CSCE 689 - Special Topics in NLP for Science

Lecture 18: Table Language Models

Yu Zhang

yuzhang@tamu.edu

March 27, 2025

Course Website: https://yuzhang-teaching.github.io/CSCE689-S25.html

Midterm Project Report (Due 3/30)

- The report should include task definition, related work, method, preliminary results, unfinished parts, and a timeline to finish them.
- The report should be 4-6 pages (ACL 2024 template, excluding references).
  - Feel free to reuse any content from your proposal.
- Submit your review as a single PDF file on Canvas. Each group only needs to submit one report.

Agenda

- TaBERT: Masked Language Modeling for Table Cell Representation
- TableLlama: Instruction Tuning for Table QA
- UniHGKR: Instruction Tuning for Table Retrieval
- TabPFN: A Tabular Foundation Model for Missing Value Prediction

Agenda

- TaBERT: Masked Language Modeling for Table Cell Representation
- TableLlama: Instruction Tuning for Table QA
- UniHGKR: Instruction Tuning for Table Retrieval
- TabPFN: A Tabular Foundation Model for Missing Value Prediction

Motivation: Natural Language Interfaces over Tabular Data

- Input (Query): Show me flights from Pittsburgh to Seattle
- Input (Tabular Data):

Flight	
FlightNo	UniqueId
Departure	foreign key
Arrival	foreign key

Airport	
Name	UniqueId
CityName	string
PublicTransport	boolean

- Output (SQL):

```
Select FlightId From Flight
Where Flight.Origin = ‘PIT’ And
      Flight.Destination = ‘SEA’
```

- Input (Query): Which US city has the largest GDP?
- Input (Tabular Data):

City	Country	Population	GDP
New York	USA	8.62M	1275B
Hong Kong	China	7.39M	341.4B
Tokyo	Japan	9.27M	1800B
London	UK	8.78M	650B
Los Angeles	USA	4.00M	941B

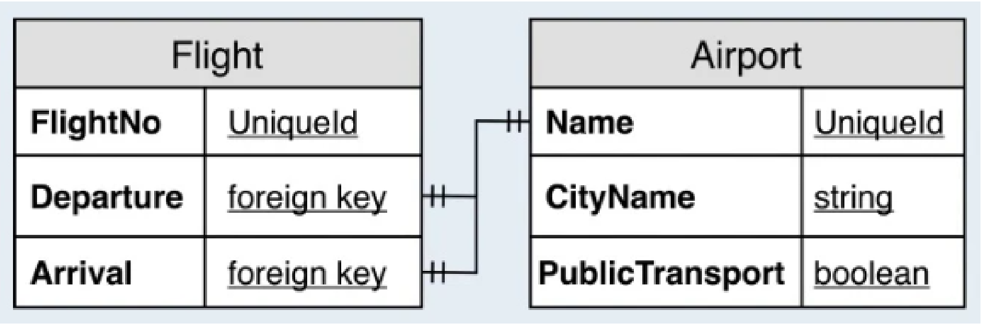
- Output (SQL):

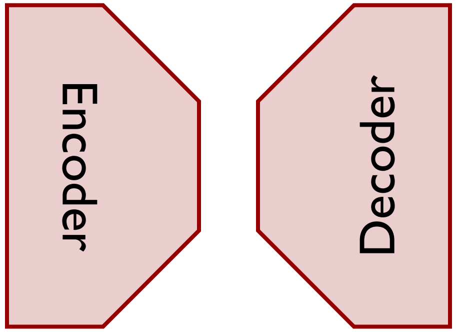
```
Table.Where(City == ‘USA’)
     .Argmax(GDP)
     .Select(City)
```

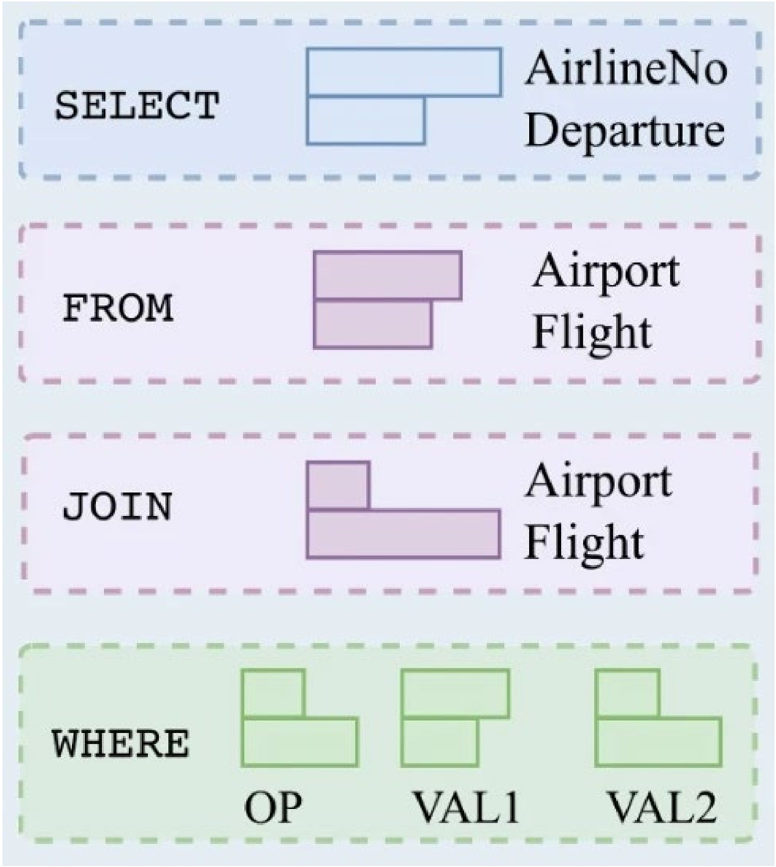
Motivation: Natural Language Interfaces over Tabular Data

- We should expect an encoder-decoder architecture.

Show me flights from
Pittsburgh to Seattle



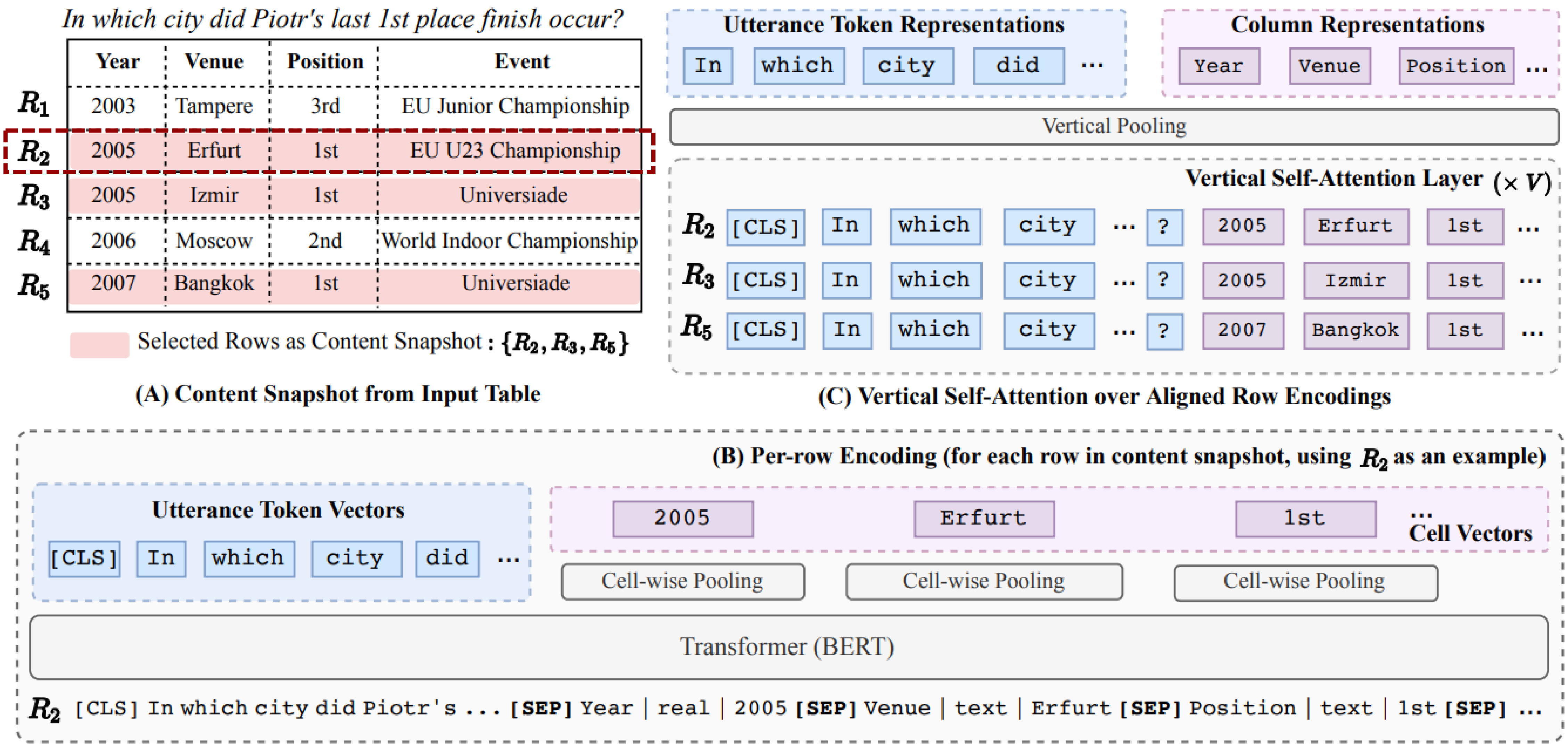




- How to encode tabular data?

