사전훈련 된 모델을 통한 한국어 임베딩 성능 비교

Comparison of Korean Embedding Performance using Pre-trained Model

박진배*,1)

Jinbae Park*,1)

[초 록]

자연어 처리 분야는 통계적 기반 방식에서 뉴럴 네트워크인 딥러닝 기반 방식이 적용 되면서 급격한 성장을 이루었다. 딥러닝 모델은 사람의 개입 없이 모델 스스로 처음부터 끝까지 이해하도록 유도하는 엔드 투 엔드 기법이 사용되었다. 하지만 자연어 처리 모델을 학습시키기 위해서는 많은 하드웨어 자원이 필요하다. 하드웨어 한계를 극복하기 위해 다양한 사전훈련 된 모델을 이용하여 파인튜닝으로 다운스트림 태스크에 적용하고 있다. 사전훈련 된 모델 중 가장 대표적인 모델을 BERT(Bidirectional Encoder Representations from Transformer) 기술이며 다양한 태스크에 적용되고 있다. 본 논문에서는 감성분류 데이터를 사전훈련 된 한국어 모델을 적용하여 기존 신경망 모델과 성능을 비교해본다.

[ABSTRACT]

The field of natural language processing achieved rapid growth as a deep learning-based method, which is a neural network, was applied from a statistical-based method. The deep learning model uses an end-to-end technique that induces the model to understand itself from start to finish without human intervention. However, in order to train a natural language processing model, a lot of hardware resources are required. In order to overcome hardware limitations, various pretrained models are used and fine tuning is applied to downstream tasks. Among the pretrained models, the most representative model is BERT (Bidirectional Encoder Representations from Transformer) technology, which is applied to various tasks. In this paper, we compare the performance with the existing neural network model by applying the pretrained Korean model to the subtractive classification data.

Key Words : Natural Language Processing(자연어 처리), Pre-trained Model(사전훈련 된 모델), RNN(순환신경망), Attention (어텐션), Transformer(트랜스포머)

1. 서 론

자연어 처리는 컴퓨터를 이용하여 자연어의 이해, 생성을 다루는 인공지능 기술이다. 자연어 이해는 일상 생활의 언어를 형태 분석, 의미 분석, 대화 분석 등을 통하여 컴퓨터가 처리할 수 있도록 변환시키는 작업이다. 자연어 생성은 컴퓨터가 처리한 결과물을 사람의 편의성에 입각하여 텍스트, 음성, 그래픽 등을 생성하는 작업이다. 이러한 자연어 처리 태스크는 언어를 이해하고 생성함으로써, AI 의사결정 지원 플랫폼 및 AI 복합지는 플랫폼으로 발전할 것으로 기대하며 앞으로 국방 분야에 지휘통제를 하기 위한 AI 요소기술로 꼽히고 있다.

자연어처리를 위한 기술로는, 컨볼루션 신경망(Convolution Neural Network, CNN), 순환 신경망(Recurrent Neural Network, RNN), 어텐션(Attention) 메커니즘이 있다. 이러한 뉴럴 모델의 장점은 특징 집합을 자동으로 추출할 수 있다는 것이다. 그래서 사람이 쉽게 자연어 처리 분야의 태스크를 해결할 수 있도록 해준다. 하지만 이러한 자연어처리 이점에도 불구하고 컴퓨터 비전(Computer Vision, CV)분야에 비해서 성능 향상이 덜 이루어지고 있는 것이 현실이다. 주된 이유로는 현재 자연어처리를 하기 위한 데이터 셋이 적기 때문이다. 딥 뉴럴 네트워크는 대부분 많은 하이퍼 파라미터를 필요로 하기 때문에 적은 데이터 셋으로는 과적합 문제로 인해 일반화를 시키기 어렵다.

