Parallel Stacked Bidirectional LSTM 모델을 이용한 한국어 영화리뷰 감성 분석

Korean Movie-review Sentiment analysis Using Parallel Stacked Bidirectional LSTM Model

저자
(Authors)

오영택, 김민태, 김우주
Yeongtaek Oh, Mintae Kim , Wooju Kim

출처
(Source)

한국정보과학회 학술발표논문집 , 2018.6, 823-825 (3 pages)

발행처
(Publisher)

한국정보과학회
KOREA INFORMATION SCIENCE SOCIETY

URL

http://www.dbpia.co.kr/Article/NODE07503164

APA Style

오영택, 김민태, 김우주 (2018). Parallel Stacked Bidirectional LSTM 모델을 이용한 한국어 영화리뷰 감성 분석. 한국정보과학회 학술발표논문집, 823-825.

Parallel Stacked Bidirectional LSTM 모델을 이용한 한국어 영화리뷰 감성 분석

오영택 ○ 김민태 김우주
연세대학교 산업공학과
yeongtaek@yonsei.ac.kr, iammt@yonsei.ac.kr, wkim@yonsei.ac.kr

Korean Movie-review Sentiment analysis Using Parallel Stacked Bidirectional LSTM Model

Yeongtaek Oh○ Mintae Kim Wooju Kim
Department of Industrial Engineering, College of Engineering, Yonsei University

요 약

감성 분석(Sentiment Analysis)은 텍스트 문서의 감성을 분류하는 분야이다. 감성 분석에 딥러닝 모델을 적용한 연구가 매우 활발히 진행되고 있다. 딥러닝을 이용한 감성 분석 방법론은 문서를 벡터화하는 과정과 벡터화된 문서를 분류하는 과정으로 나눌 수 있다. 문서를 벡터화하는 과정은 문서의 문장을 단어 단위로 토큰화하여 word2vec을 이용해 벡터화하는 방법과, 문장을 문자 단위로 토큰화하여 임베딩레이어 학습을 통해 벡터화하는 방법이 있다. 다음으로 벡터화된 문서를 분류하는 딥러닝 기반 모델은 크게 CNN (Convolutional Neural Netoworks)을 이용한 분류 모델과 RNN(Recurrent Neural Networks)을 이용한 분류 모델로 나누어진다. 이 연구에서는 한국어 문서를 형태소 분석 및 단어 단위의 토큰화 전처리 방법과 한글의 음소 단위의 토큰화 전처리 방법을 비교하고 기존 LSTM을 이용한 RNN 구조 모델을 개선하여 Parallel Stacked Bidirectional LSTM Model을 제안하였다. 제안된 모델의 성능을 기존의 딥러닝 모델 기반의 성능을 비교하기 위하여 네이버 영화 리뷰 데이터셋인 nsmc(naver sentiment movie corpus)에 대해 비교 실험을 하였다. 전처리 방법과 모델에 따른 성능을 측정하였고 전처리 방법 별 제안된 모델이 최고 성능을 보였으며, 최종적으로 단어 단위와 음소 단위 모델의 앙상블을 통해 87.75%의 분류 정확도를 달성하였다.

1. 서 론

감성 분석(Sentiment Analysis)은 텍스트 문서에서 감성을 긍정 혹은 부정으로 분류하는 방법으로 SNS상의 글이나 상품평, 리뷰글 등의 문서들로부터 대중의 의견을 분석하는 오피니언 마이닝(Opininon Mining) 분야에서 응용되어 왔다. 이산적인 단어 벡터를 연속적인 벡터로 표현하는 W2Vec 연구[1]에 의해 문서의 단어를 효율적으로 벡터화 할 수 있게 되면서 딥러닝을 이용한 자연어 처리 연구가 가속화 되어, 문서 분류 및 감성 분석 분야에 적용한 연구들이 제안되었다. 첫번째 연구[2]에서는 word2vec을 이용하여 문서를 벡터화 하고, 벡터화된 문서를 CNN(Convolutional Neural Network)을 이용한 문서 분류 방법론을 제안하였다. 또 다른 연구[3]에서는 RNN(Recurrent Neural Networks)을 이용하여 문서 분류 하는 방법론을 제안하였다. 이 연구들은 문서를 단어 단위로 전처리를 하였지만, 이외에도 문서를 문자(character)단위로 토큰화하여 문서를 분류하는 연구[4]도 진행되었다.