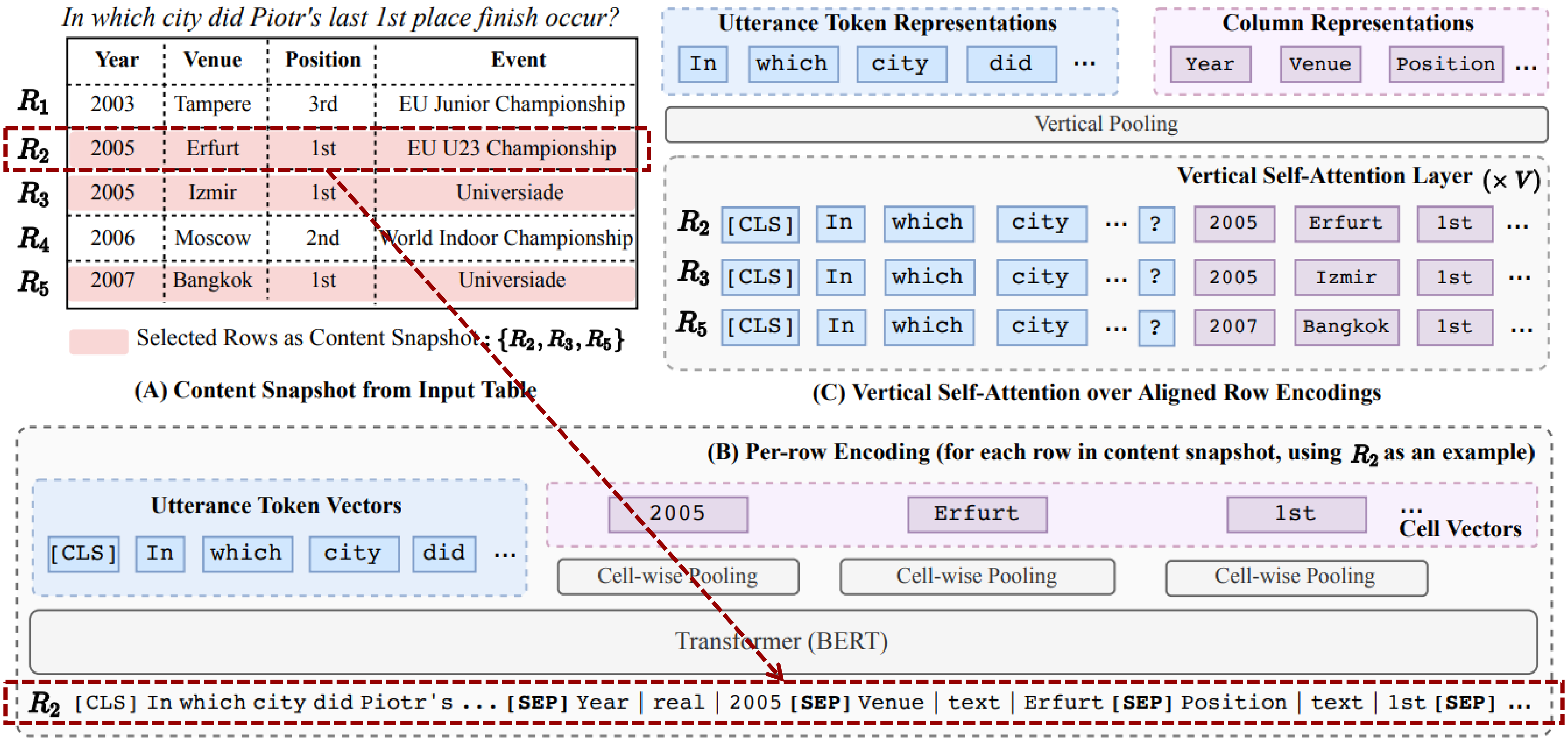
Seq2SQL: Generating Structured Queries from Natural Language using Reinforcement Learning. arXiv 2017.

TaBERT: Encoding Process



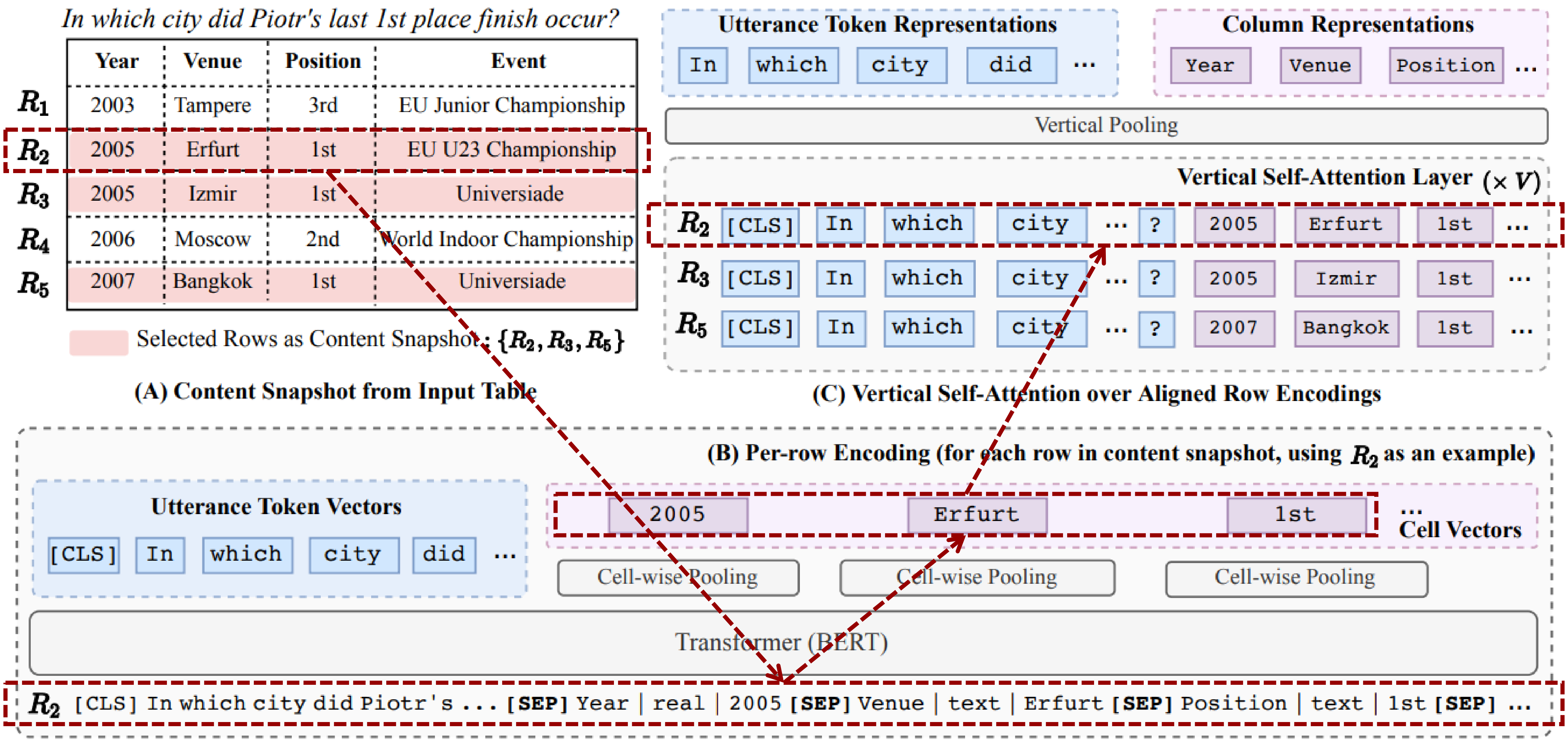
TaBERT: Learning Contextual Representations for Natural Language Utterances and Structured Tables. ACL 2020.

TaBERT: Encoding Process



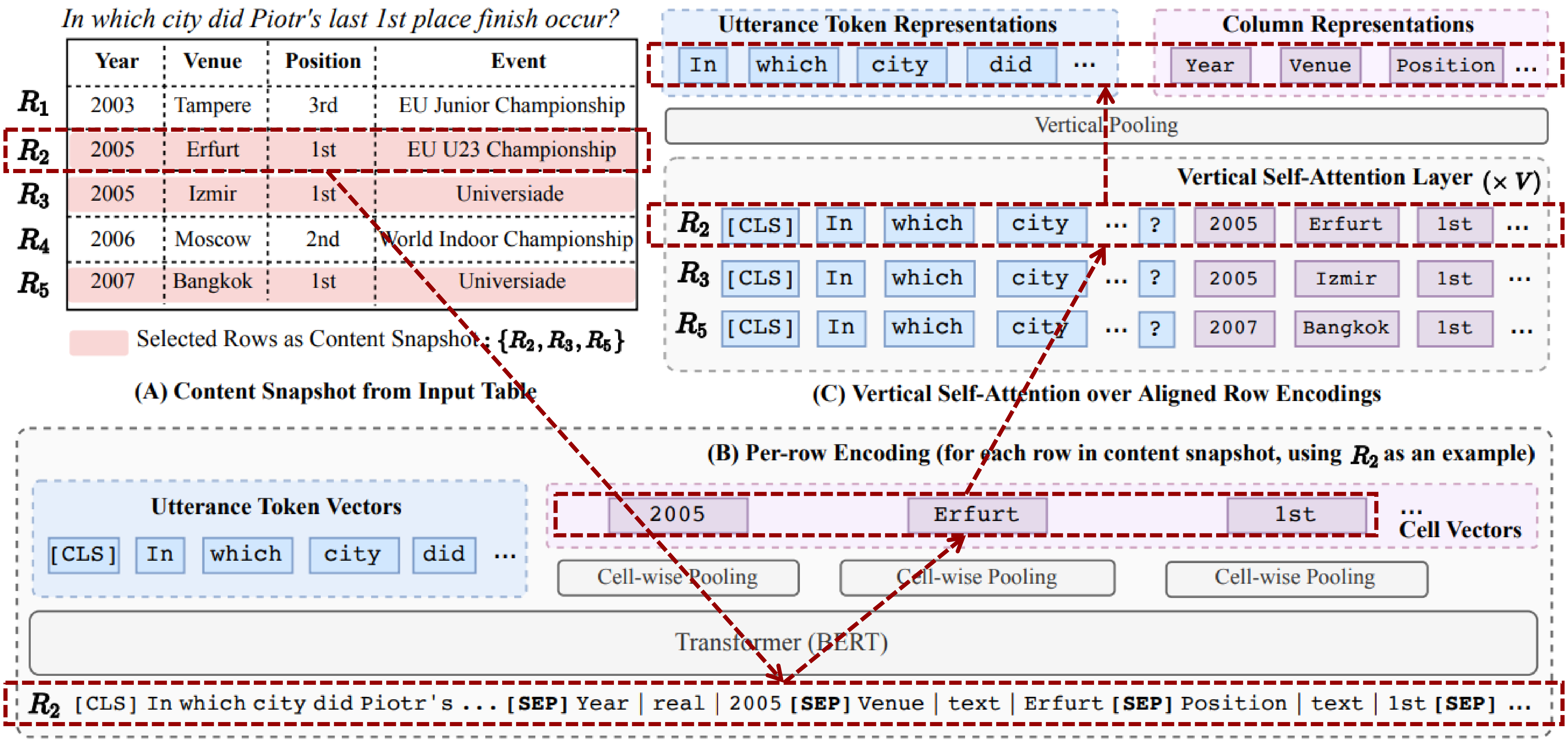
TaBERT: Learning Contextual Representations for Natural Language Utterances and Structured Tables. ACL 2020.

TaBERT: Encoding Process



TaBERT: Learning Contextual Representations for Natural Language Utterances and Structured Tables. ACL 2020.

TaBERT: Encoding Process



TaBERT: Learning Contextual Representations for Natural Language Utterances and Structured Tables. ACL 2020.

TaBERT: Pre-training Tasks

- Masked Column Prediction (MCP): Randomly select 20% of the columns in an input table, masking their names (e.g., Year) and data types (e.g., real) in each row linearization. The model needs recover the names and data types of masked columns using column representations.
- Cell Value Recovery (CVR): For each masked column, the model predicts the original tokens of each cell using its cell representation.

- TaBERT-Base and TaBERT-Large are pre-trained from uncased BERT-Base and BERT-Large, respectively.

https://github.com/facebookresearch/TaBERT

README Code of conduct License Security

TaBERT: Learning Contextual Representations for Natural Language Utterances and Structured Tables

This repository contains source code for the TaBERT model, a pre-trained language model for learning joint

TaBERT: Learning Contextual Representations for Natural Language Utterances and Structured Tables. ACL 2020.

How to select the rows?

- "K = 3": select the top-K rows in the input table that have the highest n-gram overlap ratio with the utterance
- "K = 1": create a synthetic row by selecting the cell values from each column that have the highest n-gram overlap with the utterance
  - Motivation: include as much information relevant to the utterance as possible

u: How many years before was the film Bacchae out before the Watermelon?

Input to TaBERT$_{\text{Large}}$ (K = 3) ▷ *Content Snapshot with Three Rows*

Film	**Year**	**Function**	**Notes**
The Bacchae	2002	Producer	Screen adaptation of...
The Trojan Women	2004	Producer/Actress	Documutary film...
The Watermelon	2008	Producer	Oddball romantic comedy...

