# 다중 지문 기계독해를 위한 단락 재순위화 및 세부 단락 선별 기법*

조상현[O1], 김민호[2], 권혁철[1]
부산대학교 전기전자컴퓨터공학과[1], 부산가톨릭대학교 소프트웨어학과[2]
delosycho@gmail.com, minho@cup.ac.kr, hckwon@pusan.ac.kr

# Paragraph Re-Ranking and Paragraph Selection Method for Multi-Paragraph Machine Reading Comprehension

Sanghyun Cho[O1], Minho Kim[2], Hyuk-Chul Kwon[1]
Dept. of Computer Science Pusan National University[1], Dept. of Software Catholic University of Pusan[2]

**요 약**

다중 지문 기계독해는 질문과 여러 개의 지문을 입력받고 입력된 지문들에서 추출된 정답 중에 하나의 정답을 출력하는 문제이다. 다중 지문 기계독해에서는 정답이 있을 단락을 선택하는 순위화 방법에 따라서 성능이 크게 달라질 수 있다. 본 논문에서는 단락 안에 정답이 있을 확률을 예측하는 단락 재순위화 모델과 선택된 단락에서 서술형 정답을 위한 세부적인 정답의 경계를 예측하는 세부 단락 선별 기법을 제안한다. 단락 순위화 모델 학습의 경우 모델 학습을 위해 각 단락의 출력에 softmax와 cross-entroy를 이용한 손실 값과 sigmoid와 평균 제곱 오차의 손실 값을 함께 학습하고 키워드 매칭을 함께 적용했을 때 KorQuAD 2.0의 개발셋에서 상위 1개 단락, 3개 단락, 5개 단락에서 각각 82.3%, 94.5%, 97.0%의 재현율을 보였다. 세부 단락 선별 모델의 경우 입력된 두 단락을 비교하는 duoBERT를 이용했을 때 KorQuAD 2.0의 개발셋에서 F1 83.0%의 성능을 보였다.

주제어: 다중 지문 기계독해, 단락 재순위화, 단락 선별

## 1. 서론

기계독해는 기계가 질문과 해당 질문에 대한 답을 포함하고 있는 지문을 입력받고 입력된 지문 내에서 정답을 찾는 문제이다. 다중 지문 기계독해는 질문과 여러 개의 지문을 입력하고 입력된 지문들에서 정답을 찾고 여러 정답 중에 하나의 정답을 출력하는 문제이다.

기계독해를 위한 대표적인 데이터셋으로는 SQuAD[1]가 있으며, 한국어 기계독해 데이터셋으로는 KorQuAD[2-3]가 있다. KorQuAD 1.0은 질문과 질문에 대한 답을 포함하고 있는 단락을 제공하다. KorQuAD 1.0을 확장한 KorQuAD 2.0은 질문과 질문에 대한 답을 포함하고 있는 여러 단락으로 구성된 HTML 문서를 제공하며, 단락 내의 텍스트뿐 아니라 표나 리스트에서도 정답을 찾아야 하고 단락이나 표, 리스트 전체가 정답이 될 수 있다.

긴 입력을 한 번에 처리할 수가 없어 여러 개의 입력으로 나누어 처리하는 경우, 정답이 있을 확률이 높은 단락의 정답 순위를 높이는 재순위화 방법에 따라서 전체 질의응답 성능은 크게 달라질 수 있다.

다중 지문 기계독해를 이용하는 질의응답 시스템은 입력된 문서에서 여러 개의 지문을 추출하고 질문에 대한 답을 가지고 있을 가능성이 큰 지문들을 추출하고 추출된 지문에서 기계독해 모델을 이용하여 최종 정답을 얻게 된다.

본 논문은 정답을 포함하고 있는 단락을 선택하는 단락 순위화 모델과 추출된 단락에서 세부적인 정답 단락을 선택하는 정답 단락 선택에 관한 내용을 포함하는 다중 지문 기계독해 모형을 제안한다.

## 2. 관련 연구

BERT[4]를 이용한 기계독해 연구가 높은 성능을 보이고 있다. [5]는 다중 지문 기계독해를 위해 사전 학습한 BERT로 인코딩된 문맥 정보와 Tag의 자질 정보를 인코딩하기 위해 SRU(simple recurrent unit)을 추가하고 정답이 없는 경우 정답 위치가 [CLS]의 위치를 출력하도록 하고 정답 타입이 [long, short, no]에서 no가 출력되도록 학습하였다.

