大規模言語モデルを表データ処理に活用

董 于洋

自己紹介

NEC
AIテクノロジー・サービス事業部
データサイエンスラボラトリー

主任研究員
董　于洋 (Dong Yuyang)

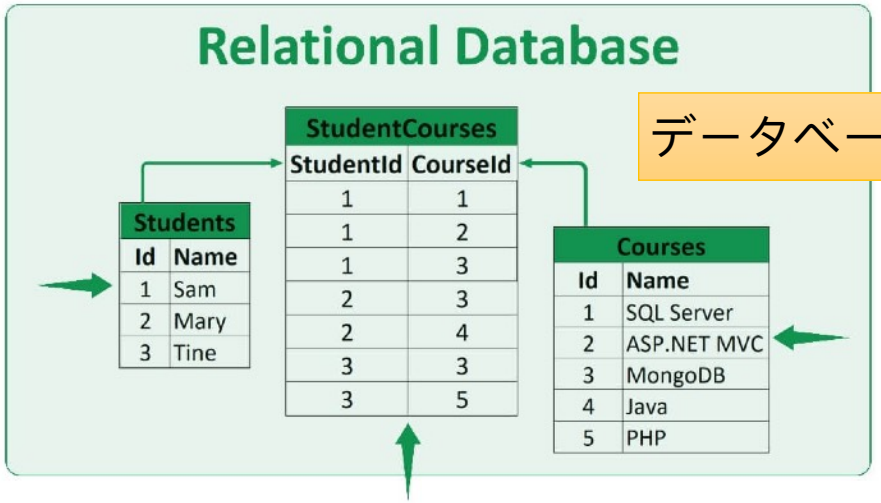
- 2014-2019 筑波大学 修士・博士 KDE lab
- 2019 NEC

最近は LLM, VLM, Agentに関する研究開発
- NEC cotomi (LLM)
- NEC 図表文脈理解技術 (VLM & Multimodal RAG)

cotomi

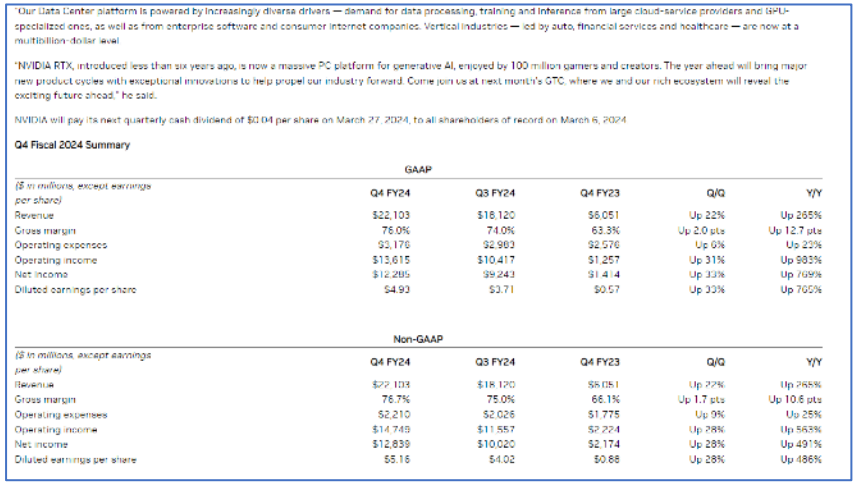
目次

背景: テーブル（表形式データ）は どこにでもある



データベースの中の表データ

(a)Relational databases



"Our Data Center platform is powered by increasingly diverse drivers — demand for data processing, training and inference from large cloud-service providers and GPU-specialized ones, as well as from enterprise software and consumer internet companies. Vertical industries — led by auto, financial services and healthcare — are now at a multibillion-dollar level.

"NVIDIA RTX, introduced less than six years ago, is now a massive PC platform for generative AI, enjoyed by 100 million gamers and creators. The year ahead will bring major new product cycles with exceptional innovations to help propel our industry forward. Come join us at next month's GTC, where we and our rich ecosystem will reveal the exciting future ahead," he said.

NVIDIA will pay its next quarterly cash dividend of $0.04 per share on March 27, 2024, to all shareholders of record on March 6, 2024.

Q4 Fiscal 2024 Summary

GAAP

($ in millions, except earnings per share)	Q4 FY24	Q3 FY24	Q4 FY23	Q/Q	Y/Y
Revenue	$22,103	$18,120	$6,051	Up 22%	Up 265%
Gross margin	76.0%	74.0%	63.3%	Up 2.0 pts	Up 12.7 pts
Operating expenses	$3,176	$2,983	$2,576	Up 6%	Up 23%
Operating income	$13,615	$10,417	$1,257	Up 31%	Up 983%
Net income	$12,285	$9,243	$1,414	Up 33%	Up 769%
Diluted earnings per share	$4.93	$3.71	$0.57	Up 33%	Up 765%

Non-GAAP

($ in millions, except earnings per share)	Q4 FY24	Q3 FY24	Q4 FY23	Q/Q	Y/Y
Revenue	$22,103	$18,120	$6,051	Up 22%	Up 265%
Gross margin	76.7%	75.0%	66.1%	Up 1.7 pts	Up 10.6 pts
Operating expenses	$2,210	$2,028	$1,775	Up 9%	Up 25%
Operating income	$14,749	$11,557	$2,224	Up 28%	Up 563%
Net income	$12,839	$10,020	$2,174	Up 28%	Up 491%
Diluted earnings per share	$5.16	$4.02	$0.86	Up 28%	Up 486%

(b) Rich documents, PDF

課題: データベース入れる前に or データベースに入れない表データ

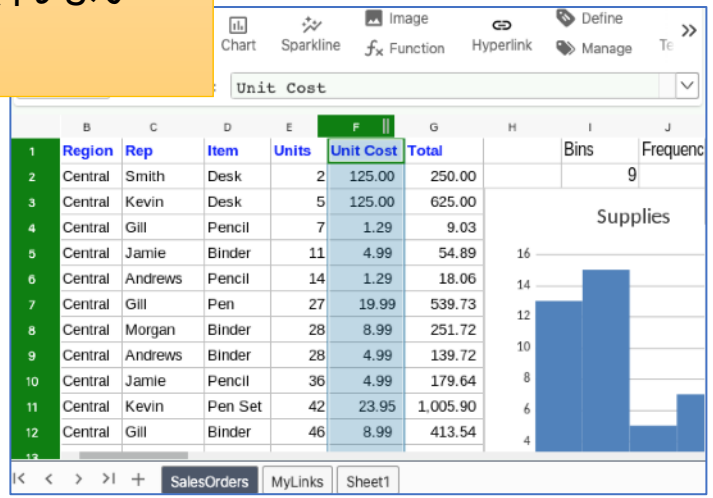
List of Large Language Models [edit]

See also: List of chatbots

For the training cost column, 1 petaFLOP-day = 1 petaFLOP/sec × 1 day = 8.64E19 FLOP. Also, only the largest model's cost is written.

Name	Release date[a]	Developer	Number of parameters (billion) [b]	Corpus size	Training cost (petaFLOP-day)	License[c]	Notes
GPT-1	June 2018	OpenAI	0.117		1[141]	MIT[142]	First GPT model, decoder-only transformer. Trained for 30 days on 8 P600 GPUs.
BERT	October 2018	Google	0.340[143]	3.3 billion words[143]	9[144]	Apache 2.0[145]	An early and influential language model.[4] Encoder-only and thus not built to be prompted or generative.[146] Training took 4 days on 64 TPUv2 chips.[147]
T5	October 2019	Google	11[148]	34 billion tokens[148]		Apache 2.0[149]	Base model for many Google projects, such as Imagen.[150]
XLNet	June 2019	Google	0.340[151]	33 billion words	330	Apache 2.0[152]	An alternative to BERT; designed as encoder-only. Trained on 512 TPU v3 chips for 5.5 days.[153]
GPT-2	February 2019	OpenAI	1.5[154]	40GB[155] (~10 billion tokens)[156]	28[157]	MIT[158]	Trained on 32 TPUv3 chips for 1 week.[157]
GPT-3	May 2020	OpenAI	175[50]	300 billion tokens[156]	3640[159]	proprietary	A fine-tuned variant of GPT-3, termed GPT-3.5, was made available to the public through a web interface called ChatGPT in 2022.[160]
GPT-Neo	March 2021	EleutherAI	2.7[161]	825 GiB[162]		MIT[163]	The first of a series of free GPT-3 alternatives released by EleutherAI. GPT-Neo outperformed an equivalent-size GPT-3 model on some benchmarks, but was significantly worse than the largest GPT-3.[163]

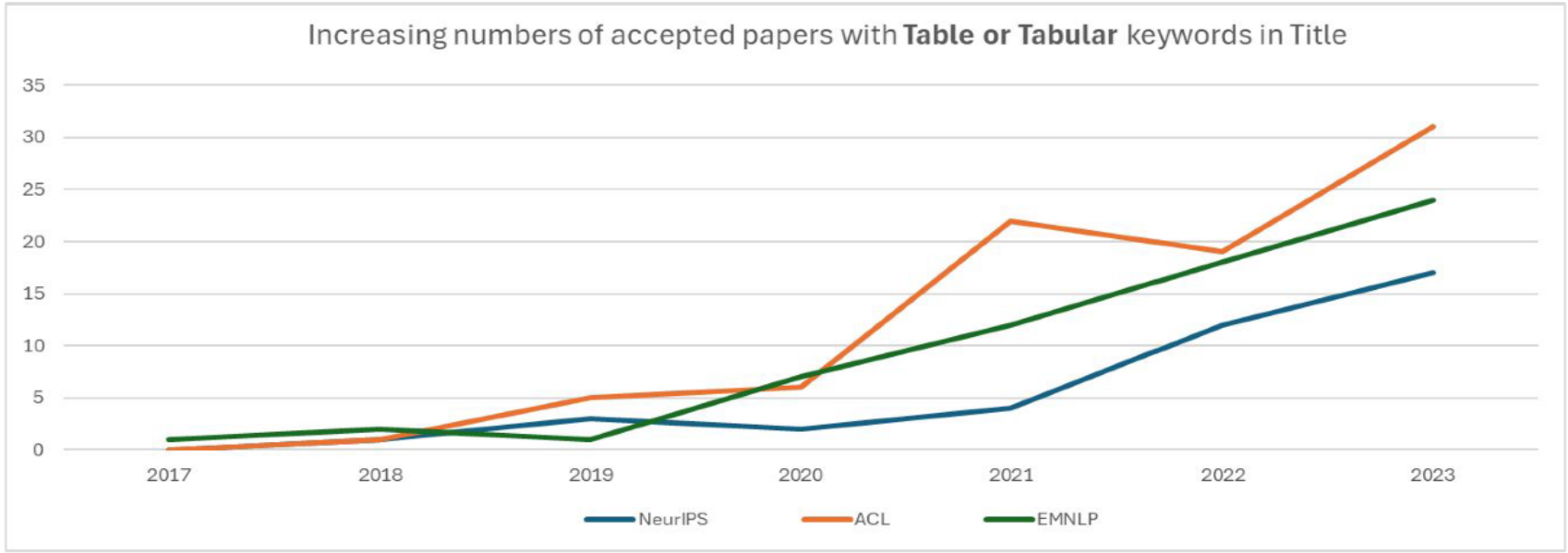
(c) webpages



(d) spreadsheet

背景: 表形式のデータに関する研究

- 近年のテーブルに関する研究の数が爆発



result from [4]