Input to TaBERT$_{\text{Large}}$ (K = 1) ▷ *Content Snapshot with One Synthetic Row*

Film	**Year**	**Function**	**Notes**
The Watermelon	2013	Producer	Screen adaptation of...

Performance of TaBERT

Previous Systems on WikiTableQuestions				
Model	DEV		TEST	
Pasupat and Liang (2015)	37.0		37.1	
Neelakantan et al. (2016)	34.1		34.2	
Ensemble 15 Models	37.5		37.7	
Zhang et al. (2017)	40.6		43.7	
Dasigi et al. (2019)	43.1		44.3	
Agarwal et al. (2019)	43.2		44.1	
Ensemble 10 Models	–		46.9	
Wang et al. (2019b)	43.7		44.5	
Our System based on MAPO (Liang et al., 2018)				
	DEV	Best	TEST	Best
Base Parser†	42.3 ±0.3	42.7	43.1 ±0.5	43.8
$w/$ BERT$_{\text{Base}}$ ($K = 1$)	49.6 ±0.5	50.4	49.4 ±0.5	49.2
– content snapshot	49.1 ±0.6	50.0	48.8 ±0.9	50.2
$w/$ TaBERT$_{\text{Base}}$ ($K = 1$)	51.2 ±0.5	51.6	50.4 ±0.5	51.2
– content snapshot	49.9 ±0.4	50.3	49.4 ±0.4	50.0
$w/$ TaBERT$_{\text{Base}}$ ($K = 3$)	51.6 ±0.5	52.4	51.4 ±0.3	51.3
$w/$ BERT$_{\text{Large}}$ ($K = 1$)	50.3 ±0.4	50.8	49.6 ±0.5	50.1
$w/$ TaBERT$_{\text{Large}}$ ($K = 1$)	51.6 ±1.1	52.7	51.2 ±0.9	51.5
$w/$ TaBERT$_{\text{Large}}$ ($K = 3$)	**52.2** ±0.7	**53.0**	**51.8** ±0.6	**52.3**

Top-ranked Systems on Spider Leaderboard		
Model	DEV. ACC.	
Global–GNN (Bogin et al., 2019a)	52.7	
EditSQL + BERT (Zhang et al., 2019a)	57.6	
RatSQL (Wang et al., 2019a)	60.9	
IRNet + BERT (Guo et al., 2019)	60.3	
+ Memory + Coarse-to-Fine	61.9	
IRNet V2 + BERT	63.9	
RyanSQL + BERT (Choi et al., 2020)	**66.6**	
Our System based on TranX (Yin and Neubig, 2018)		
	Mean	Best
$w/$ BERT$_{\text{Base}}$ ($K = 1$)	61.8 ±0.8	62.4
– content snapshot	59.6 ±0.7	60.3
$w/$ TaBERT$_{\text{Base}}$ ($K = 1$)	63.3 ±0.6	64.2
– content snapshot	60.4 ±1.3	61.8
$w/$ TaBERT$_{\text{Base}}$ ($K = 3$)	63.3 ±0.7	64.1
$w/$ BERT$_{\text{Large}}$ ($K = 1$)	61.3 ±1.2	62.9
$w/$ TaBERT$_{\text{Large}}$ ($K = 1$)	64.0 ±0.4	64.4
$w/$ TaBERT$_{\text{Large}}$ ($K = 3$)	**64.5** ±0.6	**65.2**

Ablation Studies

Linearization

Cell Linearization Template	WikiQ.	Spider
Pretrained TaBERT$_{\text{Base}}$ Models ($K = 1$)		
`Column Name`	49.6 ±0.4	60.0 ±1.1
`Column Name` | `Type`† (−content snap.)	49.9 ±0.4	60.4 ±1.3
`Column Name` | `Type` | `Cell Value`†	51.2 ±0.5	63.3 ±0.6
BERT$_{\text{Base}}$ Models		
`Column Name` (Hwang et al., 2019)	49.0 ±0.4	58.6 ±0.3
`Column Name is Cell Value` (Chen19)	50.2 ±0.4	63.1 ±0.7

Pre-training Tasks

Learning Objective	WikiQ.	Spider
MCP only	51.6 ±0.7	62.6 ±0.7
MCP + CVR	51.6 ±0.5	63.3 ±0.7

Take-Away Messages

- Feed natural language utterances, column names, column types, and cell values together into a BERT model for MLM
- Use masked column prediction and cell value recovery to replace masked token prediction
- Good column and cell representations benefit text-to-SQL generation

- Limitations
  - Only test the model performance in the text-to-SQL generation task.
  - The model should be able to perform missing cell value prediction (one of the pre-training tasks) as well, but the performance is unknown.
    - How about other table tasks?

Agenda

- TaBERT: Masked Language Modeling for Table Cell Representation
- TableLlama: Instruction Tuning for Table QA
- UniHGKR: Instruction Tuning for Table Retrieval
- TabPFN: A Tabular Foundation Model for Missing Value Prediction

Instruction-Tuning LLaMA for Table Tasks

(a) Column Type Annotation

1958 Nippon Professional Baseball season

Central League

Stat	Player	Team	Total
Wins	Masaichi Kaneda	Kokutetsu Swallows	31
Losses	Noboru Akiyama	Taiyo Whales	23
Earned run average	Masaichi Kaneda	Kokutetsu Swallows	1.3
Strikeouts	Masaichi Kaneda	Kokutetsu Swallows	311
Innings pitched	Motoshi Fujita Noboru Akiyama	Yomiuri Giants Taiyo Whales	359

Instruction:
This is a **column type annotation** task. The goal for this task is to choose the correct types for one selected column of the table from the given candidates. The Wikipedia page, ... provide important information for choosing the correct column types.

Input:
[TLE] The Wikipedia page is about 1958 Nippon Professional Baseball season. The Wikipedia section is about Central League. The table caption is Pitching leaders. [TAB] col: | stat | player | ... [SEP] row 1: | Wins | Masaichi Kaneda | ... [SEP] row 2: | Losses | ...

Question:
The column 'player' contains the following entities: ‹Masaichi Kaneda›, ‹Noboru Akiyama›, ... The column type candidates are: tv.tv_producer, astronomy.star_system_body, ... What are the correct column types for this column (column name: player; entities: ‹Masaichi Kaneda›, ... , etc)?

Response: sports.pro_athlete, baseball.baseball_player, people.person.

(b) Row Population

NBA Conference Finals

Eastern Conference Finals

Year	Champion	Coach	Result	Runner-up
1971	Baltimore Bullets	Gene Shue	4–3	New York Knicks

Instruction:
This is a table **row population** task. The goal of this task is to populate the possible entities of the selected column for a table, given the Wikipedia page title, ... You will be given a list of entity candidates. Please rank them so that the most likely entities come first.

Input:
[TLE] The Wikipedia page is about NBA conference finals. The Wikipedia section is about eastern conference finals. The table headers are: | year | champion | ... You need to populate the column: year. [SEED] The seed entity is ‹1971_NBA_playoffs›.

Question:
The entity candidates are: ‹2003_NBA_playoffs›, ‹1982-83_Washington_Bullets_season›, ‹2004_NBA_playoffs›, ‹Philadelphia_76ers›, ‹1983-84_Washington_Bullets_season›, ‹1952_NBA_playoffs›, ...

Response: ‹1972_NBA_playoffs›, ‹1973_NBA_playoffs›, ‹1974_NBA_playoffs›, ‹1975_NBA_playoffs›, ‹1976_NBA_playoffs›, ...

(c) Hierarchical Table QA

Table: Department of defense obligations for research, development, test, and evaluation, by agency: 2015-18

agency	2015	2016	2017	2018
department of defense				
rdt&e	61513.5	69306.1	70866.1	83725
total research	6691.5	7152	7178	7652.7
basic research	2133.4	2238.7	2110.1	2389.9
defense advanced research projects agency				
rdt&e	2815.6	2933.4	2894.5	3018.2
total research	1485	1535.9	1509.4	1680
basic research	359.8	378.1	391.2	458.4

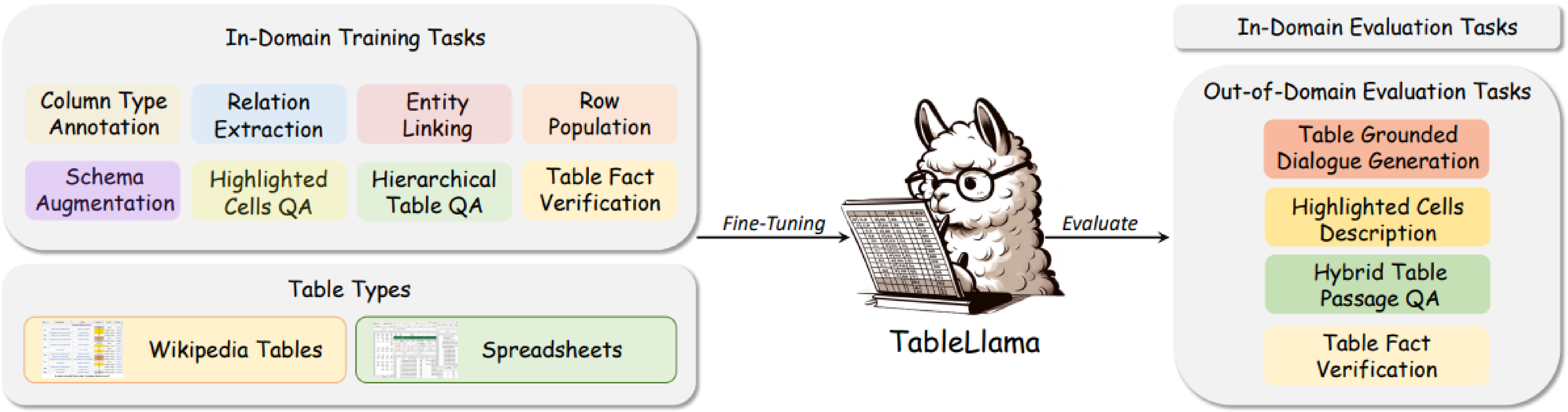
Instruction:
This is a **hierarchical table question answering** task. The goal for this task is to answer the given question based on the given table. The table might be hierarchical.

Input:
[TLE] The table caption is department of defense obligations for research, development, test, and evaluation, by agency: 2015-18. [TAB] | agency | 2015 | 2016 | ... [SEP] | department of defense | department of defense | ... [SEP] | rdt&e | 61513.5 | ... [SEP] | total research | 6691.5 | ... [SEP] | basic research | 2133.4 | ... [SEP] | defense advanced research projects agency | ...

Question:
How many dollars are the difference for basic research of defense advanced research projects agency increase between 2016 and 2018?

Response: 80.3.