[6]는 문서 순위화에 BERT를 적용했으며, 문서 순위화

* 이 논문은 2020년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임(No. 2013-0-00131, (엑소브레인-총괄/1세부)휴먼 지식증강 서비스를 위한 지능진화형 WiseQA 플랫폼 기술개발)

를 위한 데이터셋인 TREC-CAR[7]과 MS-MARCO[8]에서 높은 성능을 보였다.

[9]는 사전 학습 모델과 손실 함수에 따른 단락 재순위화에 대한 성능 비교 실험을 진행하였으며, BERT를 이용하였을 때 보다 RoBERTa를 이용했을 때 성능이 소폭 증가했으며, ELECTRA[10]를 적용했을 때 RoBERTa[11] 보다 성능이 소폭 증가했음을 보였다.

[12]는 BERT를 이용한 다층 구조의 순위화 모델을 제안했다. 다층 구조는 3단계로 이루어져 있는데, 첫 번째 단계는 많은 말뭉치에서 BM25를 이용하여 질문과 관련이 있는 문서를 추출하고 추출된 문서에서 질문과 문서의 쌍을 이용하여 학습하는 monoBERT를 이용하여 관련이 있을 순서를 결정하고 마지막으로 duoBERT를 이용하여 문서들을 재순위화한다. duoBERT는 입력으로 2개의 문서와 질문을 받으며 "질문, 관련 문서, 관련 없는 문서"의 입력에는 1의 확률이 출력되도록, "질문, 관련 없는 문서, 관련 문서"의 입력에는 0이 출력되도록 학습하여 입력된 두 문서 중 어느 문서가 더 질문과 연관이 있는지를 판단하도록 하였다.

본 연구에서는 [5]와 같이 입력된 질문에 대한 답이 단답인지 서술형인지를 결정하기 위해 [long, no, short]를 분류하도록 학습하면서 정답이 있는 문서를 찾는 정확도를 높이기 위해서 멀티 태스크 학습으로 정답이 있는 문서는 1의 확률을 출력하도록 분류기를 추가로 학습하였다. 선택된 단락에 여러 개의 세부 단락이 존재할 때, 정답이 있는 세부 단락을 찾기 위해서 [12]와 같이 "질문, 세부 단락1, 세부 단락2"의 입력으로 duoBERT를 학습하여 더 연관이 있는 세부 단락을 찾도록 하였다.

## 3. 단락 재순위화

본 논문에서는 입력된 문서를 제목 태그를 기준으로 단락으로 나누었다. 나누어진 단락들은 $P=\{p^1,p^2...,p^K\}$은 질문 $Q=\{q_0,q_1,...,q_n\}$, [CLS] 토큰 그리고 [SEP] 토큰과 함께 $X_i=\{[CLS],q_0...,q_n,[SEP],p_0^i,...,p_m^i\}$의 형태로 입력하게 되며 입력의 개수는 나누어진 단락의 개수와 같다. 모델의 출력은 출력되어야 하는 정답이 단답형인지 서술형인지를 나타내는 $Y^T=\{long,short,no\}$와 정답을 포함하고 있을 확률을 출력하는 $Y^v$이며 멀티 태스크 학습을 하였다. $Y^T$의 정답의 길이에 따라서 long과 short을 출력하며 정답 음절의 개수가 40보다 적으면 short을 출력하도록, 더 많으면 long을 출력하도록 하고 정답이 없는 경우에는 no를 출력하도록 학습하였다.

BERT에서 인코딩된 [CLS] 위치의 표현인 $h_0$를 FFNN(Feed-forward Neural Network)에 입력으로 사용하여 결과 $Y^T$와 $Y^V$을 출력한다. 이에 대한 식은 다음과 같다.

$$h_i = BERT(X_i) \in R^{(n+m+2)\times d} \quad (1)$$

$$h_i^T = FFNN^T(h_{i,0}) \in R^3 \quad (2)$$

$$h_i^V = FFNN^V(h_{i,0}) \in R^1 \quad (3)$$

$Y^v$의 경우 softmax를 이용하여 확률을 계산하는 방법과 활성화 함수로 시그모이드 함수를 적용하여 확률을 계산하는 방법을 사용하여 각각 크로스 엔트로피와 평균 제곱 오차를 이용하여 2가지의 손실 함수를 같이 최소화하도록 학습하였다. 손실 함수에 대한 수식은 다음과 같다.

