Knowledge-incorporating BERT for Response Selection

Janghoon Han
Department of Computer Science and Engineering, Sogang University
hanjh04@naver.com

Abstract

인간과 상호작용하는 대화시스템을 개발하는 것은 인공지능 분야에서 중요한 문제 중 하나이다. 이러한 문제를 해결하기 위해 대화 시스템에 외부 지식을 적용하는 연구는 꾸준히 진행되어 왔다. 하지만 외부지식을 적용하기 위해서는 구조화된 지식이 필요하며, 이 지식을 생성하려면 상당한 자원이 필요하다. 이러한 관점에서 본 연구는 retrieval-based dialogue system 에서 구조화 되지 않는 텍스트를 외부지식으로 사용하는 모델을 제안한다. 기본 모델로 사전학습된 언어모델인 BERT를 사용하고 모델이 외부지식을 학습할 수 있는 방법으로 Post-Training을 사용한다. 이후 Post-trained 된 모델을 Dialog Selection에 적용하여 문제를 해결한다. 기존 BERT 모델에 비해 외부지식을 학습한 모델의 성능이 R_{10}@1 기준 1.4% 향상된 결과를 보였다. 결과적으로 현재 Ubuntu corpus V1 데이터셋에 대해서 2위의 성능을 달성하였다.

1 Introduction

대화시스템의 목적은 자연스럽고 일관된 대화를 목적으로 한다. 대화시스템은 크게 두 가지로 나뉘는데 하나는 여러개의 후보중에 적절한 응답을 찾는 retrieval-based (or response selection) dialogue system이고, 다른 하나는 직접 응답을 생성하는 generative dialogue system 이다.

Goal-oriented 대화 시스템에서는 외부지식을 사용한다. 식당 예약 대화 시스템에서 식당에 관한 정보, 항공편 예약 대화시스템에서 항공편에 관한 정보등이 바로 그것이다. 외부 지식은 Knowledge Base와 같은 구조화된 형식으로 주어지거나 텍스트와 같이 구조화되지 않는 형식으로 주어진다. 전자는 지식의 상관 관계를 파악하기 쉽고 추출하기 쉽다는 장점을 가진다. 그러나 특정 분야에 대해서 Knowledge Base가 없을 수도 있고 새롭게 구현하기 위해선 상당한 자원이 필요하다는 단점이 있다. 후자는 텍스트이기 때문에 따로 지식을 구축하는 노력은 필요없으나 문제 해결에 필요한 지식을 추출하기 어렵다는 단점이 있다.

최근 언어모델은 자연어처리 전반 분야에서 적용되고 있다. 특히 BERT(Devlin et al., 2018)는 자연어 추론, 질의응답 시스템등 다양한 분야에서 적용되어 높은 성능을 보이고 최근 retrieval-based 대화 시스템에서도 BERT기반의 모델이 좋은 성능을 내었다. 성능향상의 이유로 언어모델이 대용량 코퍼스를 사전 학습하면서 코퍼스에 관한 일반지식을 학습한다는 연구(Petroni et al., 2019)가 제시되었다.

최근 Dialog selection 분야에서 외부지식을 사용하려는 연구는 (Chaudhuri et al., 2018)에 의해 시도되었다. 외부지식을 사용하기 위해 Ubuntu command description으로부터 특정 명령어를 서술하는 한 문장을 가져와서 사용한다. 하지만 이 연구는 제한된 크기의 지식을 가진다는 한계를 가진다.

본 연구는 정형화 되지 않는 텍스트 형식의 지식을 사용하기 위한 새로운 방법을 제안한다. 기존 Pre-Trained된 언어모델을 외부지식 텍스트를 사용하여 Post-Training 한 후 Post-Trained 된 언어모델을 Dialog selection Task에 Fine-tuning 하여 학습하는 방법이다. 이 방식은 모델의 입력으로 넣어주는 기존방식에 비해 대용량 외부지식 텍스트를 제한없이 학습할 수 있다는 장점이 있다.