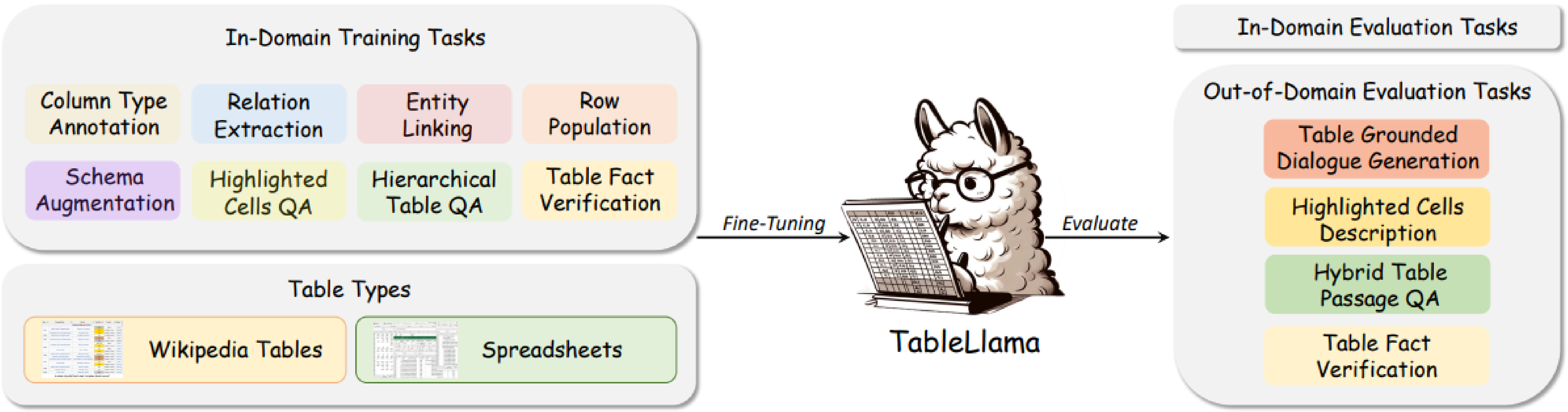
Instruction-Tuning LLaMA for Table Tasks



- Relation Extraction: Predict the correct relations between two selected columns of the table
- Entity Linking: Link the selected entity mention in the table cells to the entity in the knowledge base
- Schema Augmentation: Populate the possible headers for a table, given the table caption and the seed table header

TableLlama: Towards Open Large Generalist Models for Tables. NAACL 2024.

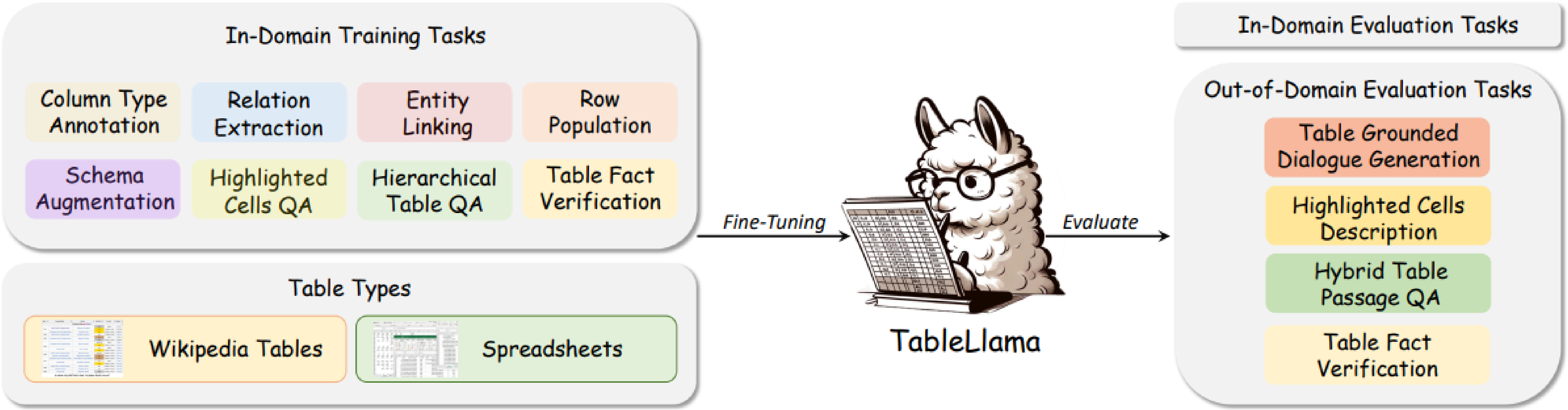
Instruction-Tuning LLaMA for Table Tasks



- Highlighted Cells QA: Answer the given question based on the given table and the highlighted cells
- Table Fact Verification: Distinguish whether the given statement is entailed or refuted by the given table
- Table QA: Answer the question given the table

TableLlama: Towards Open Large Generalist Models for Tables. NAACL 2024.

Instruction-Tuning LLaMA for Table Tasks



- Table Grounded Dialogue Generation: Generate response based on the given dialogue history and the given table
- Highlighted Cells description: Generate the language description given table cells
- Hybrid Table Passage QA: Answer the question given tables and passages

TableLlama: Towards Open Large Generalist Models for Tables. NAACL 2024.

The TableInstruct Dataset

Task Category	Task Name	Dataset	In-domain	#Train (Table/Sample)	#Test (Table/Sample)	Input Token Length min	max	median
Table Interpretation	Col Type Annot.	TURL (Deng et al., 2020)	Yes	397K/628K	1K/2K	106	8192	2613
	Relation Extract.		Yes	53K/63K	1K/2K	2602	8192	3219
	Entity Linking		Yes	193K/1264K	1K/2K	299	8192	4667
Table Augmentation	Schema Aug.	TURL (Deng et al., 2020)	Yes	288K/288K	4K/4K	160	1188	215
	Row Pop.		Yes	286K/286K	0.3K/0.3K	264	8192	1508
Question Answering	Hierarchical Table QA	HiTab (Cheng et al., 2022b)	Yes	3K/7K	1K/1K	206	5616	978
	Highlighted Cells QA	FeTaQA (Nan et al., 2022)	Yes	7K/7K	2K/2K	261	5923	740
	Hybrid Table QA	HybridQA (Chen et al., 2020b)	No	–	3K/3K	248	2497	675
	Table QA	WikiSQL (Zhong et al., 2017)	No	–	5K/16K	198	2091	575
	Table QA	WikiTQ (Pasupat and Liang, 2015)	No	–	0.4K/4K	263	2688	709
Fact Verification	Fact Verification	TabFact (Chen et al., 2020a)	Yes	16K/92K	2K/12K	253	4975	630
		FEVEROUS (Aly et al., 2021)	No	–	4K/7K	247	8192	648
Dialogue Generation	Table Grounded Dialogue Generation	KVRET (Eric et al., 2017)	No	–	0.3K/0.8K	187	1103	527
Data-to-Text	Highlighted Cells Description	ToTTo (Parikh et al., 2020)	No	–	7K/8K	152	8192	246

https://huggingface.co/datasets/osunlp/TableInstruct

Datasets: osunlp/TableInstruct like 26 Follow OSU NLP Group 91

Languages: English Size: 1M<n<10M ArXiv: arxiv:2311.09206 License: cc-by-4.0

Dataset card Data Studio Files and versions Community

How to handle large tables (i.e., long context)?

- Use LongLoRA to fine-tune LLaMA-2

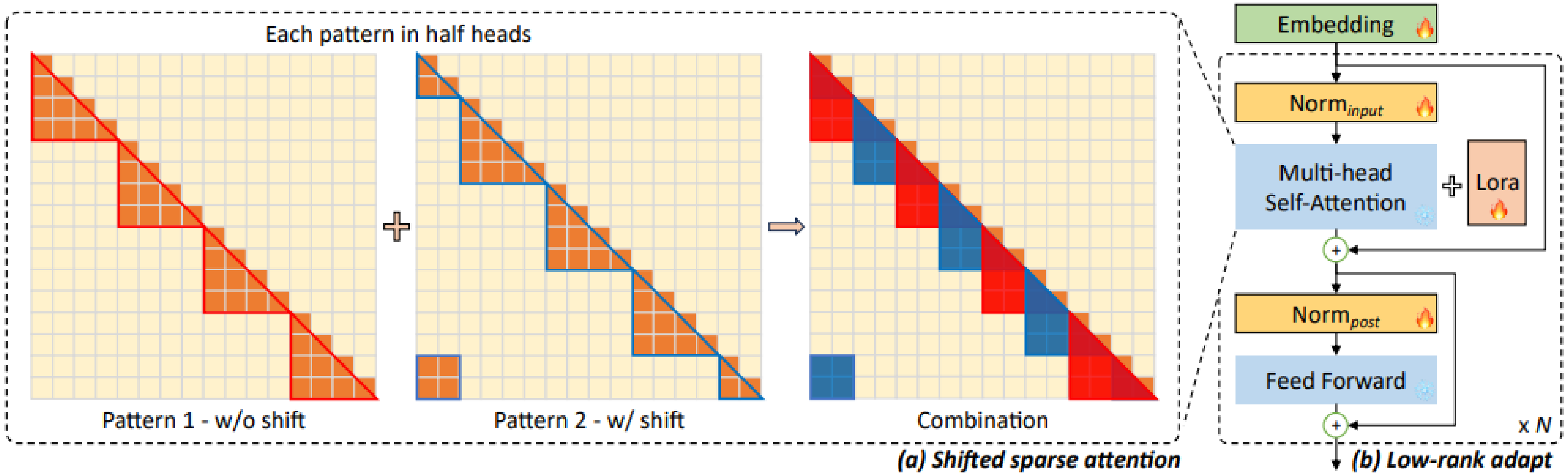


Figure 2: **Overview of LongLoRA**. We introduce Shifted Sparse Attention (S^2-Attn) during fine-tuning. The trained model retains original standard self-attention at inference time. In addition to training LoRA weights in linear layers, LongLoRA further makes embedding and normalization layers trainable. This extension is pivotal for context extension, and only introduces a minimal number of additional trainable parameters.

LongLoRA: Efficient Fine-tuning of Long-Context Large Language Models. ICLR 2024.

Performance of TableLlama: In-domain Tasks

- By simply fine-tuning a large language model on TableInstruct, TableLlama can achieve comparable or even better performance on almost all the tasks without any table pre-training or special table model architecture design.
- In particular, TableLlama displays advantages in table QA tasks.
- TableLlama achieves better performance on in-domain tasks compared with closed-source LLMs.

In-domain Evaluation						
Datasets	Metric	Base	TableLlama	SOTA	GPT-3.5	GPT-4§
Column Type Annotation	F1	3.01	94.39	**94.54***† (Deng et al., 2020)	30.88	31.75
Relation Extraction	F1	0.96	91.95	**94.91***† (Deng et al., 2020)	27.42	52.95
Entity Linking	Accuracy	31.80	**93.65**	84.90*† (Deng et al., 2020)	72.15	90.80
Schema Augmentation	MAP	36.75	**80.50**	77.55*† (Deng et al., 2020)	49.11	58.19
Row Population	MAP	4.53	58.44	**73.31***† (Deng et al., 2020)	22.36	53.40
HiTab	Exec Acc	14.96	**64.71**	47.00*† (Cheng et al., 2022a)	43.62	48.40
FeTaQA	BLEU	8.54	**39.05**	33.44 (Xie et al., 2022)	26.49	21.70
TabFact	Accuracy	41.65	82.55	**84.87*** (Zhao and Yang, 2022)	67.41	74.40

Performance of TableLlama: Out-of-domain Tasks

- By comparing with the base model, TableLlama can achieve 5-44 points gain on 6 out-of-domain datasets, which demonstrates TableInstruct can enhance the model's generalization ability.

Out-of-domain Evaluation							
Datasets	Metric	Base	TableLlama	SOTA	Δ_{Base}	GPT-3.5	GPT-4§
FEVEROUS	Accuracy	29.68	73.77	85.60 (Tay et al., 2022)	+44.09	60.79	71.60
HybridQA	Accuracy	23.46	39.38	65.40* (Lee et al., 2023)	+15.92	40.22	58.60
KVRET	Micro F1	38.90	48.73	67.80 (Xie et al., 2022)	+9.83	54.56	56.46
ToTTo	BLEU	10.39	20.77	48.95 (Xie et al., 2022)	+10.38	16.81	12.21
WikiSQL	Accuracy	15.56	50.48	92.70 (Xu et al., 2023b)	+34.92	41.91	47.60
WikiTQ	Accuracy	29.26	35.01	57.50† (Liu et al., 2022)	+5.75	53.13	68.40

Ablation Study

- The model trained on table-based QA tasks generalizes better than that trained on other tasks.
- Incorporating other tasks helps enhance the model's underlying generalization ability within the same task category.
- Individually fine-tuning models on tasks that are highly different from others tends to make models overfit and hardly generalize to others.

