CABINET: Content Relevance based Noise Reduction for Table Question Answering

ICLR 2024

*Sohan Patnaik[1,2], *Heril Changwal[1,3+], *Milan Aggarwal[1]

Sumit Bhatia[1], Yaman Kumar Singla[1], Balaji Krishnamurthy[1]

[1]MDSR Lab, Adobe [2]IIT Kharagpur [3]IIT Roorkee

Part 1. Background

- **Table Question Answering**
  - Query the table in natural language to extract desired information

Question: What was the reported mainline RPM for American Airlines in 2017?

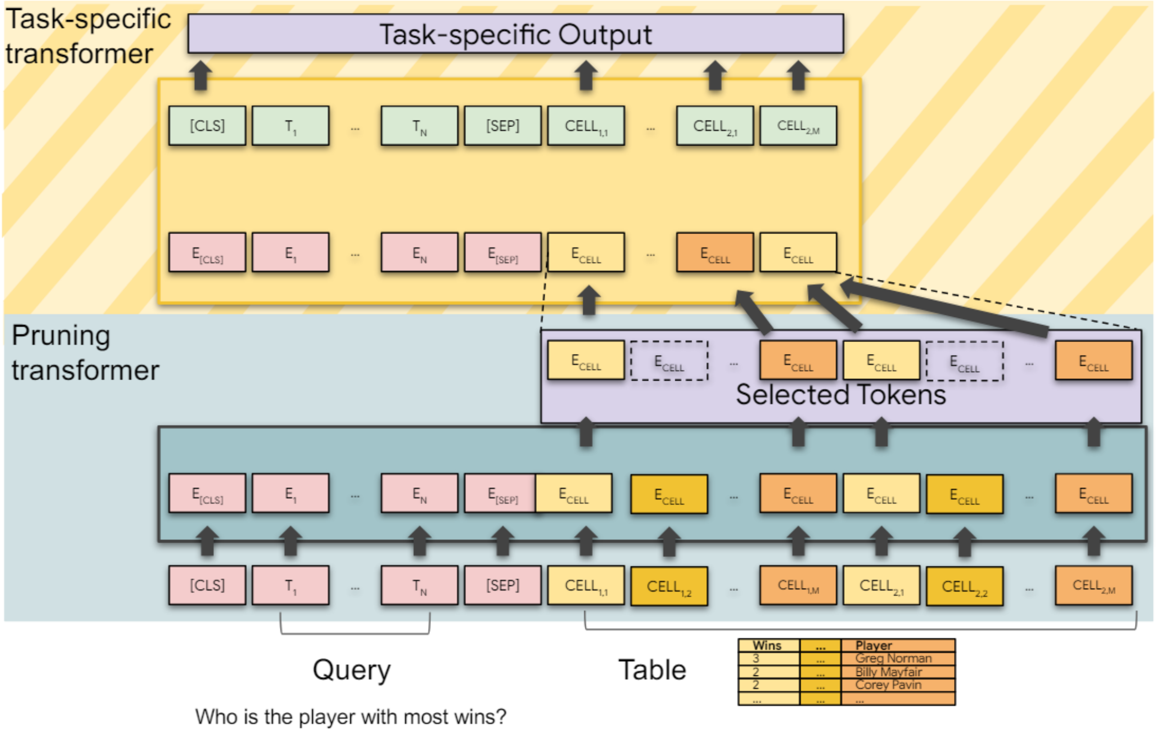
Table 1.	Year Ended December 31.		
	2017	2016	2015
Mainline			
Revenue passenger miles (millions)[a]	201,351	199,014	199,467
Available seat miles (millions)[b]	243,806	241,734	239,375
Passenger load factor (percent)[c]	82.6	82.3	83.3

  - Typical transformer-based LLMs
    - Use standard language modeling objectives
    - Do not account for the table structure and underlying compositionality of data
  - To close this gap between structured and unstructured data
    - Pre-training on table semantic parsing
    - Table-based Reasoning (In-context Learning)

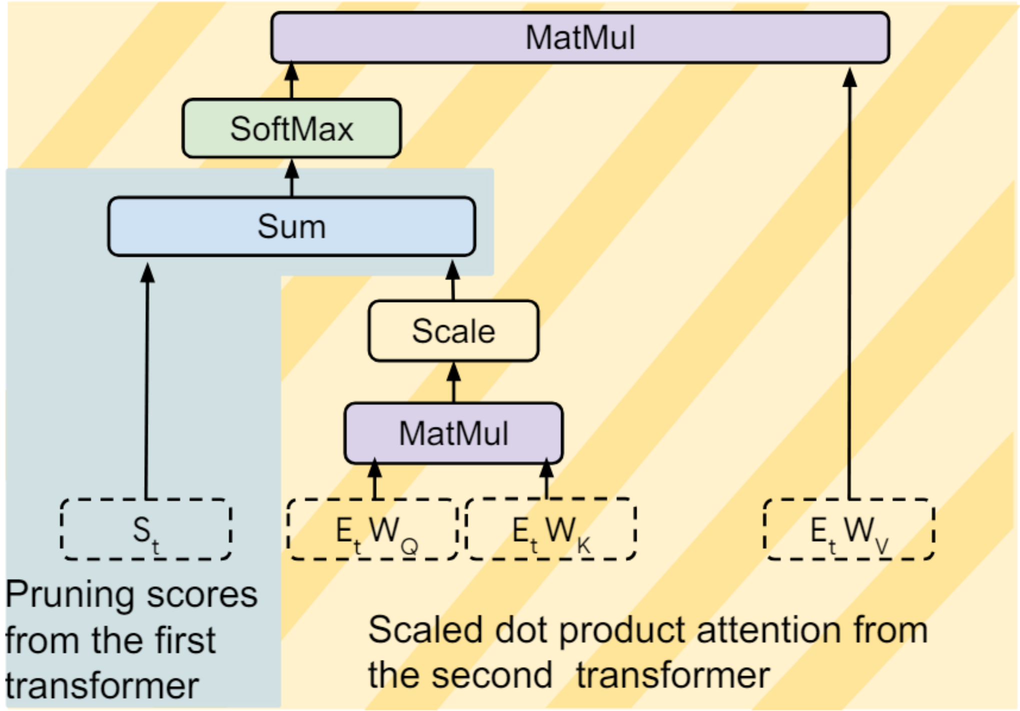
Nengzheng Jin et al. A Survey on Table Question Answering: Recent Advances. arXiv preprint arXiv:2207.05270.
Tao Yu et al. Gra{pp}a: Grammar-augmented pre-training for table semantic parsing. ICLR, 2021.

