한국어 문서 분류를 위한 신경망 구조 탐색

지병규°[1]
고려대학교 인공지능융합학과
gub115@korea.ac.kr

Neural Architecture Search for Korean Text Classification

ByoungKyu Ji°[1]
Department of [1]Applied Artificial Intelligence, Korea University

요 약

최근 심층 신경망을 활용한 한국어 자연어 처리에 대한 관심이 높아지고 있지만, 한국어 자연어 처리에 적합한 신경망 구조 탐색에 대한 연구는 이뤄지지 않았다. 본 논문에서는 문서 분류 정확도를 보상으로 하는 강화 학습 알고리즘을 이용하여 장단기 기억 신경망으로 한국어 문서 분류에 적합한 심층 신경망 구조를 탐색하였으며, 탐색을 위해 사전 학습한 한국어 임베딩 성능과 탐색한 신경망 구조를 분석하였다. 탐색을 통해 찾아낸 신경망 구조는 기존 한국어 자연어 처리 모델에 대해 4 가지 한국어 문서 분류 과제로 비교하였을 때 일반적으로 성능이 우수하고 모델의 크기가 작아 효율적이었다.

주제어: 심층 신경망, 문서 분류, 신경망 구조 탐색

1. 서론

문서 분류는 기계 번역, 질의응답 등 자연어 처리의 다양한 과제 중에서 기본적인 과제이다. 심층 신경망을 이용한 자연어 처리는 많은 수의 매개 변수를 학습하기 때문에 다른 자연어 처리 방법에 비해 학습한 모델의 크기가 크다는 단점이 있지만, 보다 좋은 성능을 보여주어 널리 활용되고 있다. 특히, Transformer [1] 구조를 이용한 BERT [2], GPT [3] 등이 제안되었고 이를 활용하여 한국어 처리에 특화된 모델들도 공개되었다. 하지만 이러한 심층 신경망을 이용한 한국어 자연어 처리 연구의 대부분은 기존 신경망 구조를 그대로 두고 영어 말뭉치를 한글 말뭉치로 변경하는 방식으로 진행되어 아직까지 한국어 자연어 처리에 적합한 신경망 구조는 밝혀지지 않았다.

Text-NAS[4]는 자연어 처리를 위한 탐색 공간을 정의하고 문서 분류에 적합한 신경망 구조를 탐색하여 찾아낸 신경망 구조로 다양한 문서 분류 데이터에 대해 전이 학습하여 성능을 분석하였다. 그러나 신경망 구조 탐색 및 문서 분류 성능을 분석한 데이터는 모두 영어 말뭉치로 한국어 문서 분류에 대한 신경망 구조 탐색은 아직 진행되지 않았다.

따라서 본 논문에서는 한국어 문서 분류에 적합한 신경망 구조를 탐색하고 탐색한 신경망 구조에 대해 분석하였으며, 기존 한국어 문서 분류 모델과의 성능 비교를 진행하였다.

2. 관련 연구

신경망 구조 탐색

신경망 구조 탐색은 과제에 적합한 인공 신경망의 구조를 자동으로 탐색하는 방법으로 이미지 처리, 자연어 처리 등 다양한 기계학습 분야에서 활용하고 있으며 탐색 공간, 탐색 전략, 성능 추정 전략 등 3 가지 차원에 따라 세부 방법을 분류할 수 있다 [5]. 강화학습을 적용하여 신경망 구조를 탐색한 [6]은 탐색 공간을 합성곱 신경망의 경우 각 계층의 구조와 skip connection으로, 순환 신경망의 경우 Cell 내부 노드의 연산으로 각각 정의하였으며 탐색 전략으로는 REINFORCE [7]를 활용하여 검증 데이터의 정확도로 신경망 구조 탐색을 위한 장단기 기억 신경망을 학습하였다. 학습한 장단기 기억 신경망을 이용하여 검증 정확도를 최대화하도록 신경망 구조를 추출하였으나 이러한 신경망 구조 탐색 방법은 많은 컴퓨팅 연산을 요구한다는 단점이 있었다.

이러한 단점을 해결하기 위하여 효율적인 신경망 구조 탐색 방법인 NASNet [8]과 ENAS [9]가 제안되었으며, NASNet [8]에서는 기존에 계층마다 탐색하던 CNN 구조를 Normal Cell과 Reduction Cell 2 개의 Cell에 대한 탐색공간으로 정의하여 구조를 탐색하고 탐색한 Cell을 중첩하여 쌓아서 효율적인 신경망 구조 탐색 방법을 제안하였다. ENAS [9]는 방향 비순환 그래프(DAG)를 활용한 가중치 공유를 통해 [6]에 비해 1000 배가 넘는 GPU 학습 시간을 절약하는 방법을 제안하였다.