2 Related work

(Lowe et al., 2015a)는 Response Selection의 새로운 벤치마크 데이터인 Ubuntu IRC Corpus와 그에 따른 Baseline 모델을 제안하였다. (Kadlec et al., 2015) 연구에서 BI-directional LSTM과 Convolutional Neural Network 적용하여 문제를 해결하고자 하였다. 2017 어텐션(Vaswani et al., 2017)의 등장으로 어텐션을 Dialog Selection에 적용한 Deep Attention Matching Network(DAM) (Zhou et al., 2018)이 제안되었다.

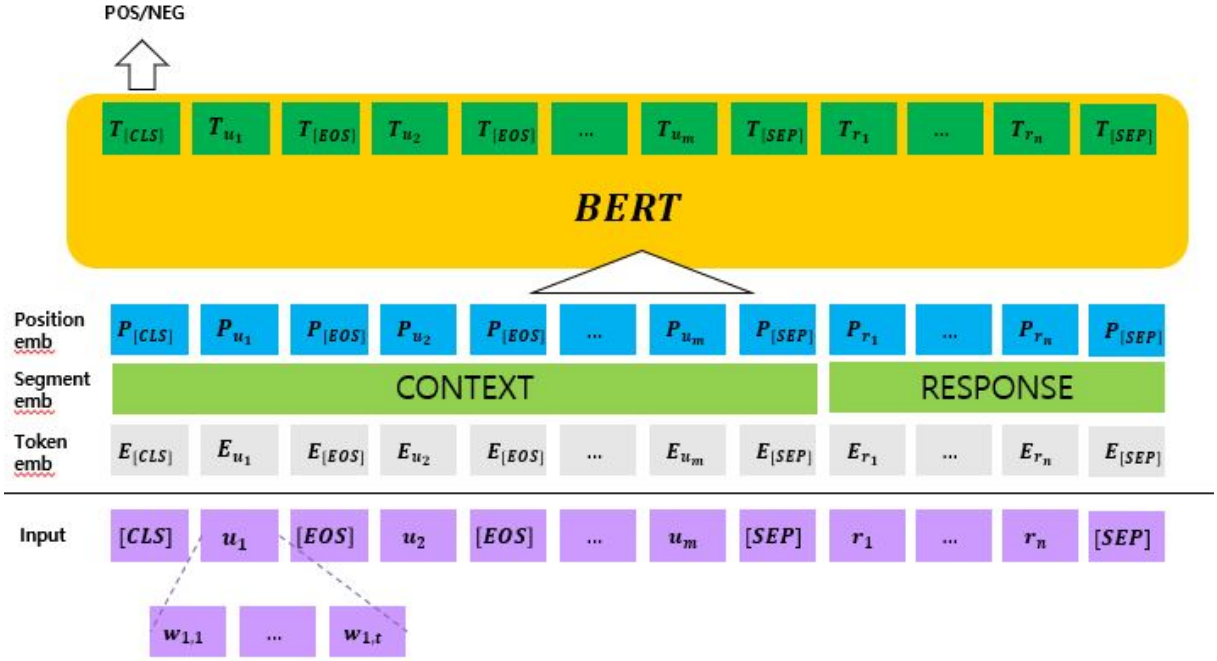

Figure 1: Response Selection Task 에서 BERT 입력

이 후 Cross Attention, Self Attention등 다양한 어텐션 기법을 적용한 모델 등이 등장한다. 최근 모델로는 자연어 추론 모델을 Response Selection 모델에 적용한 Enhanced Sequential Inference Model (Chen and Wang, 2019)있다. 이 모델은 Context와 Response에 대해 어텐션을 수행하여 가중합된 정보를 사용한다. (Tao et al., 2019)는 여러층의 어텐션 레이어를 사용하여 성능을 올리는 방법을 제안하였으며, (Yuan et al., 2019)은 여러 Hop에 대한 어텐션을 수행 하여 다양한 Context 문장을 고려함으로써 성능을 향상시켰다.