최근에는, 실질적인 업무에서 사전훈련 된 모델(Pre-trained Model, PTM)들이 대용량의 Corpus로 다양한 언어들을 학습하여 발표되었다. 이 모델들은 다운스트림 자연어처리 모델에서 좋은 성능을 보여주고 있다. 이러한 모델은

1) 한화시스템 기반기술연구소
(Infra Technology R&D Center, Hanwha Systems, Korea)
* Corresponding author, E-mail: smallchange@hanwha.com

Received: August 5, 2020 Revised:
Accepted: August 26, 2020

새로운 모델을 처음부터 훈련하지 않고, 추가 학습만으로도 일반 신경망보다 더 나은 성능이 발휘된다. 자연어 처리의 1세대 PTM은 좋은 단어 임베딩을 배우는 것을 목표로 했다. 하지만 Skip-gram, Glove는 단어 의미론적 학습은 할 수 있지만, 맥락을 파악하는 높은 수준의 개념을 이해하기는 어려웠다. 2세대 PTM은 CoVe, ELMo, OpenAI GPT, BERT와 같이 문맥을 이해하기 위한 워드 임베딩에 초점이 맞춰졌다. 이렇게 사전훈련된 인코더는 다운스트림에 의한 추가 학습으로 다양한 목적의 태스크에 사전훈련 된 모델을 적용할 수 있다.[1] 본 논문에서는 감성 분석을 RNN을 이용한 sequence-to-sequence 문제로 정의하고, 기존 신경망 기반 모델과 사전 훈련 모델 성능을 비교해 본다.

2. 자연어 처리 신경망

2.1 컨볼루션 신경망

CNN[2]은 사람의 시신경망에서 아이디어를 얻어 고안한 모델로, 다양한 패턴 인식 문제에 사용되고 있다. CNN은 Convolution Layer, Sub-sampling Layer를 번갈아 가며 학습하고, 마지막에 있는 Fully-connected layer를 이용하여 분류를 수행한다. Convolution Layer는 입력에 대해 2차원 필터링을 수행하고, Sub-sampling Layer는 맵핑 된 2차원 이미지에서 최댓값을 추출한다. 이러한 계층구조 속에서 역전파(Back propagation)를 이용, 오차를 최소화하는 방향으로 학습해나간다. 1D-CNN의 경우, 2차원 특징을 1차원으로 변환하여 학습함으로써, 자연어 또한 학습이 가능하게 된다.

2.2 순환신경망

RNN[3]은 현재 데이터가 이전 데이터의 정보를 받는 순차적인 데이터를 학습하는 모델이다. 바닐라 RNN의 경우 장기 의존성 문제가 있어, 이것을 해결하기 위해 LSTM(Long-Short Term Memory)[4]와 GRU(Gated Recurrent Units)[5] 모델이 제안되었다. LSTM은 긴 순차적인 정보를 게이트 메커니즘을 통해 저장하고 출력할 수 있다. 하지만 게이트의 추가로 복잡성이 올라간 LSTM을 개선하기 위해 파라미터 수를 줄인 GRU가 제안되었다. GRU는 리셋 게이트와 업데이트 게이트로 구성되어 있으며, 두 게이트의 상호작용을 통해 학습한다. LSTM보다 적은 파라미터를 사용하기 때문에 이론적으로는 학습 속도가 조금 더 빠르고 완전한 학습에 필요한 데이터가 LSTM보다 적게 필요하다. 그러나 실제 성능으로는 특정 태스크에서는 더 뛰어나기도 하고 뒤쳐지기도 한다.[6]

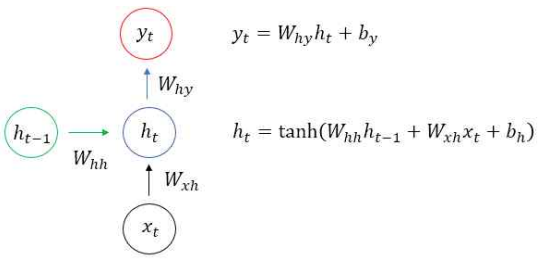

그림 1. RNN 메커니즘
Fig. 1. RNN Mechanism

2.3 어텐션 메커니즘

어텐션 모델은 최근 NLP 분야에서 질문 응답, 기계 번역, 음성 인식 등에서 다양한 분야에서 사용되고 있다. 해당 알고리즘이 각광받는 이유는 문장에서 가장 중요한 부분을 매칭해 줄 수 있다는 것이다.

RNN 기반의 메커니즘은 학습 시 입력 데이터 전부 동등하게 활용하여 적절하게 문맥 관계를 형성하기 어렵다. 반면, 어텐션 메커니즘은 학습 시 입력 데이터 전부를 동등하게 활용하지 않고 각 부분에 가중치를 매겨 필요한 입력에만 집중하게 할 수 있다.