Text-NAS [4]는 자연어 처리를 위해서 계층 별로 8 개의 탐색공간을 제안하고 Stanford Sentiment Treebank(SST) [10] 데이터에 대해서 영어 말뭉치로 사전 학습한 GloVe [11] 임베딩을 활용하여 신경망 구조 탐색을 진행하였으며 탐색한 신경망 구조를 문서 분류 뿐만 아니라 자연어 추론에도 적용하였다. Text-NAS [4]에서 발견한 신경망 구조는 ENAS [9]를 활용하여 찾은 신경망 구조나 BERT [2]보다 문서 분류와 자연어 추론에

서 좋은 성능을 내었다.

문서 분류

문서 분류는 문서가 어떤 범주에 속하는지 분류하는 과제로 스팸 분류, 감성 분류, 의도 분류 등 다양하게 응용되고 있으며 전반적인 과정은 전처리, 토큰화, 특징 값 추출, 학습 및 예측으로 진행된다. 특히, 문서의 특징 값을 추출하는 과정이 문서 분류의 성능에 많은 영향을 주며 특징 값을 추출하는 방법은 문서의 단어 빈도 수나 등장 여부로 단어-문서 행렬이나 TF-IDF를 구성하여 통계량을 직접 활용하는 방법에서 단어 수준으로 사전 학습한 임베딩을 사용하는 방법으로 발전하였다. [12, 13]

단어 수준 임베딩을 사전 학습하는 방법으로는 이전 단어가 주어졌을 때 다음 단어를 예측하거나 [14], 주어진 문맥에 맞는 단어를 잘 예측하도록 인공 신경망을 학습하는 방법 [15]이 제안되었으며 단어 수준 임베딩을 사전 학습하는 대표적인 모델로는 Word2Vec [15, 16], 단어들의 동시 발생 빈도를 활용한 GloVe [11], 단어 내부의 하위 단어도 학습이 가능한 fastText [17]가 있다. 사전 학습한 임베딩을 활용하여 문서 분류하는 방법으로는 합성곱 신경망 [18]이나 재귀 신경망을 활용하는 방법 [10], 합성곱 신경망과 순환 신경망을 모두 활용하는 방법 [19, 20]들이 제안되었다.

최근에는 문장 내부의 단어를 예측하는 Masked Language Modeling(MLM) 방식으로 말뭉치를 사전 학습하고, 하위 과제에 미세 조정하여 활용하는 딥러닝 모델인 BERT [2]가 문서 분류 뿐만 아니라 다양한 자연어 이해에서도 뛰어난 성능을 보여주었다. 하지만 한국어에 있어서는 다른 언어와 함께 사전 학습한 다국어 모델(Multilingual BERT)만이 공개되어 좋은 성능을 보여주지는 못했고, 이러한 문제를 해결하고자 한국어 말뭉치로 사전 학습하여 한국어에 특화된 KoBERT[1], KcBERT [21], KR-BERT [22] 등 다양한 모델이 공개되었으며 다국어 모델(Multilingual BERT)보다 한국어 문서 분류에 있어서 더 좋은 성능을 보여주었다.

3. 신경망 구조 탐색

3.1 탐색 공간

탐색 공간은 Text-NAS [4]와 동일하게 정의하였으며 각 계층마다 입력 계층의 선택, 이전 모든 계층에 대한 skip-connection 여부, 신경망 유형으로 크게 3 가지가 있다. 입력 계층의 선택에서는 동일한 신경망 구조의 탐색을 방지하기 위해 이전 5 개 계층으로 입력 계층을 제한하였으며 탐색하려는 신경망 유형은 8 개로 각각 1, 3, 5, 7 개의 필터 크기를 갖는 1 차원 합성곱 신경망과 3 개의 필터 크기를 갖는 Average Pooling, Max pooling, 양방향 GRU [23]와 Multi-head Self Attention [1]으로 정의하였다.