$$P_i^T = \frac{\exp(h_{i,j}^V)}{\sum_{k=0}^{3}\exp(h_k^V)} \quad (4)$$

$$P^V = \frac{\exp(h_i^V)}{\sum_{j=0}^{K}\exp(h_j^V)} \quad (5)$$

$$L = \log(P^T) + \log(P^V) + \frac{1}{K}\sum_{j=0}^{K}\left(\sigma(h_j^V) - \hat{h}_j^V\right)^2 * \gamma \quad (6)$$

여기서 $\gamma$는 0보다 크고 1보다 작은 상수를 의미한다.

질의와 단락에 대해서 형태소 분석을 하고 품사가 명사인 형태소 중에 질의와 단락에서 함께 나타나는 비율을 계산하고 최종적인 순위에 반영하도록 하였다.

### 3.1 세부 단락 선택

단답형이 아닌 서술형 정답에서는 그림2와 같이 선택된 단락 내에서 세부적인 단락 전체가 정답인 경우가 존재한다. 세부적인 단락을 찾기 위해 단답형 정답을 찾을 때 기계독해 모델을 이용하여 세부 단락의 시작 위치와 끝 위치를 찾는 방법과 단락을 세부 단락으로 나누고 [12]의 duoBERT를 이용하여 질문과 두 개의 후보 단락을 입력으로 $X_i=\{[CLS],q_0...,q_n,[SEP],p_0^i,...,p_m^i,[SEP],pj,...,p_m^j\}$와 같은 형태로 입력을 하고 $Y^T=\{0,1,2\}$ 입력된 두 개의 단락 중 첫 번째 단락만 정답에 해당하면 0을 출력하고 두 번째 단락만 정답에 해당하면 1을 출력하고 두 단락 모두 정답에 해당하면 2를 출력하도록 학습하였다. 이에 대한 수식은 다음과 같다.

$$h_i^D = FFNN^D(h_i) \in R^3 \quad (7)$$

$$P_i^D = \frac{\exp(h_{i,j}^D)}{\sum_{k=0}^{3}\exp(h_k^D)} \quad (8)$$

$$L = \log(P^D) \quad (9)$$

$$h_i^R = FFNN^R(h_i) \in R^{(n+m+2)\times 2} \quad (10)$$

$$y_i^S = softmax(h_{i,0}^R) \in R^{(n+m+2)} \quad (11)$$

$$y_i^E = softmax(h_{i,1}^R) \in R^{(n+m+2)} \quad (12)$$

$$L = \log(P^S) + \log(P^E) \quad (13)$$

수식은 기계독해 모델을 이용하여 전체 단락에서 세부

정답 단락에 대한 시작과 끝 위치를 구하는 방법이며 수식은 duoBERT를 이용하여 세부 단락을 구하는 방법이다.

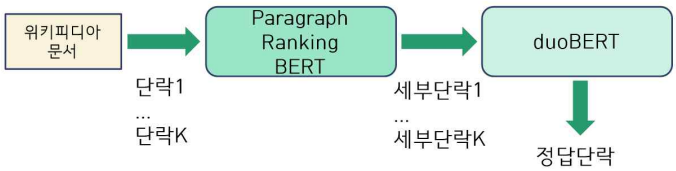


그림1. duoBERT를 이용한 세부 단락 추출 과정

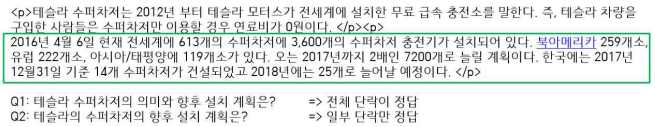


그림2. 세부 단락 추출이 필요한 질문 예시

## 4. 실험 및 결과

본 연구에서 사용된 KorQuAD 2.0 데이터셋은 83,686개의 학습 셋과 10,165개의 개발 셋을 가지고 있다. 학습 셋 데이터중 75,318개를 학습 데이터로, 8,368개를 검증 데이터로 사용했으며 개발 셋 데이터를 평가 데이터로 사용했다.

실험에 사용한 BERT 모델의 하이퍼 파라미터 설정은 BERT-base의 설정과 같게 설정했으며 “히든 레이어: 12, 히든 차원수: 768, 어텐션 헤드수: 12 ” 와 같다.

