TaPas : Weakly Supervised Table Parsing

https://arxiv.org/pdf/2004.02349

0. Introduction

- 자연어 질문에 대해 표 위에서 답변하는 작업은 보통 시맨틱 파싱 문제로 다뤄져 왔음.
- 논리 표현을 모두 수집하고 라벨링하는 것은 비용이 크며, 약지도 방식에서는 출력값만 주어짐.
- 표 + 자연어 환경에서 중간 논리 표현 없이 직접 결과값을 예측하는 방법을 마련하고자 함.
- 핵심 기여 및 차별성 :
    - TAPAS 모델은 논리표현 없이 표의 셀을 선택하고, 필요시 집계연산을 적용하여 답변을 산출함.
    - 표와 텍스트를 함께 인코딩할 수 있도록 BERT 기반 구조를 확장하고, 사전학습을 수행함.
    - 세 가지 시맨틱 파싱 데이터셋에서 실험을 진행하여 강력한 성능을 입증함.

1. Overview

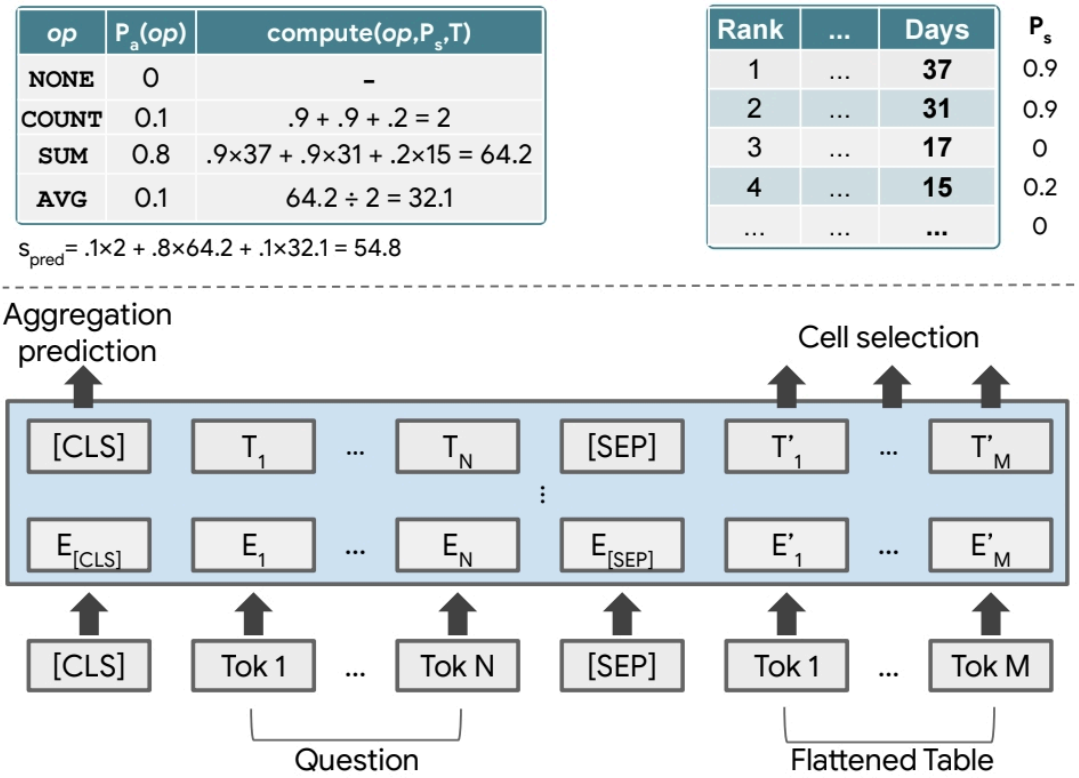
- 질문 + 표 입력을 받고, 출력으로 셀 집합 또는 셀 + 집계연산으로부터 답변을 생성. 논리 표현 생성을 생략함.
- 모델 구조 요약:
    - 입력은 "[CLS] 질문 [SEP] 표 [SEP]" 형태로 구성됨.
    - 표 내 셀과 열, 행의 위치·관계를 인식하기 위해 표 위치 임베딩을 추가.
    - 사전학습 단계 : 마스킹 학습 및 무관 연속 셀 학습을 진행함.
    - 파인튜닝 단계 : 약지도 학습을 사용하여 정답 셀 또는 집계연산을 학습함.
- 연구 목표 및 기대 효과 :
    - 낮은 라벨링 비용으로 표 기반 질의응답 모델 구축.

- 복잡한 논리 표현 없이 실용적인 성능 확보.

2. Challenges

- 약지도 환경에서 논리 표현 없이 셀 선택 및 집계연산을 학습하는 것은 어려움.
- 표는 비정형 구조이며, 행/열 구조, 셀 내부 값, 빈칸 등 다양한 요소를 포함.
- 기존 방식은 SQL 생성 등 논리표현 생성 과정이 필요했고 오류에 취약했음.
- 모델이 긴 시퀀스(표 + 질문)를 처리해야 하며, 셀 위치정보, 열/행 관계, 집계 연산을 고려해야 함.
- 사전학습을 표-텍스트 조합으로 설계하는 것이 쉽지 않음.

3. Method



- 입력 처리 및 인코딩 :
  - 질문과 표를 하나의 시퀀스로 구성.