[1] https://github.com/SKTBrain/KoBERT

3.2 탐색 전략

탐색 전략은 [6]과 동일하게 순환 신경망의 일종인 장단기 기억 신경망 [24]을 사용하여 신경망을 추출하고, 추출한 전체 신경망에 대한 검증 데이터의 정확도의 기대값을 보상으로 최대화 하도록 학습한다. 장단기 기억 신경망을 θ_c, 1 부터 T 까지 각 계층 별 추출하는 신경망 구조를 $m_{1:T}$, 검증 데이터의 정확도를 R로 하면 최대화하려는 보상의 기댓값 $J(\theta_c)$는 수식 1로 표현할 수 있다.

$$J(\theta_c) = \mathbb{E}_{\pi(m_{1:T};\theta_c)}[R] \tag{1}$$

장단기 기억 신경망을 학습하기 위하여 보상의 기댓값에 대한 정책 경사를 REINFORCE [7]로 표현하면 수식 2와 같으며 장단기 기억 신경망으로부터 신경망 구조 전체를 추출할 때마다 구할 수 있다.

$$\nabla J(\theta_c) = \sum_{t=1}^{T} \mathbb{E}_{\pi(m_{1:T};\theta_c)}[\nabla_{\theta_c} \log \pi(m_t|m_{(t-1):1};\theta_c)R] \tag{2}$$

정책 경사의 기댓값을 근사하면 수식 3로 표현할 수 있다.

$$\nabla J(\theta_c) \approx \frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T}[\nabla_{\theta_c} \log \pi(m_t|m_{(t-1):1};\theta_c)R_i] \tag{3}$$

N은 장단기 기억 신경망을 학습하기 위해 추출하는 신경망 구조의 개수이며 R_i는 i 번째 추출한 신경망 구조의 검증 데이터에 대한 정확도이다. 정책 경사의 분산을 줄이기 위하여 보상 R_i에서 기준 함수 b를 적용하면 수식 4과 같으며, 기준 함수 b는 보상에 대한 지수 이동 평균이다.

$$\nabla J(\theta_c) \approx \frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T}[\nabla_{\theta_c} \log \pi(m_t|m_{(t-1):1};\theta_c)(R_i - b)] \tag{4}$$

3.3 성능 추정 전략

장단기 기억 신경망을 활용한 [6]의 신경망 구조 탐색 방법은 컴퓨팅 시간이 과도하게 소요되기 때문에 ENAS [9]의 성능 추정 전략을 활용하여 탐색 공간의 전체 신경망 구조를 공유하여 학습하면서 성능을 추정한다. 전체 신경망 구조 ω의 학습은 확률적 경사 하강법으로 진행하며, 장단기 기억 신경망이 추출한 신경망 구조 m의 손실 함수 $\mathcal{L}(m;\omega)$의 기댓값을 최소화하도록 몬테 카를로 방법으로 추정하여 수식 5로 계산할 수 있다.

$$\nabla_{\omega}\mathbb{E}_{m\ \pi(m;\theta_c)}[\mathcal{L}(m;\omega)] \approx \frac{1}{M}\sum_{i=1}^{M}[\nabla_{\omega}\mathcal{L}(m_i,\omega)] \tag{5}$$

이 때, 손실 함수 $\mathcal{L}(m;w)$는 학습 데이터에 대한 교차 엔트로피이며 ENAS [9]의 성능 추정 전략으로 전체 신경망 구조 ω를 효율적으로 학습할 수 있다.

표 1. 한국어 문서 분류 데이터 정보

	학습 데이터	평가 데이터	범주 개수
NSMC	150,000	50,000	2
YNAT	45,678	9,107	7
Hate/Bias	7,896	974	3

4. 실험

먼저 한국어 말뭉치로 GloVe, fastText [11, 17]을 사전 학습하고, 이를 이용하여 네이버 영화 리뷰 감성 데이터(NSMC)[2]에 대해 신경망 구조 탐색을 진행하였다. 찾아낸 신경망 구조를 뉴스 주제 분류 데이터(YNAT) [25], 한국어 혐오 표현 데이터(Hate, Bias) [26]에도 전이 학습하여 성능을 비교 분석하였으며 한국어 문서 분류에 활용한 데이터는 표 1에서 확인할 수 있다.