외부지식을 Response Selection 에 사용하려는 시도는 (Lowe et al., 2015b) 의 연구에서 제안되었다. 이 연구는 연관된 문서를 TF-IDF 방식을 통해 선택하고 외부지식을 추출하여 사용한다. 또 다른 연구(Chaudhuri et al., 2018)에서는 대화 히스토리에 우분투 명령어가 있으면 명령어에 대응되는 한 문장을 Bi-GRU를 통해 인코딩 한 후 단어 임베딩과 함께 사용하는 방식을 제안하였다.

BERT는 여러 개의 층으로 이루어진 양방향 트랜스포머 인코더이다. Base 모델과 Large 모델 중 Base 모델을 사용하였으며 모델의 layer의 수를 L, 히든레이어의 크기를 H, 셀프 어텐션(self-attention)의 수를 A 라고 하였을 때 Base 모델은 L=12 H=768 A=12를 가진다. BERT는 트랜스포머와는 달리 포지션 인코딩 대신 포지션 임베딩을 사용한다. 여기에 세그먼트 임베딩과 토큰 임베딩을 추가하여 입력으로 넣는다. BERT base 모델은 12개의 인코더 블록을 지니고 있으며 각각의 인코더 블록 안에서는 멀티 헤드 어텐션(Multi-Head Attention)이 수행된다. 이후 Position-wise FFNN을 통과하여 인코더 블록의 출력이 된다. 이렇게 12개의 인코더 블록을 거쳐 BERT 토큰 단위로 최종 출력이 나오게 된다.

3 BERT for Response Selection

BERT를 Response Selection에 적용하기 위해서 Response Selection을 Binary classification으로 접근하였다. 학습 데이터를 $D = \{(c_i, r_i, y_i)\}_{i=1}^{N}$라고 하면 여러개의 Utterance로 이루어진 Context는 $c = \{u_1, u_2, ..., u_m\}$로 나타낸다. 하나의 발화 $u_i = \{w_{i,1}, w_{i,2}, ..., w_{i,l}\}$ 는 여러개의 토큰으로 이루어져 있고 l_i 는 발화의 길이를 의미한다. n개의 토큰으로 이루어진 Response 는 $ri = \{r_1, r_2, ..., r_n\}$ 로 나타낸다. $y_i \in \{0, 1\}$ 는 정답을 의미한다. 기존 BERT의 입력은 $x = ([CLS], sentence1, [SEP], sentence2, [SEP])$로 들어가는데 본 연구에서는 Context와 Response의 관계를 분류하는 것이므로 Figure 1 과 같이 입력 $x = ([CLS], u_1, [EOS], ..., u_m, [EOS], [SEP], r_1, ..., r_n, [SEP])$로 하였다. EOS는 End of sentence를 의미하며 기존의 BERT 입력과는 다르게 본 연구에서는 여러 문장을 넣어주기 때문에

구분자로 설정하였다. 이러한 입력은 포지션, 세그먼트 , 토큰 임베딩으로 BERT의 입력으로 들어가게 된다. 이 후 출력 토큰 중 $T_{[CLS]}$ 토큰을 단일 층의 신경망을 통과시켜 최종 Score $g(c, r)$을 구하게 된다.

$$g(c, r) = \sigma(WT_{[CLS]} + b) \quad (1)$$

이후 학습을 위해 Cross entropy 손실 함수를 사용하여 모델의 가중치를 업데이트 한다.

$$Loss = -\sum_{(c_i, r_i, y_i) \in D} y \log(g(c, r)) + (1 - y) \log(1 - g(c, r)) \quad (2)$$

4 External Knowledge Post-Training

우분투 데이터는 우분투 사용시 발생한 문제에 대한 채팅로그이다. 우분투 대화 데이터에서는 전문지식이 필요한 명령어가 나타나는데 이러한 명령어에 대한 Description은 우분투 Manpage에 존재한다. 본 연구에서는 외부지식 데이터를 확보하기 위해 우분투 명령어 페이지에 대해 크롤링을 수행하여 8313개의 문서를 확보하였다.

