CNN을 이용한 대화와 같은 짧은 문장에서 개체명 인식

박성재○ 차정원
창원대학교
tjdwo1289@gmail.com, jcha@changwon.ac.kr

Recognizing named entities in short sentences such as conversations using CNN

Seong-Jae Park○, Jeong-Won Cha
ChangWon National University

요 약

본 논문에서는 CNN을 이용해 대화와 같은 짧은 문장에서 개체명을 인식하는 방법을 제안한다. 하나의 문장에 하나의 개체명이 있다고 가정하여 개체명 추출 문제를 문장을 개체명으로 분류하는 문제로 변환하여 CNN에 적용하였다. 또한 문맥정보를 고려하기 위하여 현재 문장으로부터 일정길이의 이전 문장을 함께 사용하는 세그먼트를 생성하여 CNN의 입력으로 사용하였다. 실험 결과 문장단위의 입력인 경우 정확도 90.54%, 재현율 82.44%, F1=86.30%를 보였고 세그먼트의 길이가 4이고 3문장을 오버랩 한 경우 정확도 90.46%, 재현율 86.73%, F1=88.56%로 가장 높은 성능을 보였다.

1. 서 론

의미 정보를 이용하는 응용이 많아짐에 따라 문장에서 개체명을 인식하는 필요성이 증가하고 있다. 개체명을 인식하는 연구는 오래 전부터 진행되어 왔다. 영어권에서는 이미 연구실을 벗어나 실상에서 활발하게 사용되고 있다.

한국어에서도 개체명을 인식하기 위한 연구가 많이 진행되었다. 한국어는 영어권에서 사용하는 형태 정보가 단어에 나타나지 않기 때문에 문맥에 따라 파악해야 하는 어려움이 있다. 특히 대화와 같은 짧은 문장의 경우에는 문맥으로 이용할 수 있는 정보가 상대적으로 적어 개체명을 인식하는데 어려움이 더 커진다.

본 연구에서는 대화와 같은 짧은 문장에서 CNN(Convolutional Neural Networks)를 이용하여 개체명을 추출하는 것을 목표로 한다. 또한 대화의 특성을 고려하여 문맥정보를 같이 고려하는 방법을 제시한다.

2. 관련 연구

개체명 인식에 대해서는 다음과 같은 연구가 있었다. 개체명 사전을 이용하는 규칙 기반 방법[1]과 CRF를 이용한 지도학습 방법[2] 그리고 Word Embedding 자질을 이용한 방법[3]등 다양한 방법들이 연구되었다.

문장 분류에 대해서는 문서 또는 문장의 감성을 분류하는 연구들이 있었다. 그 중 Convolutional Neural Networks for Sentence Classification[4]은 문장의 감성을 분류하는 문제를 CNN을 이용해 해결하는 방법을 제시하였다. 이 방법은 CNN의 필터를 하나만 사용했음에도 불구하고 기존의 성능보다 높거나 비슷한 성능을 보여 문장을 분류하는 문제에도 CNN을 활용하는 것이 효과적임을 입증하였다.

본 논문에서는 간단한 구조로도 기존 연구들과 성능이 높거나 유사한 CNN을 이용하여 대화와 같이 짧은 문장에서 개체명을 인식하는 방법을 제안한다.

3. 제안 방법

본 논문에서는 개체명 인식 문제를 분류 문제로 변환하여 해결하는 방법을 제안한다. 이를 위해 우리는 대화에서 하나의 문장이 하나의 개체명이 있다고 가정한다. 따라서 위의 가정에 따라 그림1 과 같이 하나의 입력에 대해 하나의 개체명을 분류하는 CNN을 제안한다.

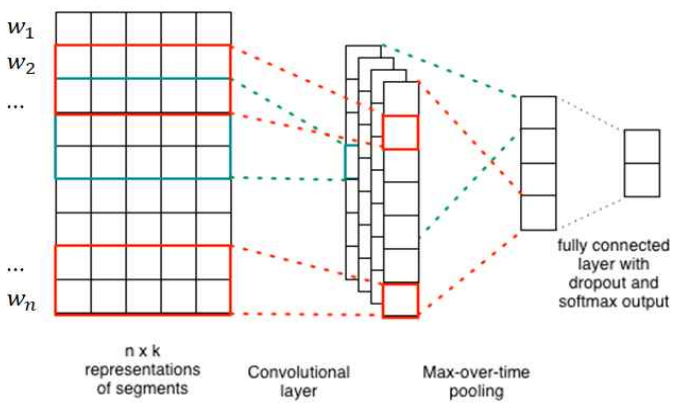

그림 1 제안 CNN 구성도

그러나 제안된 CNN은 대화 데이터의 문맥을 고려하지 못하는 단점을 가진다. 이러한 단점을 해결하기 위해 그림 2와 같이 몇 개의 문장을 묶어 세그먼트를 생성하여 문맥 정보를 사용하도록 CNN의 입력을 수정하였다.

