Research Article

pISSN : 2287-8920
https://doi.org/10.22716/sckt.2020.8.4.033

Attention에 기반한 한국어 언어모델 연구

이선정[1]
[1]국립인천대학교 컴퓨터공학부 교수

A Study on Korean Language Model Based on Attention

Sunjeong Lee[1]
[1]Professor, Incheon National University

[1]**Corresponding author:** sunjlee@inu.ac.kr

Received October 11, 2020; **Revised** November 4, 2020; **Accepted** December 17, 2020

ABSTRACT

본 논문에서는 어텐션(atention)에 기반을 둔 한국어 언어모델에 관한 연구를 수행하였다. 대표적인 어텐션 모델로 셀프 어텐션(self-attention)이 가능한 트랜스포머(transformer)가 있다. 트랜스포머는 인코더와 디코더로 구성이 되는데 언어모델로는 디코더를 일반적으로 사용한다. 한국어에 적용 실험을 하기 위해서 기본 토큰 단위로 센텐스피스(SentencePiece)를 사용하여 구하였다. AI-Hub 한국어 평가 코퍼스 60만 문장을 이용하여 성능 비교를 한 결과 5,000개의 센텐스피스 토큰을 사용한 것이 10,000개의 센텐스피스 토큰을 사용한 것과 비교하였을 경우 33.4%의 복잡도가 감소하였다. 또한 한국어 음성인식 실험을 통하여 복잡도 성능이 우수한 5,000개의 센텐스피스 토큰을 갖는 언어모델의 성능이 우수하다는 것을 보였다.

In this paper, we make a study on the language model based on attention. The representative attention model is a transformer model, which enables a self-attention. Even though the transformer model consists of encoder and decoder, decoder is usually used for language model. We build a sentence-piece model for tokenizing. The experimental result yields that the token unit number of 5000 gets the perplexity(ppl) reduction of 33.4% compared with that of 10,000 when AI-Hub corpus (https://www.aihub.org.kr) is used in the sentence-piece model. In order to prove the performance of language model with regard to perplexity, we make an experiment of speech recognition that the model with low perplexity yields better performance than that with high perplexity.

Keywords: Language model, Language model based on DNN, Transformer for language model, Sentence-piece tokenizer

1. 서 론

언어모델의 연구는 언어학자가 정의한 규칙을 근거로 한 문법 기반 언어모델과 언어의 사용성에 대한 통계를 기반으로 한 통계적 언어모델로 구성된다. 문법 기반 언어모델은 형태소분석, 구분분석 등을 위한 문법을 개발하는 것이고 통계적 기반 언어모델은 매 단어의 사용성에 대한 통계를 구하고 그것을 bigram, trigram 등으로 구성되는 n-gram으로 표현하는 것이다. 최근에 신경망 기술의 발달로 인해 신경망을 이용해서 언어모델을 구현하는 방식이 다양하게 제안되고 있다. 신경망을 활용한 언어모델 방식은 전통적인 feed forwad 신경망을 이용한 언어모델, LSTM(Long Short-Term Memory) 기반의 언어모델이 사용되고 있다[1,2]. 특히 전통적인 통계기반 언어모델[3]과 신경망 기반의 언어모델의 비교 연구가 수행되었으며 LSTM 기반의 언어모델의 우수함이 증명되었다[4].

최근에는 기계번역에 사용되고 있는 트랜스포머(transformer) 모델을 다양한 분야로 확대하고 있다[5]. 트랜스포머 모델의 장점은 셀프 어텐션(self-attention)이 가능하다는 것이다. 기존의 어텐션은 seq-to-seq 모델에서 인코더와 디코더와의 관계에서만 사용이 되었는데 셀프 어텐션은 인코더 혹은 디코더 단일로 사용 가능하다는 것이다. 이러한 방식을 이용하여 인코더만 사용하는 버트(BERT), 디코더만 사용하는 GPT2, GPT3 등이 제안되고 있다[6-8].

