Bidirectional Dynamic LSTM 을 이용한 음절 단위 개체명 추출 및 자동화된 말뭉치 구축

오성식○, 임창대, 안기호, 박외진
(주)아크릴
{Harvey, Dann, Kevin, Jin}@iacryl.com

Syllables-based Named Entity Extraction and Automatic Corpus Construction using Bidirectional Dynamic LSTM

Sungsik Oh○, Changdae Lim, Keeho Ahn, Weijin Park
Acryl Inc.

요 약

개체명 인식은 자연어 문장에서 장소, 제작물, 사람 등 분류를 통한 의미 부여가 가능한 단어를 파악하는 기술로서 의미 분석을 위한 핵심 기술이다. 현재 많은 개체명 분석 관련 연구들은 형태소 분석 결과에 의존적인 형태를 갖고 있어서, 형태소 분석 결과의 정확성이 개체명 분석 결과의 성능에 영향을 미치고 있다. 본 연구에서는 형태소 분석 과정을 거치지 않는 음절 기반의 개체명 분석 기술을 제안하여 형태소 분석의 정확도가 낮은 통신어, 신조어 분석 성능을 향상하였다. 또한, 자동화된 방법으로 음절 단위 개체명 말뭉치 및 개체명 사전을 구축하는 프로세스를 정의하여 개체명 분석의 정확도 향상 및 인지 범주의 확대를 도모하였다. 본 연구에서 제안한 개체명 인식 기술은 한국어 개체명 표준에 기반한 129가지의 개체명 분류가 가능하며, 이는 자연어 처리 기술이 필요한 산업계에서 상용화하는데 큰 기여를 할 것으로 판단된다.

주제어: 음절 단위 개체명 분석, Bidirectional dynamic LSTM, 개체명 인식 말뭉치 구축, 개체명 사전 자동생성

1. 서론

개체명 인식(Named Entity Recognition)은 자연어 문장에서 장소, 제작물, 사람 등 분류가 가능한 의미를 단어에 부여하는 기술로서 자연어 처리에서 의미 분석의 기반이 되는 기술이다.

개체명 인식은 자연어 처리의 중요 분야 중 하나로서 시계열 데이터 처리에 용이한 RNN(Recurrent Neural Network)을 이용한 순차 데이터 처리가 주로 행해진다. 최근에는 RNN의 Memory문제를 해결한 LSTM(Long Short-Term Memory)을 이용한 자연어 처리가 널리 연구되고 있으며, RNN의 양방향 순차 데이터 학습 방식인 BRNN(Bidirectional Recurrent Neural Networks)을 이용하여 순차 데이터 학습의 정확성을 향상 한다[1,2]. RNN 구현 시 입력 값의 고정된 길이를 맞추기 위하여 실제 입력 값과 맞지 않는 부분을 빈(NULL)값으로 채워 넣는데, 이는 학습 데이터의 일관성을 낮추고 정확도를 저하시키는 원인이 된다. 이런 문제를 해결하기 위하여 DRNN(Dynamic Recurrent Neural Networks)를 이용하여 가변적인 길이의 입력을 가능하게 한다[3].

기존 개체명 인식 연구 사례는 형태소 분석 후 분석된 형태소 중 개체명으로 인식될 수 있는 것을 추출하여 개체명을 분류 하였다[4]. 이러한 방법은 분석될 개체명을 추출하여 분석하므로 분석할 정보량을 비약적으로 줄이는 경제적인 방법이나 형태소 분석기의 정확도에 따라 개체명 추출이 불가능한 경우가 있다. 특히 통신어, 신조어 분석에서 낮은 정확도를 보인다. 이런 문제를 해결하기 위해 본 논문에서는 형태소 분석기에 의존하지 않는 개체명 분석기를 개발하고 음절 단위 개체명 말뭉치와 동시에 개체명 사전을 구축하는 시스템을 개발 하였다.