대부분의 연구들이 영어 문서에 대해 진행되었는데, 국내에서는 이런 방법론들을 한국어 문서에 적용하는 연구도 진행되어 왔다. Dowoo Kim[5]과 Jung-Mi Kim[6]의 연구에서는 한국어 문서를 형태소 분석기를 통해 문장을 토큰화하고 각 단어를 word2vec을 이용하여 벡터화 하였다. 각 연구는 CNN과 LSTM (Long Short-Term Memory)[7]을 이용하여 벡터화된 문서를 분류 하였다. 또한 이재준[8]의 연구에서는 문서를 한국어 음소 단위로 토큰화하여 LSTM을 통해 문서 분류를 하였다. 하지만 한국어 문서 분류에 대한 연구들이 다른 데이터에 대해 진행되어 어떤 방법론이 한국어 문서 분류에 적합한지 비교하기가 어렵다.

따라서 이 연구는 기존 모델과 전처리 방법들을 한국어 공개 데이터인 nsmc(Naver sentiment movie corpus)에 감성 분석 방법을 통해 성능 비교 실험하고 한국어에 적합한 전처리 방법과 딥러닝 모형을 찾는다. 더나아가 RNN 모델 구조를 개선한 Bidirectional LSTM[9]의 stack구조를 병렬적으로 쌓은 딥러닝 모델인 Parallel Stacked Bidirectional LSTM Model을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 사용된 모델 구조를 설명한다. 3장에서는 기존 연구 모델과 비교 실험하고 분석한다. 마지막으로 4장에서 본 연구의 결론과 향후 계획을 제시한다.

2. 모델

제안하는 모델의 구조는 그림 1과 같다. 단어 단위 혹은 음소 단위로 전처리된 시퀀스가 Input 레이어에 입력되게 되고, Embedding 레이어를 통해 각 시퀀스 데이터가 벡터화 된다. 벡터화된 데이터는 3개의 Stacked Bidirectional LSTM 레이어가 병렬적으로 구성된 층을 지나게 되고 각 Stacked Bidirectional LSTM모델의 아웃풋을 concatenate 하여 최종 Dense-Softmax 레이어를 통해 결과를 예측한다. 추가적으로 overfitting 효과를 늦추기 위해 Embedding 레이어 직후와 마지막 Dense 레이어 직전에 Dropout 기법을 적용하였다.

2.1 Embedding 레이어

문서를 단어 단위 혹은 음소단위의 전처리를 통해 고정된 길이(T)의 시퀀스 데이터 $\mathrm{x} = [x_1, x_2, x_3, \dots, x_T]$로 변환한다. 시퀀스 데이터를 전처리 방법에 대응하는 임베딩 방법을 통해 i번째 토큰 x_i에 대응하는 벡터$\mathrm{X_i}$로 변환하여 $\mathrm{X} = [X_1, X_2, X_3, \dots, X_T]$ 를 얻게 된다.

단어 단위 임베딩은 각 문서를 konlpy의 Twitter 형태소 분석기를 이용하여 문서를 형태소 분석 및 토큰화 하였다. 토큰화된 단어 시퀀스에 대한 word2vec 벡터를 CBOW 방식을 이용하여 개별 학습을 통해 생성[3]하였다. 학습된 word2vec matrix를 이용해 Embedding Layer에서 fine-tuning을 통해 각 단어를 벡터화 하였다. 이때 전체 단어 중 word2vec matrix에 존재하는 상위 18000개 단어만 모델에 사용하였다.

음소 단위 임베딩은 각 문서를 한국어 음소 단위로 분해 후 한국어 음소 및 특수문자를 포함 250개의 문자로 인덱싱한 후 임베딩 레이어를 통해 벡터화 하였다.