- Recent tutorials
  - [1] Web table extraction, retrieval and augmentation, SIGIR19
  - [2] From Tables to Knowledge: Recent Advances in Table Understanding, KDD21
  - [3] Transformers for Tabular Data Representation: A tutorial on Models and Applications VLDB22, SIGMOD23
  - [4] Large Language Models for Tabular Data: Progresses and Future Directions, SIGIR24
  - [5] On the use of large language models for table tasks, **CIKM24 (Dong, et al.)**

テーブルに関するタスク

テーブル前処理	テーブル理解	テーブル分析
“テーブルデータを良い形式に準備する”	“テーブルデータの内容を理解する”	“テーブルデータから答えを出す”
• Table cleaning • Table transformation • Table search	• Table Interpretation • Table detection • Table matching	• Table QA • Table fact verification • Table-to-text

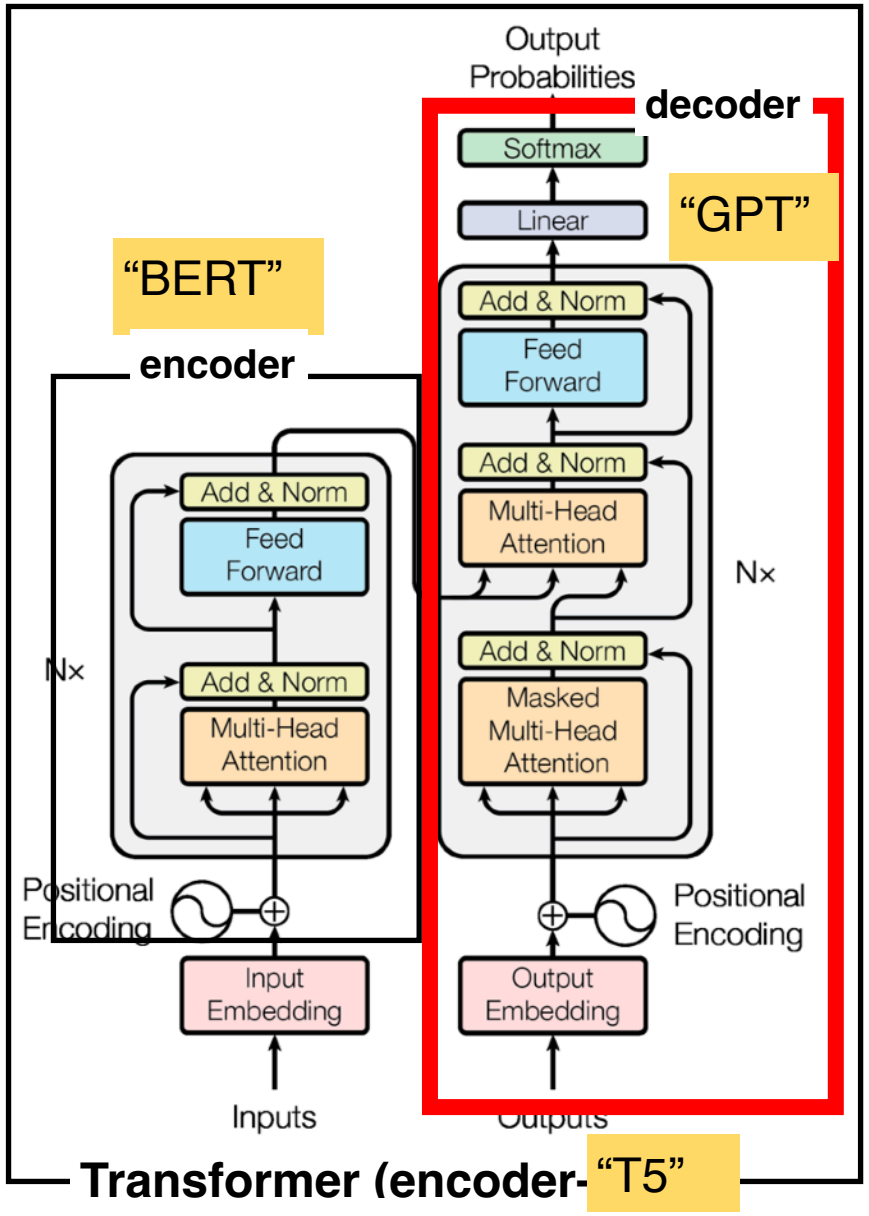
なぜ LLM使うのか?

- まず例を見ましょう
  - (別画面 Claudeの例)

- LLMでテーブルタスクに活用する理由
  - 使い方簡単
    - 要望は自然言語で入力できる
    - プログラミング能力がなくても
  - 賢い (= 精度が高い)
  - 汎用性が高い
    - 一つのモデルが複数タスクが解ける -> 調整するのは promptだけ

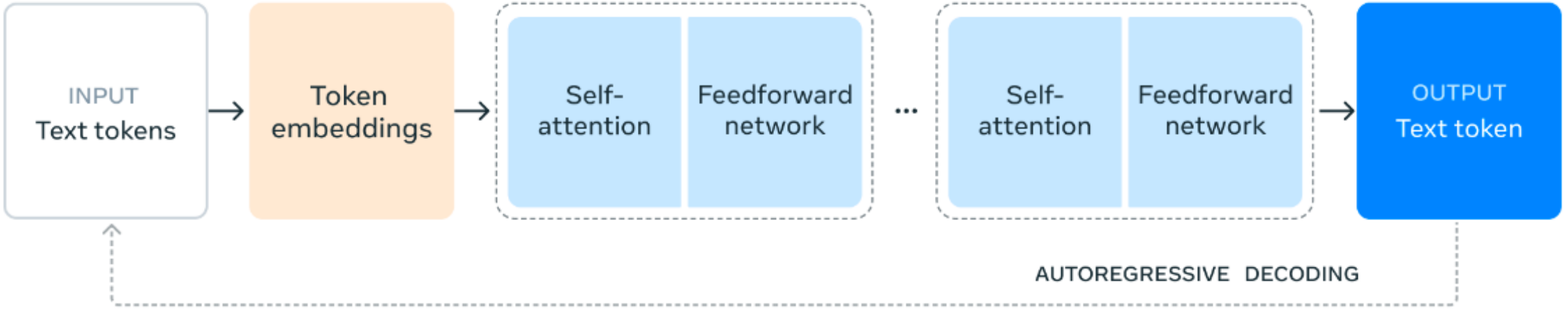
目次

LLM とは?

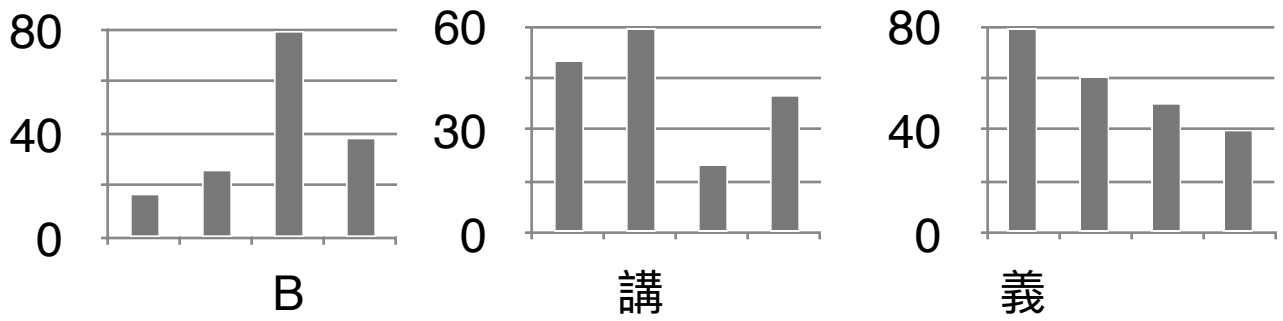


最近のLLMは通常
decoder-only の言語モデルを指す

Auto-Regressive Decoding



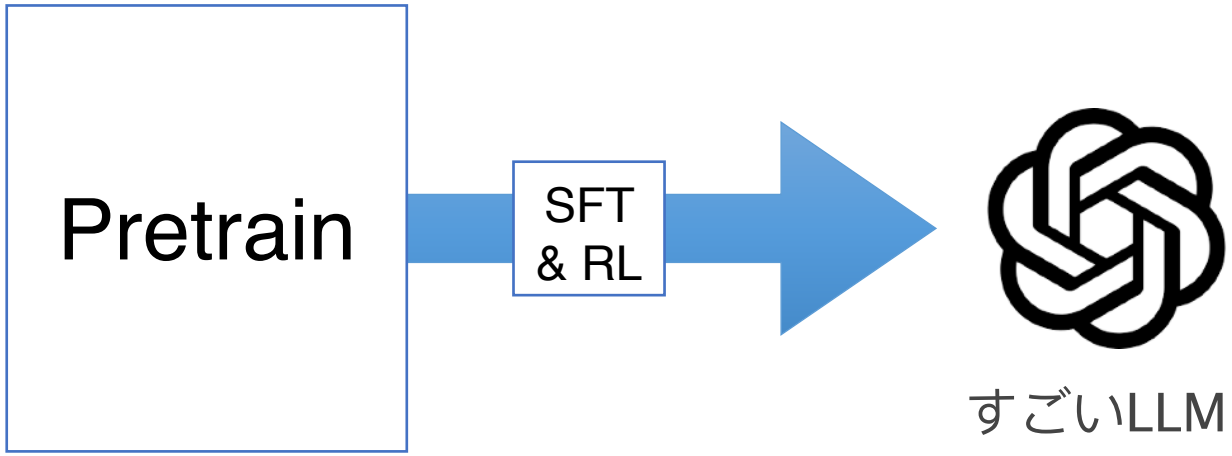
最強D -> 最強DB -> 最強DB講 -> 最強DB講義

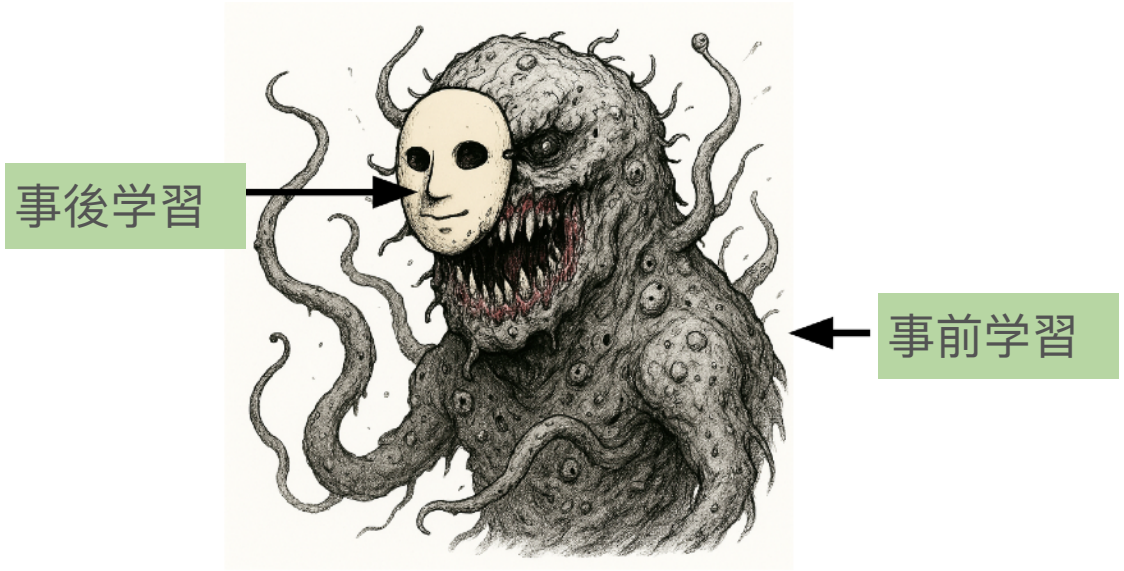


LLMの事前学習 (Pretrain) と事後学習 (SFT, RL)