Background

- **Noise Reduction for Table QA**
  - Selects relevant tokens in flattened tabular data
    - Pruning score $s_t = \log(P(t|q, T))$ and keep the top-k tokens



Double transformer



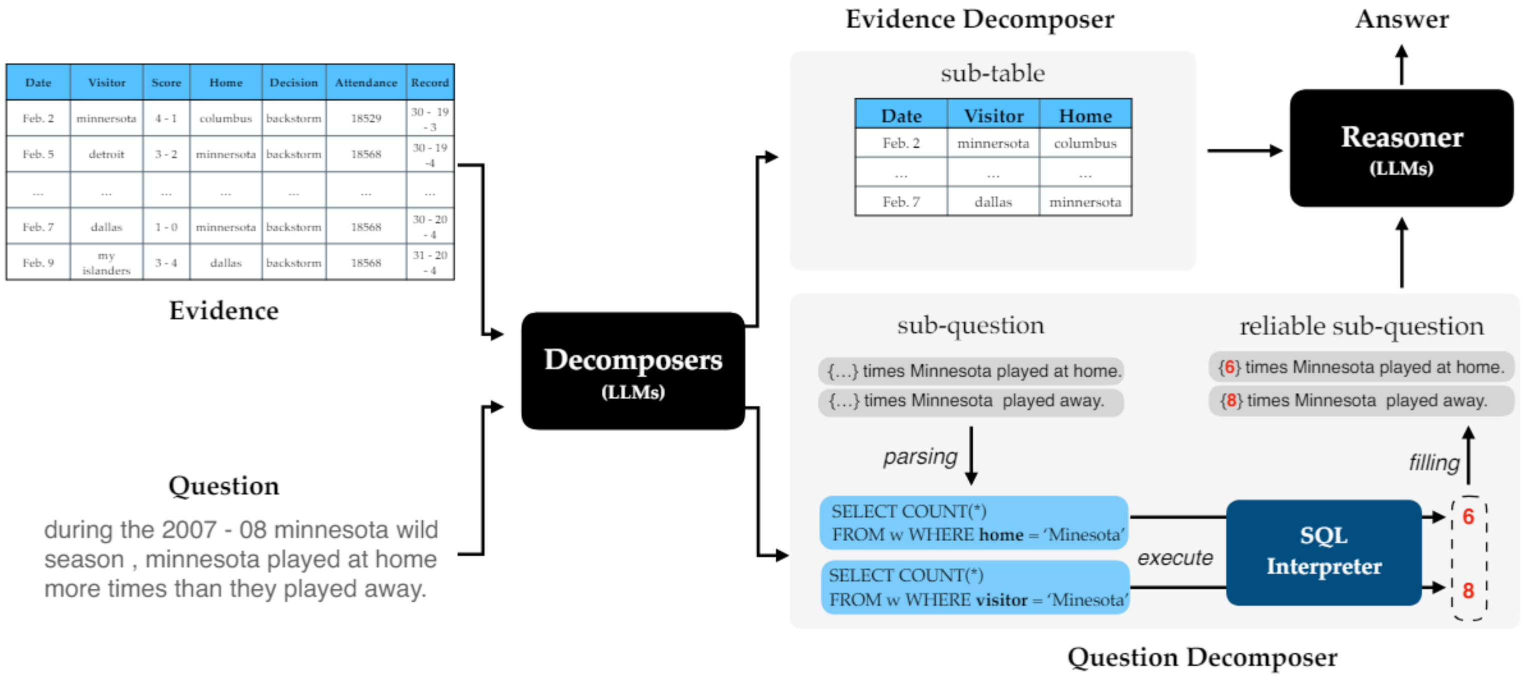
Scaled dot product attention

Syrine Krichene et al. DoT: An efficient double transformerfor NLP tasks with tables. ACL Findings, 2021.

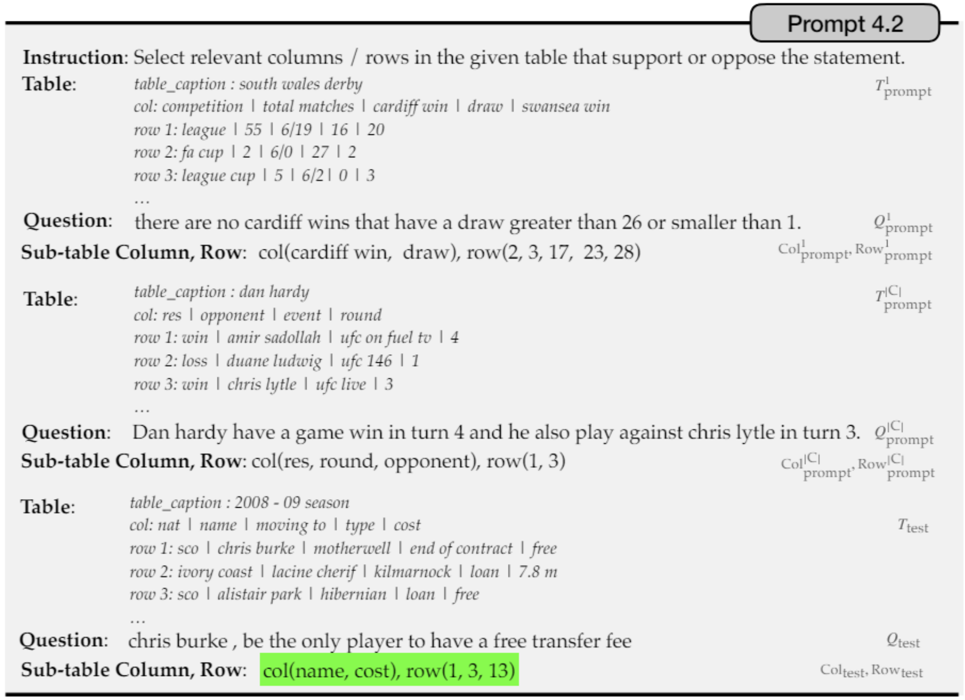
Part 1.

Background

- **DATER**
  - Extract sub-table by GPT-3 based in-context reasoning
  - Decompose a complex question into step-by-step sub-questions