본 논문에서는 한국어 개체명 인식에 BRNN, LSTM, DRNN을 병합하여 Bidirectional Dynamic LSTM을 이용한 음절 단위로 개체명을 추출하는 방식을 제안한다. 현재 개체명 분석 연구는 사람, 기관, 시간 등 주요한 10가지 이하의 개체명 분류를 대상으로 하고 있는 경우가 많다. 그러나, 실제 의미 분석을 위해서는 모든 개체명 분류를 하는 것이 필수적이며, 이를 위해 본 논문에서는 실제 개체명 분류를 모든 텍스트에서 사용하기 위해 한국어 개체명 표준(개체명 태그 세트 및 태깅 말뭉치(Tag Set and Tagged Corpus for Named Entity Recognition, TTAK.KO-10.0852))을 이용하여 개체명 분류의 분석을 수행하였다.

본 논문의 구성은 다음과 같다. 2장 에서는 관련 연구에 대해 설명한다. 3장에서는 Bidirectional Dynamic LSTM을 이용한 음절 단위 개체명 추출 방법에 대하여 설명한다. 4장에서는 실험을 통해 제안 방법에 대하여 평가한다. 5장에서는 실험 내용 분석과 차기 연구 방향에 대하여 서술한다.

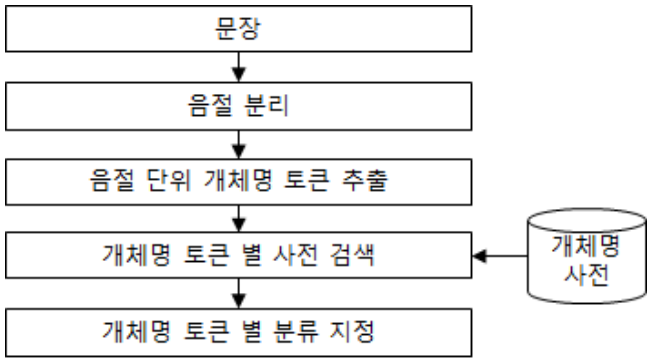
2. 관련 연구

현재까지 다양한 개체명 분석과 음절 단위 자연어 처리에 대한 연구들이 진행되었다[5-7]. [5]는 음절 단위 형태소 분석을 Bi-LSTM-CRF를 이용하여 분석하였다. 또한, [6]은 LSTM을 이용하여 개체명 분석을 시도하였으며, [7]은 Bi-LSTM-CRF를 이용하여 단어 임베딩 벡터, 품사 임베딩 벡터, 음절 입력을 이용하는 방법을 제안하였다.

3. 제안 방법

3.1 분석 시스템 구성

음절 기반의 개체명 태깅 시스템은 (그림 1)과 같은 전처리 과정과 인공 신경망의 예측 결과를 거쳐서 음절별 개체명 토큰 포함 여부를 결정하게 된다.

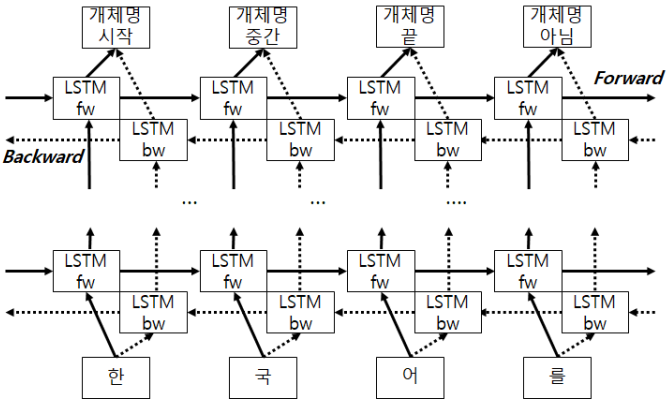


(그림 1). 음절 기반의 개체명 태깅 시스템 구성도

입력 문장을 음절 단위로 분리하여 사전에 구성되어 있는 음절 별 참조 번호로 변환 한다. 따라서, 입력 문장은 참조 번호 수열로 변환되며 이 수열은 Bidirectional Dynamic LSTM과 같은 기계학습 기반 분류기에 입력된다. 그리고 음절 별로 개체명 토큰 포함 여부를 분류 한다. 토큰으로 판명된 음절 토큰은 개체명 사전을 참조하여 개체명 토큰 별로 분류를 지정한다.