Training Data	In-domain								Out-of-domain					
	ColType	RelExtra	EntLink	ScheAug	RowPop	HiTab	FeTaQA	TabFact	FEVER.	HybridQA	KVRET	ToTTo	WikiSQL	WikiTQ
	F1	F1	Acc	MAP	MAP	Acc	BLEU	Acc	Acc	Acc	Micro F1	BLEU	Acc	Acc
Base	3.01	0.96	31.80	36.75	4.53	14.96	8.54	41.65	29.68	23.46	38.90	10.39	15.56	29.26
ColType	94.32	0	0	0	0	0.13	0.52	0	0	0	0	1.11	0.35	0.21
RelExtra	45.69	**93.96**	0.45	8.72	0.99	7.26	1.44	0	2.38	8.17	5.90	5.60	7.02	9.58
EntLink	0.86	0.03	88.45	2.31	0.94	5.37	4.79	0	39.04	3.06	0	1.76	3.42	7.07
ScheAug	-	-	-	80.00	-	-	-	-	-	-	-	-	-	-
RowPop	-	-	-	-	53.86	-	-	-	-	-	-	-	-	-
HiTab	0.20	0.14	7.15	40.81	5.45	63.19	2.07	49.46	46.81	24.70	38.70	2.45	32.86	27.97
FeTaQA	0	0.40	0	30.23	0.15	19.57	38.69	1.20	1.21	33.79	**50.69**	**23.57**	13.79	27.12
TabFact	0	0	0	0	0	0	0	74.87	56.15	0	0	0	0	0
TableInstruct	**94.39**	91.95	**93.65**	**80.50**	**58.44**	**64.71**	**39.05**	**82.55**	**73.77**	**39.38**	48.73	20.77	**50.48**	**35.01**

Take-Away Messages

- Use various table-related tasks to instruction-tune a LLaMA model for jointly dealing with text and tables
- Achieve comparable or even better performance on almost all the tasks without any special table model architecture design, particularly in table QA tasks
- Outperform closed-source LLMs in in-domain tasks

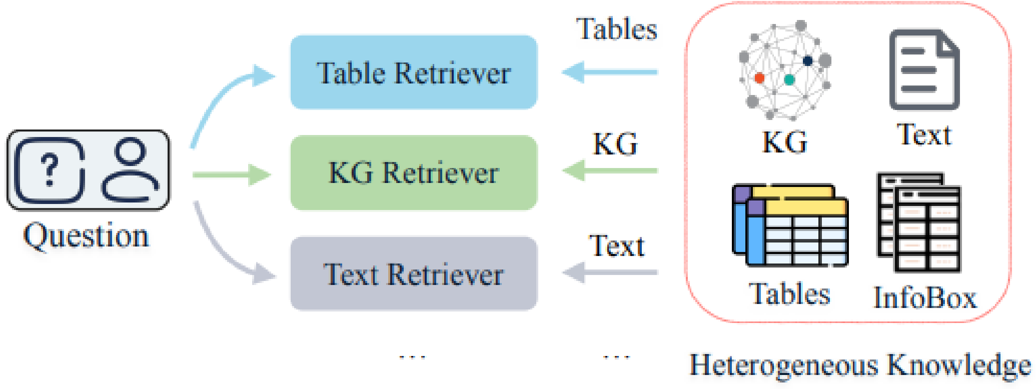
- Limitations
  - Still consistently underperform closed-source LLMs (e.g., GPT-3.5, GPT-4) in out-of-domain tasks
  - Need tricks (i.e., LongLoRA) to handle large tables; may not be able to handle super gigantic or a large collection of tables

Agenda

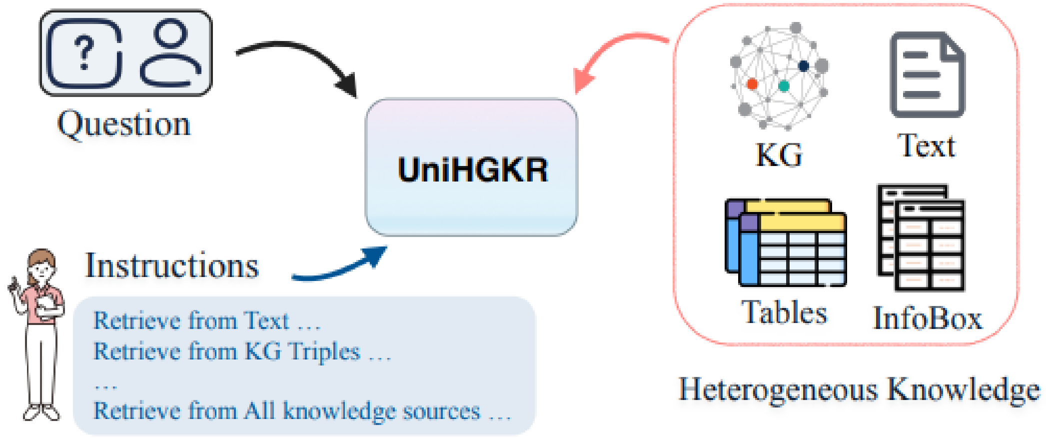
- TaBERT: Masked Language Modeling for Table Cell Representation
- TableLlama: Instruction Tuning for Table QA
- UniHGKR: Instruction Tuning for Table Retrieval
- TabPFN: A Tabular Foundation Model for Missing Value Prediction

What if you have many information sources for QA?

- Structured retrieval-augmented generation (RAG)
  - When you have multiple tables OR infoboxes OR KG entities/relations, you need to perform retrieval on a single data type.
  - When you have multiple tables AND infoboxes AND KG entities/relations, you need to perform unified retrieval on multiple data types.

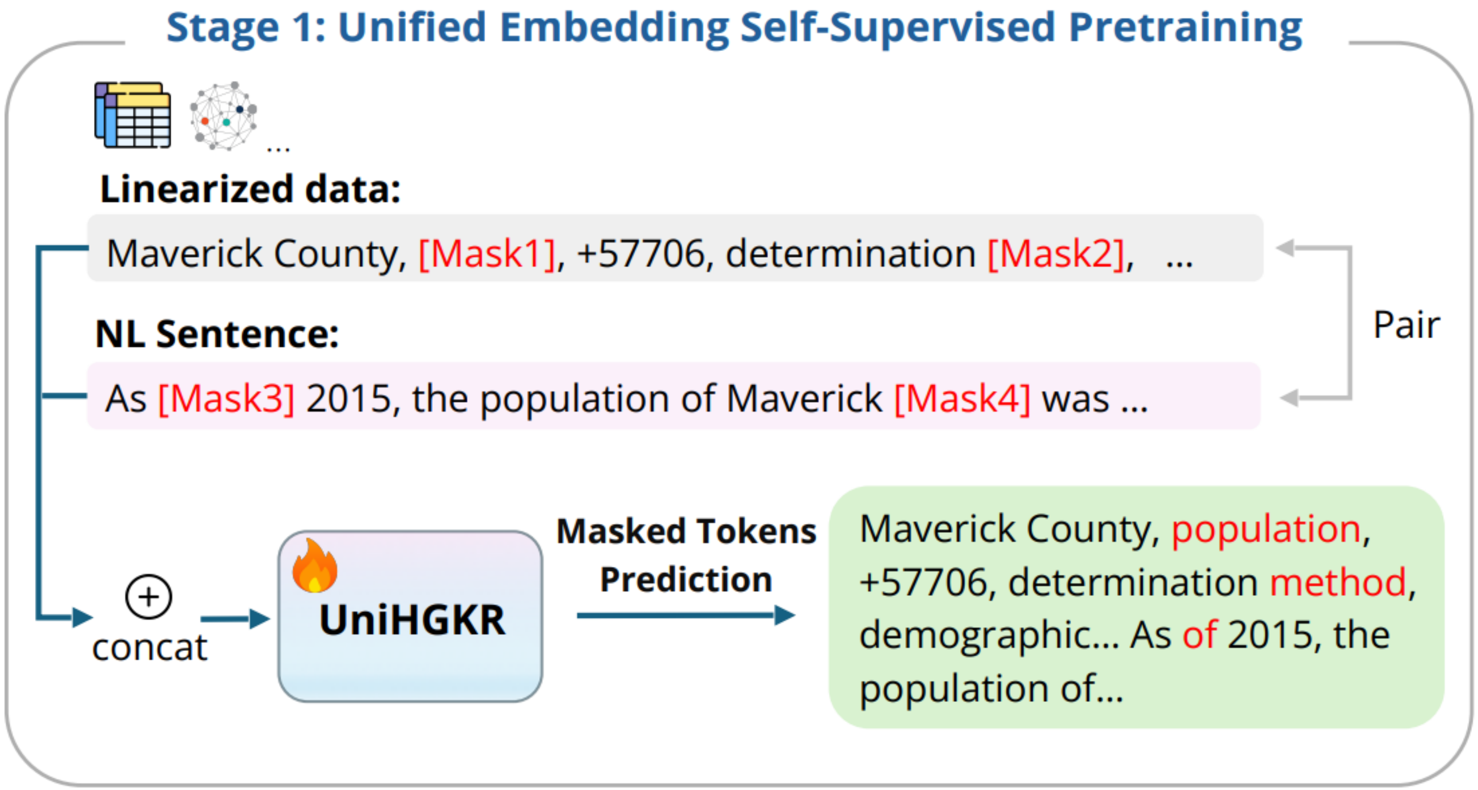


(a) Conventional retrievers focus on a single data type.



(b) UniHGKR aims to retrieval from any heterogeneous knowledge source.