2.2 Parallel Stacked Bidirectional LSTM 레이어

CNN을 모델[2]에서 필터 크기가 다른 레이어를 병렬 구성하여 모델이 깊어지지 않게 함과 동시에 내부 레이어의 앙상블을 이용하여 성능을 향상시켰다. 이와 유사하게 RNN구조 역시 메모리 역할을 하는 Cell size를 다르게 설정한 동일 구조의 레이어를 내부적으로 앙상블을 통해 성능향상을 기대 할수 있다. 따라서 벡터화된 시퀀스 데이터를 각각 Cell size가 32, 64, 128인 2-stack bidirectional LSTM 레이어에 병렬적으로 연결한다. 각 stacked layer에서의 상위 레이어의 마지막 output을 concatenate하였다. 각 Bidirectional LSTM 레이어는 다음과 같이 계산된다.

$$\overrightarrow{h_t} = LSTM(\overrightarrow{h_{t-1}}, X_t) \quad (1)$$

$$\overleftarrow{h_t} = LSTM(\overleftarrow{h_{t-1}}, X_t) \quad (2)$$

i번째 병렬 레이어의 출력은 $l_i = [\overrightarrow{h_T}; \overleftarrow{h_T}]$로 표현되며,

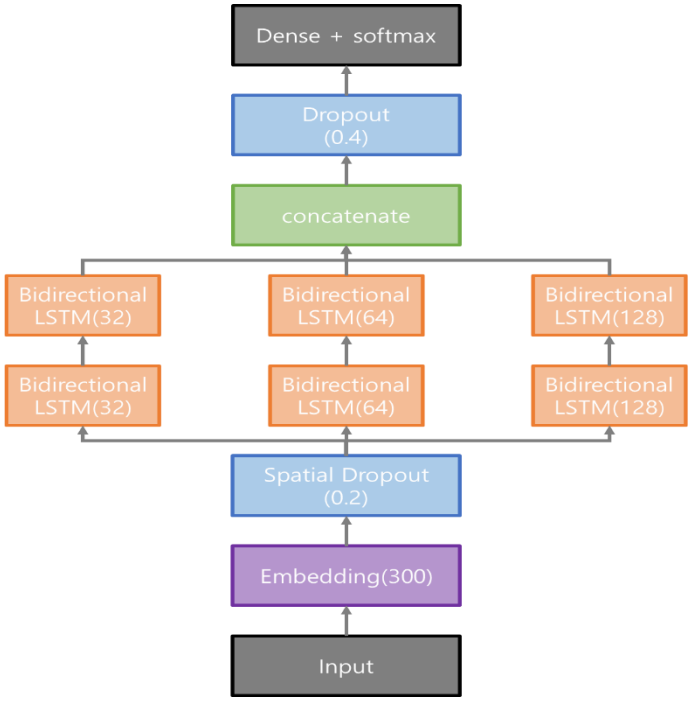

그림 1. Parallel Stacked Bidirectional LSTM 모델

concatenate된 최종 벡터c는 다음과 같다.

$$\mathrm{c} = [l_1; l_2; l_3] \quad (3)$$

2.3 Dense-softmax 레이어

마지막 Dense층은 c벡터를 입력으로 받고 fully connected층과 비선형 활성함수인 softmax를 통해 최종 클래스 별 확률 값 형태로 출력한다.

$$\hat{y} = softmax(w \cdot c + b) \quad (4)$$

비용함수는 categorical cross entropy를 사용하였다.

$$\mathrm{L}(\mathrm{y}, \hat{y}) = -\frac{1}{m}\sum_{i=1}^{m} y \cdot log(\hat{y}) \quad (5)$$

3. 실험

모델 구현은 python 3.6, Tensorflow_1.8.0와 keras_2.1.6을 이용하였다. 실험환경은 i7-7700(CPU), GeForce GTX 1080Ti(GPU)이며, 배치크기 1024, epoch은 early stopping을 통해 정하였다.

3.1 실험 데이터

모델의 성능 신뢰성과 이후 연구들의 재생산성을 위해 한국어 감성분석 공개데이터 nsmc를 사용하였다. nsmc데이터는 전체 200000개의 네이버 영화 리뷰 데이터로, Trainset 150000개, Testset 50000개 로 구성되어 그대로 사용하였다. 문서의 label은 긍정(1), 부정(0)으로 라벨링 되어있다.

