‘질문-단락’간 주의 집중을 이용한 검색 모델 재순위화 방법

장영진○, 김학수, 지혜성*, 이충희*
강원대학교 컴퓨터정보통신공학과, (주)엔씨소프트*
dan_yon@kangwon.ac.kr, nlpdrkim@kangwon.ac.kr, hyesung84@ncsoft.com*,
forever.7374@ncsoft.com*

Retrieval Model Re-ranking Method using ‘Question-Passage’ Attention

Youngjin Jang○, Harksoo Kim, Hyesung Ji*, Chunghee Lee*
Kangwon National University computer and Communication Engineering
NCSOFT Corp*

요 약

검색 모델은 색인된 문서 내에서 입력과 유사한 문서를 검색하는 시스템이다. 최근에는 기계독해 모델과 통합하여 질문에 대한 답을 검색 모델의 결과에서 찾는 연구가 진행되고 있다. 위의 통합 모델이 좋은 결과를 내기 위해서는 검색 모델의 높은 성능이 요구된다. 따라서 본 논문에서는 검색 모델의 성능을 보완해 줄 수 있는 재순위화 모델을 제안한다. 검색 모델의 결과 후보를 일괄적으로 입력받고 ‘질문-단락’간 주의 집중을 계산하여 재순위화 한다. 실험 결과 P@1 기준으로 기존 검색 모델 성능대비 5.58%의 성능 향상을 보였다.

주제어: 기계독해 모델, 검색 모델, 재순위화, 질문-단락 간 주의 집중

1. 서론

기계독해 모델(Machine Reading Comprehension: MRC)은 주어진 문서에서 질문에 대한 정답을 찾는 시스템이다. 최근에는 주어진 문서가 아닌 검색 모델을 통해 질문과 관련된 문서(단락)를 검색하고, 검색 결과 안에서 질문에 대한 정답을 찾는 연구[1-2]가 진행되고 있다. 하지만 위의 통합 모델은 검색 모델의 결과로부터 오류가 전파 될 확률이 매우 높기 때문에 후처리 등과 같은 검색 모델의 성능 보완이 필요하다. 따라서 본 논문에서는 ‘단락-질문’ 간의 주의 집중을 이용한 재순위화 방법을 제안한다. 제안 모델은 심층 학습(Deep Learning)을 통해 기존 검색 모델에서 반영하지 못한 의미적 정보를 반영하여 재순위화 한다. BiDAF(Bi-Directional Attention Flow)[3] 모델을 기반으로 구현되었으며, 포인터 네트워크[4]의 포인팅 확률 분포를 이용하여 후보 단락을 재순위화 한다.

2. 관련 연구

기존의 기계 학습을 통한 재순위화 연구는 다양하게 시도되었다. 대표적으로 쌍 별 학습(Pairwise learning)[5-7]은 문서 d_i와 d_j를 입력받아 어떤 문서가 입력 문장과 관련이 깊은지 학습한다. 하지만 쌍 별 학습은 입력 문서 수의 제한이 있어서 재순위화 해야 할 후보가 많을 경우 연산 횟수가 급격히 늘어난다는 단점이 있다. 그리고 최근 다양한 자연어 처리 분야에서 높은 성능을 보이는 BERT(Bi-directional Encoder Representations from Transformers)[8]를 이용한 단락 재순위화 연구 또한 입력 문장과 후보 단락 쌍을 입력받기 때문에 한 번에 처리할 수 있는 단락의 개수가 제한된다. 본 논문에서는 이 문제점을 해결하기 위해 검색 모델 결과인 후보 단락 전체와 질문을 같이 입력받아 한 번에 재순위화 한다.

BiDAF는 BERT가 등장하기 이전, 대부분의 기계독해 시스템의 기반[9]이 되었던 모델로 질문의 어떤 정보가 단락과 관계가 깊은지 그리고 단락의 어떤 정보가 질문과 관련이 깊은지를 양방향 주의 집중 계층을 통해 계산한다. 제안 모델은 BiDAF의 ‘질문-문맥’ 간 주의 집중 계산에서 착안하여 벡터화된 각 단락과 질문 사이의 주의 집중(Question-Passage Attention: Q-P Attention)을 scaled dot product attention을 통해 계산한다. 이 주의 집중 방법은 [10]에서 제안된 것으로, BiDAF에서 사용된 dot product attention과 유사하지만 은닉 벡터 크기 d_k의 제곱근으로 나누어 계산한다. 본 논문에서는 순환 신경망[11]을 통해 모든 후보 단락과 질문을 각각 인코딩 하고 ‘질문-단락’ 간 주의 집중 계산을 통해 질문과 가장 관련이 있는 단락을 포인터 네트워크를 통해 가리키는 재순위화 방법을 제안한다.