본 논문에서는 신경망 기반 언어모델을 설계하는 방안을 제안하고 제안된 모델을 통한 언어모델 성능뿐만 아니라 실제로 음성인식에 사용된 음성인식 성능까지 측정한다. 언어모델을 구성하기 위해서는 언어 분석을 위한 기본 토큰 유닛이 정의되어야 한다. 한국어의 경우에는 형태소가 언어처리의 기본으로 인식되고 있으나 최근에는 word piece model(WPM) 등이 많이 사용되고 있다[9]. 특히 WPM을 사용하면 전 세계 모든 언어에 활용될 수 있으므로 다국어 시스템을 개발하는 경우 도움이 될 수 있다. 구글에서는 WPM 기반으로 전 세계의 음성 인식기를 개발하고 있다[9]. 최근에는 자연어처리에 특화되고 기본 유닛을 구하는 데 시간이 빠르고 문장에서 기본 유닛을 구하는 데 필요한 인코딩 시간과 반대로 기본 유닛으로부터 문장을 복원하는데 필요한 디코딩 시간을 줄인 센텐스피스(SentencePiece)가 제안되었다[10].

본 논문에서 제안된 신경망 기반 언어모델은 자기 집중 기능이 있는 트랜스포머를 사용하는 것이다. 특히 한국어 언어모델의 기본단위로 센텐스피스를 사용하는 것을 제안한다. 2장에서는 언어모델을 위한 기본단위를 구하는 센텐스피스 모델 및 트랜스포머 알고리즘을 검토하고 3장에서는 트랜스포머 언어모델 구조 설계 및 구현에 관해서 기술한다. 4장에서는 트랜스포머에 기반한 언어모델 알고리즘 실험 결과 및 분석을 수행하고 음성인식 실험으로 확장한다. 마지막으로 5장에서 결론을 맺는다.

2. 언어모델을 위한 기본단위 구성 및 트랜스포머

2.1 언어모델을 위한 기본단위인 센텐스피스

언어모델을 위한 기본단위로는 형태소와 WPM이 있다[9]. WPM 모델은 음절을 기준으로 하여 통계적인 사용이 많으면 음절이 확대되는 방법을 취한다. 이 방식에 따르면 WPM 모델의 최대 크기는 어절이 되고 최소 크기는 하나의 음절이 된다. WPM을 이용해서 기본 토큰 단어 유닛을 찾아내는 순서는 많이 알려져 있으나 공식적으로 공개되어 있지 않다[9].

센텐스피스는 기계번역을 위한 기본 유닛으로 개발되었으며 문장이 인코더와 디코더를 거치면 원래의 문장이 되도록 즉, 손실이 없는 인코더와 디코더를 목표로 개발이 되었다. 또한 인코더, 디코더 시간을 단축하며 모든 언어에 적용하도록 개발되었다. Table 1에는 “Hello world”가 어떻게 센텐스피스로 인코딩이 되며 그것이 어떻게 원래의 문장으로 다시 디코딩되는 예제에 대한 과정을 보여주며 실제로 훈련 및 인코딩, 디코딩 등 모든 SW가 구글의 공개 SW 사이트 (https://github.com/google/sentencepiece)에 공개되어 있다.

Table 1. Examples of commands for SentencePiece

훈련 명령어 : 기본 유닛 1,000개를 input.txt에서 추출 %spm_train −input=data/input.txt --model_prefix=spm −vocab_size =1000
인코딩 명령어 : ”Hello World”를 인코딩 %echo “Hello world.”| spm_encode −model=spm.model 결과: _He ll o_ world .
디코딩 명령어 : _He ll o_ world 를 디코딩 %echo “_He ll o_ world”	spm_dncode −model=spm.model 결과:. Hello world.

2.2 트랜스포머

LSTM은 시간 축으로 변하는 패턴을 모델링하는데 우수한 결과를 제시하므로 단어와 단어간의 연결 상황을 잘 표현해 주는 언어모델에 적용될 수 있다[10]. Fig. 1에는 트랜스포머의 구조가 표현되어 있다. Fig. 1의 트랜스포머는 입력을 처리하는 인코더와 출력을 처리하는 디코더로 구성되어 있다. 인코더는 N개의 레이어로 구성되며 각 레이어의 입력은 이전 레이어의 출력으로 연결되고 각 레이어의 출력은 그 위의 다음 레이어 입력으로 연결되어 있다. 맨 아래의 레이어 입력은 인코더의 입력 문장을 기본 유닛 단위로 나누고 그것을 벡터값으로 표시하는 임베딩 과정의 결과로 이루어진다. 즉 인코더 입력 문장은 시간 축으로 연결되는 벡터의 열, 즉 매트리스로 변경된다.