표1는 KorQuAD 2.0 데이터를 이용하여 단락 순위화 모델의 분류 방법과 학습을 위한 손실 함수에 따른 성능을 비교한 것이다. 평가 방법은 선택된 단락이 데이터셋에서 태깅된 정확한 정답을 포함하고 있는 경우 맞게 선택한 것으로 평가하였다.

표1. 단락 순위화 모델의 성능 비교 (%, dev)

| 모델 | Recall | | | 유형 분류 |
|---|---|---|---|---|
| | Top1 | Top3 | Top5 | |
| 키워드매칭 | 58.5 | 84.7 | 91.8 | - |
| CE | **85.4** | 88.3 | 91.1 | - |
| 키워드매칭+CE | 81.1 | 91.7 | 93.3 | - |
| doubleLoss | 79.8 | 90.5 | 90.9 | - |
| 키워드매칭 + doubleLoss | 82.3 | **94.5** | **97.0** | - |
| 키워드매칭 +3-class 분류 | 79.2 | 93.1 | 94.7 | 92.9 |
| 키워드매칭 +멀티태스크 | 82.3 | 94.3 | 96.9 | **95.8** |

키워드 매칭의 경우 질의에 나타난 명사 형태소 중에 단락에서도 나타나는 비율을 순위화에 반영한 것을 의미한다. CE는 각 단락의 출력에 softmax를 적용하고 cross-entropy를 적용하여 학습한 것을 의미한다. doubleLoss는 각 단락의 출력에 sigmoid를 적용하고 MSE를 적용한 손실함수 값과 cross-entropy를 합하여 학습한 모델을 의미하며 상수인 $\gamma$는 0.2로 설정하였다.

CE를 사용하는 모델의 경우 단일하게 사용되었을 때보다 키워드 매칭의 결과를 함께 곱한 결과의 경우 오히려 일부 성능이 하락하거나 성능 상승의 폭이 doubleLoss를 사용한 모델과 비교하여 크지 않은 것을 확인 할 수 있었다.

3-class 분류는 [Long, Short, No]의 3가지 클래스를 분류하도록 학습한 모델이며, 멀티 태스크의 경우 doubleLoss를 학습하는 분류기와 3-class를 분류하는 분류기를 함께 학습한 모델이다. 3-class 모델의 경우 단락 순위화를 하면서 정답의 유형에 대한 예측까지 동시에 할 수 있지만 doubleLoss를 이용한 모델에 비해서 순위화의 성능이 다소 떨어지는 것을 확인했다. 3-class 모델과 doubleLoss 모델의 학습 방법을 같이 적용한 멀티태스크 모델과 비교하면, Top3의 단락을 추출할 때 3-class의 경우 유형 예측까지 적용하면 86.48%와 멀티태스크 모델의 경우 90.14%로 멀티태스크 모델이 전체 성능에서 3-class 분류 모델보다 3.66% 향상된 것을 확인할 수 있었다.

단답형과 서술형의 정답 유형에 따라서 단락 순위화 모델의 성능과 특징에 차이가 있었다. 표2는 정답의 유형에 따른 단락 순위화 모델의 성능 비교를 나타낸다. 단답형 유형 정답은 개발셋 10,165개의 데이터에서 8,256개이며, 서술형 데이터는 1909개의 데이터를 이용하여 평가했다.

표2. 정답 유형별 단락 순위화 모델의 성능 비교 (%, dev)

| 모델 | Recall | | | |
|---|---|---|---|---|
| | 유형 | Top1 | Top3 | Top5 |
| 키워드매칭 | 서술형 | 35.2 | 71.6 | 84.0 |
| | 단답형 | 63.9 | 87.8 | 93.6 |
| 키워드매칭+ 멀티태스크 | 서술형 | 66.7 | 86.7 | 93.3 |
| | 단답형 | **85.8** | **96.0** | **97.7** |
| 키워드매칭 + doubleLoss | 서술형 | **72.3** | **90.6** | **95.6** |
| | 단답형 | 85.0 | 95.4 | 97.6 |

키워드 매칭만을 이용하는 경우 단답형 유형과 서술형 유형에서의 정확도의 차이가 큰 것을 확인 할 수 있었다. 신경망을 이용한 멀티태스크 학습을 이용한 모델과 doubleLoss를 이용한 모델의 경우 키워드 매칭을 이용하는 경우와 비교해 유형에 따른 성능의 차이가 크진 않지만, 단답형 유형에서의 순위화 정확도가 더 높은 것을 확인했다.