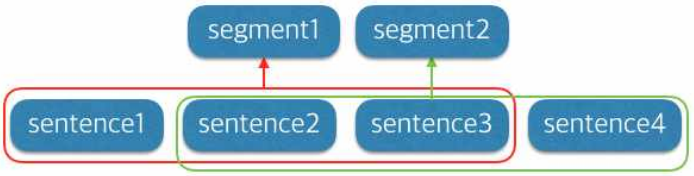

그림 2 세그먼트 생성 예제

세그먼트는 현재 문장에서 이전 방향으로 일정 길이의 문장을 결합해 생성하며 일정 길이가 오버랩 되도록 슬라이딩 윈도우 방식으로 생성한다. 생성된 세그먼트 또한 위의 가정과 동일하게 하나의 세그먼트는 하나의 개체명을 가진다고 가정한다.

4. 실험 및 토의

4.1 개체명 범주

본 논문에서 대화 데이터는 제품에 대한 문의 또는 불만에 대한 데이터이다. 우리는 문의 또는 불만의 대상이 되는 제품을 개체명으로 정의하였다. 본 논문에서 정의된 개체명은 총 138개의 범주를 가진다.

4.2 학습 코퍼스

본 논문에서는 대화 데이터를 형태소 분석하여 학습 및 평가에 사용하였다. 대화 데이터는 실제 개체명이 나타나는 문장보다 나타나지 않는 문장이 많기 때문에 None 태그의 비율이 90%가 넘어 그대로 학습을 진행하면 모델이 None태그에 편향되어 학습되는 문제가 발생할 수 있다. 따라서 None태그와 비(非)None태그의 비율을 7:3이 되도록 비율을 맞추는 작업을 진행하였다.

표 1 학습 데이터 크기 (단위: 세그먼트)

세그먼트 길이	오버랩 크기	비(非)None 태그	None 태그	총합
1	0	25,266	8,422	33,688
2	0	23,954	7,984	31,938
	1	47,907	15,969	63,876
3	0	22,757	7,585	30,342
	1	34,069	11,356	45,425
	2	68,197	22,732	90,929
4	0	21,705	7,235	28,940
	1	29,025	9,675	38,700
	2	43,318	14,439	57,757
	3	86,521	28,840	115,361
5	0	20,718	6,906	27,624
	1	25,915	8,638	34,553
	2	34,451	11,483	45,934
	3	51,650	17,216	68,866
	4	102,994	34,331	137,325

4.3 골드 코퍼스 생성

학습을 진행하기 위해서는 정답 레이블이 부착된 골드 코퍼스가 필요하다. 그러나 본 논문에서 사용된 대화 데이터는 별도로 생성된 골드 코퍼스가 존재하지 않기 때문에 골드 코퍼스를 제작하는 작업을 추가로 진행하였다. 골드 코퍼스 제작은 사람이 직접 제작하는 것이 가장 좋은 방법이지만 많은 시간과 자원이 소모되기 때문에 반자동으로 골드 코퍼스를 생성하기 위한 방법이 필요하다. 본 논문에서는 대화 데이터를 정답 클래스와 완전매치(Exact Match)를 해서 실버 코퍼스를 생성하였다. 이렇게 생성된 실버 코퍼스는 완전매치의 특성상 문맥을 통해 개체명을 유추할 수 있는 정답을 반영하지 못하는 특성을 지닌다. 이러한 실버 코퍼스의 특징 때문에 모델의 결과가 비(非)None태그이고 실버 코퍼스의 정답이 None태그인 경우 모델의 결과가 사실상 정답이 될 수 있다 따라서 성능 평가 시 Recall이 낮게 나타나는 현상이 발생할 수 있다.

4.4 실험

본 논문에서는 세그먼트를 생성하는 것이 개체명을 분류하는데 미치는 영향을 확인하기 위해 문장 단위 실험과 세그먼트 단위 실험을 진행한다. 세그먼트를 구성하는 문장의 길이가 길면 여러 개의 개체명이 존재할 수 있고 문장의 길이가 짧으면 개체명이 존재하지 않을 수 있다. 따라서 세그먼트를 구성하는 최적의 문장길이와 오버랩 크기를 확인하는 실험을 함께 진행한다.

4.5 실험 결과

표 2와 3은 세그먼트와 오버랩 크기에 따른 개체명 인식결과이다. 세그먼트를 생성하지 않고 문장으로 실험한 경우 F1이 86.30%성능을 보였다. 최고 성능으로는 세그먼트 길이가 4, 오버랩 크기가 3일 때로 88.56%로 세그먼트를 생성하지 않은 경우보다 2.26% 성능 향상을 보였다.

