표 질의응답을 위한 언어 모델 학습 및 데이터 구축

심묘섭O, 전창욱, 최주영, 김현, 장한솔, 민경구

LG AI Research
{myoseop.sim, cwjun, jooyoung.choi, hyun101.kim, hansol.jang, kyungkoo.min}@lgresearch.ai

Pre-trained Language Model for Table Question and Answering

Myoseop SimO, Changwook Jun, Jooyoung Choi, Hyun Kim, Hansol Jang, Kyungkoo Min
LG AI Research

요 약

기계독해(MRC)는 인공지능 알고리즘이 문서를 이해하고 질문에 대한 정답을 찾는 기술이다. MRC는 사전학습 모델을 사용하여 높은 성능을 내고 있고, 일반 텍스트문서 뿐만 아니라 문서 내의 테이블(표)에서도 정답을 찾고자 하는 연구에 활발히 적용되고 있다. 본 연구에서는 기존의 사전학습 모델을 테이블 데이터에 활용하여 질의응답을 할 수 있는 방법을 제안한다. 더불어 테이블 데이터를 효율적으로 학습하기 위한 데이터 구성 방법을 소개한다. 사전학습 모델은 BERT[1]를 사용하여 테이블 정보를 인코딩하고 Masked Entity Recovery(MER) 방식을 사용한다. 테이블 질의응답 모델 학습을 위해 한국어 위키 문서에서 표와 연관 텍스트를 추출하여 사전학습을 진행하였고, 미세 조정은 샘플링한 테이블에 대한 질문-답변 데이터 약 7만건을 구성하여 진행하였다. 결과로 KorQuAD2.0 데이터셋의 테이블 관련 질문 데이터에서 EM 69.07, F1 78.34로 기존 연구보다 우수한 성능을 보였다.

주제어: 테이블, 표 질의응답, 기계독해, 언어 모델

1. 서론

문서에 있는 정보 중 테이블 형식의 정보를 많이 찾아볼 수 있다. 지역별 투표율 또는 나라별 인구수와 같은 통계자료나 스포츠경기 결과 등을 기록할 때 테이블 형식이 유용하기 때문이다. 이렇게 테이블에 포함된 정답을 찾기 위해 테이블 MRC가 연구되고 있다.

테이블 MRC 연구분야의 이전 접근 방식은 의미론적 구문 분석을 통해 자연어 질문을 데이터베이스에서 사용할 수 있는 SQL 쿼리로 바꾸는 것이다[2]. 이는 테이블 정보의 연산이나 속도 면에서 이점이 있지만 웹 문서에 있는 테이블을 데이터베이스화 하는 작업과 의미적으로 유효한 쿼리를 생성하기 위한 많은 엔지니어링이 수반되어야 하며, 일반화가 어렵다.

테이블 질의응답의 접근 방식 중 하나는 BERT 아키텍처를 활용하여 질문과 테이블 형태의 데이터 구조를 함께 학습하는 방식이다[3]. 이는 위키 문서에서 표와 관련 문장의 관계를 사전학습하기 위해 입력 구조에 맞게 테이블을 변경하는 작업이 선행되어야 한다.

이와 같이 테이블의 형태를 데이터베이스 스키마와 동일하게 유지하거나, 테이블의 위치 정보를 임베딩 값으로 유지하거나, 또는 테이블의 행과 열 정보를 메타정보로 추가하여 유지하려는 이유는 테이블 데이터를 텍스트와 같이 평면적 데이터로 변환할 경우 테이블 구조를 잃어버리기 때문이다.

본 연구에서는 BERT의 임베딩 구조 변경 없이 테이블 데이터의 포맷 변경을 통해 테이블 구조를 반영하여 학습하는 방법을 소개한다.

사전학습 데이터는 위키 문서에서 추출한 위키 테이블과 인포박스 144만 건을 사용하여 텍스트와 테이블의 관계를 학습했으며, 미세 조정에는 위키 문서에서 추출한 테이블 1만 7천 건에 대한 질문-답변 데이터셋 약 7만 건을 생성하여 사용했다.

