 지식베이스 기반으로 다량의 텍스트 데이터에 부착하는 방법을 제안한다.

관련 연구

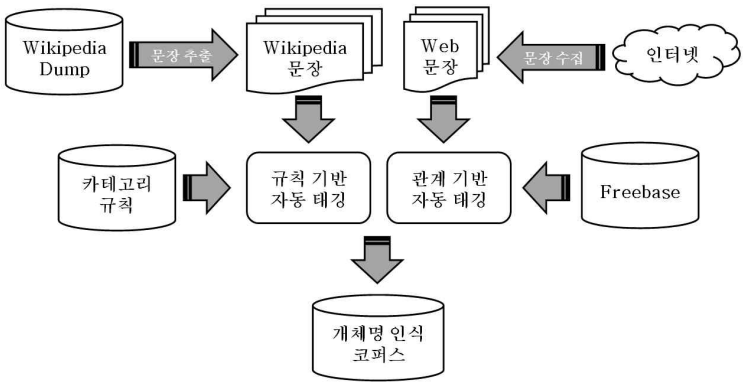
개체명 인식을 위한 학습 코퍼스 생성 방법으로는 다양한 방법이 연구되고 있으며, 대표적인 기계 학습 방법으로는 지도, 준지도, 거리 통제 방법이 있다. 개체명 태깅은 사람의 수작업을 통한 태깅이나 소량의 태깅된 데이터로부터 태깅 되지 않은 데이터를 자동 태깅시키는 부트스트래핑(Bootstrapping) 태깅 방법 또는 지식베이스(Knowledge Base)를 이용한 태깅 방법을 이용한다. 이로부터 태그가 부착된 학습 코퍼스를 얻은 후 특징을 추출하고 이를 분류기를 통해 학습시켜 개체명 인식 모델을 생성한다. 생성된 개체명 인식 모델에 새로운 데이터가 입력되면 모델은 개체명 태그가 부착된 결과를 내놓게 된다.

개체명 인식을 위한 초기의 지도 학습 방법은 사람이 수작업으로 만든 규칙(rule)을 기반으로 하였으나 이후에는 자동적으로 규칙을 생성하는 규칙 기반 시스템 또는 시퀀스 레이블링(Sequence Labeling) 알고리즘을 이용한 방법으로 발전하였다. 개체명 인식에 효과적으로 사용된 시퀀스 레이블링[1] 알고리즘으로는 초기 지도 학습인 은닉 마르코프 모델(Hidden Markov Model) 외에도 최근 영향력 있는 기계 학습 기법으로 결정 트리(Decision Tree), 최대 엔트로피 모델(Maximum Entropy Model), 지지 벡터 머신(Support Vector Machine), Conditional Random Fields(CRFs) 등이 있다(Asahara & Matsumoto, 2003; Bikel et al., 1997; Borthwick et al., 1998; McCallum & Li, 2003; Sekine, 1998). 위와 같은 모델들은 개체명 태그가 부착된 다량의 학습 코퍼스, 개체명 사전과 함께 중의적인 태그 문제를 해결 할 수 있는 규칙들이 필요하다. 다시 말해 지도 학습을 이용한 태깅 방법은 다량의 학습 코퍼스를 수작업으로 구축하여야하고 새로운 언어 현상, 개체명 등을 반영하기 위해 지속적인 추가 작업을 필요로하기 때문에 고비용의 구조를 갖는 한계가 있다.

거리 통제는 새롭게 제안된 준지도 학습 방법으로 사전에 구축된 지식베이스(Knowledge Base)의 정보들을 참조하여 훈련 데이터 셋에 대한 정답을 반자동으로 태깅하여 학습을 진행하는 방법이다. 이러한 방식을 이용하여 지도 학습에서의 단점인 훈련된 사람이 수작업으로 태깅해야 하는 비용 및 시간을 절약할 수 있