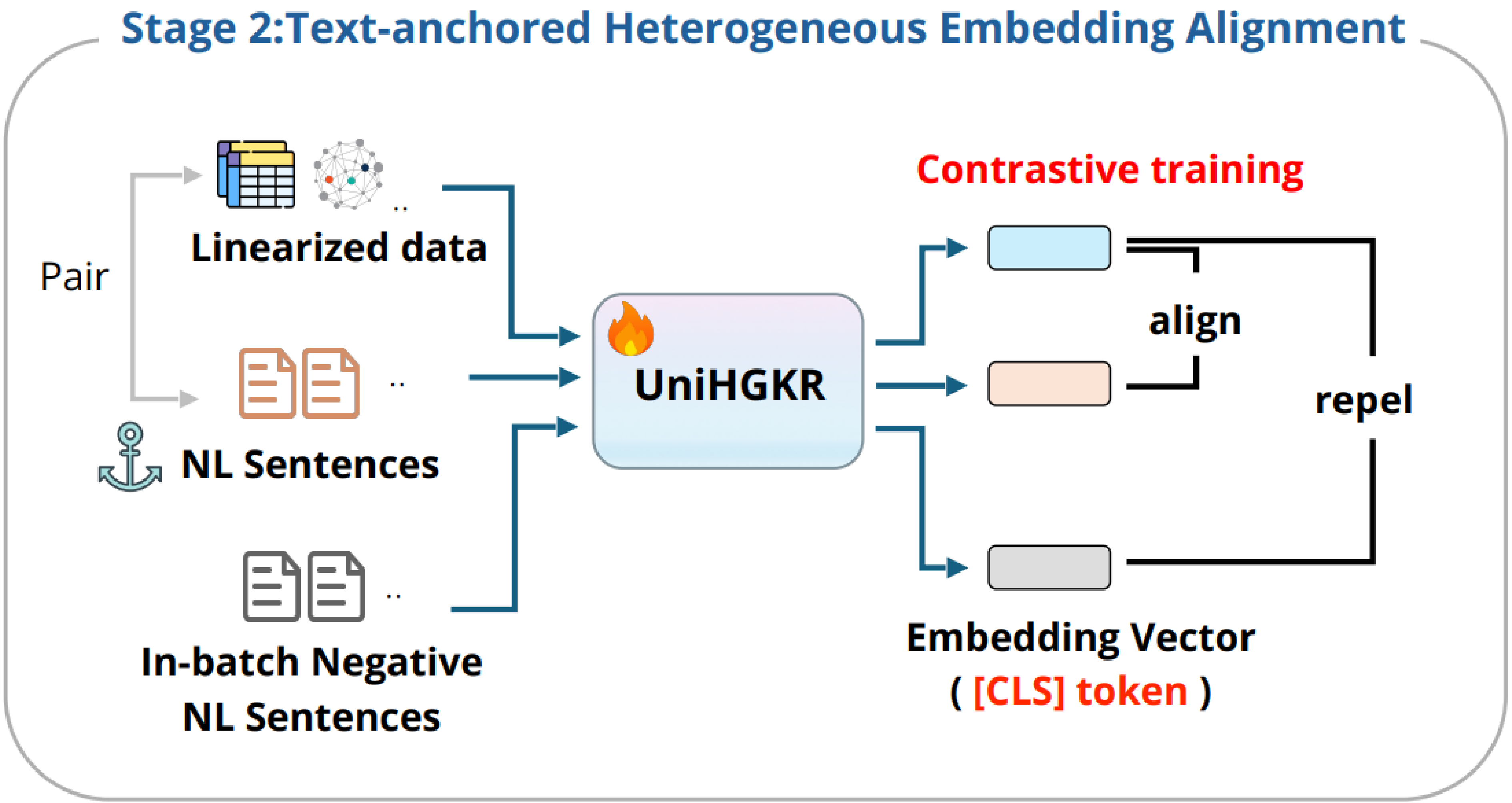
UniHGKR: Pre-training a Unified Retriever



Types	Avg. length	Count	Percentage
Text	19.86	5,916,596	57.74%
KG	11.40	2,214,854	21.61%
Table	20.32	1,043,105	10.18%
Infobox	11.05	1,072,440	10.47%
Sum	17.18	10,246,995	100.00%

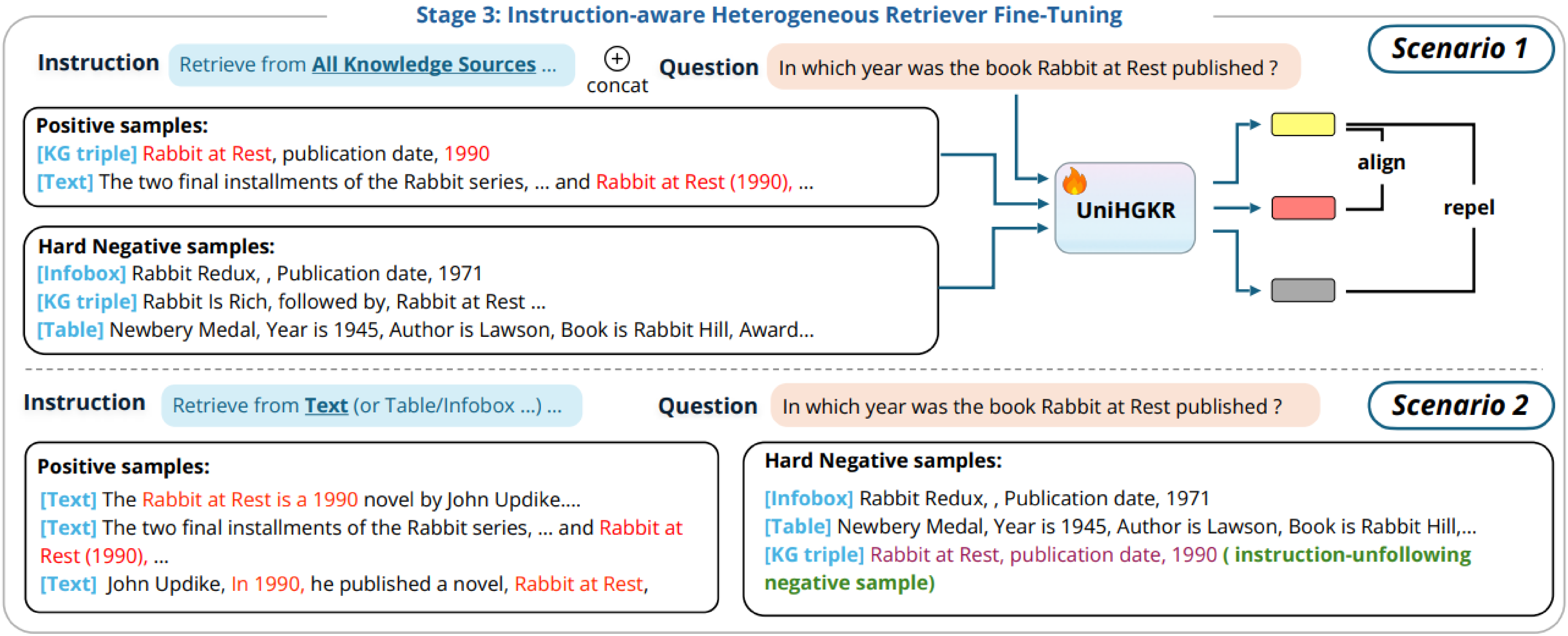
UniHGKR: Unified Instruction-aware Heterogeneous Knowledge Retrievers. NAACL 2025.

UniHGKR: Pre-training a Unified Retriever



UniHGKR: Unified Instruction-aware Heterogeneous Knowledge Retrievers. NAACL 2025.

UniHGKR: Pre-training a Unified Retriever



UniHGKR: Unified Instruction-aware Heterogeneous Knowledge Retrievers. NAACL 2025.

Data-Text Pair Collection

Data-Text Pairs Collecting

GPT-4o-mini ← Prompt

Infobox Entry: Love Hard (film), Love Hard, Production company, Wonderland Sound and Vision.

NL Sentence: The film "Love Hard" was produced by the production company Wonderland Sound and Vision.

Table Entry: New York City Ballet, Name **is** Tyler Angle, Nationality **is** United States, Training **is** Allegheny Ballet Academy School of American Ballet, Joined NYCB **is** 2004, Promoted to Principal **is** 2009.

NL Sentence: Tyler Angle, a dancer from the United States, trained at the Allegheny Ballet Academy and the School of American Ballet. He joined the New York City Ballet in 2004 and was promoted to Principal in 2009.

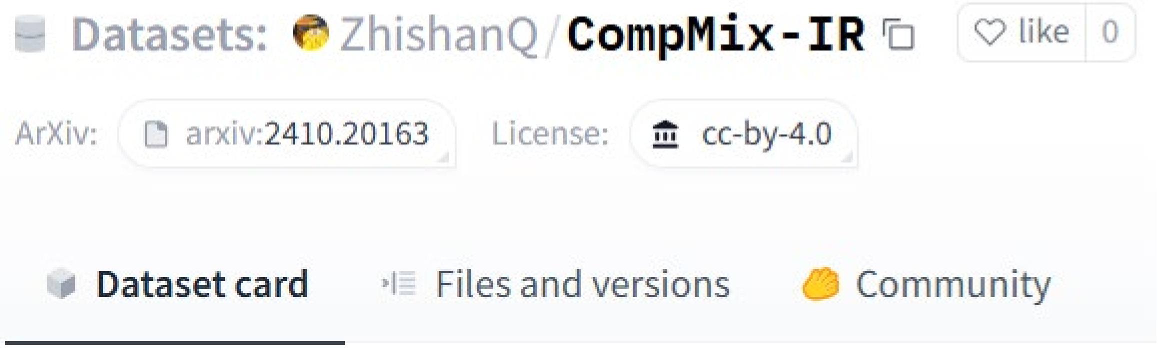
KG Triple: Maverick County, population, "+57706", determination method, demographic balance, point in time, "2015'

NL Sentence: As of 2015, the population of Maverick County was approximately 57,706, determined through demographic balance.

Instructions

Template	Given a question in the [domain] domain, retrieve relevant evidence to answer the question from the [source].
[domain] options:	books, movies, music, television series, and football
[source] options:	All Knowledge Sources, Knowledge Graph Triples, Infobox, Table, and Text
Example 1:	Given a question in the music domain, retrieve ... from Knowledge Graph Triples.
Example 2:	Given a question in the football domain, retrieve relevant ... from All Knowledge Sources.
Paraphrased 1:	For a question related to the music domain, find pertinent information from Knowledge Graph Triples.
Paraphrased 2:	For a question in the football domain, extract helpful ... to address it from All Knowledge Sources.

https://huggingface.co/datasets/ZhishanQ/CompMix-IR



Performance of UniHGKR: Retrieval with a "Small" Model

			Retrieval Scenario 1 (instruction I_{All})				Retrieval Scenario 2 (instruction I_{τ})			
Method	**Size**	**Ins**	**Hit@5**	**Hit@10**	**Hit@100**	**MRR@100**	**KG-Hit**	**Text-Hit**	**Table-Hit**	**Info-Hit**
BM25	-	✗	11.51	17.40	52.39	8.54	24.20	34.55	8.50	19.79
DPR	109M	✗	24.89	36.32	78.76	17.51	49.13	63.68	15.63	41.57
Mpnet	109M	✗	26.23	37.99	82.67	18.46	63.02	61.11	18.96	52.1
GTR-T5-base	110M	✗	24.46	36.54	80.32	16.73	57.78	59.8	22.87	46.09
Contriever	109M	✗	28.58	40.70	83.79	20.07	62.26	<u>63.86</u>	18.63	55.64
SimLM	109M	✗	25.11	37.08	80.61	17.68	59.59	59.01	17.69	52.06
Instructor-base	110M	✓	24.86	36.22	81.55	17.80	65.63	50.25	16.82	53.36
Instructor-large	336M	✓	25.98	36.87	81.51	18.54	<u>68.78</u>	44.61	17.11	53.98
BGE	109M	✓	26.66	39.04	84.15	19.40	68.42	57.96	22.58	56.58
BERT-finetuned	109M	✗	24.46	35.38	78.51	17.04	57.63	54.67	17.55	48.41
UDT-retriever	109M	✗	24.96	35.49	76.52	18.24	66.10	62.48	25.90	<u>57.05</u>
UniK-retriever	109M	✗	<u>30.68</u>	<u>43.42</u>	<u>85.20</u>	<u>21.22</u>	67.40	63.21	<u>26.74</u>	56.04
UniHGKR-base	109M	✓	**32.38**	**45.55**	**85.75**	**22.57**	**75.43**	**70.30**	**41.24**	**66.21**
▲ Relative gain			+5.54%	+4.91%	+0.65%	+6.36%	+9.67%	+10.08%	+54.23%	+16.06%