Hierarchical semantic parsing method

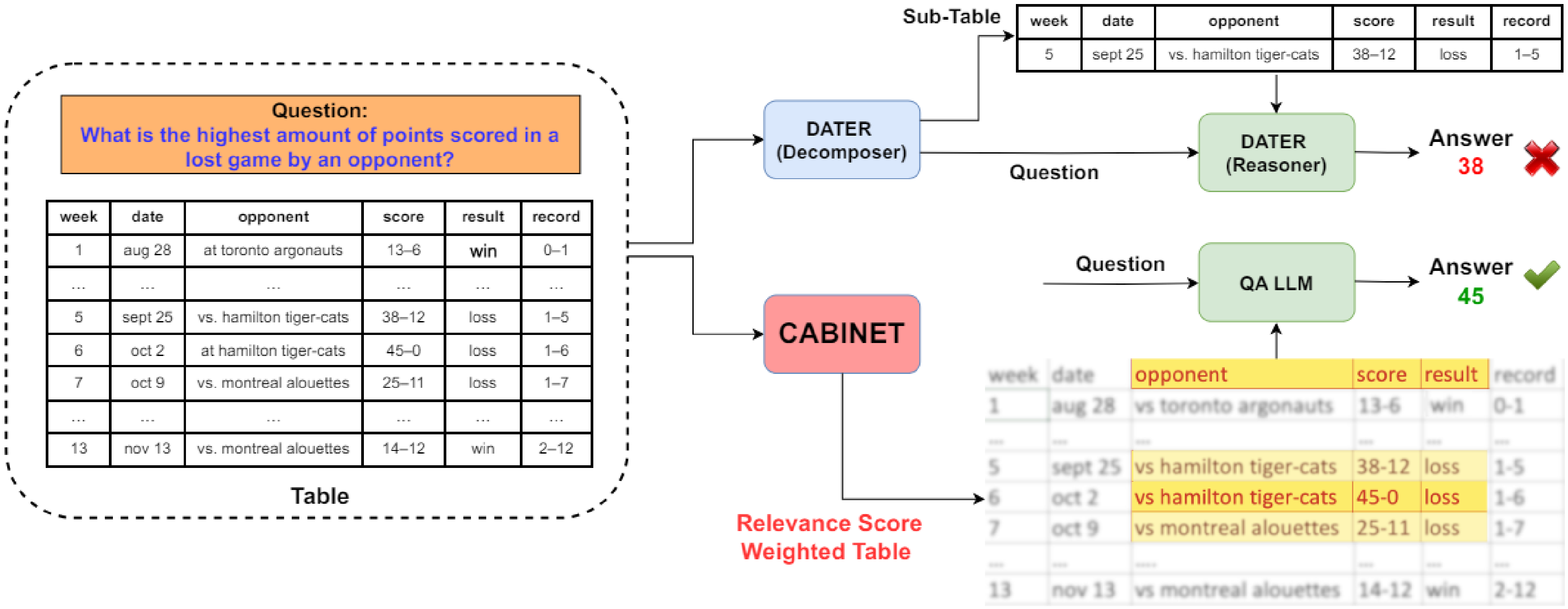


Evidence Decomposer

Yunhu Ye et al. Large language models are versatile decomposers: Decomposing evidence and questions for table-based reasoning. ACM SIGIR, 2023.

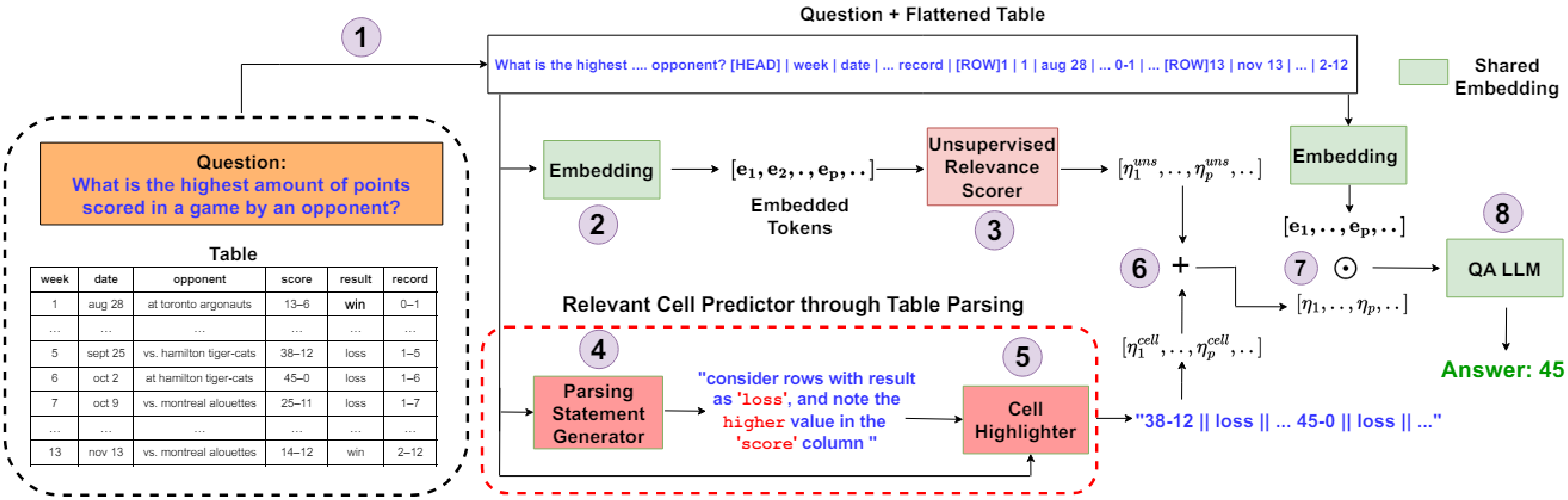
Introduction

- **CABINET**
  - Content Relevance-based Noise Reduction for Table QA
    - Weigh relevant table parts higher without removing content explicitly
    - Parsing statement generator helps unsupervised relevance scorer



Introduction

- **CABINET**
  - Content Relevance-based Noise Reduction for Table QA
    - Weigh relevant table parts higher without removing content explicitly
    - Parsing statement generator helps unsupervised relevance scorer



Method

- **Unsupervised Relevance Scorer (URS)**
  - Select top-k similar columns by cosine similarity

Input tokens

$$\mathcal{I}_{tokens} = (\mathcal{Q}_{tokens}; \mathcal{T}_{tokens}) \quad Q_{tokens} = \{q_1, q_2, \ldots, q_{|Q|}\} \; T = \{c_{ij} | \mathbf{1} \leq i \leq \mathrm{N}_{row}, 1 \leq j \leq \mathrm{N}_{col}\}$$

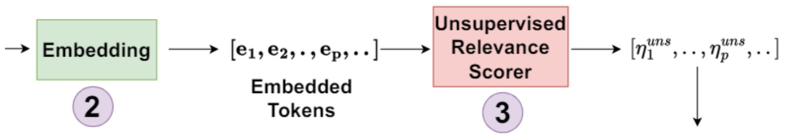
Question + Flattened Table

$$T_{flattened} = [HEAD] : c_{11} \mid c_{12} \mid \cdots \mid c_{1N_{cot}} \mid [ROW]1 : c_{21} \mid \cdots \mid c_{2N_{cot}} \mid [ROW]2 : \cdots$$

Unsupervised Relevance Score

$$e_1^{URS}, e_2^{URS}, \cdots, e_{|\mathcal{I}_{tokens}|}^{URS} = Embedding_{URS}(\mathcal{I}_{tokens})$$

$$h_1, \cdots, h_p, \cdots, h_{|\mathcal{I}_{tokens}|} = TE_{URS}(e_1^{URS}, e_2^{URS}, \cdots, e_{|\mathcal{I}_{tokens}|}^{URS})$$



Normalization

$$H_p = \phi_\mu(h_p); \; \sigma_p = \phi_\sigma(h_p) \quad z_p = \mu_p + s * \sigma_p \quad \eta_p^{uns} = sigmoid(z_p)$$

Method

- **Unsupervised Relevance Scorer (URS)**
  - T-SNE(T-Stochastic Neighbor Embedding)

Total Loss

$$\mathcal{L} = \mathcal{L}_{CE} + \lambda_{clu} * \mathcal{L}_{clu} + \lambda_{sep} * \mathcal{L}_{sep} + \lambda_{sparse} * \mathcal{L}_{sparse}$$

Separation loss

$$\mathcal{L}_{sep} = 2 - \left|\left|\mu^{clu}_{relevant} - \mu^{clu}_{irrelevant}\right|\right|^2$$