테이블 구조를 유지하며 학습된 모델은 테이블 데이터 포맷의 형태에 따라 성능 차이를 보였다. 특히 테이블의 특성을 더욱 잘 표현해주는 포맷을 사용했을 때는 모델의 구조 변경이 없더라도 TAPAS 모델을 한국어에 적용한 기존 연구[4]보다 우수한 성능을 보이는 것을 확인했다.

성능 평가는 KorQuAD2.0 에서 테이블 관련 문제만 추출한 개발 데이터셋 약 1천 건으로 진행했으며, EM 69.07, F1 78.34 의 성능을 보였다.

2. 관련 연구

테이블을 이해하기 위한 연구 방향은 크게 세 가지로 나눌 수 있다.

첫째, 검색을 활용하여 테이블 질의응답을 해결하는 연구다. [2], [5]에서는 여러 번의 반복 검색을 통해 많은 테이블 중에서 정답이 있는 테이블을 검색하는 방식으로 문제를 풀고 있다. 특히 [5]의 연구에서는 텍스트와 연관된 테이블을 연결하여 여러 개의 Fused Block들로 나누고 이 블럭들을 다중 홉으로 검색하여 답을 찾기 때문에 텍스트와 테이블이 함께 있는 문서에서도 사용할 수 있는 장점이 있다.

둘째, 자연어 질의 분석을 통해 데이터베이스에서 사용할 수 있는 SQL 쿼리로 바꾸는 연구다. [6], [7]은 BERT를 사용해서 자연어 질의를 SQL로 바꾸는 방법을 소개한다. 특히 [6]에서는 BERT의 출력을 이용해 SQL문을 구성하는 요소를 맞추는 6개의 하위 문제로 재구성해서 문제를 해결한다. 이런 접근법은 테이블 데이터가 데이

터베이스로 구성되어 있어야 하는 전제 조건이 필요하다.

마지막으로 BERT의 입력 구조를 테이블에 맞게 변경시키고, 사전학습을 통해 테이블 정보를 표현하는 모델을 만드는 연구다[3]. [3]에서는 기존 BERT 입력에 랭킹 정보, 행 정보, 열 정보를 추가하여 표의 특성을 이해하도록 학습하는 방법이다. 유사한 방법으로 [8]에서는 TAPAS와는 다르게 캡션, 헤더, 서브섹트, 오브젝트를 구분하는 타입 임베딩을 추가하여 학습하는 방법을 제안했다. 또한 [9],[10]은 BERT를 활용해 텍스트와 테이블을 학습시키는데, 이때 테이블 정보를 모두 사용하는 것이 아니라 텍스트와 가장 연관이 있는 k개의 행을 뽑아 텍스트와 함께 학습시키는 방법을 제안한다. 이 방법은 문서의 테이블 데이터를 데이터베이스로 구조화 하지 않아도 테이블의 연산이 가능하다.

본 연구에서는 BERT를 활용하여 테이블의 특성을 유지하면서도 효과적으로 입력 데이터의 포맷을 변경하여 학습시키는 방법을 제안한다.

3. 데이터 구성 및 학습

위키에서 추출한 140만 개의 텍스트 테이블 쌍을 사용해서 BERT가 텍스트를 학습하는 MLM(masked language modeling)으로 사전학습을 했다. 사용한 모델은 BERT Base 모델이며, Mask를 씌우는 비율은 15%, 이 외 세부 파라메터는 기존 BERT와 동일하게 적용하였다.

테이블 데이터를 학습하기 위해 아래와 같은 방법을 사용했다.

첫째, 데이터를 모델의 형식에 맞게 변경하여 사전학습을 진행한다. 둘째, 부족한 텍스트 데이터를 보완하고 문맥 정보를 학습하기 위해 위키 데이터셋으로 추가 학습을 한다. 마지막으로 테이블 관련 질문-답변 데이터셋을 구성하여 미세 조정한다.