- 事前学習
  - データ: 大規模のコーパス、 10-25T 規模
  - タスク: Next token prediction
  - 目的: 知識を勉強する

- 事後学習 (SFT, RL)
  - データ: QAデータ, 10-100B規模 (百万件ぐらい)
  - タスク: Next token prediction
  - 目的: 指示に従う、人間らしく話す





idea from: https://x.com/anthrupad/status/1622349563922362368/photo/1

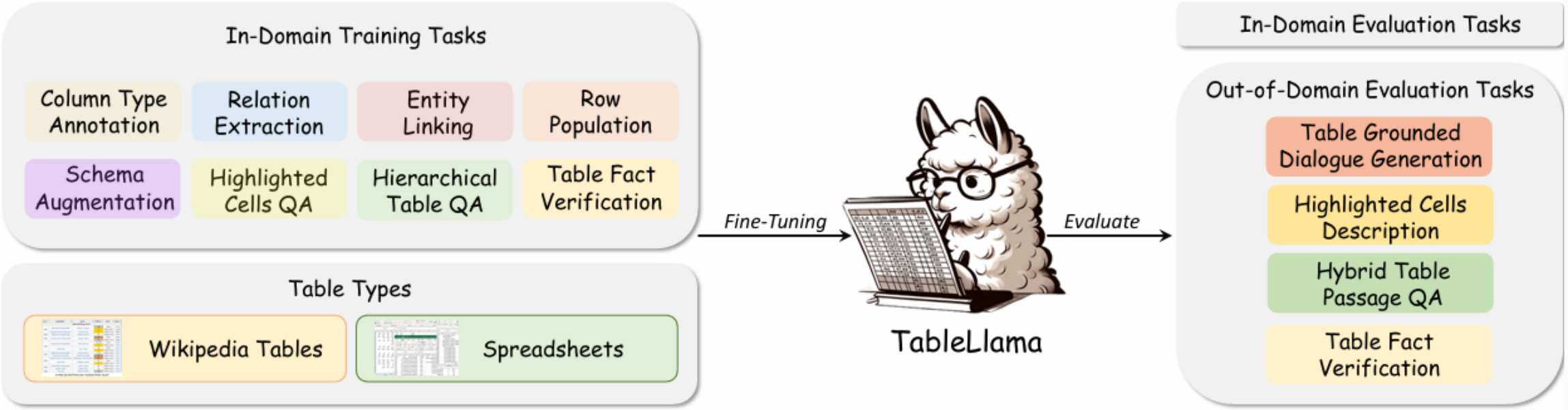
- https://www.youtube.com/watch?v=Ozos6M1JtIE

テーブルLLM 作る -> LLM 事後学習

- 目的
  - ドメイン特化 (精度向上)
  - ローカルモデル構築 (データ安全、処理速度)
- やり方
  - モデル構造は（ほぼ) 共通
  - Finetuneのスクリプト、パラメータ調整のやり方も共通
- 特化LLM作る上で重要なところ
  - タスクの定義
    - Input: Table, Question
    - Output: Answer
  - 学習データの多様性
  - 学習データの質

代表研究 1.TableLlama [NAACL'24]

- 幅広いタスク&データ TableInstruct Dataset 構築
  - 14 datasets
  - 11 tasks
  - 1.24M tables with 2.6M Q&A pairs
- LongLoRA でlong context fine tuning



TableLlamaのテーブルの入れ方 (markdown + 特殊 token)

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 | ...

Markdown

特殊token

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

Response: 80.3.

TableLlama の実験結果

Indomain は SoTAに近づいている

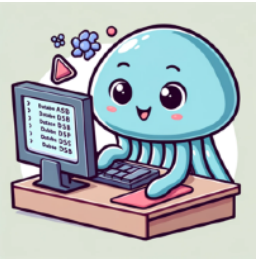
In-domain Evaluation						
Datasets	Metric	Base	TableLlama	SOTA	GPT-3.5	GPT-4§
Column Type Annotation	F1	3.01	94.39	**94.54***† (Deng et al., 2020)	30.88	31.75
Relation Extraction	F1	0.96	91.95	**94.91***† (Deng et al., 2020)	27.42	52.95
Entity Linking	Accuracy	31.80	**93.65**	84.90*† (Deng et al., 2020)	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

Out domain は baseより向上するが、SoTAにまだ距離がある

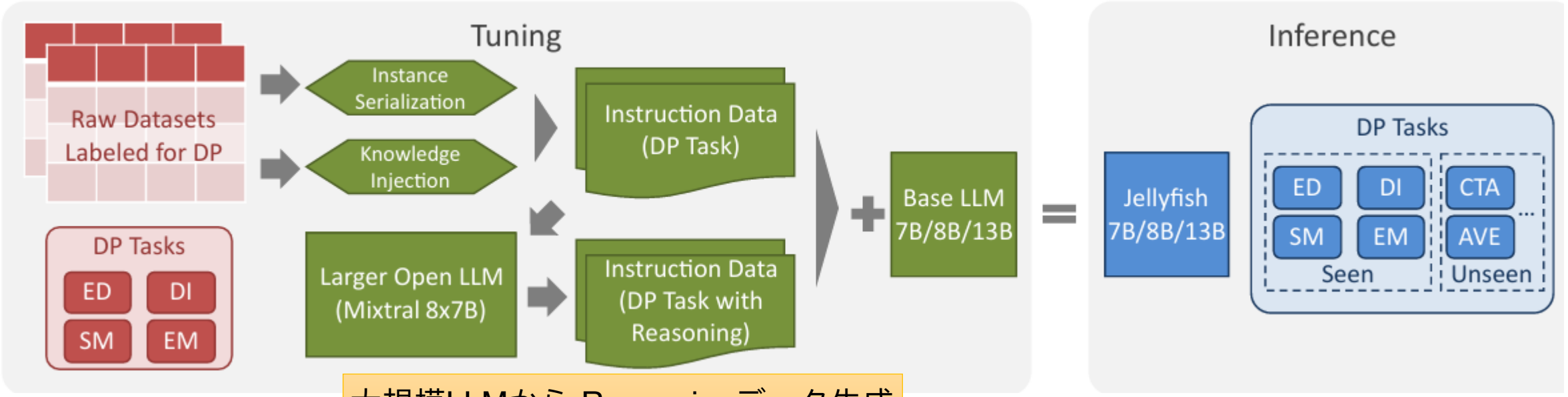
Out-of-domain Evaluation							
Datasets	Metric	Base	TableLlama	SOTA	Δ_{Base}	GPT-3.5	GPT-4§
FEVEROUS	Accuracy	29.68	73.77	85.60 (Tay et al., 2022)	+44.09	60.79	71.60
HybridQA	Accuracy	23.46	39.38	65.40* (Lee et al., 2023)	+15.92	40.22	58.60
KVRET	Micro F1	38.90	48.73	67.80 (Xie et al., 2022)	+9.83	54.56	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

代表研究 2.Jellyfish [EMNLP'24] (Dongの研究)

- 数少ないが品質が高いデータ (54K)
  - 実データ (Q,A) と 蒸留データ (Q, Reason, A) を混ぜる
- データの分布と割合で探索



https://huggingface.co/NECOUDBFM



大規模LLMから Reasoningデータ生成

Jellyfish のテーブルの入れ方

	DP Task Data	DP Task with Reasoning Data
system message	You are an AI assistant that follows instruction extremely well. User will give you a question. Your task is to answer as faithfully as you can.	You are an AI assistant that follows instruction extremely well. User will give you a question. Your task is to answer as faithfully as you can. While answering, provide detailed explanation and justify your answer.
task description	You are tasked with determining whether two Products listed below are the same based on the information provided. Carefully compare all the attributes before making your decision.	
injected knowledge	Note that missing values (N/A or "nan") should not be used as a basis for your decision.	
instance content	Product A: [name: "Sequoia American Amber Ale", factory: "Wig And Pen"] Product B: [name: "Aarhus Cains Triple A American Amber Ale", factory: "Aarhus Bryghus"]	
question	Are Product A and Product B the same Product?	
output format	Choose your answer from: [Yes, No]	After your reasoning, finish your response in a separate line with and ONLY with your final answer. Choose your final answer from [Yes, No].