3.2 실험 결과

모델의 성능을 평가하기위해 분류 성능을 모델의 Loss함수인 categorical cross entropy(Log loss)와 실제 분류 정확도 Accuracy를 기준으로 비교하였다. (표 1)

한국어 문서에 대해 단어 단위의 전처리 방법이 음소 단위의 전처리 방법보다 모든 모델에서 높은 성능을 보였다. 또한 CNN모델보다 LSTM모델이 높은 성능을 보였다. 이 연구에서 제안한 모델(Parallel Stacked Bidirectional Model)이 각 전처리 방법에서 각각 87.07%, 84.38%의 정확도로 기존 모델 대비 정확도가 1%씩 성능 향상을 보였다.

마지막으로 전처리 방법이 달리하여 학습한 모델의 결과를 단순 평균을 이용한 앙상블을 통해 싱글모델로는 가장 높은 성능을 낸 단어 단위의 Proposed Model에 보다도 약 0.7% 정확도 향상이 되어 87.75%를 달성하였다. 이를 통해 전처리를 다르게 한 모델들 간의 앙상블 결과는 단일 모델의 최고 성능보다 높은 성능을 보임을 확인하였다.

표 1 nsmc 데이터셋에 대한 성능 비교실험 결과

Model	Embedding	Log loss	Accuracy
CNN[2]	단어	0.3421	85.54
	음소	0.4251	80.08
LSTM[6]	단어	0.3251	86.04
	음소	0.3765	83.51
Proposed model	단어	0.3020	**87.07**
	음소	0.3526	**84.38**
Proposed model (Ensemble)	단어, 음소	**0.2914**	**87.75**

4. 결론 및 향후 연구

이 연구에서는 기존 문서 분류에 사용된 자연어 처리 방법론을 한국어 감성 분석에 적용하여 비교실험을 하여 한국어 자연어 처리에 적합한 모델을 분석하고 새로운 Parallel Stacked Bidirectional LSTM 구조를 제안하였다. 제안한 모델이 단어 단위, 음소 단위 전처리 방법 모두에 기존 모델보다 높은 성능을 보였으며 최종 두 가지 전처리 방법을 통한 모델 앙상블을 통해 가장 높은 성능을 보였다.

우리는 마지막에 다른 방식의 전처리 모델의 앙상블이 성능 개선에 열쇠임을 확인하였다. 영어와 달리 한국어에는 단어 및 음소 단위 전처리 방법이외에도 음절 단위의 전처리 방법도 가능하다. 따라서 음절 단위의 전처리 방법을 이용한 분류 모델과의 앙상블 연구를 진행할 것이며, 각 단일 모델의 성능 향상을 위해 깊은 LSTM 모델의 문제점인 정보 손실(Information loss)을 개선한 Attention 메커니즘을 적용하는 등의 연구를 지속 할 것이다.

참 고 문 헌

[1] Mikolov, Tomas, et al, "Efficient estimation of word representations in vector space.", arXiv preprint arXiv:1301.3781, 2013.

[2] Kim, Yoon, "Convolutional neural networks for sentence classification.", arXiv preprint arXiv:1408.5882, 2014.

[3] Lai, Siwei, et al, "Recurrent Convolutional Neural Networks for Text Classification.", AAAI, Vol. 333, 2015.

[4] Zhang, Xiang, Junbo Zhao, and Yann LeCun, "Character-level convolutional networks for text classification.", Advances in neural information processing systems, 2015.

[5] Dowoo Kim and Myoung-Wan Koo, “Categorization of Korean News Articles Based on Convolutional Neural Network Using Doc2Vec and Word2Vec”, Journal of KIISE, vol.44. no. 7, pp.742-747, 2017.

[6] Jung-Mi Kim and Ju-Hong Lee, "Text Document Classification Based on Recurrent Neural Network Using Word2vec,", Journal of Korean Institute of Intelligent Systems, Vol. 27, No. 6, pp. 560~565, Dec, 2017.

[7] Sak, Haşim, Andrew Senior, and Françoise Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling.", Fifteenth annual conference of the international speech communication association, 2014.

[8] 이재준 and 안성만, "한글 음소 단위를 적용한 딥러닝 모형의 감성분석,", 한국지능정보시스템학회 학술대회논문집, pp. 113~114, Aug, 2017.

[9] Huang, Zhiheng, Wei Xu, and Kai Yu, "Bidirectional LSTM-CRF models for sequence tagging.", arXiv preprint arXiv:1508.01991, 2015.

[10] “Naver sentiment movie corpus v1.0”, Available: https://github.com/e9t/nsmc.