3.1 사전학습 데이터 구성

위키 문서에서 일반적인 위키 테이블 그림1과 인포박스 형태의 테이블 그림2을 추출하여 약 144만건을 구성하였다.

<위키 문서내 위키테이블 예제>

그림 1. 위키 테이블

<위키 문서내 인포박스 테이블 예제>

그림 2. 인포박스 테이블

데이터는 BERT의 입력 구성과 동일하게 '[SEP]' 토큰으로 구분되는 두 개의 문장으로 구성하는데, 앞의 문장에는 테이블 연관 텍스트를 넣고, 뒤의 문장에는 테이블 정보를 넣는다.

그림1의 테이블 데이터의 경우, 테이블 연관 텍스트는 문서의 타이틀과 섹션 타이틀, 그리고 캡션을 연결해 그림3과 같이 하나의 문장으로 만들었다. 테이블의 첫번째 열을 헤더로 사용했고, 행과 열이 바뀌어 있는 테이블의 경우는 HTML 태그 정보를 사용해서 변환 후 첫번째 열을 헤더로 사용했다.

그림 3. 테이블 연관 텍스트

그림2의 인포박스 데이터의 경우, 가장 먼저 등장하는 텍스트를 연관 텍스트로 선정하고, 인포박스를 테이블로 만들었다. 인포박스를 테이블 형식으로 구성하기 위해 그림4와 같이 첫 번째 열을 헤더로 변환했다.

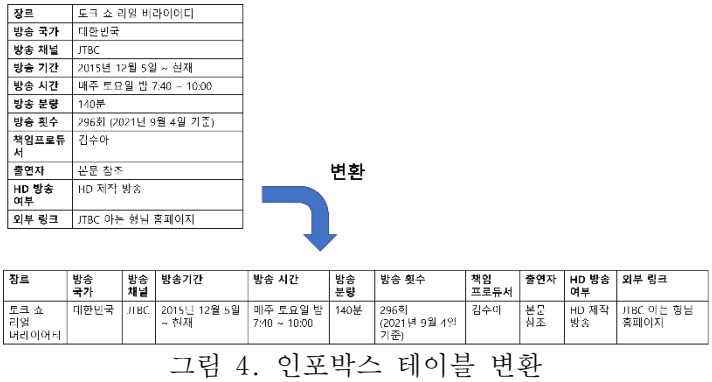

그림 4. 인포박스 테이블 변환

레벨	규 칙	예 시
레벨 1	**기준열**을 **조건**으로 **기준열 이외의 열**을 질문	UEFA 유로 2004 A조에서 포르투갈 팀의 승점은 몇점인가?
레벨 2	**기준열 이외의 열**을 **조건**으로 다른 열을 질문	UEFA 유로 2004 A조에서 승점이 3점인 팀의 골득실은 얼마인가?
레벨 3	**기준열 이외의 열**을 **조건**으로 **기준열**을 질문	UEFA 유로 2004 A조에서 승점이 6점인 팀은 어디인가?
레벨 4	**문장 변형** 적용한 질문	UEFA 유로 2004 A조에서 포르투갈이 획득한 점수를 알려주세요
레벨 5	헤더 중 순위, 시간, 날짜 등이 있을 때 최소값 또는 최근 등의 **조건으로 질문**	UEFA 유로 2004 A조에서 승점이 가장 낮은 팀은 어디인가?

그림 5. 미세조정 데이터셋 규칙 및 예시

3.2 미세조정 데이터 구성

미세조정을 위한 테이블 관련 질문-답변 데이터셋은 두 가지로 구성했다.

첫째, KorQuAD 2.0의 테이블 관련 질문-답변 데이터셋 중 단답형의 정답이 표에 존재하는 데이터셋 약 1만개를 추출하고, 8:1:1 의 비율로 학습 데이터, 검증 데이터, 평가 데이터로 사용했다. 이 데이터는 이전 연구[4]와 성능 비교를 위해 사용했다.