Key : Value

Jellyfish の実験結果

In domain 平均スコアが GPT4/ GPT-4oに超える！

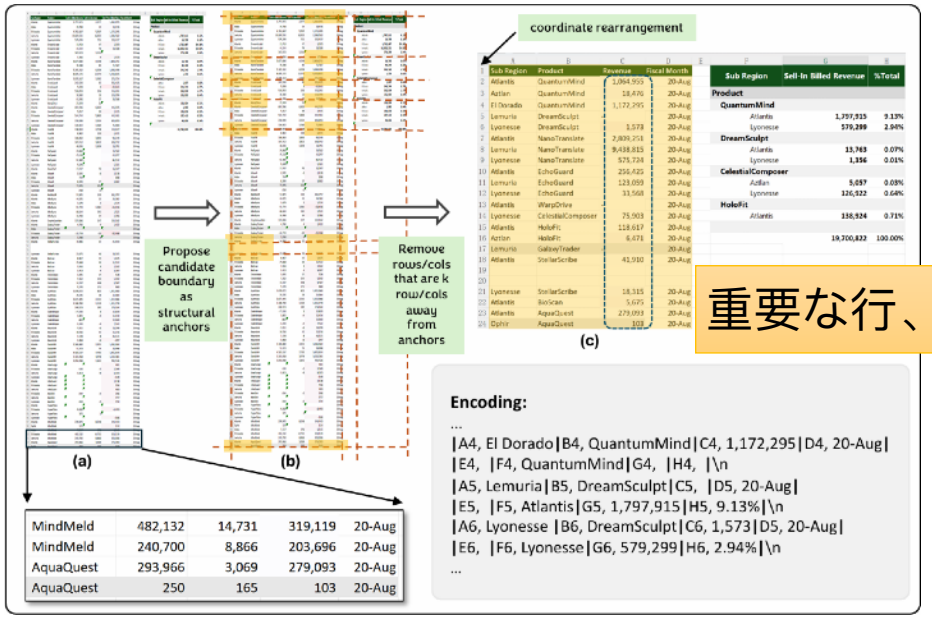
Task	Type	Dataset	Model								
			Best of non-LLM	GPT-3	GPT-3.5	GPT-4	GPT-4o	Table-GPT	Jellyfish-7B	Jellyfish-8B	Jellyfish-13B
ED	Seen	Adult	99.10	99.10	92.01	92.01	83.58	–	77.40	73.74	**99.33**
		Hospital	94.40	**97.80**	90.74	90.74	44.76	–	94.51	93.40	95.59
	Unseen	Flights	81.00	–	–	**83.48**	66.01	–	69.15	66.21	82.52
		Rayyan	79.00	–	–	81.95	68.53	–	75.07	81.06	**90.65**
DI	Seen	Buy	96.50	98.50	98.46	**100**	**100**	–	98.46	98.46	**100**
		Restaurant	77.20	88.40	94.19	**97.67**	90.70	–	89.53	87.21	89.53
	Unseen	Flipkart	68.00	–	–	**89.94**	83.20	–	87.14	87.48	81.68
		Phone	86.70	–	–	**90.79**	86.78	–	86.52	85.68	87.21
SM	Seen	MIMIC-III	20.00	–	–	40.00	29.41	–	**53.33**	45.45	40.00
		Synthea	38.50	45.20	57.14	**66.67**	6.56	–	55.56	47.06	56.00
	Unseen	CMS	50.00	–	–	19.35	22.22	–	42.86	38.10	**59.29**
EM	Seen	Amazon-Google	75.58	63.50	66.50	74.21	70.91	70.10	**81.69**	81.42	81.34
		Beer	94.37	**100**	96.30	**100**	90.32	96.30	**100.00**	**100.00**	96.77
		DBLP-ACM	**98.99**	96.60	96.99	97.44	95.87	93.80	98.65	98.77	98.98
		DBLP-GoogleScholar	95.70	83.80	76.12	91.87	90.45	92.40	94.88	95.03	**98.51**
		Fodors-Zagats	**100**	**100**	**100**	**100**	93.62	**100**	**100**	**100**	**100**
		iTunes-Amazon	97.06	98.20	96.40	**100**	98.18	94.30	96.30	96.30	98.11
	Unseen	Abt-Buy	89.33	–	–	**92.77**	78.73	–	86.06	88.84	89.58
		Walmart-Amazon	86.89	87.00	86.17	**90.27**	79.19	82.40	84.91	85.24	89.42
Average			80.44	-	-	84.17	72.58	-	82.74	81.55	**86.02**

Task	Dataset	Model								
		RoBERTa (159 shots)	RoBERTa (356 shots)	Stable Beluga 2 70B	SOLAR 70B	GPT-3.5	GPT-4	GPT-4o	Jellyfish-7B	Jellyfish-8B	Jellyfish-13B
CTA	SOTAB	79.20	89.73	–	–	89.47	**91.55**	65.06	83.00	76.33	82.00
AVE	AE-110k	–	–	52.10	49.20	**61.30**	55.50	55.77	56.09	59.55	58.12
	OA-Mine	–	–	50.80	55.20	62.70	**68.90**	60.20	51.98	59.22	55.96

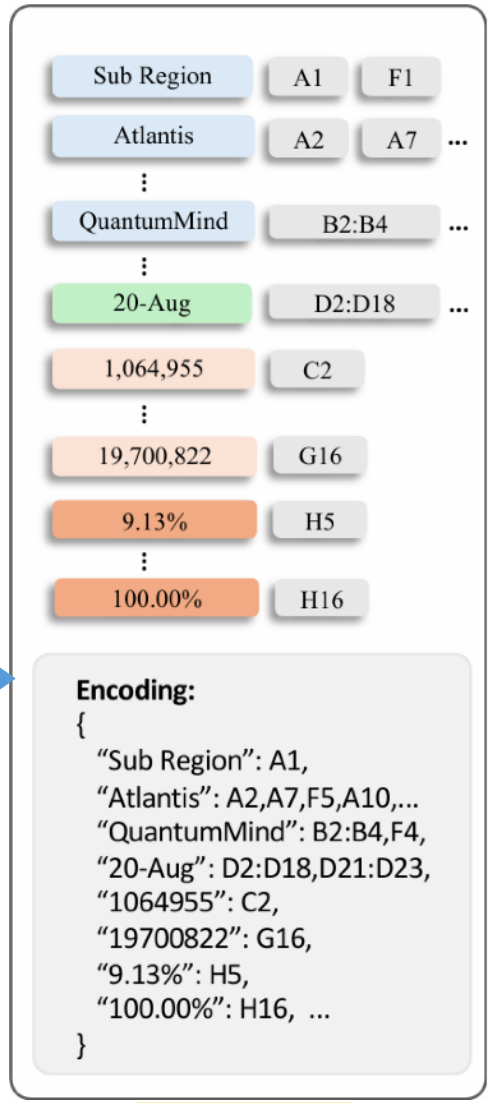
Out domain は まだ距離がある

代表研究 3. SpreadSheetLLM [EMNLP'24]

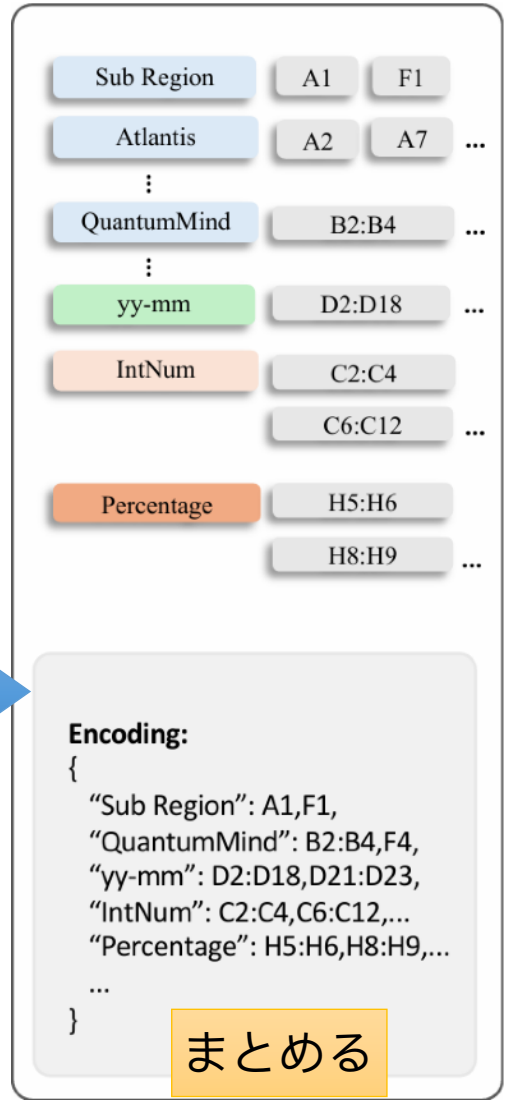
- テーブルデータの入れ方を工夫
  - 目的: 大きいテーブルを圧縮する
  - やり方
    - 重要な行・列だけ選ぶ
    - "Inverted index 的に" json形式に変換
    - 統計情報入れる、データまとめ



重要な行、列



転置索引



まとめる

SpreadSheetLLM の実験結果

独自定義の形なので in context learningが効かない

Fine tuneしたらよくなってた

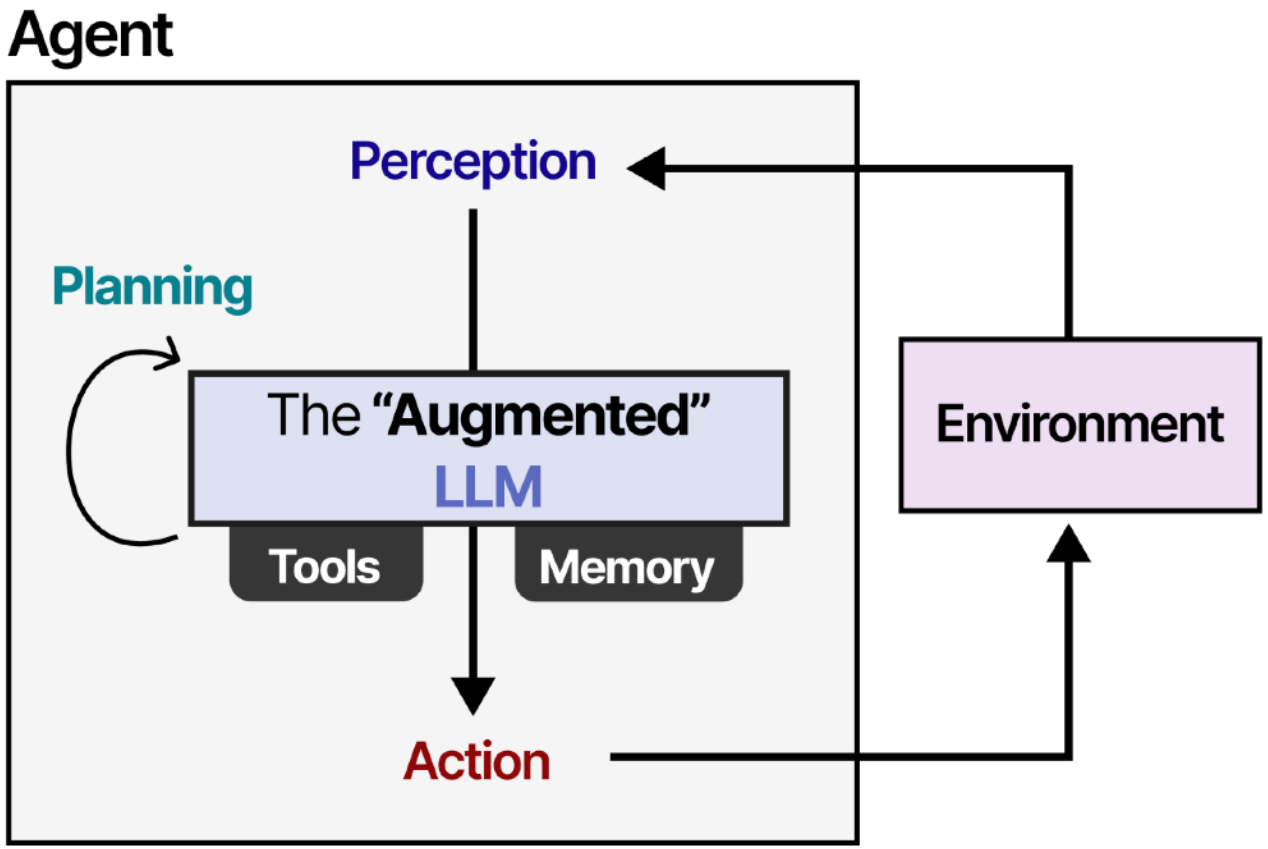
Model & Method	Small	Medium	Large	Huge	All
ICL					
Mistral-v2	0.071	0.013	0.029	0.017	0.036
GPT4	0.318	0.292	0.090	0.000	0.154
GPT4-compress	0.480	0.454	0.373	0.330	0.410
Fine-tune					
Llama3	0.715	0.765	0.290	0.000	0.471
Llama2	0.557	0.378	0.107	0.000	0.280
Phi3	0.604	0.481	0.201	0.130	0.330
Mistral-v2	0.700	0.784	0.472	0.123	0.542
GPT4	0.779	0.707	0.288	0.000	0.520
Llama3-compress	0.825	0.768	0.664	0.617	0.719
Llama2-compress	0.710	0.722	0.633	0.578	0.660
Phi3-compress	0.800	0.673	0.624	0.675	0.689
Mistral-v2-compress	0.778	0.729	0.686	0.744	0.726
GPT3.5-compress	0.795	0.649	0.600	0.680	0.680
GPT4-compress	0.810	**0.832**	0.718	0.690	0.759
-w/o Aggregation	**0.864**	0.816	**0.739**	**0.753**	**0.789**
TableSense-CNN	0.785	0.788	0.567	0.561	0.666