Clustering loss

$$\mathcal{L}_{clu} = \frac{1}{B}\sum_b KL(Z||Q) = \frac{1}{B}\sum_b\sum_p\sum_j z_{pj} log\frac{z_{pj}}{q_{pj}}$$

Clustering Latent Vectors

$$q_{pj} = \frac{(1+||h_p - \mu^{clu}_j||^2/\alpha)^{-\frac{\alpha+1}{2}}}{\sum_{j'}(1+||h_p - \mu^{clu}_{j'}||^2/\alpha)^{-\frac{\alpha+1}{2}}}$$

$$\mu^{clu}_0 = \mu^{clu}_{relevant} \qquad \mu^{clu}_1 = \mu^{clu}_{irrelevant}$$

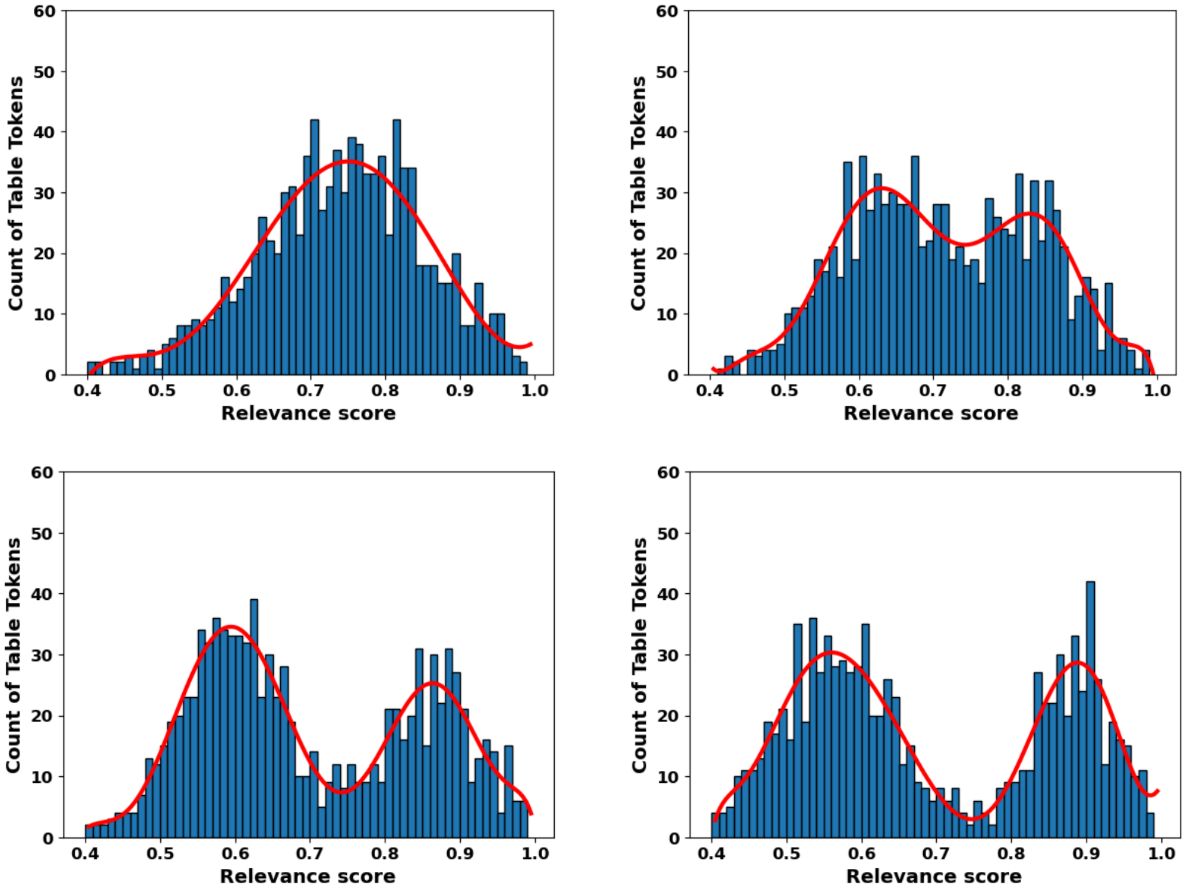
Target distribution

$$z_{pj} = \frac{q^2_{pj}/f_{pj}}{\sum_{j'} q^2_{pj'}/f_{pj'}}$$

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. JMLR, 2008.

Method

- **Unsupervised Relevance Scorer (URS)**
  - Ablation Study (Left: Without Loss / Right: With Loss)



Separation loss

$$\mathcal{L}_{sep} = 2 - \left|\left|\mu^{clu}_{relevant} - \mu^{clu}_{irrelevant}\right|\right|^2$$

Clustering loss

$$\mathcal{L}_{clu} = \frac{1}{B}\sum_{b} KL(Z||Q) = \frac{1}{B}\sum_{b}\sum_{p}\sum_{j} z_{pj} log \frac{z_{pj}}{q_{pj}}$$

Method

- **Unsupervised Relevance Scorer (URS)**
  - Get relevance scores Lower for tokens in one cluster

Total Loss

$$\mathcal{L} = \mathcal{L}_{CE} + \lambda_{clu} * \mathcal{L}_{clu} + \lambda_{sep} * \mathcal{L}_{sep} + \lambda_{sparse} * \mathcal{L}_{sparse}$$

Sparsification Los

$$\mathcal{L}_{sparse} = \frac{1}{|\mathcal{T}_{tokens}|} \sum_{p} \mathrm{e}^{-z_p^2}; \ |\mathcal{Q}_{tokens}| + 1 \le p \le |\mathcal{Q}_{tokens}| + |\mathcal{T}_{tokens}| \qquad z_p = \mu_p + s * \sigma_p$$

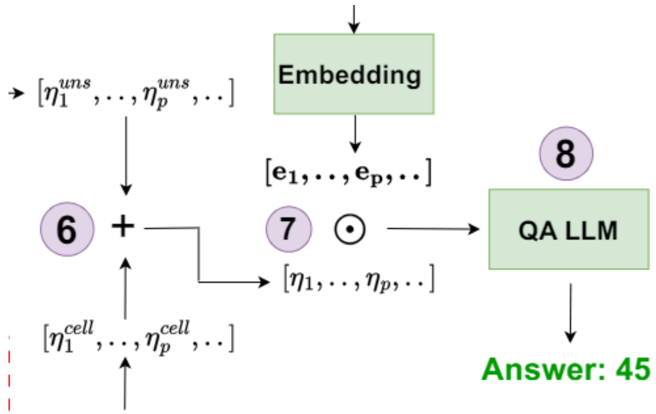
When providing input to QA LLM

$$e_1, e_2, \cdots, e_{|\mathcal{I}_{tokens}|} = Embedding_{QA}(\mathcal{I}_{tokens})$$

$$e_p^{'} = \eta_p \odot e_p; \ |\mathcal{Q}_{tokens}|+1 \le p \le |\mathcal{Q}_{tokens}| + |\mathcal{T}_{tokens}|$$

$$h_1^{'}, \cdots, h_{|\mathcal{I}_{tokens}|}^{'} = TE_{QA}(e_1^{'}, e_2^{'}, \cdots, e_{|\mathcal{I}_{tokens}|}^{'})$$