둘째, 위키 테이블 데이터 중 약 1만 7천건의 테이블에 대해 질문-답변 데이터셋을 그림5의 규칙[11]을 적용하여 약 7만건의 데이터셋을 만들었다. 이는 양질의 데이터셋 추가에 따른 모델의 성능 향상을 확인하기 위해 사용했다.

4. 실험 및 결과

테이블 기계독해 실험은 테이블 포맷에 따른 성능 비교 실험과 미세조정 데이터셋 추가에 따른 성능 비교 실험 두 가지를 진행했다. 첫번째 실험은 사전학습 데이터와 KorQuAD 2.0에서 추출한 미세조정 데이터셋에 두 가지 테이블 포맷을 적용하여 성능을 비교했고, 두번째 실험은 양질의 미세조정 데이터셋 추가에 따른 성능 비교 실험이다.

4.1 테이블 포맷 비교 실험

테이블 데이터를 텍스트 데이터로 평면화 할 때, 테이블의 구조와 특성을 유지하는 것이 중요하다.

그림6은 테이블 형태를 그대로 유지하며 테이블 데이터를 텍스트로 변환한 방법이다. 이를 단순 테이블 포맷으로 정한다.

단순 테이블 포맷은 테이블의 헤더와 테이블 값을 매핑하여, 변환된 텍스트가 테이블과 유사한 구조로 되어 있고, 테이블 한 행을 텍스트로 변경했을 때 길이가 상대적으로 짧은 장점이 있다.

그림7은 테이블 데이터의 특성을 반영하는 방법이다. 이를 연관 테이블 포맷으로 정한다.

연관 테이블 포맷은 테이블 헤더에 첫 번째 기준 열 정보를 추가한 후 테이블 값을 매핑하여, 테이블에서 중요한 정보를 반영하는 장점이 있다. 하지만, 헤더의 개수가 많아질수록 헤더와 기준 열이 반복적으로 들어가기 때문에 변환되는 텍스트 데이터의 길이가 길어지는 단점도 있다.

{방송 채널|SBS}{프로그램|일일드라마 달려라 장미}{방송 시간|평일 저녁 7:20 ~ 밤 8:00 (40분)}
{방송 채널|SBS}{프로그램|일일드라마 돌아온 황금복}{방송 시간|평일 저녁 7:20 ~ 밤 8:00 (40분)}
{방송 채널|KBS 1TV}{프로그램|그대가 꽃}{방송 시간|월요일 저녁 7:30 ~ 밤 8:25 (55분)}
{방송 채널|KBS 1TV}{프로그램|이웃집 찰스}{방송 시간|화요일 저녁 7:30 ~ 밤 8:25 (55분)}
{방송 채널|KBS 1TV}{프로그램|엄마의 탄생}{방송 시간|수요일 저녁 7:30 ~ 밤 8:25 (55분)}
{방송 채널|KBS 1TV}{프로그램|한국인의 밥상}{방송 시간|목요일 저녁 7:30 ~ 밤 8:25 (55분)}
{방송 채널|KBS 1TV}{프로그램|똑똑한 소비자 리포트}{방송 시간|금요일 저녁 7:30 ~ 밤 8:25 (55분)}
{방송 채널|KBS 2TV}{프로그램|2TV 저녁 생생정보}{방송 시간|월 ~ 목 저녁 6:30 ~ 7:50 (80분)}
{방송 채널|KBS 2TV}{프로그램|뮤직 뱅크}{방송 시간|금요일 저녁 6:30 ~ 7:50 (80분)}
{방송 채널|KBS 2TV}{프로그램|일일드라마 달콤한 비밀}{방송 시간|평일 저녁 7:50 ~ 밤 8:30 (40분)}
{방송 채널|KBS 2TV}{프로그램|일일드라마 오늘부터 사랑해}{방송 시간|평일 저녁 7:50 ~ 밤 8:30 (40분)}

그림 6. 단순 테이블 포맷

두 포맷에서 변환된 텍스트의 구분자 기호 '{}' 은 하나의 셀을 나타내고 구분자 기호 '|' 은 헤더정보와 값을 구분하는 역할을 한다.