目次

- 背景
- テーブル LLM
- **テーブル Agent**
- テーブル VLM
- まとめと今後の課題

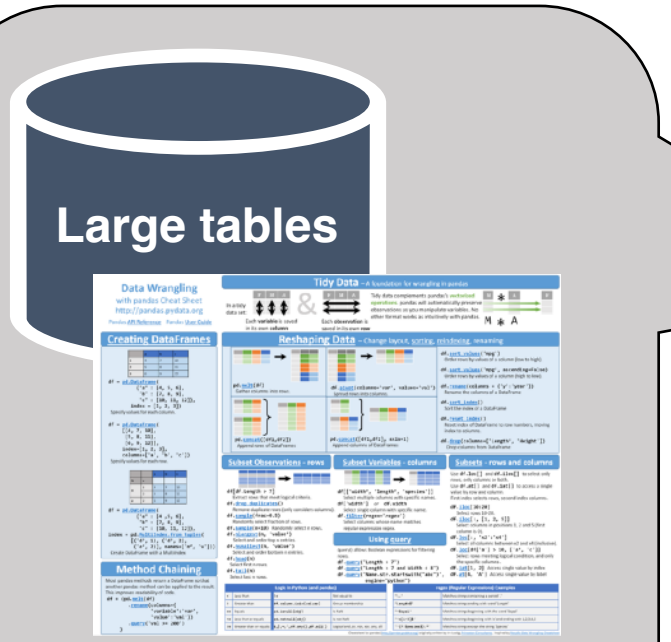
Agent とは?

- 目的を達成するために、外部環境とやりとりしながら、自律的に判断・行動できるモデル集合
  - Observation (Perception)
  - Planning
  - Tool use
  - Memory

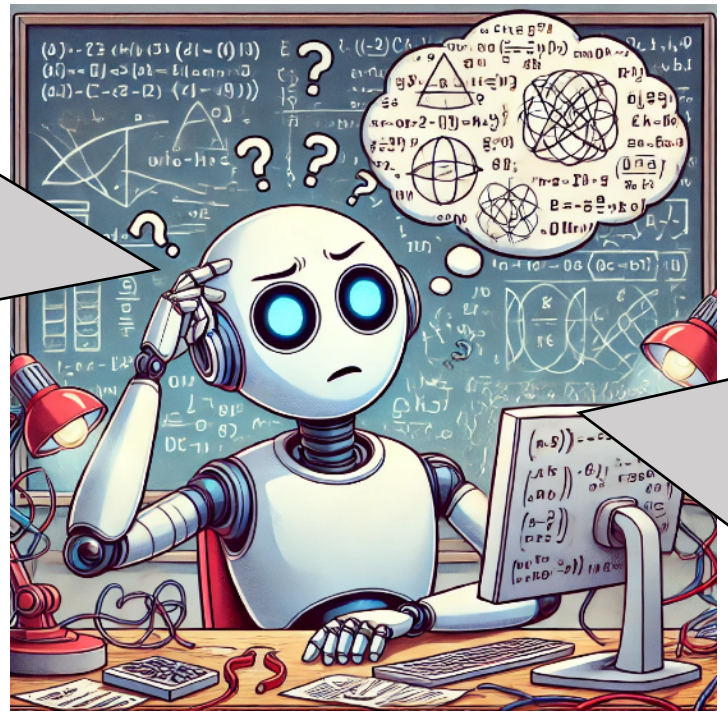


なぜ Agentが必要なのか?

- テーブルは複雑なので、一回の推論では複雑な問題を解くのが難しい
  - 知識が足りない -> **Tool: RAG**
  - 能力が足りない -> **Tool: Python, MCP, etc..**
  - 計画が足りない -> **Planning**



巨大なテーブルの理解



square beads	$2.97 per kilogram
oval beads	$3.41 per kilogram
flower-shaped beads	$2.18 per kilogram
star-shaped beads	$1.95 per kilogram
heart-shaped beads	$1.52 per kilogram
spherical beads	$3.42 per kilogram
rectangular beads	$1.97 per kilogram

Question: If Tracy buys 5 kilograms of spherical beads, 4 kilograms of star-shaped beads, and 3 kilograms of flower-shaped beads, how much will she spend? (unit: $)
Answer: 31.44
Solution:
Find the cost of the spherical beads. Multiply: $3.42 × 5 = $17.10.
Find the cost of the star-shaped beads. Multiply: $1.95 × 4 = $7.80.
Find the cost of the flower-shaped beads. Multiply: $2.18 × 3 = $6.54.
Now find the total cost by adding: $17.10 + $7.80 + $6.54 = $31.44.
She will spend $31.44.

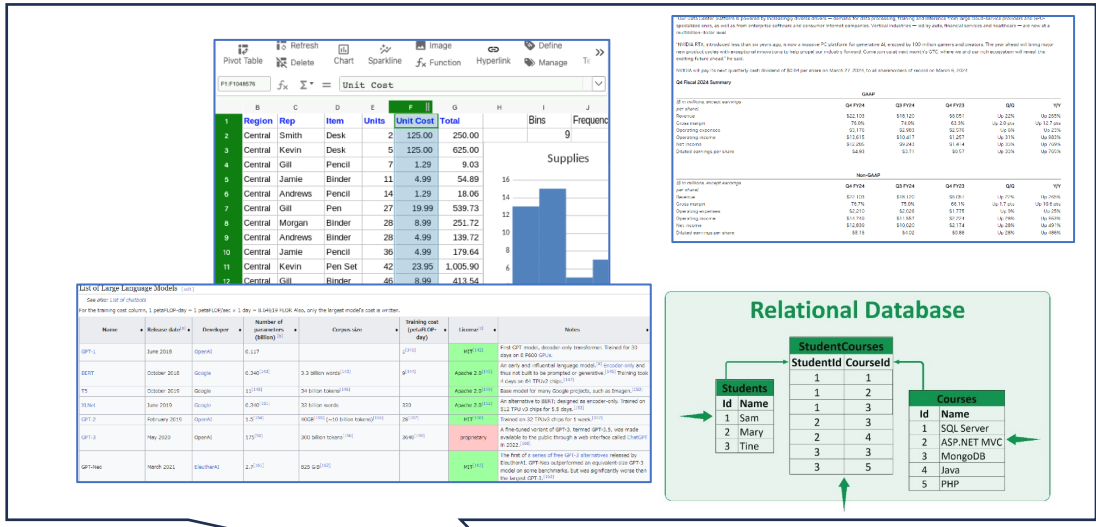
Sandwich sales		
Shop	Tuna	Egg salad
City Cafe	6	5
Sandwich City	3	12
Express Sandwiches	7	17
Sam's Sandwich Shop	1	6
Kelly's Subs	3	4

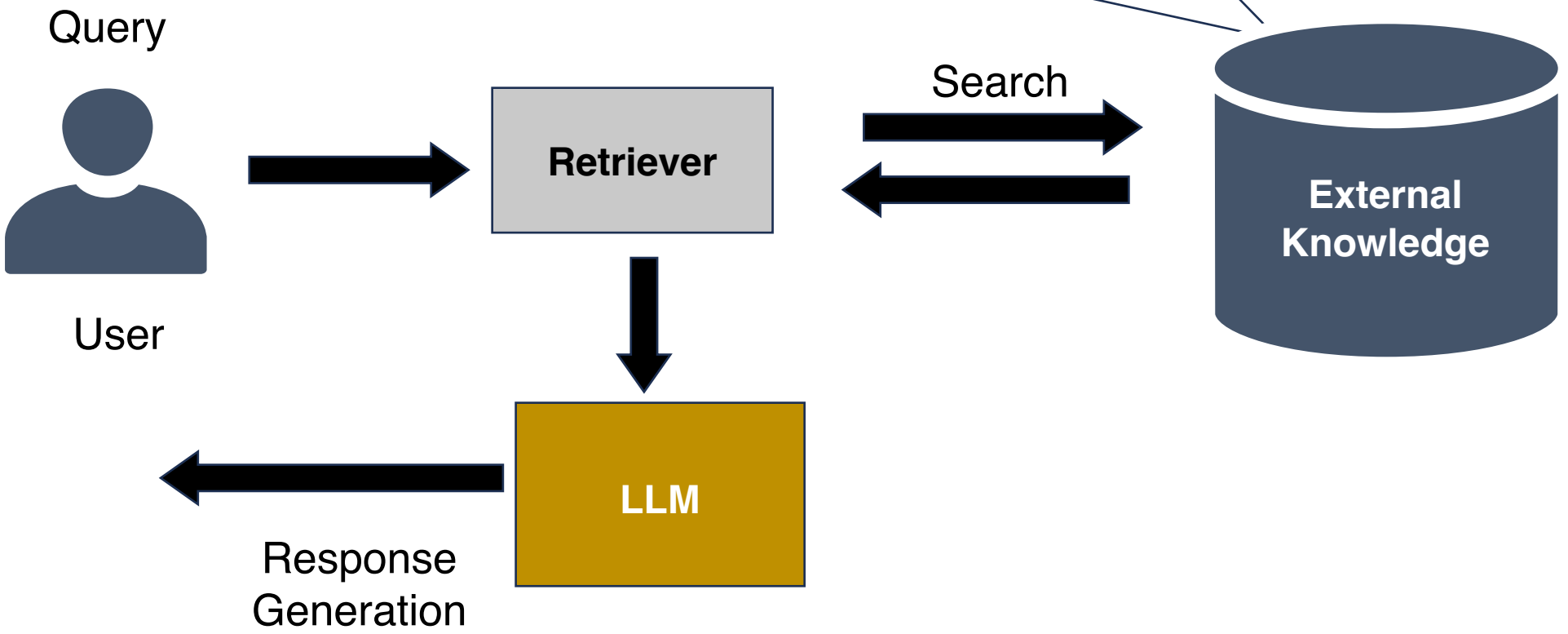
Question: As part of a project for health class, Cara surveyed local delis about the kinds of sandwiches sold. Which shop sold fewer sandwiches, Sandwich City or Express Sandwiches?
Options: (A) Sandwich City (B) Express Sandwiches
Answer: (A) Sandwich City
Solution:
Add the numbers in the Sandwich City row. Then, add the numbers in the Express Sandwiches row.
Sandwich City: 3 + 12 = 15. Express Sandwiches: 7 + 17 = 24.
15 is less than 24. Sandwich City sold fewer sandwiches.