3.2 Bidirectional Dynamic LSTM을 이용한 개체명 추출 방식

입력 음절을 참조 번호로 변환하여 Bidirectional Dynamic LSTM에 입력한다. 본 과정에서, 입력 음절의 distributed representation 의 성질을 이용하기 위하여 벡터로 변환하는 음절 임베딩 알고리즘을 이용하지 않았는데, 이는 Bidirectional RNN의 각 LSTM Cell의 stack을 복수로 쌓으면 distributed representation이 자동으로 반영된다고 판단한 것에 기인한다. 개체명 추출을 위한 Bidirectional Dynamic LSTM에서는 5개의 Cell Stack을 이용하였다. 각 입력 음절에 대하여 개체명 토큰 포함 여부를 예측 하게 하였는데 '㉠ 개체명 시작', '㉡ 개체명 중간', '㉢ 개체명 끝' 으로 음절 별 개체명 토큰 포함 여부를 예측하게 하였고 '㉣ 개체명 아님' 으로 개체명에 속하지 않는 음절을 분리해 내었다. 개체명으로 판명된 음절 토큰은 개체명 사전을 참조하여 개체명 토큰 별로 개체명 분류를 지정한다. (그림 2)는 음절 입력 방식의 개체명 추출을 위한 Bidirectional Dynamic LSTM을 도식화 한 그림이다.



(그림 2). 음절 입력 방식의 Bi-Dynamic-LSTM 예

3.3 학습 말뭉치 및 개체명 사전 구축

3.3.1 학습 말뭉치 구축

음절 별 개체명 학습 말뭉치를 구축하기 위해 실제로 웹에서 사용되고 있는 문장을 수집하여 개체명 태깅을 진행하였다. 말뭉치 구축에는 30명의 사용자들이 약 17만 개의 문장에 대하여 문장에서 개체명에 해당하는 음절을 추출하고 개체명 분류 정보를 태깅 하였다. 개체명 분류는 정보통신단체표준인 '개체명 태그 세트 및 태깅 말뭉치(Tag Set and Tagged Corpus for Named Entity Recognition, TTAK.KO-10.0852)' 를 가공하여 18개의 대분류, 129개의 소분류로 개체명 분류 말뭉치를 구축 하였다[9].

3.3.2 개체명 사전 구축

말뭉치를 구축하면서 사용자들이 표기한 개체명은 개체명 사전으로 저장 된다. 구축된 개체명 사전은 개체명 추출 알고리즘의 결과에 참조되어 추출된 개체명의 분류를 지정해 준다. (그림 3)은 음절 별 개체명 설정과 분류 사전 구축 예이다.

(그림 3). 음절 별 개체명 말뭉치, 개체명 사전 구축 예

사용자들이 웹 수집 말뭉치를 기반으로 구축한 개체명 사전과 동시에 위키피디아의 표제어에 대해서도 개체명 사전을 함께 구축하였다. DBPedia ontology를 현재 사용중인 개체명 분류로 변환 하여 개체명 사전을 확장하였다[10]. 이를 통하여 45,000건의 사용자 태깅 개체명 사전과 위키피디아 표제어를 이용한 750,000건의 개체명 확장 사전을 구축했다.

4. 실험 및 성능 평가

Bidirectional Dynamic LSTM을 구현하기 위하여 Tensorflow 1.2.1을 사용하였으며 개체명 말뭉치 구축을 위한 클라이언트에서 Komoran3.0을 사용하였다[11]. 개체명 사전의 데이터베이스로 ElasticSearch 5.1.1를 사용하였다.