3. 검색 모델 재순위화 모델

그림 1은 본 논문에서 제안하는 검색 모델 재순위화

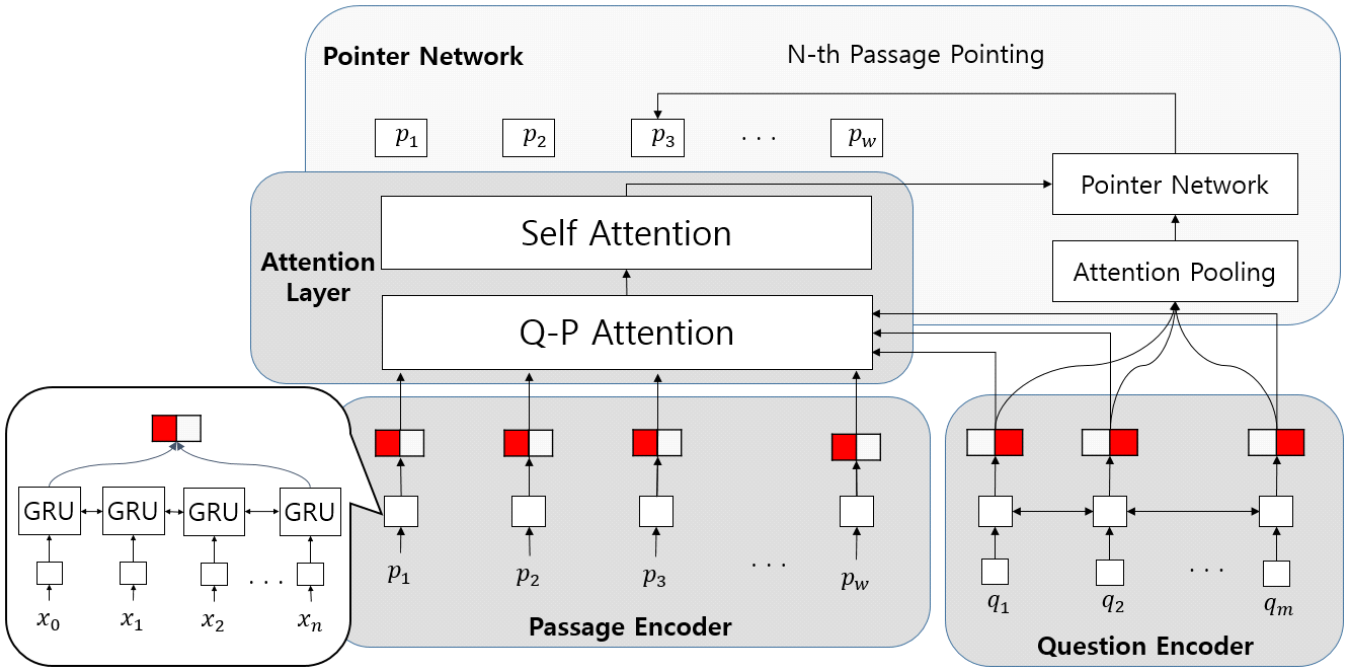

그림 1. 제안 모델 구조도

모델의 구조를 보여준다. 제안 모델은 단락 인코더, 질문 인코더, 주의 집중 계층, 그리고 포인터 네트워크로 구성되어있다. 단락 인코더는 양방향 순환 신경망[12]으로 구성되어 있으며, 순방향, 역방향 순환 신경망의 마지막 상태(State) 값을 연결(Concatenate)하여 각 단락을 하나의 벡터로 표현한다. 질문 인코더 또한 양방향 순환 신경망으로 이루어져 있으며, 단락 인코더와 가중치를 공유한다. 각각의 인코더를 통해 인코딩된 단락 벡터와 질문 벡터는 '질문-단락' 주의 집중을 통해 어떤 단락이 질문과 관련이 있는지 계산하고, 그 정보를 자가 주의 집중(Self Attention)을 통해 강조한다. 주의 집중 계층을 통해 계산된 벡터와 주의 집중 풀링(Attention Pooling)을 통해 계산된 질문 벡터는 각각 포인터 네트워크의 입력 값과 초기 값으로 넣어준다. 포인터 네트워크를 통해 생성된 벡터 값은 softmax 함수를 통해 입력받은 모든 후보 단락에 대한 확률 분포를 출력한다.