数学&推論タスクの解く

RAG: Table Retriever

- RAGの対象は表データもあり





RAG: Table Retriever

- **Traditional retriever**
  - Keyword-based (BM25)
  - SQL for relational database

id: 1, name: Apple, loc: CA, employee: 154,000
id: 2, name: IBM, loc: NY, employee: 282,000

Text

Table serialization

id	name	loc	employee
1	Apple	CA	154,000
2	IBM	NY	282,000

Schemas

Keyword-based retrievers

Databases

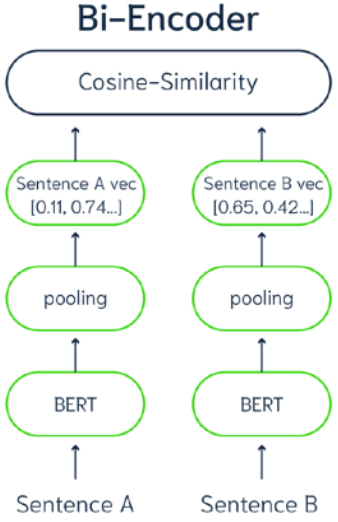
RAG: Table Retriever

- **Dense Retriever: Encoder + vector database**
  - テキストモデルを使う

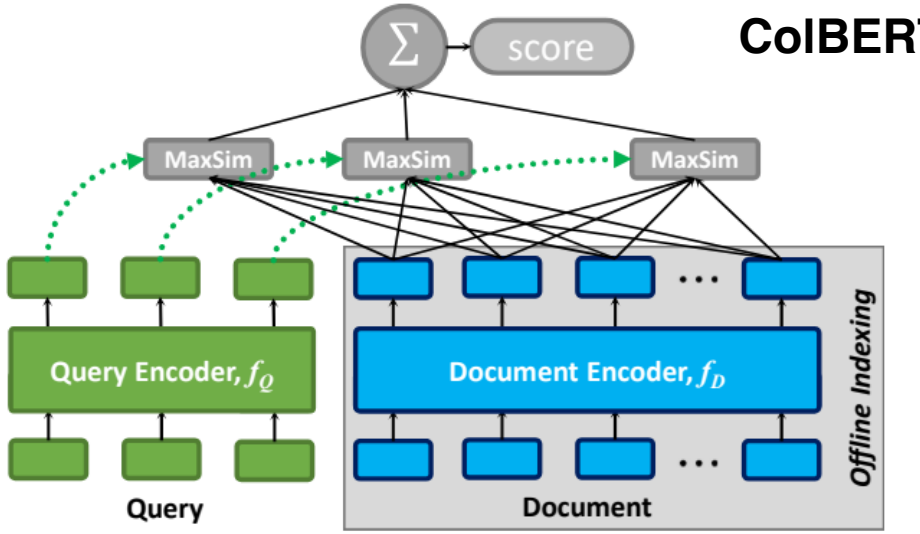
	Model-architecture	serialization
T-RAG [arxiv22]	DPR	rows
Deep-join[VLDB23] (Dongの研究)	Sentence-BERT	columns
LI-RAGE [ACL23]	ColBERT	table
ITR [ACL23]	DPR	located row & col

DPR

SBERT

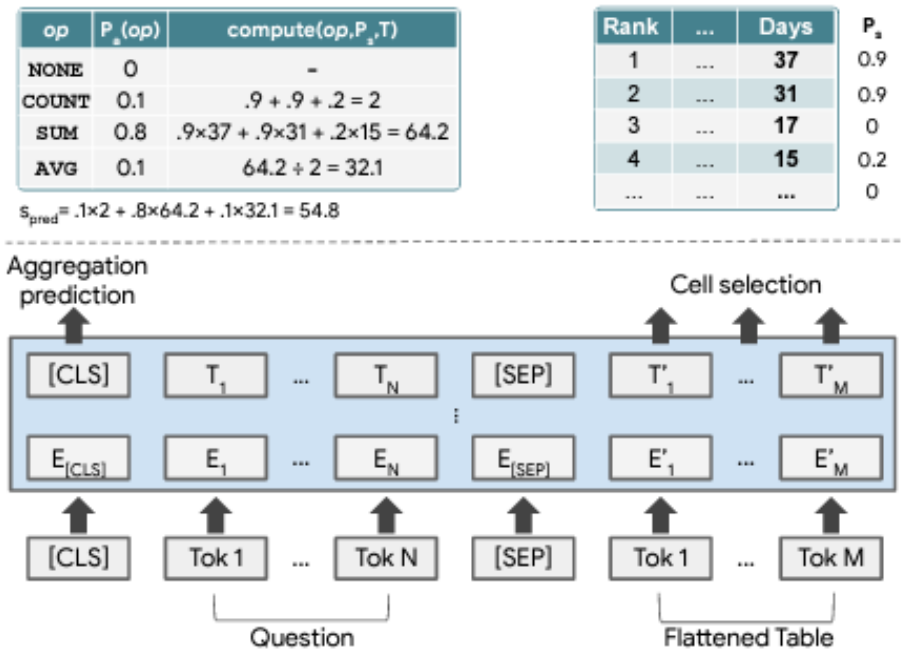


ColBERT

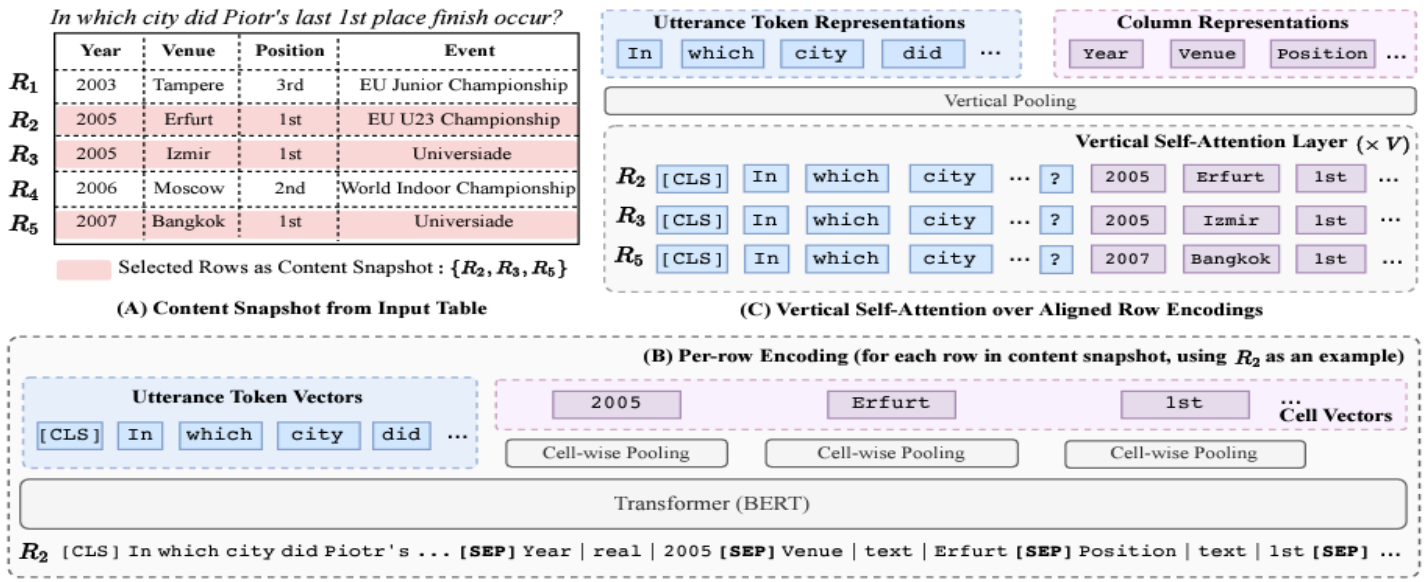


RAG: Table Retriever

- **Dense Retriever: LM encoder + vector database**
  - テーブル専用モデル:
    - query -> text encoder
    - table -> table-specific encoder



TAPAS



TaBERT

TaPas: Weakly Supervised Table Parsing via Pre-training, ACL20

TaBERT: Pretraining for Joint Understanding of Textual and Tabular Data, ACL20

RAG: Table Retriever

- テーブル専用モデルが効果あるのか? -> テキストモデルと差がないよ!
- Table Retrieval May Not Necessitate Table-specific Model Design [ACL workshop22]
- Open-WikiTable [ACL23]

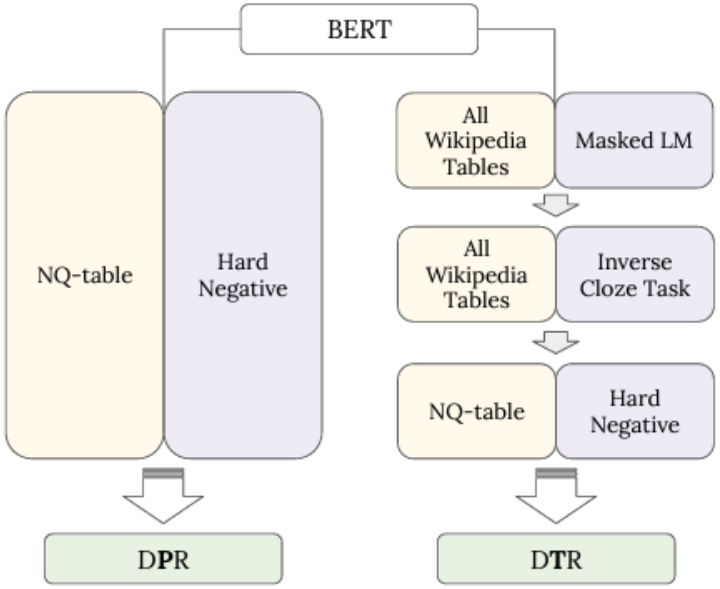


Figure 3: Comparison of DPR and DTR training.

Model	Retrieval Accuracy @1	@5	@10	@20	@50
DTR (medium)	62.32	82.51	86.75	91.51	94.26
DTR (large)	63.98	84.27	**89.65**	**93.48**	**95.65**
BERT-table	60.97	79.81	85.51	88.20	91.62
DPR	57.04	80.54	86.13	89.54	92.34
DPR-table	**67.91**	**84.89**	88.72	90.58	92.86