4.1 한국어 임베딩 사전 학습

한국어 임베딩 사전 학습을 위하여 한국어 위키피디아, 나무위키, KcBERT [21] 사전 학습에 사용한 네이버 뉴스 댓글 데이터로 구성한 말뭉치를 이용하였다. 한국어 위키피디아와 나무위키는 특수 문자, 이메일, 웹 페이지 주소 등을 제거하고 [13] KSS [27]로 문장을 분리하였다. 모든 말뭉치는 mecab-ko[3]을 이용해서 형태소 단위로 토큰화하고 임베딩 차원은 300으로 하여 GloVe, fastText [11, 17]로 학습하였다. 표 2는 학습 방법 및 어휘 개수에 따른 한국어 임베딩과 기존 한국어 임베딩 모델 [28, 29]을 단어 유사도 평가와 의미론적 질의에 대한 단어 유추 평가 [28]를 진행하여 비교하였다. 한국어 말뭉치로 학습한 fastText [17]가 GloVe [11]보다 단어 유사도 평가에서 좋은 성능을 보였으나, 단어 유추 평가에서는 GloVe [11]가 좋은 성능을 보였다. fastText [17]는 어휘의 수가 감소하면 단어 유사도 평가 성능이 악화되지만 단어 유추 평가 성능은 개선되었고, GloVe [11]는 반대로 단어 유사도 평가 성능이 개선되고 단어 유추 평가 성능은 악화되었다. 한국어 말뭉치를 학습한 fastText [17]은 기존 한국어 임베딩 모델 [28, 29]에 비해 단어 유사도 평가와 단어 유추 평가에서 모두 좋은 성능을 보였다.

4.2 신경망 구조 탐색

신경망 구조 탐색은 학습한 어휘의 수가 많고 단어 유사도, 유추 평가에서 좋은 성능을 보인 표 2의 fastText(736k)를 6, 12, 24 계층에 대해 각각 가중치 공유 방식으로 NSMC 데이터를 학습하면서 진행했다. NSMC 데이터를 한 번 학습할 때마다 장단기 기억 신경망을 20 번 학습하고 신경망 구조를 추출했으며,

[2]https://github.com/e9t/nsmc

[3]https://bitbucket.org/eunjeon/mecab-ko/src/master/

표 2. 한국어 임베딩 성능 비교

	단어 유사도 평가		단어 유추 평가
학습 방법 (어휘 수)	spearman	pearson	의미론적 질의
fastText (736k)	0.665	0.643	83.3%
fastText (60k)	0.618	0.593	85.0%
GloVe (736k)	0.552	0.580	91.0%
GloVe (60k)	0.611	0.622	88.6%
Lee et al. (60k) [28]	0.561	0.559	66.2%
cc.ko.300. (2M) [29]	0.585	0.564	77.7%

추출한 신경망 구조 중에서 검증 데이터에 대해 가장 정확도가 높은 신경망 구조를 다른 문서 분류 과제에 전이 학습하였다. 24 계층에 대해 탐색한 신경망 구조는 그림 1과 같고 사각형의 첫 번째 줄은 장단기 기억 신경망이 추출한 계층의 순서이며 다음 줄은 탐색 공간에서 추출한 신경망 구조로 검은 실선은 입력, 회색 점선은 skip connection, 회선 실선은 각 계층의 출력의 선형 결합이 마지막 출력으로 더해지는 것을 표현하였다. 처음 문서의 입력은 임베딩과 합성곱 신경망을 거쳐 추출한 신경망 구조의 입력으로 들어가며, 추출한 모든 신경망 구조의 출력에 대한 선형 결합이 마지막 출력이 되어 global max pooling과 fully connected 계층을 거쳐 문서를 분류한다. 장단기 기억 신경망이 추출한 24 계층 신경망 구조는 합성곱 신경망 10 개, pooling 6 개, 양방향 GRU 3 개, self attention 5 개로 이루어져 있으며 영어 데이터인 SST [10]에 비슷한 신경망 구조 탐색 방법을 사용한 Text-NAS [4]와 비교하였을 때 합성곱 신경망의 개수는 적고 self-attention의 개수가 많다는 차이점과 양방향 GRU와 self-attention이 차례로 이어서 나오는 경우가 없이 앞뒤로 합성곱 신경망이나 pooling 계층이 나타나고 양방향 GRU가 전체 신경망 구조의 초기에 보이는 공통점이 있었다.