테이블 포맷에 따른 성능 차이는 표1과 같이 연관 테

이블 포맷이 EM 점수는 4.5점, F1 점수는 3.8점 높은 점수를 보였다.

{방송 채널 SBS 프로그램|일일드라마 달콤한 장미}{방송 채널 SBS 방송 시간|평일 저녁 7:20 ~ 밤 8:00 (40분)}
{방송 채널 SBS 프로그램|일일드라마 돌아온 황금복}{방송 채널 SBS 방송 시간|평일 저녁 7:20 ~ 밤 8:00 (40분)}
{방송 채널 KBS 1TV 프로그램|그대가 꽃}{방송 채널 KBS 1TV 방송 시간|월요일 저녁 7:30 ~ 밤 8:25 (55분)}
{방송 채널 KBS 1TV 프로그램|이웃집 찰스}{방송 채널 KBS 1TV 방송 시간|화요일 저녁 7:30 ~ 밤 8:25 (55분)}
{방송 채널 KBS 1TV 프로그램|엄마의 탄생}{방송 채널 KBS 1TV 방송 시간|수요일 저녁 7:30 ~ 밤 8:25 (55분)}
{방송 채널 KBS 1TV 프로그램|한국인의 밥상}{방송 채널 KBS 1TV 방송 시간|목요일 저녁 7:30 ~ 밤 8:25 (55분)}
{방송 채널 KBS 1TV 프로그램|똑똑한 소비자 리포트}{방송 채널 KBS 1TV 방송 시간|금요일 저녁 7:30 ~ 밤 8:25 (55분)}
{방송 채널 KBS 2TV 프로그램|2TV 저녁 생생정보}{방송 채널 KBS 2TV 방송 시간|월 ~ 목 저녁 6:30 ~ 7:50 (80분)}
{방송 채널 KBS 2TV 프로그램|뮤직 뱅크}{방송 채널 KBS 2TV 방송 시간|금요일 저녁 6:30 ~ 7:50 (80분)}
{방송 채널 KBS 2TV 프로그램|일일드라마 달콤한 비밀}{방송 채널 KBS 2TV 방송 시간|평일 저녁 7:50 ~ 밤 8:30 (40분)}
{방송 채널 KBS 2TV 프로그램|일일드라마 오늘부터 사랑해}{방송 채널 KBS 2TV 방송 시간|평일 저녁 7:50 ~ 밤 8:30 (40분)}

그림 7. 연관 테이블 포맷

이와 같은 결과가 나온 이유는 기준 열에 중요한 정보를 갖는 테이블의 특성 때문이다. 그림6을 예로 들면, 단순 테이블 포맷의 경우, 중요한 정보인 'SBS'는 '방송 시간' 헤더와 텍스트상에서 거리가 멀다. 이는 헤더 개수가 많아질수록 더욱 멀어지기 때문에 두 값에 대한 연관성을 모델이 학습할 수 없다. 따라서 연관 테이블 포맷이 단순 테이블 포맷보다 효과적으로 테이블의 특성을 반영하는 것을 확인할 수 있었다.

또한, 연관 테이블 포맷으로 사전학습 후 KorQuad2.0의 테이블 데이터로 미세조정한 모델의 성능은 기존 연구[4]의 EM 63.6점과 F1 76.0점 보다 높은 성능을 내는 것을 확인하였으며, 본 연구에서 제안하는 테이블 데이터 포맷 변경을 통한 테이블 MRC 모델이 효과적인 방식임을 확인할 수 있었다.