한편 사전학습된 BERT는 다른 Task에 Fine-tuning 하는 방식으로 적용된다. 이러한 BERT는 대용량 코퍼스(위키피디아)로 사전 학습되었기 때문에 일반적인 단어나 구에 관한 정보는 있으나 특정 지식에 대한 정보는 부족하다. 특정 외부 지식을 모델에 학습하기 위해 앞서 크롤링한 데이터에 대해 Post-training을 수행한다. 모델은 두 가지의 목적함수를 통해 Post-training 되는데 Masked Language Model (MLM) 방식과 Next Sentence Prediction (NSP)을 사용한다. 첫번째 목적함수의 경우 문장에 특정 토큰을 mask 하고 mask된 토큰을 예측하는 방식이다. 기존의 mask 방식과는 다르게 본 연구에서는 명령어 토큰을 학습하기 위해 명령어 집합에 포함된 토큰만 mask를 하여 학습하도록 하였다. 두번째 함수는 두 가지 문장을 입력으로 하고 두번째 문장이 첫번째 문장의 다음 문장인지 예측하도록 학습한다. 연속된 문장은 각 문서에서 선택하도록 하였다. 두 가지 학습 방법을 통해 모델은 일반적인 코퍼스에서 나타나지 않는 명령어(단어)의 정보를 학습하게 된다. 첫 번째 방법에서는 토큰 레벨에서의 외부지식을 학습하게 되고, 두 번째 방식에서는 Response Selection 필요한 다음 문장과 관련성을 학습하게 된다. 최종적인 함수로써 2가지 방법을 융합해서 손실함수로 사용하게 된다.

$$L_{final} = L_{MLM} + L_{NSP} \quad (3)$$

Model	$R_{10}@1$	$R_{10}@2$	$R_{10}@2$
DAM	0.767	0.874	0.969
IoI-local	0.796	0.894	0.974
ESIM	0.796	0.894	0.975
MSN	0.800	0.899	0.978
$BERT_{Base}$	0.810	0.900	0.977
$BERT_{Knowledge}$	0.824	0.908	0.978

Table 1: Model comparison on Ubuntu Corpus V1.

5 Experiment

5.1 Datasets and Training Setup

모델을 평가하기위해 Ubuntu IRC(Internet Relay Chat) corpus V1을 사용하였다. 우분투 코퍼스에서 Training set은 100k의 Context-Response pair로 1:1의 비율로 정답과 오답으로 이루어져 있다. Validation과 Test set의 경우 50K의 데이터 쌍으로 1:9 비율로 정답과 오답이 존재한다. 평가하기 위한 Metric으로써 Recall을 사용하는데 10개의 후보 응답중에서 정답이 상위 k개의 후보들 중에 존재하는지 평가한다. $R_{10}@k$로 나타내고 $R_{10}@1, R_{10}@2, R_{10}@5$을 사용하였다.

5.2 Baseline Methods

성능 비교를 위한 Baseline은 다음과 같다.

DAM은 트랜스포머 인코더 기반 모델으로 셀프 어텐션과 크로스 어텐션을 적용하였다.

IoI-local은 Aggrigation에서 Multi-Interaction layer를 적용한 모델이다.

ESIM은 NLI의 ESIM 모델을 Response Selection에 적용하였다.

MSN은 Selection Process를 추가하여 k개의 Context Selector에 대해 어텐션을 수행한다.

$BERT_{Base}$는 사전학습된 BERT에 Post-training을 진행하지 않은 Vanilla 모델이다.

$BERT_{Knowledge}$ post-traning을 진행한 모델로서 Masked language model (MLM) 방식과 Next sentence prediction (NSP) 으로 학습한 뒤, Response Selection task에 대한 Fine-tuning을 진행하였다.