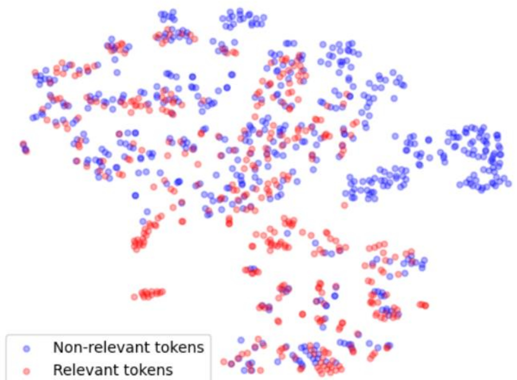
$$a_1, a_2, \cdots, a_N = TD_{QA}(h_1^{'}, \cdots, h_{|\mathcal{I}_{tokens}|}^{'})$$

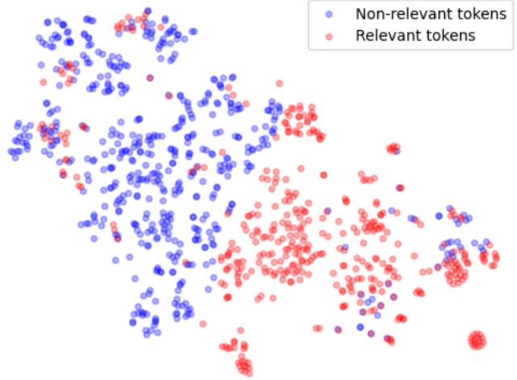


Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. JMLR, 2008.

Method

- **Unsupervised Relevance Scorer (URS)**
  - Ablation Study (Left: Without Loss / Right: With Loss)

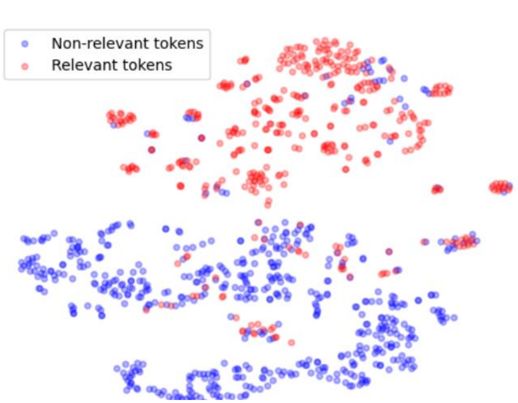


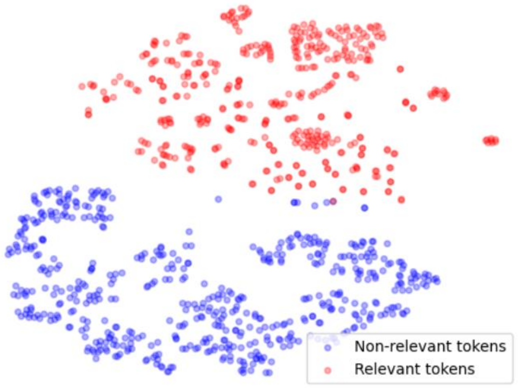


Clustering loss

$$\mathcal{L}_{clu} = \frac{1}{B}\sum_{b} KL(Z||Q) = \frac{1}{B}\sum_{b}\sum_{p}\sum_{j} z_{pj} log \frac{z_{pj}}{q_{pj}}$$

$$z_p = \mu_p + s * \sigma_p$$





Sparsification Loss

$$\mathcal{L}_{sparse} = \frac{1}{|\mathcal{T}_{tokens}|}\sum_{p} \mathrm{e}^{-z_p^2};$$

$$|\mathcal{Q}_{tokens}| + 1 \leq p \leq |\mathcal{Q}_{tokens}| + |\mathcal{T}_{tokens}|$$

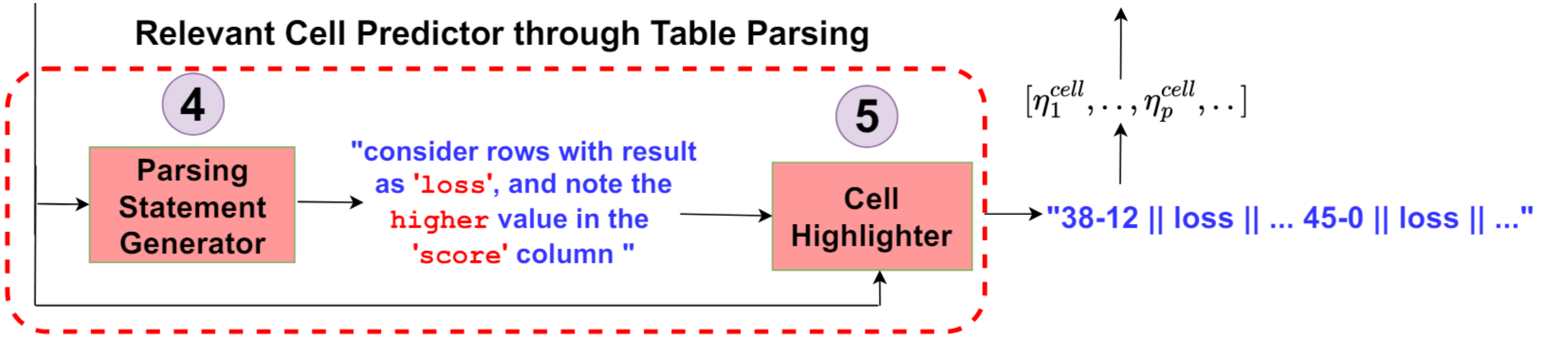
$$z_p = \mu_p + s * \sigma_p$$

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. JMLR, 2008.

Method

- **Parsing Statement Generator (PSG)**
  - Flan T5-xl is pre-trained to WikiTableQuestions (WikiTQ)
    - The most complex QA dataset containing a variety of samples
    - We manually annotate parsing statement
  - Pre-trained PSG model is fine-tuend to datasets of each experiments

Cluster	Question	Answer	Parsing Statement
1	how many episodes had a nightly rank of 11?	3	to find number of episodes with nightly rank of 11, we need to look at the column named ”nightly rank” and count number of times the value 11 occurs.