초기 어텐션 LSTM 모델의 내부 상태 값들의 시퀀스에 가중치를 매기는데 사용했지만, 최근에는 LSTM 모델 자체가 어텐션 모델들로 대체되고 있다. 이렇게 인코더와 디코더를 어텐션 레이어로만 구성한 모델들이 기존 모델들보다 뛰어난 성능을 보이고 있다. 뿐만 아니라 RNN 모델들은 시퀀스가 전부 입력되어야 Back Propagation Through Time(BPTT)로 학습할 수 있는데 비해 어텐션 모델들은 병렬성이 높아 같은 시간동안 더 많은 양의 데이터를 학습할 수 있다.

2.4 트랜스포머

기존 LSTM 모델을 통한 임베딩은 입력을 순차적으로 처리하기 때문에, 앞의 입력들만을 이용해 현재 입력에 대한 상태를 생성한다. 이를 보완하기 위해 Bidirectional LSTM 모델[7]과 셀프 어텐션(Self Attention) 모델[8]이 존재한다.

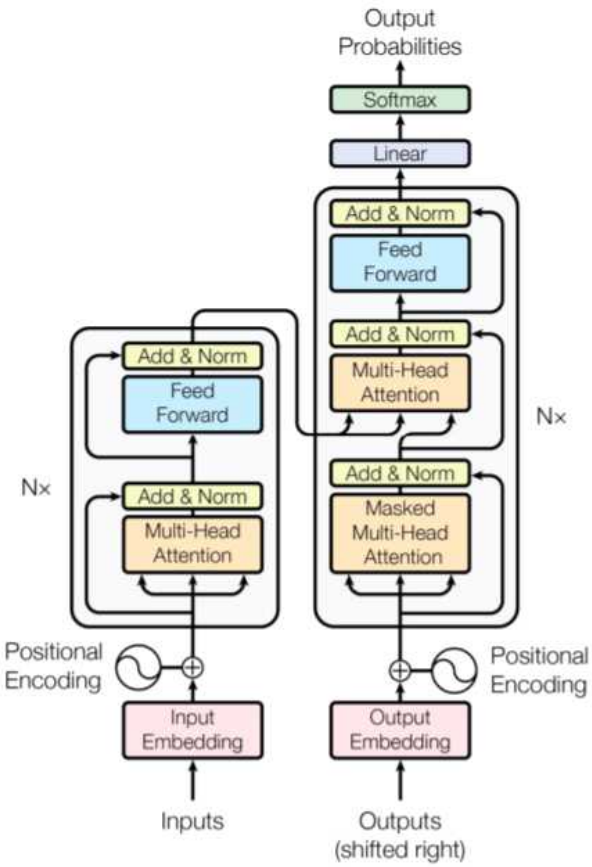

그림 2. 트랜스포머 구조
Fig. 2. Transformer structure

트랜스포머 모델은 Fig. 2와 같이 포지셔널 인코딩과 멀티헤드 어텐션 레이어, feed forward 레이어로 구성된 Nx개의 인코더들과 마스크 된 멀티헤드 어텐션, 멀티헤드 어텐션, feed forward 레이어를 적용한 Nx개의 디코더들로 구성된다. 셀프 어텐션 모델은 쿼리(Query)와 모든 키(Key)의 유사도를 가중치로 적용하여 키와 맵핑되어 있는 각각의 값(Value)에 반영한다. 이를 통해 문장 안에서 어떤 한 단어의 의미를 다른 단어들과의 연관성 가중치 정도를 이용해 판단할 수 있다. 셀프 어텐션은 Fig. 3에서와 같이 학습이 잘 되었을 경우, 'it'의 의미를 파악할 때 가장 높은 가중치를 가진 입력 'animal'을 통해 'it'의 의미를 파악할 수 있다. 기존의 seq2seq의 어텐션 구조로는 찾을 수 없었지만 셀프 어텐션을 적용하여 가능해진 메커니즘이다. 멀티 헤드 어텐션은 이러한 셀프 어텐션을 헤드 개수만큼 학습시켜 더 나은 성능을 얻고자 하는 모델이다. 그러나 이러한 셀프 어텐션 모델은 기존 RNN 모델에 비해 시퀀스 안에서의 입력 순서를 감안하지 못하는 문제가 있다. 따라서 이를 보완하기 위해 위치 인코딩(positional encoding)이 사용됐다. 이는 각 입력의 순서에 따라 생성한 위치 벡터 값을 더하는 방식으로 이루어진다.