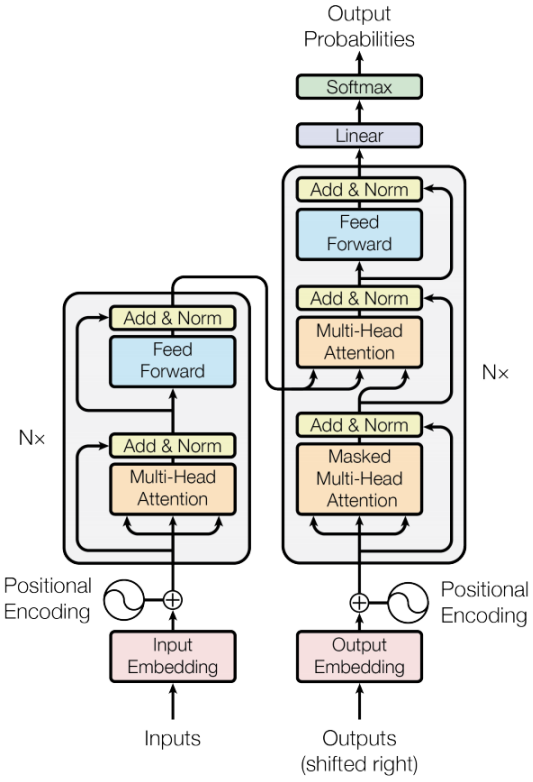

Fig. 1. The architecture of transformer

각 레이어는 어텐션 관계를 Q(query), K(key),V(value) 매트리스로 표시하는 스케일드 닷 프로덕트 어텐션(scaled dot-product attention)과정과 그것을 이용한 멀티헤드 어텐션으로 구성되어 있다. Fig. 2는 스케일드 닷 프로덕트 어텐션의 구조도이며 Q,K,V 계산 흐름도로 표시되어 있다. 기본 개념은 Q,K,V 를 매트리스로 구성시키고 그것을 수식 (1)과 같이 계산해 주면 전통적인 곱셈 어텐션(multiplicative attention)이 된다는 것이다[5].

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

어텐션의 종류는 덧셈 어텐션[11]과 곱셈 어텐션으로 나누어지는데 프랜스포머는 곱셈 어텐션이다.

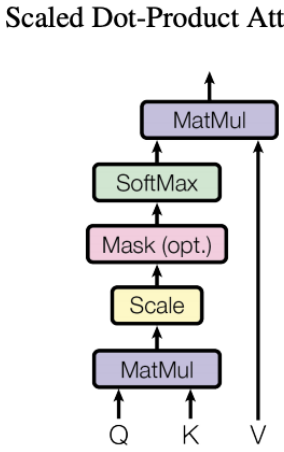

Fig. 2. Flow of computation for Q,K,V

Fig. 3은 각 레이어를 구성하고 있는 멀티 헤드 어텐션의 구성도를 나타내고 있다. QKV로 이루어진 스케일드 닷 프로덕트 어텐션을 h개로 구성하고 이것을 모아 W 매트릭스로 곱한 것이 멀티 헤드 어텐션이 된다는 것을 보인다. 수식 (2)는 멀티 헤드 어텐션을 Q,K,V 및 W로 표시하였다.

$$MltiHead(Q,K,V) = Concat(head_1,..,head_h)\,W^Q$$
$$where\ \ head_i = \mathrm{Attentio}\left(\mathrm{QW}_i^{\mathrm{Q}}, \mathrm{KW}_i^{\mathrm{K}}, \mathrm{VW}_i^{\mathrm{V}}\right) \tag{2}$$

트랜스포머의 가장 큰 특징은 셀프 어텐션이 가능하다는 것이다. 즉 Q,K,V를 모두 입력 혹은 출력으로 정하면 Q와 가장 연관이 깊은 K를 구할 수 있으며 이것을 적당한 K값으로 스케일을 할 수 있는 구조이다. 또한 인코더의 출력을 디코더의 입력으로 정하면 전통적인 seq2seq의 어텐션이 되는 것이다. 특히 Fig. 3의 디코더에서는 셀프 어텐션의 Q에 관련이 있는 K를 구할 때 미래의 값에 영향을 받지 않도록 코절리티(causality)가 될 수 있도록 마스크드 멀티 어텐션이 구성되도록 하였다.