4.1 데이터 셋

음절 단위 개체명 태깅을 시행하기 위한 데이터로서 웹 수집 텍스트를 활용하여 개체명 태깅, 학습, 평가를 시행하였다. 음절 단위 개체명 분석기의 개발 취지에 맞게 형태소 분석이 용이하지 않은 통신어, 문법 파괴어의 학습과 평가를 하였다. 태깅된 말뭉치는 약 17만 문장이며 16만 문장을 가지고 학습하였고 평가는 4,000문장에 대하여 시행하였다.

4.2 성능 분석

성능 분석은 첫째로 각 입력 음절 분류인 '㉠ 개체명 시작', '㉡ 개체명 중간', '㉢ 개체명 끝', '㉣ 개체명 아님' 의 4종의 분류 정확도 측정과 둘째로 추출된 개체명의 분류가 말뭉치와 얼마나 일치하는지를 보는 크게 2가지 평가 방식으로 나뉜다.

표 1. 음절 단위 개체명 분류 예

⋮
한 - ㉠ 개체명 시작
국 - ㉡ 개체명 중간
어 - ㉢ 개체명 끝
를 - ㉣ 개체명 아님
⋮

개체명으로 시작과 끝으로 인식 된 문자열을 개체명 사전에서 검색하여 개체명 분류 태그를 부착한다.

4.2.1 평가 결과

평가 방식에 따라 세 가지 평가 척도를 상정 하여 평가하였다.

1) 개체명 음절 추출 정확도

$$= \frac{\text{분류가 맞은 음절수}(B)}{\text{전체 문장 음절수}(A)}$$

$$= \frac{\sum_{j=1}^{\text{문장별 음절수}} \text{일치음절수}(\text{말뭉치분류}_j, \text{예측치분류}_j)}{\sum_{i=1}^{4} \text{음절수}(\text{말뭉치}, \text{음절분류}_i)}$$

$$\left(\text{일치 음절수}(a,b) = \begin{cases} 1 & (a=b) \\ 0 & (a \neq b) \end{cases}\right)$$

2) 개체명 음절 인식 정확도

$$= \frac{\text{개체명 해당 음절수 중 분류가 맞은 음절수}(B)}{\text{개체명에 해당되는 음절수}(A)}$$

$$= \frac{\sum_{j=1}^{\text{문장별 음절수}} \text{일치음절수}(\text{말뭉치분류}_j, \text{예측치분류}_j)}{\sum_{i=1}^{3} \text{음절수}(\text{말뭉치}, \text{음절분류})}$$

(음절분류가 ㉣인 경우는 일치음절수에서 계산하지 않음.)

3) 개체명 분류 인식 정확도

$$= \frac{\text{개체명 분류가 맞은 토큰수}(B)}{\text{문장별 개체명수}(A)}$$

평가 방식 1), 2), 3) 에 따라 평가한 결과는 표 2와 같다.

표 2. 음절 단위 개체명 분류 결과

	정확도	(A)	(B)
1) 개체명 음절 추출	0.889	101523	90265
2) 개체명 음절 인식	0.847	25216	21367
3) 개체명 분류 인식	0.406	10929	4433