1) 레이블링(labling)과 태깅(tagging)은 동일한 의미로 사용

다. 거리 통제는 다양한 자연어처리 분야에 응용할 수 있으며 특히 관계 추출 코퍼스 생성 분야에서 뛰어난 성능을 보여주었다(Mintz, 2009). 거리 통제를 이용한 관계 추출 코퍼스 생성 기법은 지식베이스를 활용한다. 예를 들어 <Microsoft, Organizations founded, Bill Gates>라는 트리플이 지식베이스에 있다고 가정 한다면 수집된 문장에서 Microsoft와 Bill Gates가 함께 포함된 문장들은 모두 Organizations founded 관계로 가정하여 태깅을 수행한다. 이 외에 거리 통제를 사용하여 관계 추출 분야와 다른 방향으로 진행된 연구가 다양하다. 두 개체 사이의 의미관계에 시간 개념을 도입하는데 Garrido의 연구에서 거리 통제 기법을 사용하였으며, Surdeanu는 특정 관계의 다양한 정답 요소를 인정하여 2개 이상의 정답을 부여할 수 있는 방식의 접근을 통한 연구를 진행하였다(Garrido et al., 2012; Surdeanu et al., 2012). 또한 Surdeanu는 같은 접근 방식을 슬롯 채우기 문제(Slot Filling Task)에도 적용하여 성능 향상의 효과를 거두었다(Surdeanu et al., 2010).



(그림 1) 전체 시스템 구조

학습 코퍼스 자동 생성 기법

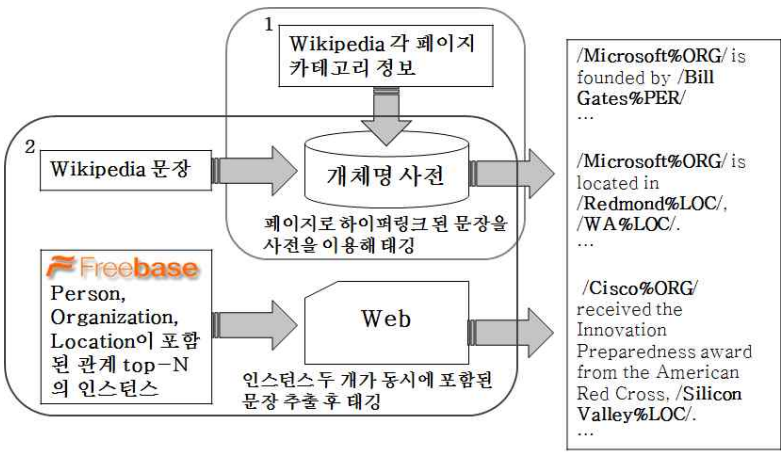
본 논문에서 제안하는 전체 시스템의 구조는 (그림 1)과 같이 지식베이스를 기반으로 크게 위키피디아 기반 방법과 프리베이스 기반 방법으로 구분된다. 위키피

〈표 1〉 개체명 종류에 따른 분류 규칙

학습 코퍼스	개체종류
Person	Births, Deaths, Living people, Occupations을 나타냄, Fictional characters를 포함, Redirect하는 위키피디아 페이지가 인명을 가리키는 경우
Organization	Military, Organizations, Corps, Censorship, Television stations, Companies, Council, Subsidiary organs, Agencies 등의 Organization 정보를 포함, Redirect하는 위키피디아 페이지가 Organization을 가리키는 경우
Location	Settlement type, Geology, Valleys, Planets, Countries, Continents, Stations, Airports, Oceans, River, Geography, Mountain 등의 위치 정보를 포함, Redirect하는 위키피디아 페이지가 Location을 가리키는 경우

디아(en.wikipedia.org)를 기반으로 생성하는 방법은 규칙 기반 반자동 태깅을 이용하며 프리베이스(www.freebase.com)를 기반으로 생성하는 방법은 관계 기반 자동 태깅을 이용한다. (그림 2)는 본 논문에서 제안하는 코퍼스 자동 생성의 예를 보여준다.