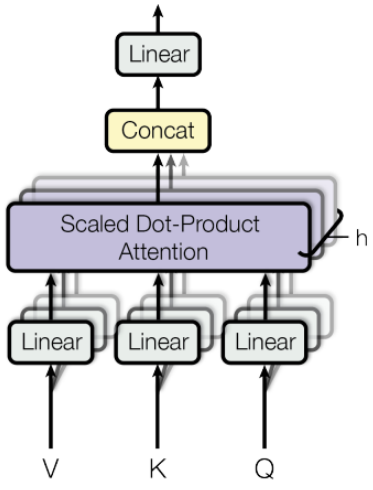

Fig. 3. The architecture for multi-head attention

3. 트랜스포머 기반 언어모델 구조 설계

Fig. 4에는 트랜스포머에 기반한 언어모델의 적용 개념도가 그려져 있다. Fig. 4는 Fig. 1의 트랜스포머 기본 구조도에서 디코더만 사용한 것이고 마스크드 멀티헤드 어텐션이 적용된 것이다. 본 논문에서는 훈련 코퍼스로 AI-Hub에 있는 코퍼스만 사용하였는데 만약 코퍼스를 한국어에 대한 많은 코퍼스로 대처를 할 경우에는 GPT2가 되는 것이다[7].

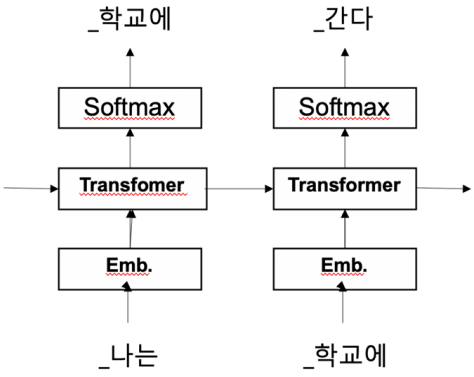

Fig. 4. An example of a language model based on transformer in Korean

Fig. 4에는 입력이 특정 시간 t에서 "_나는" 이 트랜스포머에 입력될 때 출력이 "_학교에" 가 되고 그 단어가 특정 시간 t+1에 트랜스포머의 입력이 되면 출력이 "_간다"가 되어 언어모델이 예측되는 결과를 보여준다. 여기서 softmax 는 출력 단어에 대한 최고 확률을 구하기 위한 정규화 수식을 나타낸 것이다. Emb.는 "_나는" 등과 같이 기본 토큰 단위를 단어표현 벡터로 변경시키는 임베딩 과정을 의미한다. 일반적으로 트랜스포머에서는 단어표현 벡터 차원이 정해지면 임의 값으로 초기화를 한다.

4. 한국어 언어모델 알고리즘 시뮬레이션

4.1 한국어 코퍼스 분류

AI Hub(https:// aihub.or.kr)에는 자연언어, 음성, 영상 등의 데이터가 공개되어 있다. 본 논문에서는 한국어 자유 발화 학습데이터를 이용하였다. 이 데이터는 조용한 환경에서 623명이 발성한 한국어 대화 음성 622,545문장으로 구성되며 두 사람이 일상, 쇼핑, 정치, 경제 등 다양한 주제로 자유롭게 대화하는 음성을 녹음하고 전사한 것을 사용하였다. Table 2에는 본 논문에서 사용하고 있는 한국어 코퍼스 구성을 나타내었다.

Table 2. Corpus in Korean

훈련 문장	검증 문장	실험 문장
592,545 (593화자)	15,000 (15화자)	15,000 (15화자)

4.2 센텐스피스 기반 언어모델

4.1절에 기술한 한국어 코퍼스를 활용하여 먼저 그래픽 기반 전처리를 수행한 후에 2.1절에 기술한 센텐스피스 SW를 활용하여 5,000개 및 10,000개의 기본 유닛을 만들었으며 실험 문장에 대한 언어모델 복잡도를 계산하였다.

Table 3에는 한국어 실험 코퍼스 내의 동일 문장에 대해서 5,000개 및 10,000개의 기본 유닛 사전을 활용해서 어떻게 기본 유닛으로 인코딩이 되는지를 보여 준다.