표 2 개체명 인식 결과

세그먼트 길이	오버랩 크기	예측 / 정답	비(非)None 태그(정답/오답)	None 태그	총합
1	0	Tag	5,224/527	586	6,337
		None	0/19	2,093	2,112
		Total	5,770	2,679	8,449

2	0	Tag	4,621/617	780	6,018
		None	0/12	1,994	2,006
		Total	5,250	2,774	8,024
	1	Tag	10,268/912	840	12,020
		None	0/23	3,983	4,006
		Total	11,203	4,823	16,026
3	0	Tag	4,224/642	844	5,710
		None	0/10	1,893	1,903
		Total	4,876	2,737	7,613
	1	Tag	6,978/853	730	8,561
		None	0/38	2,815	2,853
		Total	7,869	3,545	11,414
	2	Tag	14,654/1,462	972	17,088
		None	0/47	5,649	5,696
		Total	16,163	6,621	22,784
4	0	Tag	3,837/709	905	5,451
		None	0/9	1,808	1,817
		Total	4,555	2,713	7,268
	1	Tag	5,431/836	1,016	7,283
		None	0/16	2,411	2,427
		Total	6,283	3,427	9,710
	2	Tag	8,758/1,201	917	10,876
		None	0,68	3,557	3,625
		Total	10,027	4,474	14,501
	3	Tag	18,787/1,926	949	21,662
		None	0/55	7,165	7,220
		Total	20,768	8,114	28,882
5	0	Tag	3,493/764	949	5,206
		None	0/18	1,717	1,735
		Total	4,275	2,666	6,941
	1	Tag	4,728/847	932	6,507
		None	0/15	2,154	2,169
		Total	5,590	3,086	8,676
	2	Tag	6,558/993	1,097	8,648
		None	0/16	2,866	2,882
		Total	7,567	3,963	11,530
	3	Tag	10,383/1,557	1,016	12,956
		None	0/48	4,270	4,318
		Total	11,988	5,286	17,274
	4	Tag	21,934/2,540	1,300	25,774
		None	0/83	8,508	8,591
		Total	24,557	9,808	34,365

표 3 개체명 인식 결과

세그먼트 길이	오버랩 크기	Precision	Recall	F1
1	0	90.54%	82.44%	86.30%
2	0	88.02%	76.79%	82.02%
	1	91.65%	85.42%	88.43%
3	0	86.63%	73.98%	79.80%
	1	88.68%	81.51%	84.94%
	2	90.66%	85.76%	88.14%
4	0	84.24%	70.39%	76.69%
	1	86.44%	74.57%	80.07%
	2	87.34%	80.53%	83.80%
	3	90.46%	86.73%	88.56%
5	0	81.71%	67.10%	73.68%
	1	84.58%	72.66%	78.17%
	2	86.67%	75.83%	80.89%
	3	86.61%	80.14%	83.25%
	4	89.32%	85.10%	87.16%

5. 결론 및 향후 연구

본 논문에서는 대화와 같이 짧은 문장에서 멀티 클래스를 분류하는 문제에서 CNN을 사용하여 개체명을 추출하였다. 또한 CNN의 특성상 시퀀스 데이터 처리에 적합하지 않은 문제를 해결하기 위해 세그먼트를 도입하였다. 세그먼트가 시퀀스 데이터 문제를 해결하는데 영향을 미친다는 것을 증명하기 위해 문장 단위의 실험과 세그먼트를 생성한 실험을 비교하였고 세그먼트 생성 시 문장 단위 분류보다 2.26% 성능 향상을 보였다.

향후 연구로 이전 세그먼트에서 추출된 개체명과 이벤트가 다음 세그먼트에 영향을 미치는 것을 확인하고 이전 세그먼트에서 나온 정보를 LSTM(Long short-term memory)을 사용하여 현재 세그먼트의 개체명과 이벤트를 추출하는 연구를 진행할 계획이다.

참 고 문 헌

[1] Joo-Young Lee, Young-In Song and Hae-Chang Rim, 2004, “Title Named Entity Recognition based on Automatically Constructed Context Patterns and Entity Dictionary,“ 한국정보과학회 언어공학연구회 학술발표 논문집, Vol. 16, No. 1, pp. 40~45.

[2] Changki Lee, Yi-Gyu Hwang, Hyo-Jung Oh, Soojong Lim, Jeong Heo, Chung-Hee Lee, Hyeon-Jin Kim, Ji-Hyun Wang and Myung-Gil Jang, 2006, “Fine-Grained Named Entity Recognition using Conditional Random Fields for Question Answering,“ 한국정보과학회 언어공학연구회 학술발표 논문집, , pp. 268~272.

[3] Yunsu Choi and Jeongwon Cha, 2016, “Korean Named Entity Recognition and Classification using Word Embedding Features,“ Journal of KIISE, Vol. 43, No. 6, pp. 678~685.

[4] KIM, Yoon. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.