4.2.1 결과 분석

129종의 개체명 분류 대상에 대하여 '1) 개체명 음절 추출' 이 90%에 가까운 정확도를 보이고 '2) 개체명 음절 인식' 의 정확도가 80%를 상회 하는 것으로 나타났으며, 음절 단위의 개체명 추출을 Bidirectional Dynamic LSTM을 이용한다면 개체명 추출 및 인식에 높은 성능을 보이고 있음을 확인하였다. 그러나 개체명 분류 인식 성능은 40% 정도로 나타났는데, 이러한 성능의 원인은 다음과 같은 이유로 판단된다. 첫째, 추출된 개체명의 분류가 개체명 사전에 없는 경우이다. 이것은 지속적인 개체명 사전 구축을 통해 해결 될 수 있을 것으로 판단된다. 둘째로는, 개체명 말뭉치 구축을 30명의 사용자가 진행하였으므로 실제로 예측된 분류가 맞음에도 불구하고 말뭉치에 기재된 개체명 분류와 실험을 통해 예측한 분류가 달라서 오답으로 기재 된 경우에 기인한다. 4,000건의 테스트 데이터에 대하여 모두 개체명 분류의 정확성 판별을 직접 할 수 없었기 때문에 '3) 개체명 분류 인식' 의 정확도는 추후 지속적인 실험이 필요할 것으로 판단된다.

5. 결론

본 논문에서는 Bidirectional Dynamic LSTM 기반 한국어 개체명 분류를 위해 음절 단위 개체명 분류 방법 및 개체명 사전 구축, 개체명 말뭉치 구축 방법론을 제안하였다. 실험 결과, 제안 방법은 129가지의 개체명 추출에서 현재 10가지 이하의 개체명 분류 방식에서 도달한 수준과 같은 개체명 추출 정확도를 구현하였다[7]. 별도의 휴리스틱 알고리즘과 word2vec 없이 음절 단위 입력만으로도 개체명 추출에서 우수한 성능을 보임을 확인 할 수 있었다. 향후에는 개체명 사전을 참조하는 현재의 방식에서 현재 사용하는 RNN의 구조를 바꾸지 않고 state 벡터를 이용한 개체명 자동 분류 방식에 대한 연구가 고려되어야 할 것으로 판단된다.

감사의 글

본 연구는 미래창조과학부 및 정보통신기술진흥센터의 정보통신·방송 연구개발사업[R0114-16-0012, 개인화 서비스를 위한 통합 인지 컴퓨팅 플랫폼 개발]과 2017국어정보처리시스템경진대회(국립국어원) 일환으로 수행하였음.

참고문헌

[1] Sepp Hochreiter, Jurgen Schmidhuber, "LONG SHORT-TERM MEMORY", Neural Computation 9(8), pp.1735-1780, 1997.

[2] Schuster, Mike, and Kuldip K. Paliwal, "Bidirectional recurrent neural networks", IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL. 45, NO. 11, pp.2673-2681, 1997.

[3] Barak A. Pearlmutter, "Dynamic recurrent neural networks", Tech. Rep. CMU-CS-90-196, 1990.

[4] 이창기, 김준석, 김정희, 김현기, "딥 러닝을 이용한 개체명 인식", 한국정보과학회 제 41 회 정기총회 및 동계학술발표회, pp. 423-425, 2014.

[5] 김혜민, 윤정민, 안재현, 배경만, 고영중, "품사 분포와 Bidirectional LSTM CRFs를 이용한 음절 단위 형태소 분석기", 제28회 한글 및 한국어 정보처리 학술대회 논문집, pp.3-8, 2016.

[6] 이창기, "Long Short-Term Memory 기반의 Recurrent Neural Network를 이용한 개체명 인식", 한국정보과학회 2015 한국컴퓨터종합학술대회 논문집,pp.645-647, 2015.

[7] 유홍연, 고영중, "품사 임베딩과 음절 단위 개체명 분포 기반의 Bidirectional LSTM CRFs를 이용한 개체명 인식", 제28회 한글 및 한국어 정보처리 학술대회 논문집, pp.105-110, 2016.

[8] Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, "Efficient Estimation of Word Representations in Vector Space", arXiv:1301.3781, 2013.

[9] Acryl, https://goo.gl/o1pQrT, 2017.

[10] DBPedia, http://mappings.dbpedia.org/server/ontology/classes/, 2017.

[11] SHINEWARE, https://github.com/shin285/KOMORAN, 2017.