3. 사전 훈련된 모델(Pre-trained Model)

최근에 자연어 처리 분야에서 트랜스포머 구조를 이용해 더욱 성능을 높인 새로운 모델들이 나타나고 있다. 대표적으로 구글의 BERT가 그 중 하나이다. BERT는 트랜스포머의 인코더 부분만을 이용해 입력 시퀀스를 벡터들로 표현하는 모델이다. BERT는 원하는 목적의 데이터로 학습하기 전에 라벨링 되지 않은 문장 데이터들을 이용해 미리 언어 모델을 학습(Pre-training)시키고, 그 후에 원하는 목적의 데이터로 모델을 최적화 시키는 방식으로 학습된다.

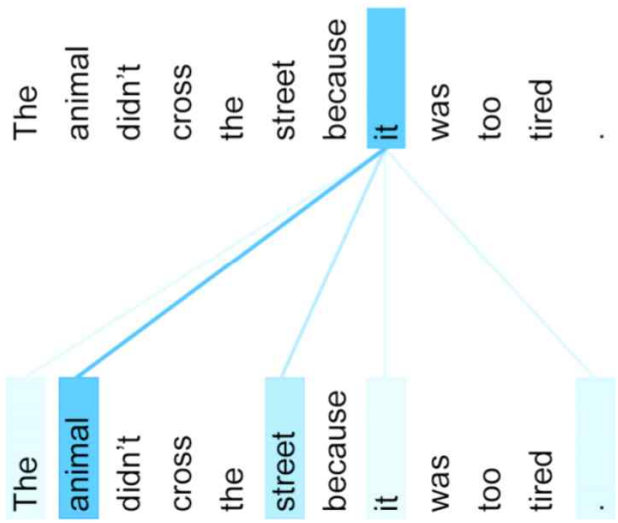

그림 3. 셀프 어텐션 내 가중치 관계
Fig. 3. Weight relationship in self attention

BERT는 사전 학습 단계에서 입력 단어들 중 15%를 랜덤으로 마스킹하고, 마스킹 된 단어를 모델이 예측하는 방식으로 학습시킨다. 그래서 단어가 포함된 문맥에 따라 그 단어의 임베딩 값을 변화시키도록 만들었다. 이 모델을 SQuAD(The Stanford Question Answering Dataset) 데이터 셋을 통해 학습시킨 결과, 최초로 자연어 처리 분야에서 인간을 뛰어 넘는 성능을 보여주었다.[9] SQuAD 데이터 셋을 학습한 모델 중 BERT 이전에 높은 성능을 보여주었던 ELMo는 BERT와 비슷하게 많은 양의 문장을 이용해 사전학습을 거쳤으나, BERT와는 다르게 사전 학습 방식이 간소했고, Bidirectional LSTM 기반으로 상대적으로 낮은 성능을 보여주었다. BERT 이전의 또 다른 모델로 OpenAI에서 만든 GPT(Generative Pre-Training) 모델이 있다. GPT 모델은 트랜스포머를 이용해 만들어졌지만 ELMo와 비슷하게 사전 학습 방식이 간소해서 BERT와 같은 성능을 내지는 못하였다. BERT의 경우 Seq2Seq에서 사용된 인코더 (Encoder)를 이용해 언어를 이해하도록 미리 학습하고, GPT는 디코더(Decoder)를 이용해 사전 학습을 진행한다.

4. 성능비교

본 논문에서는 성능 비교를 위한 데이터로 NSMC(Naver Sentiment Movie Corpus)를 이용하였다. NSMC의 학습 데이터는 15만개, 평가 데이터는 5만개로 구성된다. RNN과 CNN의 경우, 단어 토큰은 형태소 단위로 분리하였다. 형태소 분석기는 Okt를 적용하고, 한글/공백을 제외한 문자 제거, 불용어 제거 등 데이터 전처리를 진행하였다. BERT는 Wordpiece기반 토크나이저를 적용한다.
하이퍼 파라미터 설정은 다음과 같다. 최대 문장 길이는 128로 설정하였고, 배치크기는 32, 히든 스테이트 차원 수는 768으로 설정하였다. CNN의 경우, 필터크기는 128로 설정하였다. 사전 훈련된 모델은 가장 다양한 모델에 적용되고 있는 BERT(Multilingul-cased) 모델을 적용하였다. KoBERT는 SKTBrain팀에서 한국어 코퍼스로 사전 학습한 모델을 적용하였다.