NQ-tables

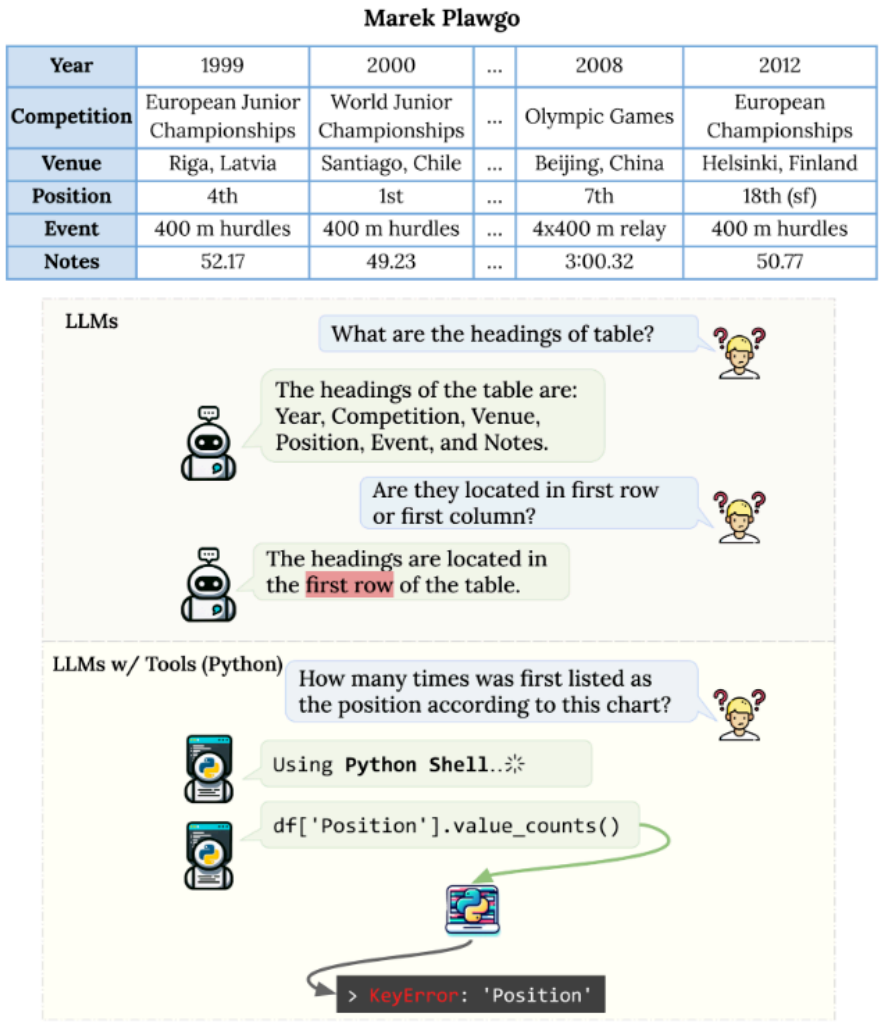
Encoder Text	Encoder Table	Data	k=5	k=10	k=20
BM25		Original	6.6	8.0	10.3
		Decontextualized	45.5	52.9	59.7
		Paraphrased	42.2	48.9	56.1
BERT	BERT	Original	25.0	34.1	45.1
		Decontextualized	91.6	96.0	97.8
		Paraphrased	89.5	95.0	97.3
BERT	TAPAS	Original	19.4	28.1	38.5
		Decontextualized	88.2	94.5	97.3
		Paraphrased	84.0	91.4	95.6

Open-WikitaTables

Tool use: プログラミングは必須 tool

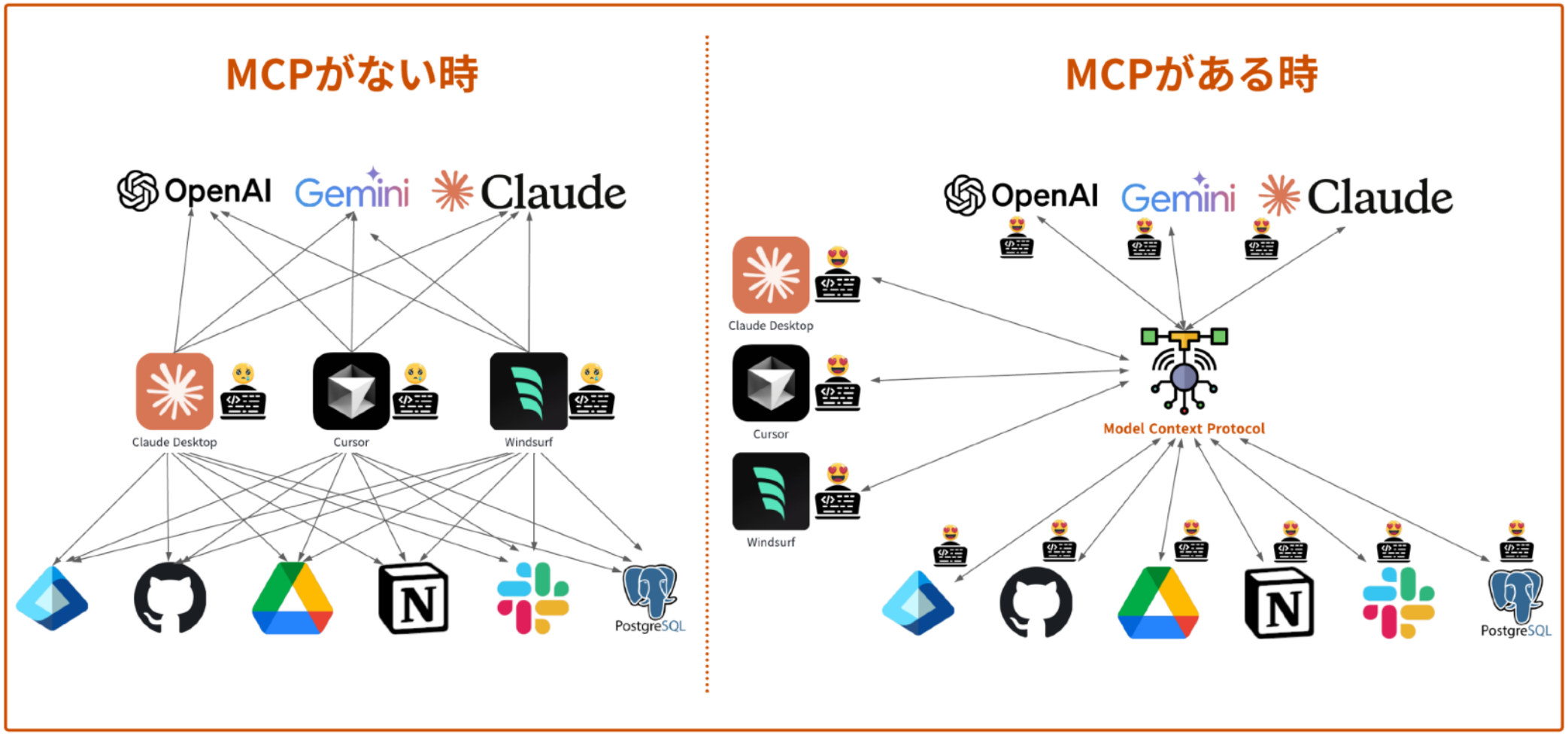
- LLMに対して、long context、数値処理に対するのは苦手
- Python、pandasなどを使って正しく処理できる

	Task	SQL	Python	Retriever	Database	Math	Graph
PyAgent	TableQA		○				
ToolQA	TableQA	○	○	○	○	○	○
ReAcTable	TableQA	○	○				
LEVER	TableQA	○					
Binder	TableQA, TableFact	○	○	○			
Chain-of-Table	TableQA, TableFact		○				
ToolWriter	TableQA	○	○				
SheetAgent	Table Transformation	○	○	○			
AutoTQA	TableQA, TableFact	○	○	○	○		
TableRAG	TableQA, TableFact		○	○			



Rethinking Tabular Data Understanding with Large Language Models NAACL24

Tool use: 最近は MCPによってtool use簡単化



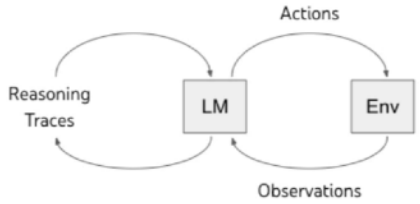
MCPとは何か ～AIエージェントの為の標準プロトコル～

https://blog.cloudnative.co.jp/27994/

Planning

- 目的
  - Divide and concur: 複雑なタスクを分解することで、それぞれ解決できて、全体のゴールを達成できる
- 手法
  - Workflow
    - 人間の知識を生かして、このタスクを解けるために、どんなサブタスクを事前に定義する

  - ReAct (CodeAct) [ICLR24]
    - Reasoning + Acting
    - Thinking -> Action -> Observation で繰り返す

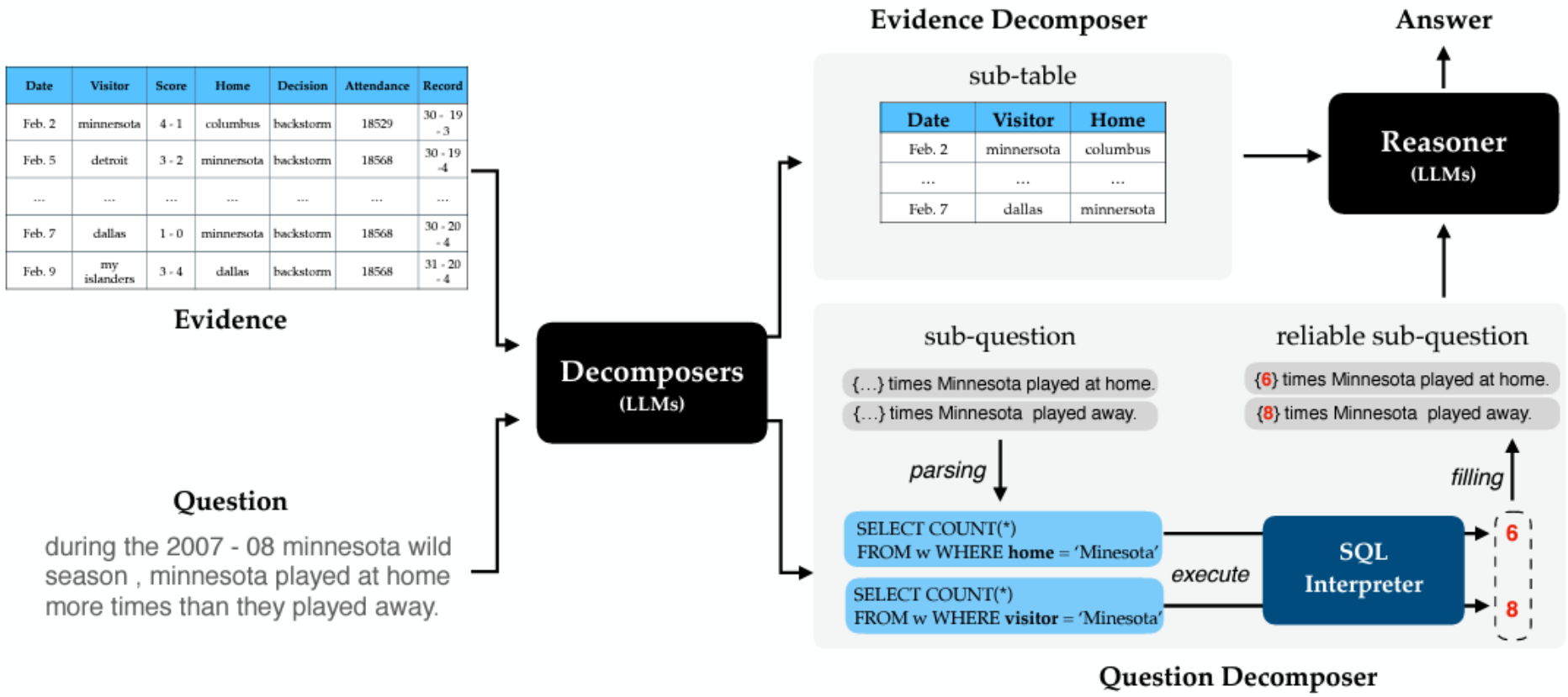


ReAct (Reason + Act)

  - Plan-and-Act
    - Top-down planning list
    - まず全体の計画（手順）を立ててから、順番に実行する。

Planning: Workflow

- Datar [SIGIR23]: Table QA
  - Question decomposition and table decomposition



https://arxiv.org/pdf/2301.13808

Planning: Workflow

- DIN-SQL [NeurIPS23]: Text-to-SQL
  - 事前に定義した 4 sub tasksに分解
    - Schema linking
    - Generate 3-levels SQL with 3 sub tasks
  - Self-correction the final answer