Table 3. Experimental result for corpus in Korean

훈련 코퍼스의 센텐스피스 개수	복잡도
5,000	70.15
10,000	103.62

Table 4를 보면 5,000개의 유닛은 개수가 적기 때문에 문장을 더 세부적으로 나누는 경향이 생긴다. 예를 들면 "쿠팡"일 경우 10,000개의 사전에는 "_쿠팡"이 하나의 기본 유닛으로 인식이 되지만 5,000개의 사전을 사용할 때 "_쿠" , "팡"이 되어서 두 개의 기본 유닛으로 인식되고 있다.

Table 4. Examples of encoding for sentencepiece in Korean experimental corpus

기준 문장(6어절)	지금쯤 뭐 하고 있었을까 쿠팡에 있었을까
5,000 개 유닛(11 유닛)	_지금 쯤 _뭐 _하고 _있었 을까 _쿠 팡 에 _있었 을까
10,000 개 유닛(10 유닛)	_지금 쯤 _뭐 _하고 _있었을 까 _쿠팡 에 _있었을 까

4.3 언어모델 기반 음성인식 실험

복잡도가 작은 언어모델이 음성인식 실험 결과에서도 우수한 결과를 얻는지를 확인하기 위해서 음성인식 실험도 수행하였다. 음성인식의 성능을 측정할 때 일반적으로 기본 유닛의 단위로 측정이 되는데 기본 유닛이 다르면 성능의 차이 비교가 어려워지므로 인식 결과에서 센텐스피스로 디코딩된 것을 활용하여 성능을 비교하였다. 센텐스피스로 디코딩되면 띄어쓰기 단위로 결과가 나오기 때문에 WER(word error rate)는 어절 단위의 오인식률이 된다. Table 5에는 5000개, 10,000개 단위에 따른 기본단위 오인식률, 어절 오인식률을 나타내었다.

Table 5. Performance in speech recognition according to basic units

기본단위 개수	기본단위 WER	어절 WER
5,000	14.0 %	16.7 %
10,000	30.2 %	31.6 %

Table 5를 보면 언어모델 복잡도가 작은 5,000 단위 개수가 음성인식 성능도 좋음을 알 수 있다. Fig. 5에는 기본 유닛 단위로 인코딩된 문장이 5,000 및 10,000 각각에 대해서 어떤 결과가 나왔는지에 대한 인식 결과를 나타내었다.

```
Scores: (#C #S #D #I) 12 0 0 0
REF:  _지 금  쯤  _뭐  _하 고  _있 었  을 까  _ 쿠  팡  에  _있 었  을 까
HYP:  _지 금  쯤  _뭐  _하 고  _있 었  을 까  _ 쿠  팡  에  _있 었  을 까
Eval:
```

Fig. 5(a). Speech recognition result using basic units(5,000)

```
Scores: (#C #S #D #I) 8 1 1 0
REF:  _지 금  쯤          _뭐  _하 고  _있 었 을  까  _쿠 팡  에  _있 었 을  까
HYP:  ********* _그 거 랑  _뭐  _하 고  _있 었 을  까  _쿠 팡  에  _있 었 을  까
Eval: D         S
```

Fig. 5(b). Speech recognition result using basic units(10,000)

Fig. 6(a)는 Fig. 5(a)의 기본 단위 유닛을 어절로 디코딩한 후의 인식 성능의 예를 보여준다. Fig. 5(a)는 12개의 기본단위 유닛을 모두 정확히 인식한 결과를 보여주고 있으며 Fig. 6(a)는 6개의 어절이 정확히 인식되는 것을 볼 수 있다. 반면 Fig. 6(b)에서는 6개의 어절에서 1개의 어절 "지금쯤"이 "그거랑"으로 오인식이 되었다는 것을 보인다. 이렇게 동일 한국어 텍스트를 다른 기본 유닛으로 변경하였을 경우 언어모델의 복잡도가 달라지고 그것은 곧 음성인식 성능에 영향을 미치게 된다. 그러므로 복잡도가 작은 언어모델의 설계가 언어처리뿐만 아니라 한국어 음성인식 성능 향상에 중요한 역할을 한다고 말할 수 있다.

```
Scores: (#C #S #D #I) 6 0 0 0
REF:  지 금 쯤  뭐  하 고  있 었 을 까  쿠 팡 에  있 었 을 까
HYP:  지 금 쯤  뭐  하 고  있 었 을 까  쿠 팡 에  있 었 을 까
Eval:
```