Performance of UniHGKR: Retrieval with a “Large” Model

	Retrieval Scenario 1 (instruction I_{All})				Retrieval Scenario 2 (instruction I_{τ})			
Method	**Hit@5**	**Hit@10**	**Hit@100**	**MRR@100**	**KG-Hit**	**Text-Hit**	**Table-Hit**	**Info-Hit**
UniHGKR-base	32.38	45.55	85.75	22.57	75.43	70.30	41.24	66.21
E5-mistral-7B	31.3	43.49	83.36	22.97	69.03	41.46	33.03	62.92
LLARA-passage	37.45	51.59	86.61	26.11	68.23	<u>70.48</u>	<u>37.88</u>	60.64
LLARA-finetuned	<u>42.19</u>	<u>55.35</u>	<u>87.81</u>	<u>30.83</u>	<u>74.38</u>	69.86	36.40	<u>64.40</u>
UniHGKR-7B	**49.78**	**59.23**	**88.21**	**38.20**	**81.80**	**76.05**	**49.57**	**73.88**
▲Relative gain	+17.99%	+7.01%	+0.46%	+23.91%	+9.98%	+7.90%	+30.86%	+14.72%

Performance of UniHGKR: Retrieval-Augmented QA

Methods	Retriever	Reader	P@1	MRR
BM25+FiD	BM25	FiD	25.3	27.5
QuReTeC	QuReTeC	FiD	28.2	28.9
CONVINSE	CLOCQ+BM25	FiD	34.3	37.8
EXPLAIGNN	CLOCQ+BM25	GNN	<u>40.6</u>	<u>47.1</u>
Ours	**UniHGKR-base**	FiD	42.4	46.6
▲Abs. gain			+8.10	+8.80
	UniHGKR-7B	FiD	**46.5**	**51.4**
▲Abs. gain			+12.20	+13.60
▲SOTA gain			**+5.90**	**+4.30**

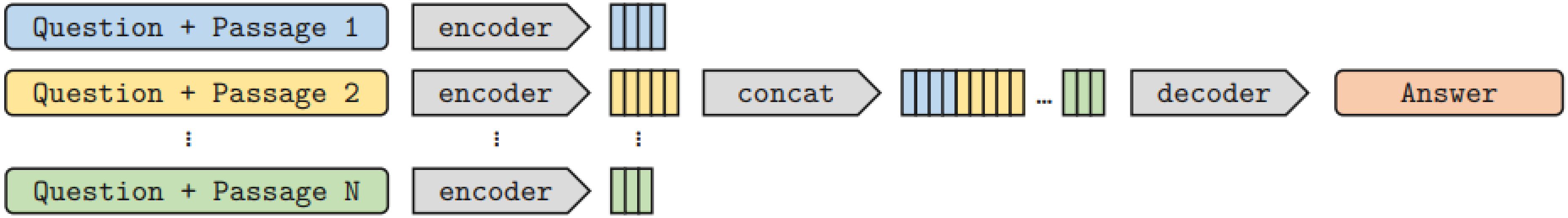


Figure 2: Architecture of the Fusion-in-Decoder method.

Leveraging Passage Retrieval with Generative Models for Open Domain Question Answering. EACL 2021.

Take-Away Messages

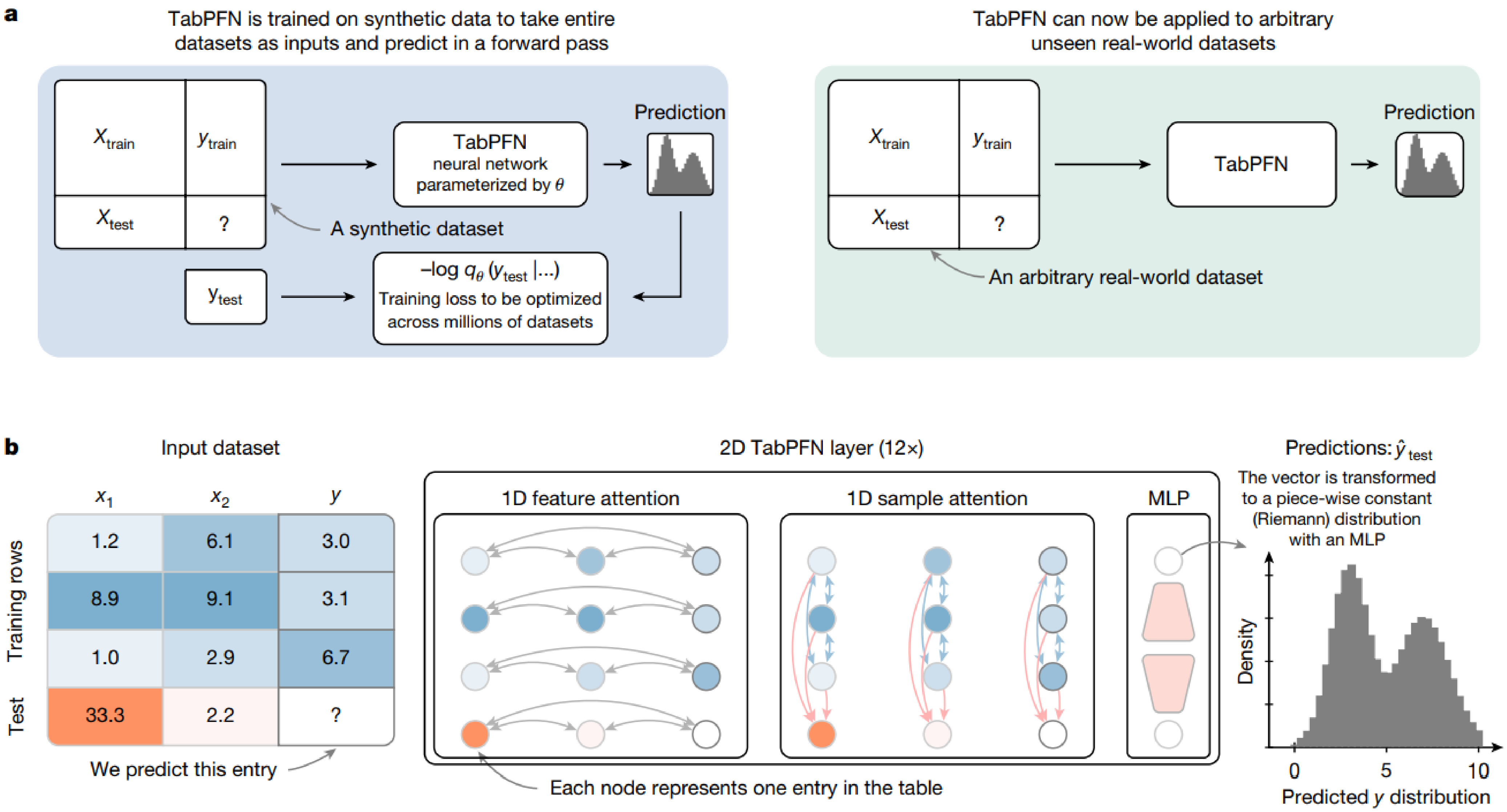
- Use various table/infobox/KG-text pairs to instruction-tune a retriever for retrieving heterogeneous knowledge to facilitate QA
- Outperform retrieval baselines in (1) retrieving evidence from all types of knowledge; and (2) retrieving type-specific evidence with different model sizes
- Benefit open-domain QA

- Limitations
  - Users might want to instruct the retriever to return a combination of evidence from multiple knowledge sources, such as text and tables.
  - More modalities such as image, audio and interleaved image and text can be considered and incorporated, possibly using the Mixture-of-Experts architecture.

Agenda

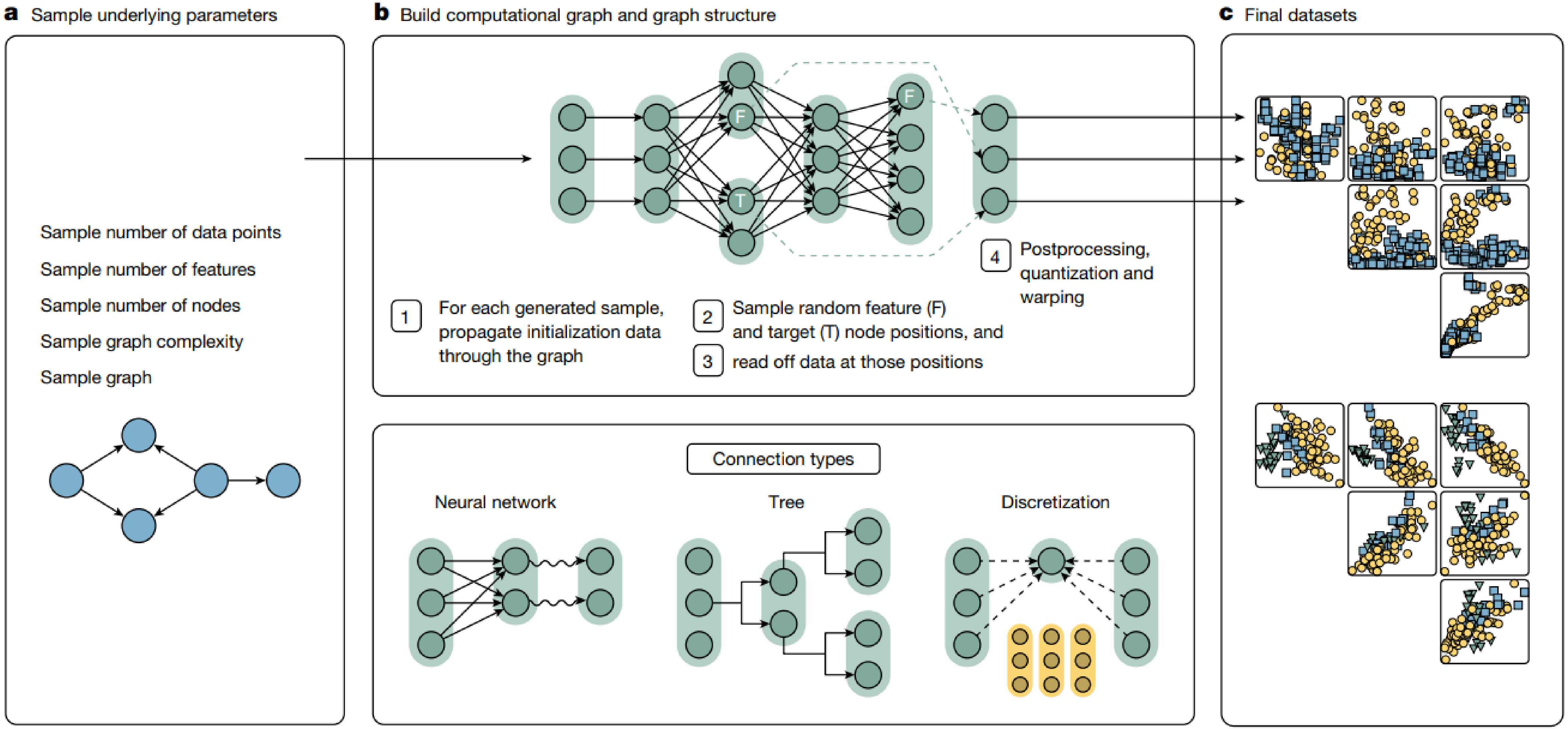
- TaBERT: Masked Language Modeling for Table Cell Representation
- TableLlama: Instruction Tuning for Table QA
- UniHGKR: Instruction Tuning for Table Retrieval
- TabPFN: A Tabular Foundation Model for Missing Value Prediction

Pre-training a Tabular Foundation Model



Accurate Predictions on Small Data with a Tabular Foundation Model. Nature 2025.