첫 번째 학습 코퍼스 생성 방법은 구축된 개체명 컨셉 사전을 이용하여 위키피디아의 모든 문장들을 태깅하는 방법이다. 위키피디아의 문장들에는 다른 페이지 엔트리로 연결되는 링크가 존재한다. 이를 이용하여 링크된 부분을 개체명 컨셉



(그림 2) 학습 코퍼스 생성 예

사전과 비교하여 인명(PER), 기관명(ORG), 지명(LOC)과 일치하는 경우 태깅한다. (그림 2)의 1번 부분이 이 작업에 해당한다. 2번은 지식베이스에 따라 두 가지 방향으로 진행되는 학습 코퍼스 생성 방법을 나타낸다. 위키피디아 본문의 모든 문장 추출은 Evan Jones의 Extracting Text from Wikipedia(wikipedia2text)를 이용한다 (Evan, 2009). XML 형태의 파일에는 위키피디아의 페이지에는 텍스트 이외의 그림, 표 등의 정보가 포함되어 있으므로 wikipedia2text를 이용하여 모든 페이지로부터 텍스트 이외의 정보를 제거한 데이터를 추출한다. 추출 후에는 텍스트를 문장 단위로 분할하기 위해 WikiXMLSAXParser[2]를 이용한다. 위키피디아 컨셉 사전은 각 페이지 엔트리 하단에 포함된 분류 정보를 이용한다. 분류 정보 내에는 해당 페이지를 나타내는 특징을 표현하는 주요 단어들로 이루어져 있다. <표 1>은 본 논문에서 구축한 개체명 컨셉 사전의 개체명 종류와 각 개체명에 따른 분류 규칙을 보여준다.

두 번째 방법은 프리베이스를 지식베이스로 활용한다. 프리베이스는 위키피디아, NNDB(Notable Names Database)를 비롯한 다양한 자원으로부터 수집한 데이터를 정리한 지식베이스이다. 각 엔트리는 어떤 두 개체와 그 사이의 관계를 표시하고 있다. 예를 들어 <Petrus Bertius, Place of birth, Beberan> 엔트리에서 두 개체 Petrus Bertius와 Beberan은 Place of birth의 관계를 갖는다는 의미이고, 각 개체는 Person/Person의 타입으로 표시된다. 현재 프리베이스는 인명, 기관명, 지명뿐만 아니라 영화명, 음악명 등 다양한 개체들 간의 관계가 수집되는 중이며 2015년 6월 기준으로 약 29억개의 개체로 구성되어 있다. 본 논문에서는 먼저 프리베이스의 관계에서 인명(PER), 기관명(ORG), 지명(LOC)의 개체를 포함하고 있는 관계들 중 각 개체별로 인스턴스를 많이 보유하고 있는 상위 N개의 관계를 선택한다. 상위 N개의 관계를 추출하기 위해 프리베이스 이지(Freebase Easy)[3]를 사용하며 프리베이스 이지의 Query를 인스턴스 타입 즉 개체명 타입으로 입력하면 입력한 타입이 포함된 관계들을 결과로 얻을 수 있다. 예를 들어 프리베이스 이지에 Query를 Person으로 입력하면 인명(PER) 타입을 포함하고 있는 관계를 인스턴스가 많은 순으로 정렬해서 보여준다. Person의 경우 Gender 관계 1,976,747개, Date of birth 관계

2) Wikipedia XML SAX Parser, https://code.google.com/p/wikixmlj/

3) 프리베이스 검색 엔진, http://freebase-easy.cs.uni-freiburg.de

1,274,974개, Profession 관계 999,587개 Place of birth 관계 885,071개, Country of nationality 관계 775,486개의 인스턴스를 보유하고 있으므로 인명(PER) 타입을 포함한 상위 5개의 관계임을 알 수 있다. 프리베이스의 관계에서 상위 N개의 관계를 추출한 후 각 관계에 있는 인스턴스가 포함 되어있는 문장을 웹(Web)[4]으로부터 수집한 후 프리베이스에 정의된 개체 타입으로 태깅한다. 예를 들어 웹에서 "Micorosoft is located in Redmond WA ..." 라는 문장이 추출되었고 상위 N개의 관계 중 /Microsoft%ORG/-locate-/Redmond WA%LOC/ 관계가 포함되어 있다면 Microsoft와 Redmond WA는 locate 관계를 가질 확률이 높기 때문에 각각 ORG와 LOC으로 태깅한다.