3.1. 주의 집중 계층

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

본 논문의 주의 집중 계층에서 사용한 주의 집중 수식은 (1)과 같다. d_k는 은닉 벡터의 크기를 나타내며, V는 K와 동일한 벡터를 가진다. '질문-단락' 주의 집중 단계에서 Q와 K는 각각 단락 벡터, 질문 벡터를 나타내고 자가 주의 집중 단계에서 Q와 K는 모두 '질문-단락' 주의 집중 값을 나타낸다. '질문-단락' 주의 집중을 통해 어떤 단락이 질문과 관련이 있는지 계산하고, 자가 주의 집중을 통해 정보를 더 강조한다. 생성된 벡터는 포인터 네트워크의 입력 값으로 사용된다.

3.2. 주의 집중 풀링과 포인터 네트워크

$$W_q = FNN(Output_{GRU}(Q),1)$$
$$Attention\,Pooling(Q) = \sum(W_q \times Output_{GRU}(Q)) \quad (2)$$

본 논문에서 사용한 주의 집중 풀링의 수식은 (2)와 같다. $Output_{GRU}(Q)$는 질문 인코더를 통해 생성된 [질문 최대 길이, 은닉 벡터 크기]의 행렬이며 W_q는 $Output_{GRU}(Q)$를 전방 전달 신경망을 통해 [질문 최대 길이, 1]의 크기로 계산된 가중치 행렬이다. W_q와 $Output_{GRU}(Q)$는 수식 (2)의 가중 합 연산을 통해 [은닉 벡터 크기]의 벡터를 생성한다. 주의 집중 풀링 계층을 통해 생성된 벡터는 포인터 네트워크의 초기 값으로 사용된다. 포인터 네트워크는 주의 집중 계층을 통해 생성된 벡터와 주의 집중 풀링을 통해 생성된 벡터를 입력받고, 질문과 관련이 깊은 단락을 가리킨다. 이때, 포인터 네트워크에서 생성된 [입력 단락 크기]의 확률 분포 값은 '질문-각 단락' 의 유사도 점수로 간주한다. 본 논문에서는 위의 유사도 점수를 통해 입력 받은 단락을 재순위화 한다.

4. 실험

4.1. 실험 준비

본 논문에서 실험한 데이터는 나무위키, 위키피디아 1,361개 문서를 기반으로 수집한 기계독해 데이터 58,980쌍으로 이루어져있다. 모델에 입력되는 후보 단락

은 BM25[13]를 통해 질문에 대한 상위 10개, 20개, 30개의 단락을 검색하여 구축했다. 학습 데이터와 실험 데이터는 무작위로 9:1 비율로 나누어 52,885 쌍, 6,095 쌍으로 사용했다. 실험에 사용한 단어 임베딩은 뉴스 기사 20GB로 사전 학습한 GloVe[14]를 사용했다. 제안 모델의 파라미터는 표 1과 같다.

표 1. 모델 파라미터

배치 크기	64
학습 횟수	300
은닉 크기	100
단어 임베딩 크기	100
질문 최대 길이	128
단락 최대 길이	512
후보 단락 수	10, 20, 30
학습률	0.004
드랍아웃	0.3

4.2. 실험 평가

성능 평가는 모델이 재순위화 한 단락 후보 상위 n개 안에 정답 단락이 있는지 확인하는 방법으로 진행했다. 실험 결과는 표 2와 같다.

표 2. 실험 결과

	[13]	window 10	window 20	window 30
P@1	0.6872	0.7205	**0.7430**	0.7033
P@2	0.7717	0.7548	**0.7871**	0.7472
P@3	0.8052	0.7957	**0.8149**	0.7785
P@4	0.8263	0.8168	**0.8351**	0.8077
P@5	0.8439	0.8279	**0.8488**	0.8266
P@6	0.8555	0.8454	**0.8588**	0.8442
P@7	0.8639	0.8544	**0.8651**	0.8546
P@8	**0.8718**	0.8635	0.8704	0.8637
P@9	0.8781	0.8711	**0.8797**	0.8714
P@10	**0.8829**	**0.8829**	0.8827	0.8776