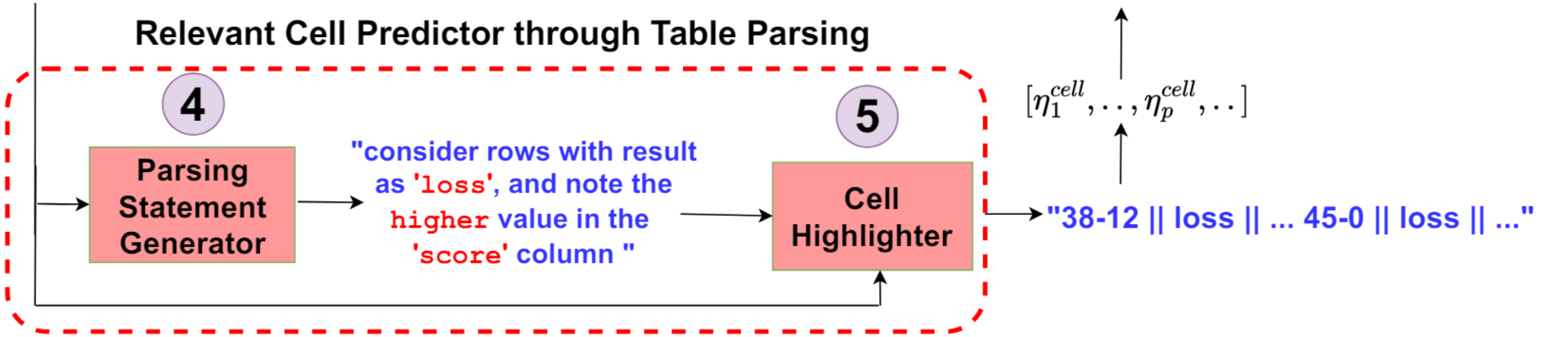


Panupong Pasupat and Percy Liang. Compositional semantic parsing on semi-structured tables. 2015. URL http://arxiv.org/abs/1508.00305.

Method

- **Cell Highlighting**
  - Flan T5-xl is fine-tuned to ToTTo
    - Given the parsing statement, predictor generates highlighted cells
  - TOTTO
    - Open-domain Controlled generation task
    - Given a Wikipedia table and a set of highlighted cells
    - To produce a single sentence description

$$c_1^{highlighted} \mid\mid \cdots \mid\mid c_M^{highlighted} = Cell\_Highlighter_{LLM}(\mathcal{T}, text_{parse})$$



Ankur Parikh et al. ToTTo: A controlled table-to-text generation dataset. EMNLP, 2020.

Experiment

- **Implementation Details**
  - Employ OmniTab (Jiang et al., 2022) backbone comprising of BART-Large
  - Hidden dimension of TE_{URS} is 1024
  - Optimize with cosine annealing through AdamW

Clustering loss

Dataset Statistics

Dataset	# Train samples	# Validation samples	# Test samples
WikiTQ	11321	2831	4344
WikiSQL	56355	8421	15878
FeTaQA	7326	1001	2003

hengbao Jiang et al. OmniTab: Pretraining with natural and synthetic data for few-shot table-based question answering. NAACL, 2022.
Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. ICLR, 2017.

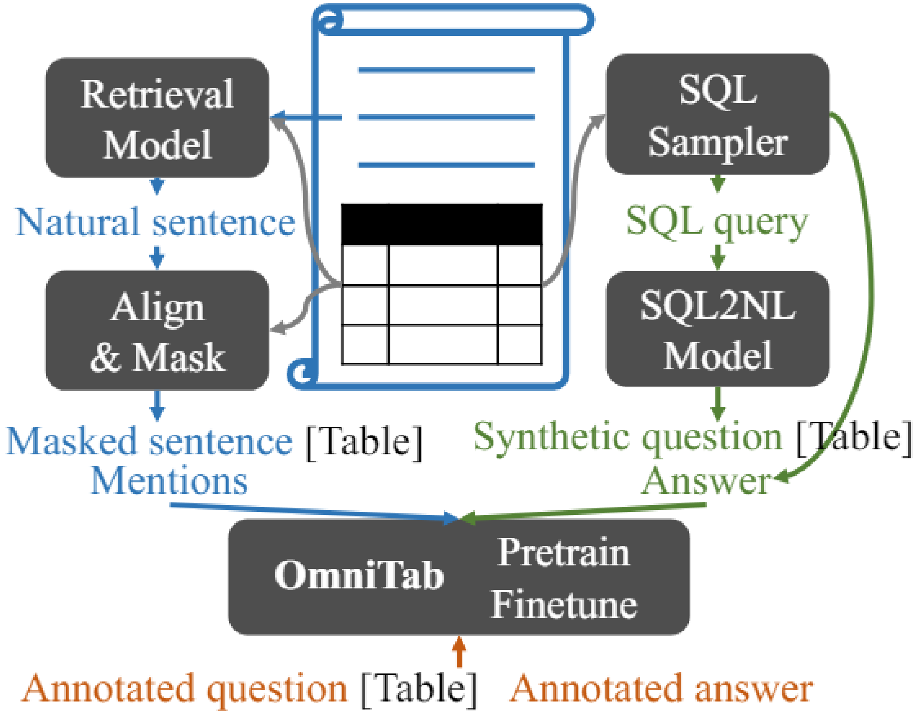
Experiment

- **OmniTab (Jiang et al., 2022)**
  - Employ TAPEX (Liu et al., 2021) backbone comprising of BART-Large
  - Pretrain with natural data, synthetic data
  - Finetune with limited annotated questions

Title: List of 2002 box office number-one films in the United States

#	Date	Film	Gross
1	January 6, 2002	The Lord of the Rings: The Fellowship of the Ring	$23,006,447
5	February 3, 2002	Black Hawk Down	$11,112,555
6	February 10, 2002	Collateral Damage	$15,058,432
18	May 4, 2002	Spider-Man	$114,844,116
20	May 19, 2002	Star Wars Episode II: Attack of the Clones	80,027,814

		Input	*Output*
Pretrain	*Natural*	~~Spider-Man~~__ with its ~~$114.8 million~~__ mark established a new opening weekend **record**. [Table]	Spider-Man, $114.8 million
	Synthetic	SELECT film WHERE gross > (SELECT gross WHERE film = 'Star Wars') Which **film** has **grossed more than Star Wars**? [Table]	Spider-Man
	Annotated Finetune	What was the **previous film** to **air before Collateral Damage**? [Table]	Black Hawk Down



hengbao Jiang et al. OmniTab: Pretraining with natural and synthetic data for few-shot table-based question answering. NAACL, 2022.
Qian Liu et al. TAPEX: table pre-training via learning a neural SQL executor. ICLR, 2022.

Experiment

- **Experiment**
  - CABINET achieves SoTA performance
  - Metric: Sacre-BLEU (S-BLEU)

Generation Task on FeTaQA

Method	S-BLEU	# params
Fine-tuning Table-specific LLMs		
PeaQA (Pal et al., 2022)	33.5	406 M
TAPEX (Liu et al., 2022)	34.7	406 M
OmniTab (Jiang et al., 2022)	34.9	406 M
Fine-tuning text-based LLMs		
T5-small (Nan et al., 2022)	21.6	60 M
T5-base (Nan et al., 2022)	28.1	222 M
T5-large (Nan et al., 2022)	30.5	738 M
T5-3b (Xie et al., 2022)	33.4	2.9 B
FlanT5-xl	36.2	2.9 B
Few/zero shot Prompting of LLMs		
Codex-COT (Chen, 2023)	27.0	175 B
Codex (Ye et al., 2023)	27.9	175 B
DATER (Ye et al., 2023)	30.9	175 B
CABINET (Ours)	**40.5**	560 M

