한국어에서 Attention 모델과 Naïve Bayes 모델 기반의 어휘 말뭉치 구축 및 응용에 관한 연구

윤주성○, 김현철

고려대학교
{xelloss705, hkim64}@gmail.com

Attention and Naïve Bayes Models based Lexicon Corpus and Applications for Korean

Joosung Yoon○, Hyeoncheol Kim
Korea University

요 약

감성 분석에서 어휘 말뭉치는 기존의 전통적인 기계학습 방법에서 중요한 특징으로 사용되었다. 최근 딥러닝의 발달로 hand-craft feature를 사용하지 않아도 되는 End-to-End 방식의 학습이 등장했다. 하지만 모델의 성능을 높이기 위해서는 여전히 어휘말뭉치와 같은 특징이 모델의 성능을 개선하는데 중요한 역할을 하고 있다. 본 논문에서는 이러한 어휘 말뭉치를 Attention 모델과 Naïve bayes 모델을 기반으로 구축하는 방법에 대해 소개하며 구축된 어휘 말뭉치가 성능에 끼치는 영향에 대해서 Hierarchical Attention Network 모델을 통해 분석하였다

주제어: 한국어 어휘 말뭉치, Attention, Naïve bayes

1. 서론

감성 분석(sentiment analysis)는 자연어처리 분야에서 가장 기본적인 태스크 중에 하나로써, 해당 문서의 감성이 긍정, 부정, 중립 중 어디에 속했는지 구분하는 것을 의미한다. 감성 분석을 위한 전통적인 기계학습 방법으로는 SVM[1], Naïve bayes[3] 등이 사용되었고 어휘(lexicon)기반의 특징(feature)을 활용한 연구도 많이 진행되었다[2].

최근 딥러닝(deep learning)의 발달[4]로, 이러한 어휘 기반의 특징인 hand-craft feature를 이용하는 것이 아닌 End-to-End 방식으로 문제를 해결하는 것이 가능해졌다. 하지만 End-to-End 방식으로 모든 문제가 해결되는 것이 아니며, 학습 데이터가 부족하거나 노이즈가 많을 경우 딥러닝을 써도 성능이 높게 나오지 않을 수 있다. 이러한 경우 전통적인 기계학습 방법에 사용되었던 hand-craft feature를 추가적으로 사용하여 모델을 개선할 수 있다.최근 연구 결과에 따르면 어휘 특징(lexicon feature)은 문서 분류를 위한 딥러닝 모델에서도 모델의 성능을 높이기 위해 여전히 유용한 특징으로 알려져 있다[2]. 그러므로 이러한 어휘 말뭉치(lexicon corpus)를 구축하는 일은 모델의 성능을 향상시키기 위해 필요하다고 할 수 있다.

본 논문에서는 Attention 모델과 Naïve bayes 모델을 사용으로 어휘 말뭉치를 구축하고 그것의 효과에 대해서 연구하였다.

본 논문의 구성은 2장에서 관련 연구 및 모델에 대해서 설명하고, 3장에서 실험에 대해서 서술한다. 4장에서는 결론 및 향후 연구에 대해서 다룬다.

2. 관련 연구

2.1 Naïve Bayes (NB)

Naïve bayes 모델은 아래와 같이 문서 d가 어떤 클래스 $c \in C$에 속하는지 확률을 사용해서 분류하는 모델이다[3].

$$c^{predict} = argmax_c P_{NB}(c|d)$$

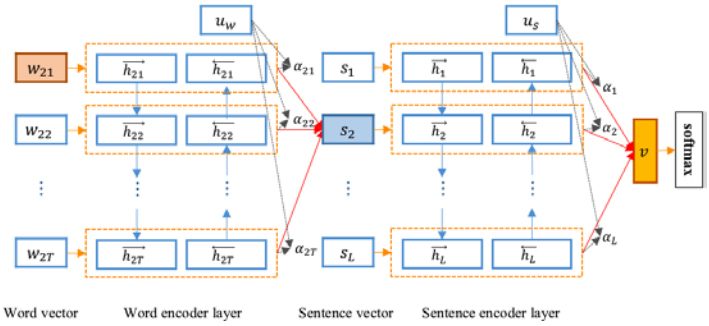
$$P_{NB}(c|d) := \frac{P(c)\sum_{i=1}^{m} P(f|c)^{n_i(d)}}{P(d)}$$

위 식에서 f는 특징(feature)을 의미하며, $n_i(d)$는 f_i 특징의 횟수를 의미한다. Naïve bayes 모델에서 f이 모델에 끼치는 영향을 분석해서 어휘 말뭉치를 구축하는데 사용했다.