표 1. 테이블 포맷에 따른 모델 성능 비교

모델		EM	F1
TAPAS를 이용한 사전학습 언어 모델[4]	MRC	46.8	63.8
	col&row selection	60.2	74.3
	MRC + col&row embedding	62.8	75.2
	MRC + NE + col&row embedding	63.6	76.0
Our	단순 테이블 포맷	64.53	74.52
	연관 테이블 포맷	**69.07**	**78.34**

4.2 미세조정 데이터셋 추가에 따른 성능 비교 실험

3.2절에서 소개한 방식으로 생성한 데이터를 모델에 적용하였을 때 테이블 MRC의 성능을 비교 실험했다. 미세조정 데이터셋에도 단순 테이블 포맷과 연관 테이블 포맷을 적용했을 때, 표2와 같이 연관 테이블 포맷이 EM 87.15, F1 91.15로 단순 테이블 포맷의 EM 84.12, F1 88.50 에 비해 높은 성능을 보였다. 또한, 약 7만건의 테이블 데이터 적용 시 KorQuAD 2.0 데이터셋으로 미세조정한 모델보다 EM 성능이 약 20점 향상되었다.

이 실험을 통해 3.2절에서 소개한 다섯 레벨 질문-답변 데이터셋의 구성 방식이 효과적임을 확인할 수 있었다.

하지만, 표3의 질문 레벨 별 EM 결과를 통해 테이블 값의 연산 또는 정렬이 필요한 레벨 5 질문에 대해서는 보완이 필요함을 알 수 있었다.

표 2. 미세조정 데이터 추가에 따른 모델 성능 비교

모델	EM	F1
단순 테이블 포맷	64.53	74.52
연관 테이블 포맷	69.07	78.34
단순 테이블 포맷 + 미세조정 데이터 추가	84.12	88.50
연관 테이블 포맷 + 미세조정 데이터 추가	**87.15**	**91.15**

표 3. 테이블 질문 레벨별 성능

질문 레벨	EM
레벨 1	89.6
레벨 2	89.1
레벨 3	86.1
레벨 4	81.7
레벨 5	67.8

5. 결론

본 연구에서는 테이블 MRC 모델에서 테이블의 특성을 반영하기 위한 테이블 데이터 포맷 변경 방법을 제안하였다. 테이블 사전 학습에서는 테이블의 기준 열과 헤더와의 관계를 명시해줄 때 더욱 우수한 성능을 보였다. 또한 성능 향상을 위한 테이블 미세 조정 데이터의 구성 방식을 제안하였다. 하나의 테이블에 대해 다섯 가지 레벨의 다양한 질문이 성능 향상에 효과적임을 알 수 있었다.

향후에는 테이블 데이터의 연산 또는 정렬에 관한 질문을 해결할 수 있는 연구를 할 계획이다.

참고문헌

[1] J Devlin, MW Chang, K Lee and K Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding ”, arXiv preprint arXiv:1810.04805, 2018.

[2] Dasigi, Pradeep, et al . "Iterative search for weakly supervised semantic parsing." Proceedings of the 2019 Conference of the North American Chapter of the Association f or Computational

Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). 2019.
[3] Herzig, Jonathan, et al. "TaPas: Weakly supervised table parsing via pre-training." arXiv preprint arXiv:2004.02349, 2020.
[4] 조상현, 김민호, 권혁철, "TAPAS를 이용한 사전학습 언어 모델 기반의 표 질의응답, 제32회 한글 및 한국어 정보처리 학술대회 논문집, 2020.
[5] Chen, Wenhu, et al. "Open question answering over tables and text." arXiv preprint arXiv:2010.10439, 2020.
[6] Hwang, Wonseok , et al . " A comprehensive exploration on wikisql with table -aware word contextualization." arXiv preprint arXiv:1902.01069, 2019.
[7] Lyu, Qin, et al. "Hybrid ranking network for text-to-sql." arXiv preprint arXiv:2008.04759, 2020.
[8] Deng, Xiang , et al . "Turl: Table understanding through representation learning." arXiv preprint arXiv:2006.14806, 2020.
[9] Yin, Pengcheng, et al . "TaBERT: Pretraining for joint understanding of textual and tabular data." arXiv preprint arXiv:2005.08314, 2020.
[10] Chen, Wenh u, et al . "Tabfact: A large -scale dataset for table-based fact verification." arXiv preprint arXiv:1909.02164, 2019.
[11] 박소윤, et al . "TabQA: 표 양식의 데이터에 대한 질의응답 모델." HCLT 2018, pp.263-269, 2018.