Extraction Task on WikiTQ

Method	Acc.	# params
Fine-tuning Table-specific LLMs		
TAPAS (Herzig et al., 2020)	86.4	345 M
GraPPa (Yu et al., 2021)	84.7	355 M
DoT (Krichene et al., 2021)	85.5	299 M
TAPEX (Liu et al., 2022)	86.4	406 M
OmniTab (Jiang et al., 2022)	87.9	406 M
UTP (Chen et al., 2023b)	88.1	345 M
ReasTAP (Zhao et al., 2022)	88.8	406 M
Fine-tuning text-based LLMs		
T5-3b (Xie et al., 2022)	85.9	2.9 B
FlanT5-xl	87.8	2.9 B
Few/zero shot Prompting of LLMs		
ChatGPT (Jiang et al., 2023)	51.6	175 B
StructGPT (Jiang et al., 2023)	54.4	175 B
CABINET (Ours)	**89.5**	560 M

Experiment

- **Robustness to noise and irrelevant information**
  - Perform four types of perturbations
    - Row Addition (RA), Row Permutation (RP)
    - Column Permutation (CP)
    - Cell Replacement (CR)

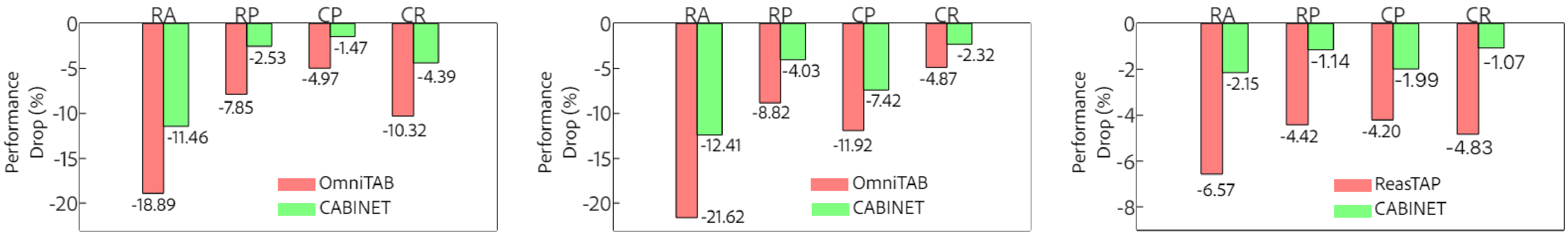
Extraction Task on WikiTQ

Method	Acc.	# params
Fine-tuning Table-specific LLMs		
TAPAS (Herzig et al., 2020)	48.8	345 M
TaBERT (Yin et al., 2020)	52.3	345 M
MATE (Eisenschlos et al., 2021)	51.5	340 M
GraPPa (Yu et al., 2021)	52.7	355 M
DoT (Krichene et al., 2021)	54.0	299 M
TableFormer (Yang et al., 2022)	52.6	345 M
TAPEX (Liu et al., 2022)	55.5	405 M
ReasTAP (Zhao et al., 2022)	58.6	406 M
TaCube (Zhou et al., 2022)	60.8	406 M
OmniTab (Jiang et al., 2022)	62.7	406 M
Fine-tuning text-based LLMs		
T5-3b (Xie et al., 2022))	49.3	2.9 B
FlanT5-xl (Chung et al., 2022a)	64.4	2.9 B
Few/zero shot Prompting of LLMs		
Codex (Ye et al., 2023)	47.6	175 B
Codex-COT (Chen, 2023)	48.8	175 B
Binder (Cheng et al., 2023)	64.6	175 B
LEVER (Ni et al., 2023)	65.8	175 B
DATER (Ye et al., 2023)	65.9	175 B
ChatGPT (Jiang et al., 2023)	43.3	175 B
StructGPT (Jiang et al., 2023)	48.4	175 B
CABINET (Ours)	**69.1**	560 M

Experiment

- **Robustness to noise and irrelevant information**
  - Perform four types of perturbations
    - Row Addition (RA), Row Permutation (RP)
    - Column Permutation (CP)
    - Cell Replacement (CR)

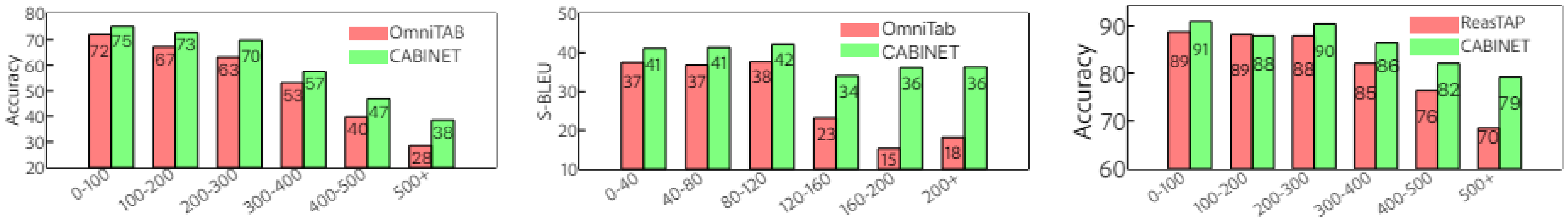
Relative performance drop with perturbations



(a) WikiTQ (b) FeTaQA (c) WikiSQL

Experiment

- **Impact of Table Size on Performance**
  - Entire information is usually not required to answer
  - Distracting information causes performance drop



(a) WikiTQ (b) FeTaQA (c) WikiSQL

Experiment

- **Effect of Clustering Table Tokens**

$\mathcal{L}_{clu}$	$\mathcal{L}_{sep}$	$\mathcal{L}_{sparse}$	WikiTQ	FeTaQA	WikiSQL
✗	✗	✗	60.8	35.1	86.2
✗	✗	✓	60.9	35.1	86.3
✓	✗	✗	62.7	35.0	88.9
✓	✗	✓	61.0	35.0	**89.5**
✓	✓	✗	61.0	35.1	89.1
✓	✓	✓	**65.6**	**35.8**	89.3

λ_{uns}	λ_{cell}	WikiTQ	FeTaQA,	WikiSQL
1	0	65.6	35.8	**89.2**
0.7	0.3	**69.1**	**40.5**	**89.2**
0.5	0.5	68.6	**40.5**	88.9
0.3	0.7	67.0	38.9	88.8
0	1	37.6	24.2	34.1

Total Loss

$$\mathcal{L} = \mathcal{L}_{CE} + \lambda_{clu} * \mathcal{L}_{clu} + \lambda_{sep} * \mathcal{L}_{sep} + \lambda_{sparse} * \mathcal{L}_{sparse}$$

Separation loss

$$\mathcal{L}_{sep} = 2 - \left|\left|\mu^{clu}_{relevant} - \mu^{clu}_{irrelevant}\right|\right|^2$$

Clustering loss

$$\mathcal{L}_{clu} = \frac{1}{B}\sum_{b} KL(Z||Q) = \frac{1}{B}\sum_{b}\sum_{p}\sum_{j} z_{pj} log\frac{z_{pj}}{q_{pj}}$$

Sparsification Los

$$\mathcal{L}_{sparse} = \frac{1}{|\mathcal{T}_{tokens}|}\sum_{p} \mathrm{e}^{-z_p^2}; \ |\mathcal{Q}_{tokens}| + 1 \leq p \leq |\mathcal{Q}_{tokens}| + |\mathcal{T}_{tokens}| \qquad z_p = \mu_p + s * \sigma_p$$

Part 4.