2.2 Hierarchical Attention Networks (HAN)

HAN 모델[5]은 GRU[6]와 Attention이 계층적으로 이루어진 모델이며 문장 단위의 GRU와 문장을 이루는 단어 단위의 GRU로 이루어져있다. 각 GRU layer 위에 Attention layer가 있으며 단어에 대한 Attention과 문장에 대한 Attention을 계산한다. 단어에 대한 Attention은 모델이 학습될 때 같이 학습되는 단어의 컨텍스트 벡터 u_w가 어떻게 학습되는지에 따라 달라지며 본 논문에서는 단어에 대한 Attention을 어휘 말뭉치를

구축하는데 사용했다.



<그림 1: Hierarchical Attention Network 모델 구조>

2.2 형태소 분석

형태소 분석은 KoNLPy를 사용했다[8]. HAN에서 문서를 문장 단위로 나눠줄 때는 꼬꼬마 형태소 분석기[9]를 사용했으며 문장 내에서 형태소 분석을 할 때는 한국어 트위터 형태소 분석기[10]를 사용했다.

3. 실험

3.1 평가 방법

평가를 위한 기준은 Accuracy를 사용했으며, HAN 모델과 HAN 모델에 어휘 말뭉치 특징을 추가했을 때의 성능 변화를 비교했다.

3.2 말뭉치

말뭉치는 네이버로부터 얻은 영화 평점 데이터[1]를 사용했으며 중립 리뷰는 포함하지 않았다. 긍정적인 리뷰의 경우 9-10점 리뷰들로 구성되어있으며 부정적인 리뷰는 1-4점 리뷰로 구성되어 있으며 데이터의 크기는 <표 1>과 같다. 각 리뷰는 영화당 100개의 140자평을 초과하지 않는다. 모델에 사용한 단어 종류는 16,931개이며 등장 빈도가 4이하인 단어는 제외했다.

<표 1: 영화 리뷰 데이터>

	긍정	부정	문장당 평균 단어 수
Train	750,000	750,000	11.34
Test	25,000	25,000	11.37

3.3 어휘 말뭉치 구축

말뭉치로부터 어휘 말뭉치를 구축하기 위해 Naïve baye 모델과 HAN 모델을 학습 후 사용했다. Naïve bayes 어휘 말뭉치의 경우, Naïve bayes 모델에 가장 영향을 많이 끼친 단어 상위 2,200개를 다음과 같은 기준으로 선택했다.

$$w = argmax_w\left(\frac{max\ P_{NB}(w|c_i)}{min\ P_{NB}(w|c_i)}\right)$$

Naïve bayes 어휘 말뭉치에서 각 단어에 대한 감성 점수는 $max\ P_{NB}(w|c_i)\ /\ min\ P_{NB}(w|c_i)$의 값들에 대해서 0에서 1사이로 정규화 해서 나타냈다.

Attention 어휘 말뭉치의 경우, Attention모델이 문서를 분류 할 때 단어에 대한 Attention weight가 높은 단어에 대해서 상위 2,200개를 선택하고 weight기반으로 각 단어에 점수를 부여했다. 이때 형태소 분석기로부터 Josa, Verb, Punctuation이 태깅 된 단어들은 제외하였다. 어휘 말뭉치는 단어와 단어에 대응되는 감성 점수로 구성하였으며 각 점수는 0에서 1사이로 정규화 하였다.