여기서 두 개의 자동 태깅 방법을 비교하면, 위키피디아 기반의 방법은 편집자들이 직접 태깅을 하였기 때문에 비교적 정확한 성능을 보장한다. 반면 위키피디아는 대부분 정형화된 문장들로 이루어져 있기 때문에 다양한 문장 형태를 추출하기 어렵다. 프리베이스 기반의 방법의 경우 거리 통제의 가정을 이용한 것이기 때문에 상대적으로 오류가 발생할 확률은 높지만 다양한 형태의 문장을 수집할 수 있다는 장점이 있다. 따라서 두 방법은 상호 보완적인 관계에 있다고 할 수 있다.

CRFs를 이용한 개체명 인식

제안하는 코퍼스 생성 기법이 효과적이라는 것을 증명하기 위해서는 생성된 코퍼스를 기존의 개체명 인식 모델의 학습에 적용시켰을 때 성능 향상이 있음을 보여야 한다. 따라서 본 논문에서는 기존의 개체명 인식 모델 중 뛰어난 성능을 제공하는 것으로 알려진 CRFs(Conditional Random Fields)를 사용한다(심광섭, 2011; McCallum & Li, 2003). CRFs는 입력 데이터 열의 조건부 확률 값을 계산하기 위한 비방향성 그래프 모델이다. CRFs 모델은 은닉 마르코프 모델의 단점인 독립 가정을 해결하여 다양한 특징(feature)을 사용할 수 있으며 최대 엔트로피 마르코프 모델(Maximum Entropy Markov Model)의 단점인 레이블 편향 문제를 완화할 수 있

4) Google에서 제공하는 데이터를 사용, https://code.google.com/p/relation-extraction-corpus/

는 장점이 있다. $\mathrm{x}=x_1 \cdots x_n$를 입력 데이터 열에 대한 확률 변수라고 하고, $\mathrm{y}=y_1 \cdots y_n$를 입력 데이터 열에 대응하는 표지 열의 화률 변수라고 하면 매개 변수 $\Lambda=(\lambda_1, \lambda_1, ..., \mu_1, \mu_2, ...)$를 갖는 선형 체인 구조의 CRFs는 다음과 같은 조건부 확률로 정의된다(Peng et al., 2004).

$$P_\Lambda(\mathrm{y}|\mathrm{x}) = \frac{1}{Z(x)} \exp\left(\sum_{\mathrm{j}} \sum_{i=1}^{\mathrm{n}} \lambda_j t_j(y_{i-1}, y_i, \mathrm{x}, i) + \sum_k \sum_{i=1}^{n} \mu_k s_k(y_i, \mathrm{x}, i)\right) \quad (1)$$

여기서 $Z(x)$는 입력 데이터 열에 대한 표지 열의 확률 값의 합이 1이 되도록 하는 정규화 상수이다. $t_j(y_{i-1}, y_i, \mathrm{x}, i)$는 전이 특징 함수(transition feature function)이며, $s_k(y_i, \mathrm{x}, i)$는 상태 특징 함수(state feature function)이다. λ_j와 μ_k는 각 특징 함수에 대한 가중치로서 태깅이 된 학습용 데이터로부터 구할 수 있다. 매개변수 Λ는 MLE(Maximum Likelihood Estimation)를 사용하여 구하는데, 다른 알고리즘 보다 수렴 속도가 빠른 BFGS(Broyden Fletcher Goldfarb Shanno)알고리즘이 주로 사용된다. 학습 코퍼스로부터 매개 변수 Λ를 구하고 나면, 주어진 입력 데이터 열 x에 대하여 가장 가능성이 높은 열 y^*은 다음과 같이 구할 수 있으며 y^*는 동적 프로그래밍 기법인 Viterbi 알고리즘에 의해 계산된다(Peng et al., 2004).

$$\mathrm{y}^* = \mathrm{argmax}_{\mathrm{y}} P_\Lambda(\mathrm{y}|\mathrm{x}) \quad (2)$$

본 논문에서 사용한 특징은 다음과 같다.