Experiment

- **Ablation Study**
  - Unsupervised Relevance Scorer (URS) VS BERT based similarity metric
  - With or without highlighted cells

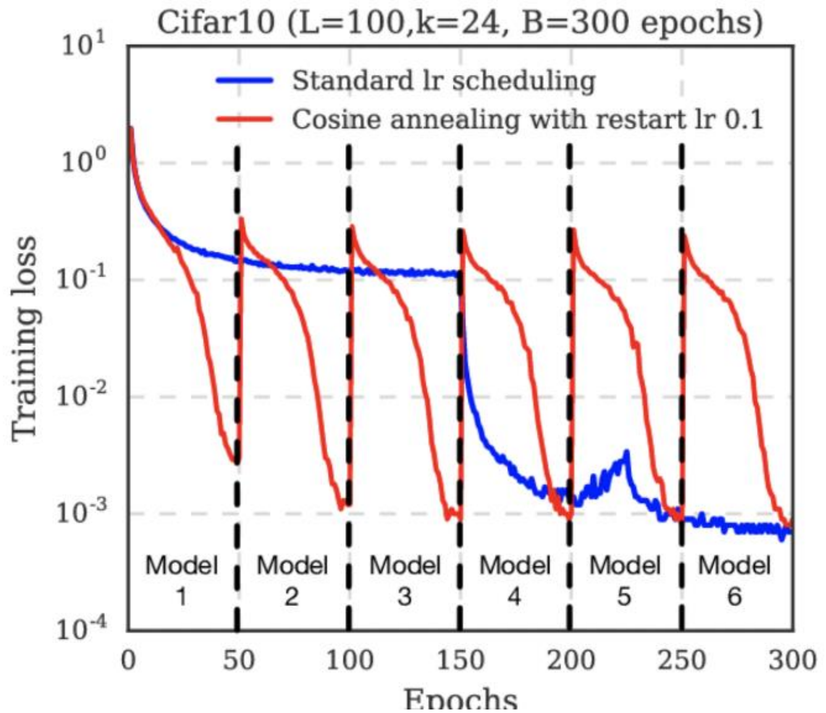
Method	WikiTQ	FeTaQA	WikiSQL
OmniTab	63.1	35.9	85.8
CABINET w parsing statement as input to QA model instead of highlighting corresponding cells	66.2	34.9	85.9
CABINET with BERT based relevance scoring (as discussed above) without cell highlighter	61.8	34.9	83.7
CABINET with BERT based relevance scoring (as discussed above) with cell highlighter	64.5	36.7	85.1
CABINET with question as input to cell highlighter	63.7	34.4	85.7
CABINET with URS only and without cell highlighter	65.6	35.8	89.3
CABINET	**69.1**	**40.5**	**89.5**

Conclusion

- **CABINET**
  - Vulnerability to noise, distracting information leads to lower performance
  - Weigh the table content based on its relevance to the question
  - Outperforms with much larger GPT-3 scale models based in context learning

Appendix

- **Cosine annealing learning rate schedule**
  - Learning rate changes between cosine maximum and minimum
  - Deviate from local minimum
  - Improve generalization of model performance



Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. ICLR, 2017.

Part 6.

Appendix

- **Free-form Table Question Answering**
  - Both questions and answers is natural and grounded in the context of the entire table
  - Retrieving and reasoning over relations of multiple entities

<table>
<tr><th colspan="5">Page Title: Hawaii demographics - ancestry</th></tr>
<tr><th>Racial composition</th><th>1970</th><th>1990</th><th>2000</th><th>2010</th></tr>
<tr><td>White</td><td>38.80%</td><td>33.40%</td><td>24.30%</td><td>24.70%</td></tr>
<tr><td>Asian</td><td rowspan="2">57.70%</td><td rowspan="2">61.80%</td><td>41.60%</td><td>38.60%</td></tr>
<tr><td>Native Hawaiian and other Pacific Islander</td><td>9.40%</td><td>10.00%</td></tr>
<tr><td>Black</td><td>1.00%</td><td>2.50%</td><td>1.80%</td><td>1.60%</td></tr>
<tr><td>Native American and Alaskan native</td><td>0.10%</td><td>0.50%</td><td>0.30%</td><td>0.30%</td></tr>
<tr><td colspan="2">Q: What ethnic groups are the majorities back in 1970?</td><td colspan="3">A: In 1970, Hawaii's population mainly consists of 38.8% white and 57.7% asian, native hawaiian and other pacific islander.</td></tr>
</table>

Dataset	Answer Format	Avg # Words in Answer
SQuAD (Rajpurkar et al., 2016)	Text-span	3.2
HotpotQA (Yang et al., 2018)	Short-form entity	2.2
NarrativeQA (Kočiský et al., 2018)	Free-form text	4.7
ELI5 (Fan et al., 2019)	Free-form text	130.6
WikiTableQuestions (Pasupat and Liang, 2015)	Short-form entity	1.7
SequenceQA (Saha et al., 2018)	Short-form entity	1.2
HybridQA (Chen et al., 2020e)	Short-form entity	2.1
FeTaQA	Free-form text	18.9

Linyong Nan et al. FeTaQA: Free-form table question answering. TACL. 2022.

Appendix

- **WikiTableQuestion (WikiTQ)**
  - Answer a question using an HTML table as the knowledge source
  - For each question, we put one of the 36 generic prompts

Year	City	Country	Nations
1896	Athens	Greece	14
1900	Paris	France	24
1904	St. Louis	USA	12
...	...	...	...
2004	Athens	Greece	201
2008	Beijing	China	204
2012	London	UK	204

x_1: *"Greece held its last Summer Olympics in which year?"*
y_1: $\{2004\}$

x_2: *"In which city's the first time with at least 20 nations?"*
y_2: $\{\text{Paris}\}$

x_3: *"Which years have the most participating countries?"*
y_3: $\{2008, 2012\}$

x_4: *"How many events were in Athens, Greece?"*
y_4: $\{2\}$

x_5: *"How many more participants were there in 1900 than in the first year?"*
y_5: $\{10\}$

Panupong Pasupat and Percy Liang. Compositional semantic parsing on semi-structured tables. 2015. URL http://arxiv.org/abs/1508.00305.

Appendix

- **WikiSQL**
  - Inputs consist of a table and a question
  - Outputs consist of a SQL query and the result from execution

Table: CFLDraft

Pick #	CFL Team	Player	Position	College
27	Hamilton Tiger-Cats	Connor Healy	DB	Wilfrid Laurier
28	Calgary Stampeders	Anthony Forgone	OL	York
29	Ottawa Renegades	L.P. Ladouceur	DT	California
30	Toronto Argonauts	Frank Hoffman	DL	York
...	...	...	...	...

Question:

How many CFL teams are from York College?

SQL:

```
SELECT COUNT CFL Team FROM
CFLDraft WHERE College = "York"
```

Result:

2

Victor Zhong et al. Seq2sql: Generating structured queries from natural language using reinforcement learning. CoRR, abs/1709.00103, 2017.