<표 2: Attention과 NB의 어휘 말뭉치 상위 단어 비교>

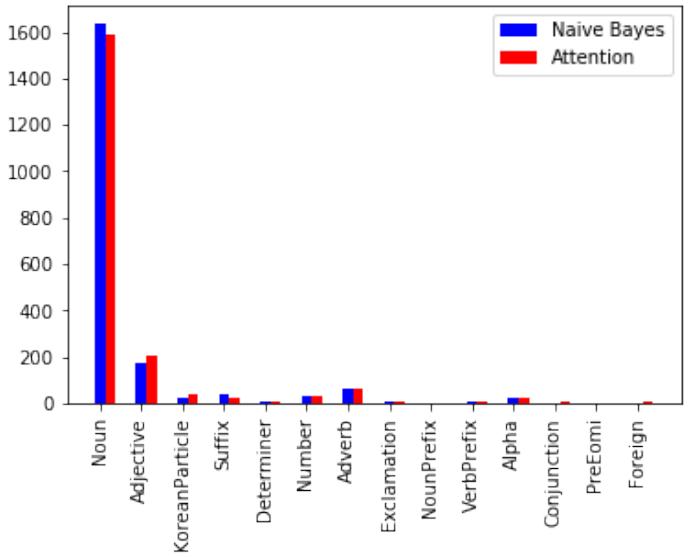
	Attention	Naïve Bayes
긍정	영화/Noun	울컥/Adverb
	재밌다/Adjective	♥/Foreign
	ㅋㅋ/KoreanParticle	Good/Alpha
	좋다/Adjective	펑펑/Noun
	너무/Noun	♥♥♥/Foreign
	정말/Noun	꿀잼/Noun
	최고/Noun	♡/Foreign
	재미있다/Adjective	아련하다/Adjective
	있다/Adjective	최고다/Noun
	ㅠㅠ/KoreanParticle	쵝오/Noun
부정	영화/Noun	최악/Noun
	없다/Adjective	낚였/Noun
	ㅋㅋ/KoreanParticle	낭비/Noun
	재미없다/Adjective	반개/Noun
	아깝다/Adjective	빵점/Noun
	점/Noun	노잼/Noun
	너무/Noun	하품/Noun
	이/Noun	기세/Noun
	쓰레기/Noun	이도/Noun
	진짜/Noun	개뿔/Noun

구축된 Attention 기반 어휘 말뭉치에서 긍정과 부정의 첫번째 단어가 영화/Noun 가 나온 것을 확인할 수 있었다. 이는 빈도수 기반의 Naïve bayes와는 달리 학습된 컨텍스트 벡터(context vector)를 통해 단어에

[1] http://github.com/e9t/nsmc/

Attention을 주는 모델의 특성이 반영된 것으로 보인다.

<그림 2>에 의하면, 전체적으로는 Noun이 태깅 된 단어는 Attention 어휘 말뭉치보단 Naïve bayes 어휘 말뭉치에서 더 많이 분포되어 있는 것으로 나타났다. 그리고 Naïve bayes 어휘 말뭉치에서는 Noun이 태깅 된 단어가 Attention 어휘 말뭉치에 비해 더 상위 순위에 있는 것을 확할 수 있었으며 Attention의 경우 Adjective, KoreanParticle등 다양한 품사를 가진 단어들이 상위에 분포하는 것을 확인 할 수 있었다.



<그림 2: 어휘 말뭉치 태깅 종류 분포>

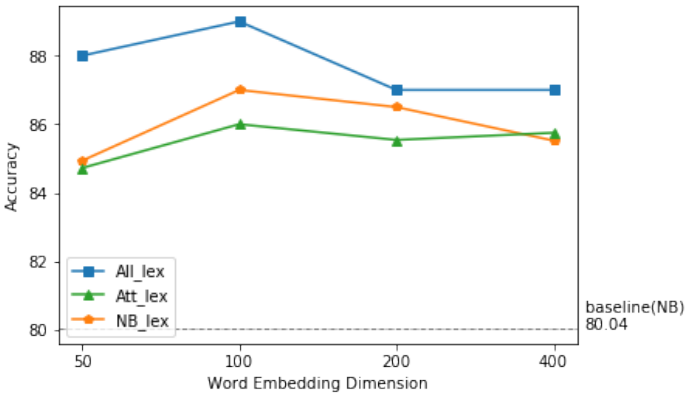
3.4 학습 파라미터

HAN 모델에 대한 학습 파라미터는 기본적으로 Zichao가 제안한 구성[5]과 같다. 모델 내의 Word embedding은 200 차원, Bidirectional GRU는 100차원, 단어와 문장의 컨텍스트 벡터는 100차원이다. 학습을 위한 파라미터는 다음과 같다. 모델 훈련을 위한 파라미터는 Stochastic gradient descent의 learning rate는 0.1, 모멘텀은 실험적으로 0.6의 파라미터로 설정했다. 모든 모델의 훈련을 위한 epoch은 20, batch size는 65로 설정했다.