표 1. NSMC를 활용한 성능비교
Table 1. Performance comparison using NSMC

Model	Test Acc(%)
LSTM(128 Layer)	85.79
Bidirectional LSTM(128 Layer)	84.67
1D-CNN(128 Layer)	84.72
BERT(Multilingul-cased)	87.00
KoBERT(SKTBrain)	89.59

Table 1을 참조하면, 사전 학습 모델 중 한국어로 학습한 KoBERT 적용 시 성능이 가장 좋음을 볼 수 있다. BERT(Multilingul-cased)의 다국어로 학습을 했기 때문에, 한국어 학습 데이터가 Kobert에 비해 적어서 성능이 더 낮게 나

왔다. 그리고 Bidirectional LSTM의 성능이 LSTM보다 더 성능이 낮게 나왔는데, 하이퍼파라미터 수치가 Bidirectional LSTM에 최적화 되어 있지 않다는 뜻이다. 이렇듯, 딥 뉴럴 네트워크에서는 태스크에 맞는 하이퍼 파라미터 튜닝이 중요한 것을 알 수 있다. 하이퍼 파라미터 튜닝에 따라 테스트 정확도 결과 값이 달라질 수 있다.

5. 결 론

본 논문에서는 기존 신경망과 사전 훈련된 모델을 통해 NSMC를 활용한 성능을 비교하였다. 사전 훈련된 모델 사용은 기존 신경망 모델보다 더 나은 성능을 보이고, AI 개발자가 모델을 설계하는 시간을 줄여준다는 면에도 효율적이다.
또한, 매년 데이터 량은 늘어 현재 ZB 수준의 데이터 량을 보이고 있다. 사람이 직접 모든 데이터를 처리할 수 없기 때문에, 미래에 AI 역할은 클 것으로 예상된다. 또한 미래 전에서도 사람이 모든 상황을 판단하기는 어렵기 때문에 AI의 중요성이 점점 부각될 것이다. 이러한 사전훈련 된 모델을 이용하여 민관이 협력해 AI 플랫폼을 개발한다면, 선도국과의 기술적인 격차를 해소하면서 효율적인 국방 시스템을 구축할 수 있을 것이다.

References

[1] Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. “Pre-trained Models for Natural Language Processing: A Survey”, CoRR, abs/2003.08271, 2020.
[2] LeCun, Yann, and Yoshua Bengio. "Convolutional networks for images, speech, and time series," In M.A. Arbib (Ed.), The handbook of brain theory and neural networks, Cambridge, MA: MIT Press, pp.255-258, 1995
[3] Goller, Christoph, and Andreas Kuchler. "Learningtask-dependent distributed representations bybackpropagation through structure." Neural Networks,IEEE International Conference on. Vol. 1. IEEE, 1996.
[4] Hochreiter, Sepp, and Jürgen Schmidhuber. "Longshort-term memory." Neural computation 9.8,pp.1735-1780, 1997
[5] Cho, Kyunghyun, et al. "Learning phraserepresentations using RNN encoder-decoder forstatistical machine translation." arXiv preprintarXiv:1406.1078, 2014.
[6] Jozefowicz, Rafal, Wojciech Zaremba, and IlyaSutskever. "An empirical exploration of recurrentnetwork architectures." Proceedings of the 32ndInternational Conference on Machine Learning(ICML-15). 2015.
[7] Schuster, Mike, and Kuldip K. Paliwal. "Bidirectional recurrent neural networks.", IEEE Transactions on Signal Processing 45.11, pp.2673-2681, 1997.
[8] Cheng, Jianpeng, Li Dong, and Mirella Lapata. "Long short-term memory-networks for machine reading.", arXiv preprint arXiv:1601.06733, 2016.
[9] “The Stanford Question Answering Dataset”, https://rajpurkar.github.io/SQuAD-explorer/