표3의 경우 서술형의 정답을 출력할 때 단락 내에서 정확한 정답 단락의 경계를 예측하는 모델의 성능 비교를 나타낸다.

표3. 세부 단락 정답 경계 예측 정확도 (%, dev)

| 방법 | F1 |
|---|---|
| 전체 단락 출력 | 81.1 |
| duoBERT | 83.0 |
| MRC | 76.6 |

서술형 정답에서 정답 경계 예측에 대한 평가를 위한 입력으로 정답 단락을 입력으로 주었다. MRC의 경우 단순하게 단락 전체를 정답으로 출력하는 것에 비해서 오히려 F1 점수가 하락한 것을 보였으며, duoBERT를 이용한 경우 전체 단락을 출력하는 것에 비해서 1.9%의 성능 향상이 있었다.

## 5. 결론

본 연구에서는 여러 개의 지문이 입력되는 다중 지문 기계독해에서 정답이 있는 단락을 찾기 위한 단락 순위화 모델과 서술형의 긴 정답이 출력되어야 하는 경우 선택된 단락에서 세부적인 정답 경계를 찾는 방법을 제안하였다. 단락 순위화 모델에서 softmax와 cross-entropy를 이용하여 학습한 모델이 sigmoid와 MSE 손실 함수 값을 추가하여 학습한 모델에 비하여 더 높은 성능을 보였지만 모델의 출력 값에 키워드 매칭의 결과를 함께 적용하여 예측하는 경우 두 가지 손실 함수 값을 함께 학습시키는 것이 정확도가 더 높은 것을 보였다. 선택된 단락에서 정답 경계를 찾기 위해 기계독해 모델을 적용한 것 보다 duoBERT를 이용하는 것이 더 높은 정확도를 보였지만 두 방법 모두 전체 단락을 출력하는 것에 비해서 큰 성능 향상을 보이지 않는 것을 보였다.

향후 연구로는 선택된 단락 내에서 정답 경계를 찾기 위한 더 향상된 모델에 관한 연구를 할 계획이다.

## 참고문헌


[1] I. Rajpurkar, P., Zhang, J., Lopyrev, K. and Liang, P., Squad: 100,000+ questions for machine comprehension of text. arXiv preprint arXiv:1606.05250, 2016.

[2] 임승영, 김명지, 이주열, KorQuAD: 기계독해를 위한 한국어 질의응답 데이터셋. 한국정보과학회 학술발표논문집, 539-541, 2018.

[3] 김영민, 임승영, 이현정, 박소윤, 김명지. (2020). KorQuAD 2.0: 웹문서 기계독해를 위한 한국어 질의응답 데이터셋. 정보과학회논문지, 47(6), 577-586.

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

[5] 이동헌, 박천음, 이창기, 임승영, BERT-SRU와 HTML Tag 자질을 이용한 다중 지문 기계 독해 모델. 한국정보과학회 학술발표논문집, 383-385, 2019.

[6] Nogueira, R. and Cho, K., Passage Re-ranking with BERT. arXiv preprint arXiv:1901.04085, 2019.

[7] Dietz, L., Verma, M., Radlinski, F. and Craswell, N., TREC Complex Answer Retrieval Overview. In TREC, 2017.

[8] Bajaj, P., Campos, D., Craswell, N., Deng, L., Gao, J., Liu, X. and Rosenberg, M., Ms marco: A human generated machine reading comprehension dataset. arXiv preprint arXiv:1611.09268, 2016.

[9] Han, S., Wang, X., Bendersky, M. and Najork, M., Learning-to-Rank with BERT in TF-Ranking. arXiv preprint arXiv:2004.08476, 2020.

[10] Clark, K., Luong, M. T., Le, Q. V. and Manning, C. D., Electra: Pre-training text encoders as discriminators rather than generators. arXiv preprint arXiv:2003.10555, 2020.

[11] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D. and Stoyanov, V., Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692, 2019.

[12] Nogueira, R., Yang, W., Cho, K. and Lin, J., Multi-stage document ranking with BERT. arXiv preprint arXiv:1910.14424, 2019.