4.3 전이 학습 결과

탐색한 신경망 구조를 NSMC[2], YNAT [25], 한국어 혐오 표현 데이터(Bias, Hate) [26]에 하이퍼 파라미터 최적화를 활용하여 전이 학습하고 기존 모델과 성능 및 크기를 비교 분석하였다. 전이 학습에 사용한 한국어 임베딩은 각 문서 분류 학습 및 검증 데이터에 나타난 형태소를 활용하여 과제마다 파라미터의 개수가 다르며 표 3에는 그 중에서 가장 큰 값을 표현하였다. 표 3은 4 가지 한국어 문서 분류 과제를 6, 12, 24 계층의 신경망 구조 탐색으로 찾아낸 모델과 기존 모델에 대해 비교하였다. 비교한 기존 모델은 Kc-BERT [21]과 모델 경량화를 위한 지식 증류 [30, 31]를 KoBERT[1]에 적용한 Distil-KoBERT [32]로 KcBERT [21]는 분류 범주 개수가 적은 데이

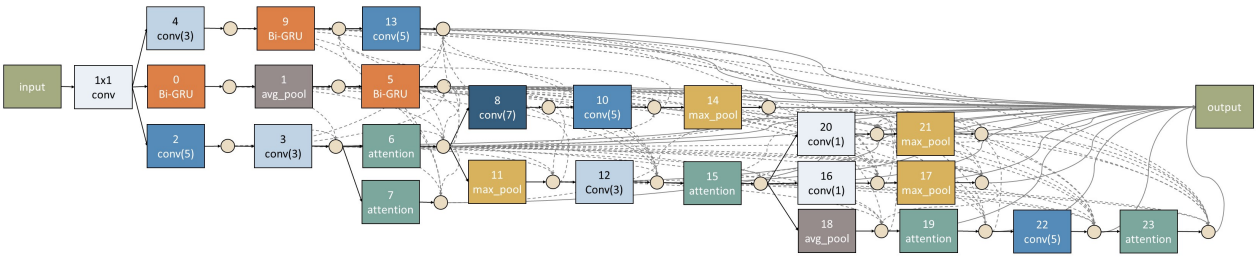

그림 1. 한국어 문서 분류를 위한 신경망 구조

터에 대해서는 좋은 성능을 보이지만, 분류해야하는 범주 개수가 많은 YNAT [25] 데이터에 대해서는 좋지 않은 성능을 보였으며 Distil-KoBERT [32]는 반대로 YNAT [25]에 대해 좋은 성능을 보이며 나머지 데이터에 대해 안 좋은 성능을 보였다. 신경망 구조 탐색으로 찾아낸 모델의 성능이 기존 모델에 비해 4 가지 문서 분류 과제에 일반적으로 우수하였고 크기는 감소하였으나 찾아낸 모델은 한국어 임베딩을 사전 학습한 736k 개의 형태소 단위 임베딩을 모두 불러오지 않고 각각의 문서 분류 데이터에 나타난 형태소만을 파라미터 개수로 계산하였고, 하이퍼 파라미터 최적화를 적용하지 않고 비지도학습 방법으로 토큰화하여 [33] 임베딩에 활용한 기존 모델 [21, 32]와 엄밀하게 비교 가능하지 않다는 한계가 있다.

5. 결론

본 논문에서는 한국어 문서 분류를 위한 신경망 구조 탐색을 위해 GloVe, fastText [11, 17]로 한국어 임베딩을 새로 학습하여 기존 모델과 비교하였으며, NSMC[2] 데이터에 대한 검증 정확도를 보상으로 REINFORCE [7] 알고리즘과 가중치 공유 방식 [9]을 활용하여 장단기 기억 신경망을 학습하였다. 학습한 장단기 기억 신경망으로 추출한 신경망 구조를 다른 한국어 문서 분류 과제에도 전이 학습하여 그 성능을 기존 모델과 비교하였으며 찾아낸 신경망 구조의 한국어 문서 분류 성능은 기존 모델에 비해 일반적으로 우수하고 모델의 크기는 작아 효율적이었다. 그러나 신경망 구조 탐색을 자연어 처리 과제 중에서 문서의 길이가 짧은 문서 분류 과제에만 적용하였으며, 활용한 단어 수준 임베딩은 문맥을 파악할 수 없는 한계가 있으므로 향후에는 문맥까지 학습할 수 있는 문장 수준으로 학습하는 BERT [2], GPT [3]의 사전 학습과 다양한 자연어 처리 과제에 대해서도 한국어 말뭉치에 대하여 신경망 구조 탐색을 진행할 계획이다.

참고문헌

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, Vol. 30, 2017.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[3] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever *et al.*, "Improving language understanding by generative pre-training," 2018.