5.3 Results and Analysis

실험은 우분투 데이터셋으로 진행되었다. Table 1은 Baseline에 대한 비교와 제안 모델의 최종 성능을 보여준다. $BERT_{Knowledge}$는 Baseline 모델에 비해 $R_{10}@1, R_{10}@2$ 가 각각 2.4% 0.9% 성능향상을 보였다. 또한 $BERT_{base}$ 모델에 비해서도 1.4% 0.8%의 성능 향상이 있는것으로 보아 외부 지식으로 Post-training 한 $BERT_{Knowledge}$가 Domain Specific

Loss	$R_{10}@1$	$R_{10}@2$	$R_{10}@2$
MLM	0.818	0.905	0.978
$MLM_{command}$	0.821	0.906	0.978
NSP	0.817	0.904	0.978
$MLM + NSP$	0.822	0.906	0.978

Table 2: ablation with Loss function.

한 대화 시스템에 도움이 된다는것을 확인하였다.

5.4 Ablation Study

본 연구는 모델에 대해 Ablation 실험을 수행하였다. Table 2는 Post-training의 학습 방법 차이에 따른 Response Selection 모델의 성능 변화이다. Masked language model (MLM) 방법은 기존의 MLM과 명령어 토큰만 Mask 한 $MLM_{command}$를 실험 하였고, 기존의 Next sentence prediction (NSP) 방법과 두 방법을 융합한 $MLM + NSP$를 비교하였다. 기존 MLM 방식과 NSP 방식이 가장 성능이 낮았고 특정 명령어 토큰만을 mask한 $MLM_{command}$ 방식이 성능향상을 보였다. 또한 각각 학습방식으로는 성능 이 낮았으나 MLM, NSP를 함께 학습한 모델이 성능이 가장 높았다.

6 Conclusion

본 연구에서는 정형화되지 않는 외부지식을 Multi-turn Response Selection에 적용하는 방법으로 Post-training 제안하고 평가 하였다. 외부 지식을 사용한 모델이 그렇지 않은 모델보다 성능이 향상됨을 확인하였다. 제안한 모델은 Benchmark 데이터셋인 Ubuntu corpus V1에 대해 **2위**의 성능을 달성하였다.

7 Contribution

120190211 한장훈 100%

References

Debanjan Chaudhuri, Agustinus Kristiadi, Jens Lehmann, and Asja Fischer. 2018. Improving response selection in multi-turn dialogue systems. In *CoNLL*.

Qian Chen and Wen Wang. 2019. Sequential attention-based network for noetic end-to-end response selection. *ArXiv*, abs/1901.02609.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Rudolf Kadlec, Martin Schmid, and Jan Kleindienst. 2015. Improved deep learning baselines for ubuntu corpus dialogs. *ArXiv*, abs/1510.03753.

Ryan Lowe, Nissan Pow, Iulian Serban, and Joelle Pineau. 2015a. The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. In *SIGDIAL Conference*.

Ryan Thomas Lowe, Nissan Pow, Laurent Charlin, and Joelle Pineau. 2015b. Incorporating unstructured textual knowledge sources into neural dialogue systems.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H. Miller, and Sebastian Riedel. 2019. Language models as knowledge bases? *ArXiv*, abs/1909.01066.

Chongyang Tao, Wei Wu, Can Xu, Wenpeng Hu, Dongyan Zhao, and Rui Yan. 2019. One time of interaction may not be enough: Go deep with an interaction-over-interaction network for response selection in dialogues. In *ACL*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *ArXiv*, abs/1706.03762.

Chunyuan Yuan, Wen jie Zhou, MingMing Li, Shangwen Lv, Fuqing Zhu, Jizhong Han, and Songlin Hu. 2019. Multi-hop selector network for multi-turn response selection in retrieval-based chatbots. In *EMNLP/IJCNLP*.

Xiangyang Zhou, Lu Li, Daxiang Dong, Yi Liu, Ying Chen, Wayne Xin Zhao, Dianhai Yu, and Hua Wu. 2018. Multi-turn response selection for chatbots with deep attention matching network. In *ACL*.