- 단어 특징 : $(i-2, i-1, i, i+1, i+2)$ 위치에 해당하는 단어 정보
- 품사 특징 : $(i-2, i-1, i, i+1, i+2)$ 위치에 해당하는 품사 정보
- 접두사/접미사 특징 : (i) 위치에 해당하는 단어의 접미사/접두사 N-gram, 여기서 N은 3이 사용됨. 예를 들어 “Young” 라는 단어에 대해 “<Y”. “<Yo”, “<You”, “ung>”, “ng>”, “g>” 와 같은 특징들을 추출함.
- 단어 패턴 특징 : (i) 위치에 해당하는 단어의 문자열 패턴. 패턴은 대문자(X), 소문자(x), 특수문자(-), 숫자(#)를 사용. 예를 들어 “Peir-39” 라는 단어에 대해 “Xxxx-##”와 같은 특징을 추출함.

실험 및 평가

본 장에서는 지식베이스에 따른 개체명 태깅 성능과 추출된 코퍼스에 의해 학습된 개체명 인식 모델에 대한 성능을 비교 평가한다. 실험 데이터는 ontoNotes[5]에서 제공하는 태깅된 데이터 약 1M개의 문장과 위키피디아 기반 방법으로 태깅한 약 1M개 문장 그리고 프리베이스 기반 방법으로부터 태깅한 약 6k 문장이다. 개체명 인식을 위한 모델은 ontoNotes 데이터로만 학습시킨 ontoNotes 모델, 위키피디아로부터 추출한 문장으로 학습시킨 Wikipedia 모델, 웹으로부터 추출한 문장들로 학습시킨 프리베이스 모델, 세가지 영역의 학습 코퍼스의 모두 학습시킨 All 모델들로 총 4개의 개체명 인식 모델을 생성하였다. 이때 코퍼스 중 각 영역 별로 200문장씩, 총 600문장을 추출하여 수동 태깅 후 테스트 문장으로 사용하였다. 추출된 600개의 테스트 문장 중 특히 웹에서 추출한 200문장은 기존의 개체명 인식 코퍼스에 비해 새로운 언어현상, 새로운 개체명들이 포함되기 때문에 제안하는 모델의 장점을 측정하는데 효과적이라고 할 수 있다.

코퍼스 자동 생성 기법의 태깅 성능은 <표 2>와 같다. 제안 방법으로 생성된

〈표 2〉 지식베이스에 따른 개체명 태깅 성능

지식베이스	개체종류	Precision	Recall	F_1-Measure
Wikipedia	Person	1	0.6082	0.7563
	Organization	0.9194	0.5249	0.6682
	Location	0.8640	0.4611	0.6013
	Average	0.9404	0.5452	**0.6898**
Freebase	Person	0.9854	0.5232	0.6835
	Organization	0.9183	0.1196	0.2117
	Location	0.8461	0.1633	0.2738
	Totals	**0.9496**	0.3821	0.5203

5) Linguistic Data Consortium에서 제공하는 코퍼스, https://catalog.ldc.upenn.edu/LDC2013T19

코퍼스로 학습한 모델들은 재현율(Recall)은 다소 낮았으나 평균적으로 약 94% 이상의 높은 정확률(Precision)을 보였다. 이러한 현상은 제안하는 방법의 특성에서 기인하는 것이라고 할 수 있다.

위키피디아 기반 방법의 경우 문서의 링크들이 비교적 정확하게 태깅 되어 있어 정확률이 높은 경향을 보였지만 편집자들이 모든 개체명에 대해 태깅을 하지 않는다는 점이 재현율을 낮추는 원인이 되었다. 프리베이스 기반의 방법은 프리베이스의 관계있는 두 개체가 동시에 나타난다면 높은 확률로 해당 개체명이라고 할 수 있으나 개체명이 한 개만 출현한 경우 또는 서로 관계가 없는 개체명들이 출현한 경우는 태깅하지 못하는 점이 재현율을 낮추는 원인이 되었다. 두 문제점을 비

〈표 3〉 학습 코퍼스에 따른 개체명 인식 모델 성능

학습코퍼스	개체종류	Precision	Recall	F1-Measure
ontoNotes	Person	0.8199	0.6299	0.6278
	Organization	0.3453	0.5597	0.4271
	Location	0.6258	0.6299	0.6278
	Average	0.5468	**0.6079**	0.5757
Wikipedia	Person	0.9291	0.3471	0.5054
	Organization	0.8113	0.2935	0.4311
	Location	0.7826	0.2922	0.4255
	Average	0.8488	0.3124	0.4562
Web	Person	0.6792	0.3176	0.4329
	Organization	0.8750	0.0239	0.0465
	Location	0.6333	0.0617	0.1124
	Average	0.6802	0.1424	0.2355
All	Person	0.9247	0.5059	0.6540
	Organization	0.9270	0.4334	0.5907
	Location	0.8639	0.4123	0.5582
	Average	**0.9064**	**0.4527**	**0.6038**