    - 표의 셀은 플래튼되어 입력되며, 각 셀은 행/열 위치 임베딩 및 텍스트 임베딩을 가짐.
- 모델 아키텍처 :
    - BERT 기반 인코더를 확장.
    - 표 구조를 반영하기 위해 추가 임베딩과 마스킹 전략 도입.
- 사전학습 :
    - 마스킹된 셀 텍스트 및 무작위 셀 쌍 예측 등의 목적함수 사용.
    - 대량의 위키피디아 표 + 텍스트 자료에서 학습.
- 파인튜닝 :
    - 약지도 setting : 각 예제는 질문, 표, 정답 형태.
    - 모델은 먼저 열을 선택한 뒤 해당 열 안에서 셀을 선택하는 계층적 전략 사용.
    - 손실 함수로 열 선택 이진 교차엔트로피 및 셀 선택 손실 포함.
- 집계연산 처리 :
    - 셀 선택 외에 SUM, AVERAGE 등 집계 연산을 자동으로 예측.

4. Experiments

	WikiSQL	WikiTQ	SQA
Logical Form	✓	✗	✗
Conversational	✗	✗	✓
Aggregation	✓	✓	✗
Examples	80654	22033	17553
Tables	24241	2108	982

- 사용 데이터셋 :
    - SQA

    - WikiSQL
    - WikiTQ
- 실험 설계 및 변수 :
    - 사전학습 후 각 데이터셋에 맞춰 파인튜닝 진행.
    - 비교 대상 : 기존 논리표현 생성 방식 및 기타 표 기반 질의응답 모델.
    - 평가 지표 : 정확도 및 특정 셀 선택/집계 정확도.
- 추가 분석 :
    - 전이학습 실험에서 성능 개선 확인.
    - Ablation study를 통해 표 위치 임베딩, 마스킹 전략, 집계 연산 처리의 영향 분석.

5. Results

Model	Dev	Test
Liang et al. (2018)	71.8	72.4
Agarwal et al. (2019)	74.9	74.8
Wang et al. (2019)	79.4	79.3
Min et al. (2019)	84.4	**83.9**
TAPAS	**85.1**	83.6
TAPAS (fully-supervised)	88.0	86.4

Table 3: WIKISQL denotation accuracy[4].

Model	Test
Pasupat and Liang (2015)	37.1
Neelakantan et al. (2017)	34.2
Haug et al. (2018)	34.8
Zhang et al. (2017)	43.7
Liang et al. (2018)	43.1
Dasigi et al. (2019)	43.9
Agarwal et al. (2019)	44.1
Wang et al. (2019)	44.5
TAPAS	42.6
TAPAS (pre-trained on WIKISQL)	48.7
TAPAS (pre-trained on SQA)	48.8

Model	ALL	SEQ	Q1	Q2	Q3
Pasupat and Liang (2015)	33.2	7.7	51.4	22.2	22.3
Neelakantan et al. (2017)	40.2	11.8	60.0	35.9	25.5
Iyyer et al. (2017)	44.7	12.8	70.4	41.1	23.6
Sun et al. (2018)	45.6	13.2	70.3	42.6	24.8
Müller et al. (2019)	55.1	28.1	67.2	52.7	46.8
TAPAS	**67.2**	**40.4**	**78.2**	**66.0**	**59.7**

Table 5: SQA test results. ALL is the average question accuracy, SEQ the sequence accuracy, and QX, the accuracy of the X'th question in a sequence.

	SQA (SEQ)		WIKISQL		WIKITQ	
all	39.0		84.7		29.0	
-pos	36.7	-2.3	82.9	-1.8	25.3	-3.7
-ranks	34.4	-4.6	84.1	-0.6	30.7	+1.8
-{cols,rows}	19.6	-19.4	74.1	-10.6	17.3	-11.6
-table pre-training	26.5	-12.5	80.8	-3.9	17.9	-11.1
-aggregation	-		82.6	-2.1	23.1	-5.9

Table 6: Dev accuracy with different embeddings removed from the full model: positional (pos), numeric ranks (ranks), column (cols) and row (rows). The model without table pre-training was initialized from the original BERT model pre-trained on text only. The model without aggregation is only trained with the cell selection loss.

	all	text	header	cell
all	71.4	68.8	96.6	63.4
word	74.1	69.7	96.9	66.6
number	53.9	51.7	83.6	53.2

- SQA에서 정확도 55.1% → 67.2%로 향상.
- WikiSQL / WikiTQ에서도 기존 모델과 동등하거나 더 나은 성능.
- 모델 크기 및 학습 효율성: 단순 구조로 높은 성능 구현.
- 실무 적용 가능성 및 한계점 :
  - 표 기반 질의응답 응용에 적합.
  - 다만 매우 큰 표나 복잡한 연산이 필요한 경우 제한 있음.

6. Insight

- 논리 표현 생성을 생략하고도 우수한 성능 달성 가능.
- 사전학습이 표 구조 인식에 중요한 역할.
- 낮은 라벨링 비용, 단순 구조, 표-텍스트 융합 모델 가능성 제시.
- 입력 길이 제한으로 매우 큰 표에는 적용 어려움, 복잡한 연산이나 다중 테이블 QA에는 최적화 부족.
- 표 기반 애플리케이션에 바로 적용 가능.
- 입력 길이, 배치 크기, 열 선택 파라미터 설정이 성능에 큰 영향.