How to collect synthetic data?



Accurate Predictions on Small Data with a Tabular Foundation Model. Nature 2025.

Fitting Simple Functions

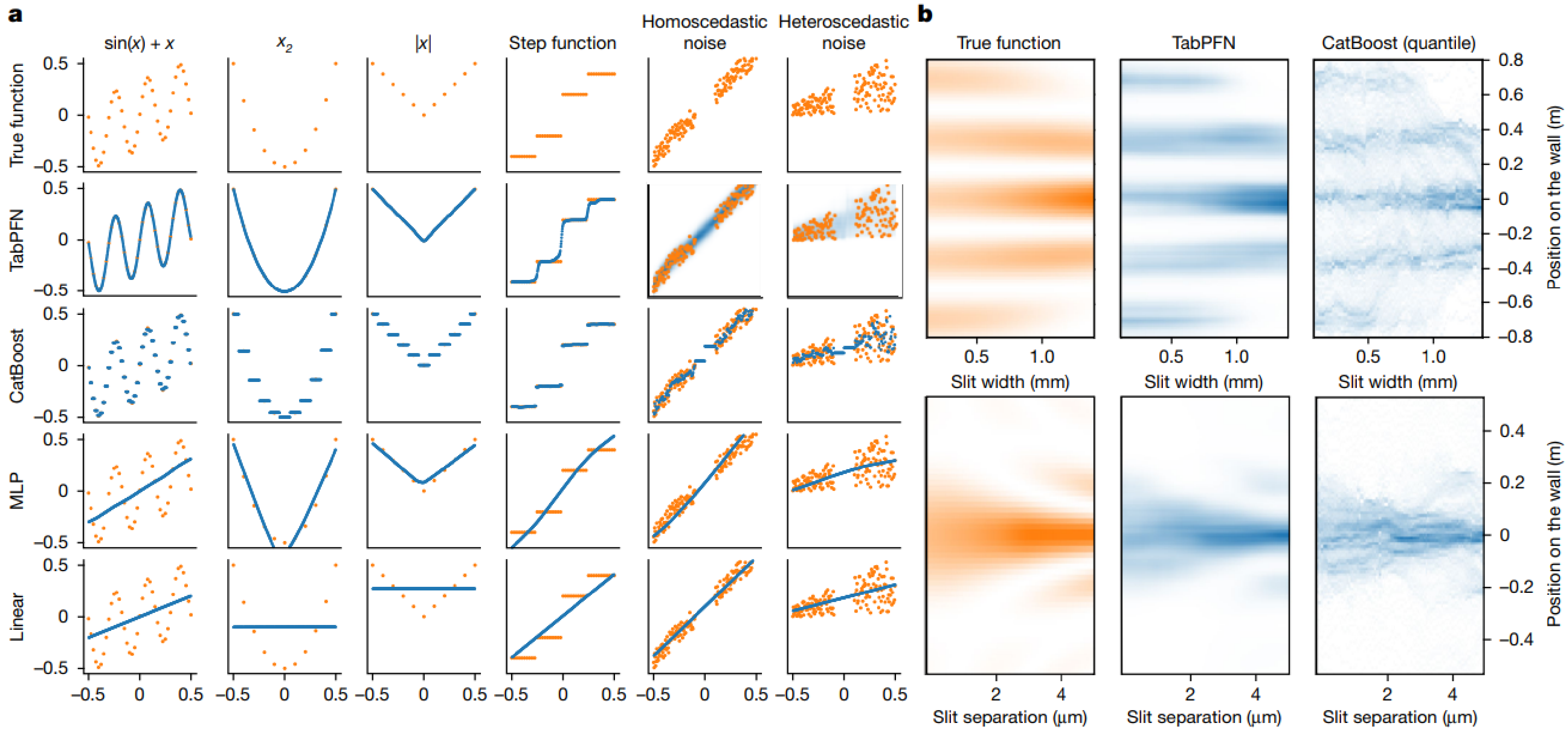
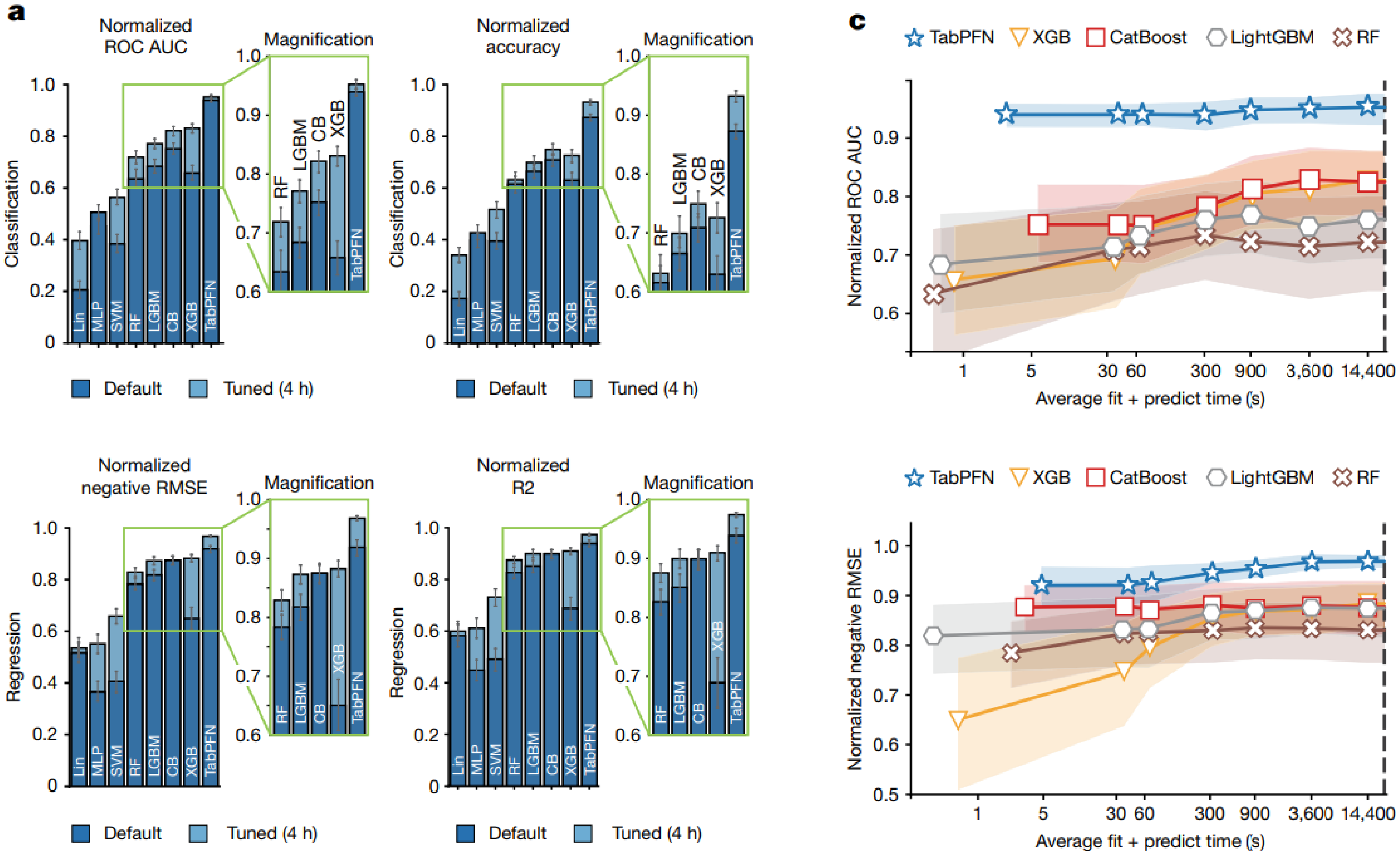


Fig. 3 | The behaviour of TabPFN and a set of baselines on simple functions. In all plots, we use orange for the ground truth and blue for model predictions. **a**, Each column represents a different toy function, each having a single feature (along the *x*-axis) and a target (along the *y*-axis). TabPFN can model a lot of different functions, including noisy functions. **b**, TabPFN can model distributions over outputs out of the box, which is exemplified by predicting the light intensity pattern in a double-slit experiment after observing the positions of 1,000 photons.

Accurate Predictions on Small Data with a Tabular Foundation Model. Nature 2025.

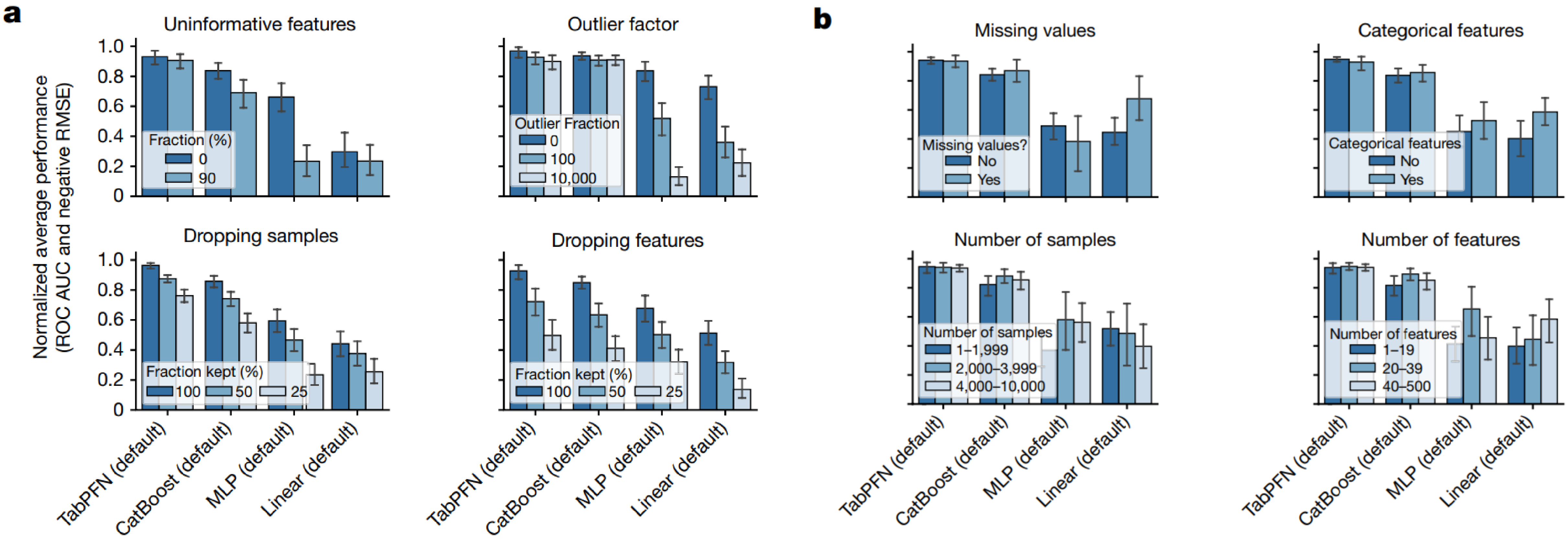
Comparison with Baselines

A pioneering work that uses foundation models/pre-training to beat traditional baselines for tabular data (e.g., RF, XGBoost, LGBM)!!!



Accurate Predictions on Small Data with a Tabular Foundation Model. Nature 2025.

Model Robustness



Accurate Predictions on Small Data with a Tabular Foundation Model. Nature 2025.

Thank You!

Course Website: https://yuzhang-teaching.github.io/CSCE689-S25.html