위의 표 2에서 [13]은 기본 검색 모델의 성능을 의미하며, window N은 제안 모델에 N개의 단락을 입력한 실험 결과를 의미한다. P@N은 각 비교 모델 결과의 상위 N개의 단락 안에 정답 단락이 존재하는 지에 대한 성능을 의미한다. 표 2에 따르면 제안 모델의 성능이 기본 검색 모델 성능과 비교하여 높은 성능을 보이는 것을 확인할 수 있다(P@1 기준). 그리고 window 10과 window 30의 실험은 P@1을 제외한 모든 성능이 기본 검색 모델과 비교하여 성능이 떨어지는 반면, window 20의 성능은 기본 검색 모델 성능과 비교하여 대부분의 성능이 높게 나오는 것을 확인할 수 있다. 따라서 제안 모델은 입력받는 단락의 수가 적거나 많을 경우에는 오히려 성능이 하락하는 것을 확인할 수 있다.

5. 결론

본 논문에서는 기계독해 모델과 검색 모델의 통합 성능을 보완하기 위해 검색 모델의 성능을 올려줄 수 있는 단락 후보 재순위화 모델을 제안했다. 쌍 별 학습을 기반으로 한 재순위화 모델의 일괄처리 불가 문제를 보완할 수 있었다. 4절의 성능 표에 따르면 후보 단락의 개수가 20개 일 때, 기존 검색 모델의 성능을 가장 잘 보완해주는 것을 확인할 수 있었다. 추후 실험으로는 단락의 인코딩 방법을 단순히 순방향, 역방향 순환 신경망의 상태 값을 연결하여 사용하는 것이 아닌 트랜스포머나 주의 집중 풀링 방식을 이용하여 인코딩하는 방법으로 실험할 예정이다.

감사의 글

본 연구는 엔씨소프트 산학연구용역 과제의 지원을 받아 수행되었음.

참고문헌

[1] 김시형, 김진태, 김학수, 최맹식, "질의 확장 및 재순위화를 이용한 기계독해용 검색 모델", *2018 한국어소프트웨어종합학술대회 논문집*, pp.620-622, 2018.

[2] M.Hardalov, I.Koychev, P.Nakov, "Machine Reading Comprehension for Answer Re-Ranking in Customer Support Chatbots", *arXiv preprint arXiv:1902.04574v1*, 2019.

[3] M. Seo, A. Kembhavi, A. Farhadi and H.Hajishirzi, "Bidirectional attention flow formachine comprehension.", *arXiv preprint arXiv:1611.01603*, 2016.

[4] O. Vinyals, M. Fortunato and N. Jaitly, "PointerNetworks", *Neural Information Processing Systems(NIPS)*, pp.2692-2700, 2015

[5] T. Joachims, "Optimizing search engines using clickthrough data," I*n Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery anddata mining*, pp. 133-142, 2002

[6] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds,N. Hamilton, and G. Hullender, "Learning to rank using gradient descent," *In Proceedings of the 22nd internationalconference on Machine learning* ,pp. 89-96, 2005.

[7] B. Song, "Deep Neural Network for Learning to RankQuery-Text Pairs," *arXiv preprint arXiv:1802.08988*, 2018

[8] J. Devlin, M. Chang, K. Lee, K. Toutanova, “BERT:Pre-training of Deep Bidirectional Transformers for Language Understanding”, *arXiv prepring arXXiv:1810.04805v2*, 2019.

[9] 이현구, 김학수, 이연수, “GF-Net: 자질 선별을 통한 고성능 기계독해”, *2018 한국컴퓨터종합학술대회 논문집*, pp.598-600, 2018.

[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit,L. Jones, A.N. Gomez, L. Kaiser and

I.Polosukhin, "Attention Is All You Need", *NeuralInformation Processing Systems (NIPS)*,pp.5998-6008, 2017

[11] J. Chung, C. Gulcehre, K. Cho and Y. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, *arXiv preprint arXiv:1412.3555*, 2014

[12] M. Schuster and K. K. Paliwal, Bidirectional recurrent neural networks, *IEEE Transactions on Signal Processing*, pp. 2673-2681, 1997

[13] S. E. Robertson, S. Walker, S. Jones, M. M. Beaulieu, and M. Gatford, “Okapi at TREC-3”. *Proceedings of TREC-3* ,pp. 109-126). 1994.

[14] J. Pennington, R. Socher and C. D. Manning."GloVe: Global Vectors for WordRepresentation", *2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*,pp.1532-1543, 2014