[4] Y. Wang, Y. Yang, Y. Chen, J. Bai, C. Zhang, G. Su, X. Kou, Y. Tong, M. Yang, and L. Zhou, "Textnas: A neural architecture search space tailored for text representation," *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34, No. 05, pp. 9242–9249, 2020.

[5] F. Hutter, L. Kotthoff, and J. Vanschoren, *Automated machine learning: methods, systems, challenges*. Springer Nature, 2019.

[6] B. Zoph and Q. V. Le, "Neural architecture search with reinforcement learning," *arXiv preprint arXiv:1611.01578*, 2016.

[7] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine learning*, Vol. 8, pp. 229–256, 1992.

[8] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," *Proceedings of the IEEE conference on computer*

표 3. 한국어 문서 분류 성능 비교

	NSMC	YNAT	Hate	Bias	Model size
학습 모델	accuracy	Macro-F1	Macro-F1	Macro-F1	# parameters
NAS 24-layers	89.20	85.36	67.73	78.20	15.2M
NAS 12-layers	89.44	85.15	68.19	78.16	13.1M
NAS 6-layers	89.34	85.21	66.12	77.80	11.5M
KcBERT-large [21]	90.68	83.23	68.60	75.07	334M
KcBERT-base [21]	89.62	82.23	68.08	75.90	108M
Distil-KoBERT [32]	88.41	85.38	60.58	70.96	27.8M

vision and pattern recognition, pp. 8697–8710, 2018.

[9] H. Pham, M. Guan, B. Zoph, Q. Le, and J. Dean, "Efficient neural architecture search via parameters sharing," *International conference on machine learning*, pp. 4095–4104, 2018.

[10] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 1631–1642, 2013.

[11] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.

[12] 임희석, *자연어처리 바이블: 핵심이론 응용시스템 딥러닝*. 휴먼싸이언스, 2019.

[13] 이기창, *한국어 임베딩: 자연어 처리 모델의 성능을 높이는 핵심 비결 Word2Vec에서 ELMo, BERT까지*. 에이콘출판사, 2019.

[14] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," *Journal of Machine Learning Research*, Vol. 3, pp. 1137–1155, 2003.

[15] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[16] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in neural information processing systems*, Vol. 26, 2013.

[17] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the association for computational linguistics*, Vol. 5, pp. 135–146, 2017.

[18] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint arXiv:1408.5882*, 2014.

[19] C. Zhou, C. Sun, Z. Liu, and F. Lau, "A c-lstm neural network for text classification," *arXiv preprint arXiv:1511.08630*, 2015.

[20] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification," *Proceedings of the AAAI conference on artificial intelligence*, Vol. 29, No. 1, 2015.

[21] J. Lee, "Kcbert: Korean comments bert," *Proceedings of the 32nd Annual Conference on Human and Cognitive Language Technology*, pp. 437–440, 2020.

[22] S. Lee, H. Jang, Y. Baik, S. Park, and H. Shin, "Kr-bert: A small-scale korean-specific language model," *ArXiv*, Vol. abs/2008.03979, 2020.

[23] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[24] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, Vol. 9, No. 8, pp. 1735–1780, 1997.

[25] S. Park, J. Moon, S. Kim, W. I. Cho, J. Han, J. Park, C. Song, J. Kim, Y. Song, T. Oh *et al.*, "Klue: Korean language understanding evaluation," *arXiv preprint arXiv:2105.09680*, 2021.

[26] J. Moon, W. I. Cho, and J. Lee, "BEEP! Korean corpus of online news comments for toxic speech detection," *Proceedings of the Eighth International Workshop on Natural Language Processing for Social*

Media, pp. 25–31, Jul. 2020. [Online]. Available: https://www.aclweb.org/anthology/2020.socialnlp-1.4

[27] H. Ko and S.-k. Park, "Kss: A toolkit for korean sentence segmentation," https://github.com/hyunwoongko/kss, 2021.

[28] D. Lee, Y. Lim, and T. Kwon, "Morpheme-based efficient korean word embedding," *Journal of KIISE*, Vol. 45, No. 5, pp. 444–450, 2018.

[29] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, and T. Mikolov, "Learning word vectors for 157 languages," *arXiv preprint arXiv:1802.06893*, 2018.

[30] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[31] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019.

[32] J. Park, "Distilkobert: Distillation of kobert," https://github.com/monologg/DistilKoBERT, 2019.

[33] T. Kudo and J. Richardson, "Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing," *arXiv preprint arXiv:1808.06226*, 2018.