〈표 4〉 학습 코퍼스에 따른 개체명 인식 모델 문장단위 성능

학습코퍼스	ontoNotes	Wikipedia	Web	All
exactly score	0.4983	0.3616	0.1550	**0.5033**

교하면 위키피디아 기반의 방법은 문서의 질에 대한 문제이고 프리베이스 기반의 방법은 방법론의 한계라고 할 수 있다.

<표 3>은 실제 개체명 인식 모델에 각 코퍼스를 적용한 후 측정한 성능이다. 또한 <표 4>는 테스트 문장 중 한 문장의 모든 개체명을 정확하게 맞춘 수의 비율이다. 개체명 인식 모델의 ontoNotes 코퍼스의 경우 사람이 직접 태깅한 학습 코퍼스로 내부 데이터로만 성능을 평가한 경우 모든 개체 분류에서 90%이상의 성능을 보였다. 하지만 위키피디아와 웹으로부터 추출한 문장들과 함께 테스트한 결과 성능이 현저히 떨어짐을 볼 수 있다. 이러한 현상의 원인은 기존의 수동으로 구축된 코퍼스가 새로운 언어현상을 반영하지 못하는 것으로 분석된다. 또한, 새로운 개체명들도 성능하락에 크게 작용한 것을 알 수 있다. 즉 높은 성능의 개체명 인식모델을 유지하기 위해서는 계속해서 발생되는 문장들을 태깅하는 작업을 필요로 한다. 위키피디아 모델의 경우 ontoNotes 모델과 비교해서 매우 높은 정확률을 보여주었지만 상대적으로 낮은 재현율을 나타내고 있다. 이것은 앞서 <표 2>에서 언급한 개체명 태깅 방식의 낮은 재현율이 반영된 것으로 설명할 수 있다. 프리베이스 모델의 경우 위키피디아 모델과 유사한 경향을 보였지만 전체적으로 성능이 더 낮은 결과를 보였다. 이러한 현상은 추출된 문장의 수가 더 적다는 점(위키피디아-1백만 문장, 프리베이스-약 6천 문장)이 결정적으로 작용하였다. 하지만 프리베이스 모델은 웹에서 추출한 문장에 대해서는 가장 높은 성능을 보여주었기 때문에 의미 있는 문장들이 수집되었다고 할 수 있다. 마지막으로 세 코퍼스를 모두 합하여 학습한 All 모델의 성능은 가장 높은 F_1-Measure를 보여주었는데 이것은 제안 방법이 수동 태깅 코퍼스의 단점을 어느 정도 보완해 줄 수 있다고 할 수 있다.

결 론

본 논문에서는 지식베이스를 활용하여 자동으로 개체명 인식 코퍼스를 생성하는 두 가지 방법을 제안하였다. 첫 번째 방법은 위키피디아 문서들의 태깅정보를 이용하는 방법으로서 비교적 높은 정확률을 보여주었다. 두 번째 방법은 거리 통제를 적용하여 프리베이스를 이용해 웹에서 수집된 문장에 태깅을 하는 방법이다. 이 방법은 기존의 수동 태깅이나 위키피디아와 비교하여 최근에 생성된 문장에서 더 좋은 성능을 보여주었다. 또한 수동 태깅된 ontoNotes와 위키피디아, 프리베이스 모델의 코퍼스를 모두 합쳤을 때 가장 높은 성능을 보여주는 것으로 제안하는 방법이 수동 코퍼스의 단점을 보완해준다는 것을 확인하였다.