Fig. 6(a). Final speech recognition result using basic units(5,000)

```
Scores: (#C #S #D #I) 5 1 0 0
REF:  지 금 쯤  뭐  하 고  있 었 을 까  쿠 팡 에  있 었 을 까
HYP:  그 거 랑  뭐  하 고  있 었 을 까  쿠 팡 에  있 었 을 까
Eval: S
```

Fig. 6(b). Final speech recognition result using basic units(10,000)

5. 결 론

본 논문에서는 신경망 기반 언어모델을 설계하는 방안을 제안하고 제안된 모델을 통한 언어모델 성능뿐만 아니라 실제로 음성인식에 사용된 음성인식 성능까지 측정하였다. 이를 위해서 어텐션 기반 신경망을 이용한 한국어 언어모델에 관한 연구를 수행하였다. 최근에 다양한 분야에 많이 사용되고 있는 트랜스포머는 승합적 어텐션이고 매트리스 단위로 처리가 되므로 속도가 빠르다는 장점이 있어서 이를 활용한 한국어 어텐션 기반 언어모델 연구를 수행하였다. 먼저 한국어 언어모델에 적용하기 위해서는 한국어에 대한 기본 토큰 단위를 결정해야 한다. 영어의 경우에는 단어를 기본 토큰 단위로 사용한다. 하지만 한국어는 단어를 구분하기가 어렵고 띄어쓰기 단위는 어절로 되어있으나 어절을 기본 단위로 사용하면 너무 개수가 많아지므로 서브워드 단위로 분할하여야 한다. 본 연구에서는 최신에 기계번역에 많이 사용되고 있는 센텐스피스 모델을 활용하여 기본 단위 사전을 구축하였다.

AI-hub에 공개된 한국어 음성 코퍼스를 활용하여 5,000개와 10,000개의 사전을 구축하였고 그것을 이용하여 언어모델 복잡도를 구하였다. 5,000개의 센텐스피스 모델이 10,000개의 센텐스피스 모델과 비교해 언어모델 복잡도가 33.4% 감소하였다. 또한 한국어 음성인식 실험을 통하여 복잡도 성능이 우수한 5,000개의 센텐스피스 토큰을 갖는 언어모델의 성능이 우수하다는 것을 입증하였다. 최종적인 음성인식 성능은 어절 오인식률 16.7%를 얻었다. 트랜스포머 기반으로 한국어 언어모델 연구를 최초로 수행하였으나 향후 좀 더 다양한 실험을 통해서 한국어 기본 단위를 선정하는 프로세스를 검증할 필요가 있겠다.

Acknowledgement

※ 본 논문은 인천대학교 교내연구비 지원으로 연구되었음.

References

1. W. Zaremba, I Sutskever, O.Vinyals, “Recurrent Neural Network Regularization,” arXiv:1409.2329, 2014, arxiv.org
2. 김양훈 외 3인 “LSTM 언어모델 기반 한국어 문장생성”, 한국통신학회 논문지, 제41권 제5호, pp.592-601, 2016.
3. CMU statistical language modeling toolkit(SRILM), http://www.speech.sri.com/pipermail/srilm-user/2003q4/000153.html
4. 이선정, “Long Short-Term Memory에 기반한 한국어 언어모델 연구”, 융복합지식학회 논문지, 제 8권 제1호, pp.19-26, 2020.
5. Ashish Vaswani, et al. “Attention is All You Need,” In proceddings of NIPS 2017.
6. Jacob Devlin, et al., “BERT:Pre-training of Deep Bidirectional Transformers for Language Understanding,” In proceedings of NAACL, pp.171-4186, 2019.
7. Alec Radford, et al.,“Language Models are Unsupervised Multitask Learners,” 2019, openai.com
8. Tom B. Brown, et al.,“Language Models are Few-Shot Learners,” arXiv:2005.14165, May, 2020.
9. MikeSchuster and Kaisuke Nakajima, “Japanese and Korean Voice Search,”, in Proceeding of ICASSP, pp.5149-5152, 2012.
10. Taku Kudo and John Richardson, “SentencePiece: A Simple and Language Independent Subword Tokenizer and Detokenizer for Neural Text Processing,” In proceedings of EMNLP, pp. 66-71, Oct. 2018.
11. Dzmitry Bahdanau, et al.,”Neural Machine translation by Jointly Learning to Align and Translate,” CoRR. Abs/1703.03906, 2017.