Word embedding의 차원의 크기에 따른 성능 비교를 위해 차원의 크기를 50, 100, 200, 400으로 나눠서 실험하였다.

3.4 실험 모델

Attention과 Naïve bayes 모델을 통해 구축한 어휘 말뭉치가 감성 분석에 끼치는 영향을 알아보기 위해 HAN 모델에 각각의 어휘말뭉치를 단어에 대한 점수 벡터로 변환하여 Word embedding 벡터[11]에 붙여서 모델을 구성했다. <그림 3>에 의하면 Attention 기반 어휘 말뭉치와 Naïve bayes 기반 말뭉치를 모두 사용했을 때 가장 높은 성능을 나타냈으며 Word embedding이 100차원일 때 가장 성능이 높은 것으로 나타났다. 대체적으로 Attention 기반 어휘 말뭉치보다는 Naïve bayes 기반 어휘 말뭉치가 감성분석에서 성능을 높이는데 더 큰 영향을 주는 것으로 나타났다.



<그림 3: 어휘 말뭉치에 따른 모델 성능>

4. 결론 및 향후 연구

본 논문에서는 Attention과 Naïve bayes모델을 기반으로 어휘 말뭉치를 구축하고 그 효과에 대해서 감성 분석을 통해 평가하였다. 실험결과 Attention 어휘말뭉치는 Noun, Adjective, KoreanParticle등 다양한 형태소로 구성되었고 Naïve bayes 어휘 말뭉치는 주로 Noun 형태소로 구성된 것을 확인 할 수 있었다. 각 어휘말뭉치를 HAN 모델에 적용하여 감성 분석 실험결과 대체적으로 Attention과 Naïve bayes 어휘 말뭉치를 모두 사용한 모델이 가장 성능이 높았으며 각각을 따로 적용한 경우 Naïve bayes 어휘 말뭉치가 더 성능이 높음을 확인 할 수 있었다.

본 연구에서는 긍정과 부정을 분류하는 감성 분석을 통해 어휘말뭉치의 효과를 검증했지만 각 클래스당 어휘말뭉치를 생성해서 텍스트 분류(text classification)[12]에서는 이러한 연구가 진행된 바가 없다. 대부분의 분류 문제는 클래스가 2개 이상이므로 향후 연구에서는 텍스트 분류에 대해서도 적합한 어휘말뭉치를 구축하고 적용하는 연구가 필요할 것으로 보인다.

감사의 글

" 이 논문은 2017년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (NRF-2017R1A2B4003558)."

참고문헌

[1] Mullen, Tony, and Nigel Collier. "Sentiment Analysis using Support Vector Machines with Diverse Information Sources." EMNLP. Vol. 4. 2004.

[2] Shin, Bonggun, Timothy Lee, and Jinho D. Choi. "Lexicon integrated cnn models with attention for sentiment analysis." arXiv preprint arXiv:1610.06272, 2016.

[3] Go, Alec, Richa Bhayani, and Lei Huang. "Twitter

sentiment classification using distant supervision." CS224N Project Report, Stanford 1, 12, 2009.
[4] LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. "Deep learning." Nature 521.7553: 436-444, 2015.
[5] Yang, Z., Yang, D., Dyer, C., He, X., Smola, A. J., & Hovy, E. H. Hierarchical Attention Networks for Document Classification. In HLT-NAACL, pp. 1480-1489, 2016.
[6] Bahdanau, D., Cho, K., &Bengio, Y. Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473, 2014.
[7] Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555, 2014.
[8] 박은정, 조성준, “KoNLPy: 쉽고 간결한 한국어 정보처리 파이썬 패키지”, 제 26회 한글 및 한국어 정보처리 학술대회 논문집, 2014.
[9] 이동주, 연종흠, 황인범, and 이상구. “꼬꼬마: 관계형 데이터베이스를 활용한 세종 말뭉치 활용 도구.” 정보과학회논문지: 컴퓨팅의 실제 및 레터 16, no. 11, pp. 1046-1050, 2010.
[10] 트위터에서 만든 오픈소스 한국어 처리기, Github, twitter/twitter-korean-text, https://github.com/twitter/twitter-korean-text, 2016.
[11] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems pp. 3111-3119, 2013.
[12] Kim, Yoon. "Convolutional neural networks for sentence classification." arXiv preprint arXiv:1408.5882, 2014.