하지만 제안하는 모델들은 몇 가지 문제점을 가지고 있다. 위키피디아 기반 방법은 편집자가 엄밀히 태깅하지 않기 때문에 낮은 재현율을 발생시킨다는 점과 백과사전이라는 특성 상 문장이 정형적이기 때문에 일반 문장들에서는 낮은 성능을 보인다. 거리 통제 기반 방법은 문장에서 한 개의 개체명만 출현하거나, 서로 관계가 없는 개체명이 출현한 경우 태깅을 하지 못한다는 문제가 있다. 향후 지속적인 연구를 통해 이러한 문제를 해결한다면 수동 태깅에 가까운 성능을 제공하는 개체명 인식 코퍼스를 생성할 수 있을 것이다.

참고문헌

심광섭 (2011). CRF를 이용한 한국어 자동 띄어쓰기. **인지과학, 22**(2), 217-233.

Asahara, M., & Matsumoto, Y. (2003). Japanese named entity extraction with redundant morphological analysis. *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, 8-15.

Bikel, D. M., Miller, S., Schwartz, R., & Weischedel, R. (1997). Nymble: a high-performance learning name-finder. *Proceedings of the fifth conference on Applied natural language processing,* 194-201.

Blum, A. (2015). *Semi-supervised Learning.* Springer, 1-7.

Borthwick, A., Sterling, J., Agichtein, E., & Grishman, R. (1998). NYU: Description of the MENE named entity system as used in MUC-7. *Proceedings of the 7th Seventh Message Understanding Conference.*

Evan J. (2009). Generating a plain text corpus from Wikipedia(Wikipedia2text), http://blog.afterthedeadline.com/2009/12/04/generating-a-plain-text-corpus-from-wikipedia/

Garrido, G., Penas, A., Cabaleiro, B., & Rodrigo, A. (2012). Temporally anchored relation extraction. *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics,* 107-116.

Godbole, V., Liu, W., & Togneri, R. (2015). An Investigation of Neural Embeddings for Coreference Resolution. *Computational Linguistics and Intelligent Text Processing,* 241- 251.

McCallum, A., & Li, W. (2003). Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. *Proceedings of the seventh conference on Natural language learning at HLT-NAACL,* 188-191.

Mintz, M., Bills, S., Snow, R., & Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP,* 1003-1011.

Peng, F., Feng, F., & McCallum, A. (2004). Chinese segmentation and new word detection using conditional random fields. *Proceedings of the 20th international conference on Computational Linguistics,* Article No.562.

Sekine, S. (1998). NYU: Description of the Japanese NE system used for MET-2. *Proceedings of the 7th Message Understanding Conference.*

Surdeanu, M., McClosky, D., Tibshirani, J., Bauer, J., Chang, A. X., Spitkovsky, V. I., & Manning, C. D. (2010). A simple distant supervision approach for the TAC-KBP slot filling task. *Proceedings of Text Analysis Conference Workshop.*

Surdeanu, M., Tibshirani, J., Nallapati, R., & Manning, C. D. (2012). Multi-instance multi-label learning for relation extraction. *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning,* 455-465.

1차원고접수 : 2015. 08. 10
1차심사완료 : 2015. 11. 10
2차원고접수 : 2015. 12. 14
2차심사완료 : 2016. 03. 07
3차원고접수 : 2016. 03. 16
최종게재승인 : 2016. 03. 17

(*Abstract*)

Automatic Training Corpus Generation Method of Named Entity Recognition Using Knowledge-Bases

Youngmin Park Yejin Kim Sangwoo Kang Jungyun Seo

Computer Science and Engineering Sogang University

Named entity recognition is to classify elements in text into predefined categories and used for various departments which receives natural language inputs. In this paper, we propose a method which can generate named entity training corpus automatically using knowledge bases. We apply two different methods to generate corpus depending on the knowledge bases. One of the methods attaches named entity labels to text data using Wikipedia. The other method crawls data from web and labels named entities to web text data using Freebase. We conduct two experiments to evaluate corpus quality and our proposed method for generating Named entity recognition corpus automatically. We extract sentences randomly from two corpus which called Wikipedia corpus and Web corpus then label them to validate both automatic labeled corpus. We also show the performance of named entity recognizer trained by corpus generated in our proposed method. The result shows that our proposed method adapts well with new corpus which reflects diverse sentence structures and the newest entities.

Key words : Named Entity Recognition, Semi-supervised Learning, Knowledge Base, Corpus Generation.