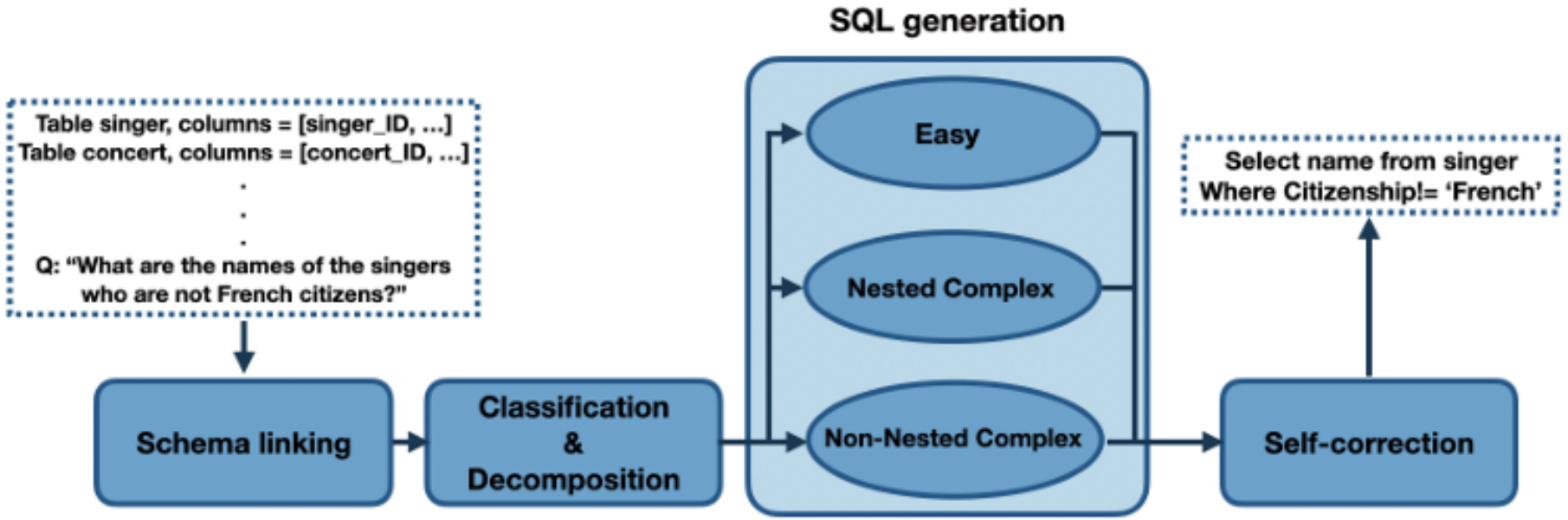


Figure 2: An overview of the proposed methodology including all four modules

https://arxiv.org/pdf/2304.11015

Planning: Dynamic Planning, ReAct

- ReAcTable [VLDB24]
  - During action, use SQL and python shell

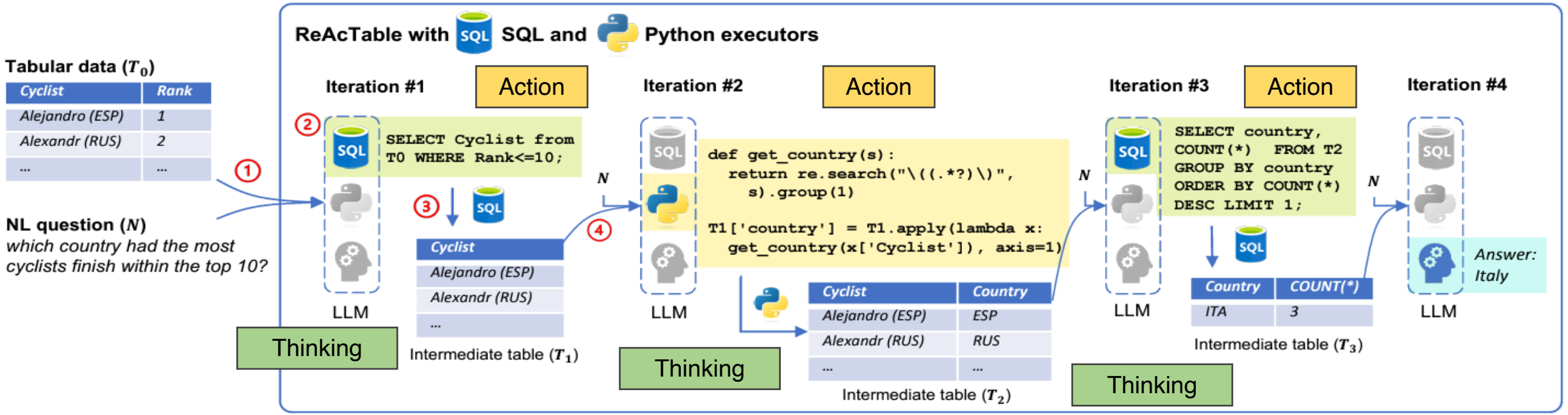


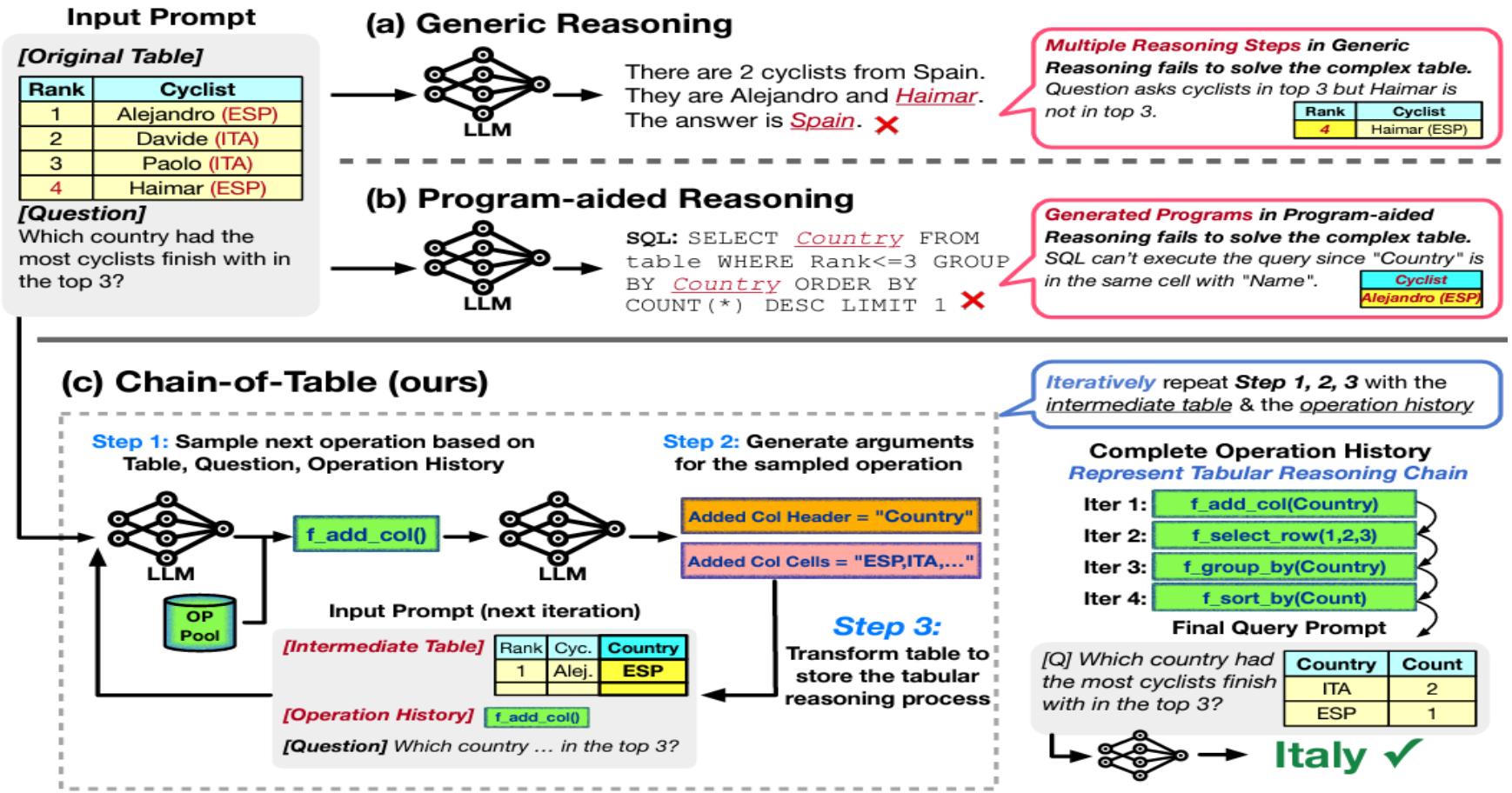
Figure 1: Overview of the ReAcTable framework with SQL and Python code executors.

https://www.vldb.org/pvldb/vol17/p1981-zhang.pdf

Planning: Dynamic Planning, Plan-and-Act

- Chain-of-table [ICLR24]:
  - 全体 Plan 生成
  - 各 Actionの arguments生成
  - それぞれ実行

https://arxiv.org/pdf/2401.04398

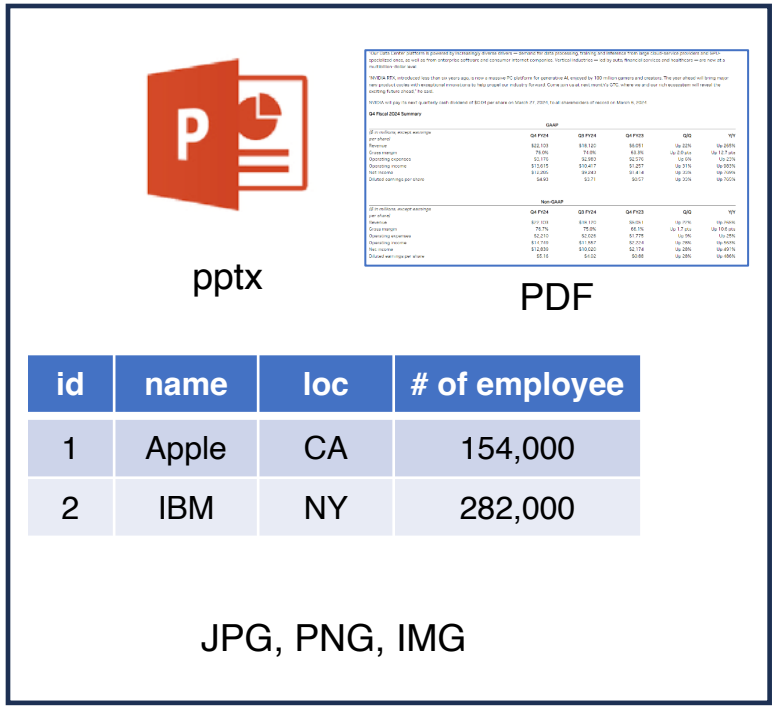


目次

- 背景
- テーブル LLM
- テーブル Agent
- **テーブル VLM**
- まとめと今後の課題

VLMとは?

入力: 画像 (image), question (text)
出力: Response (text)



pptx

PDF

id	name	loc	# of employee
1	Apple	CA	154,000
2	IBM	NY	282,000

JPG, PNG, IMG

VLM

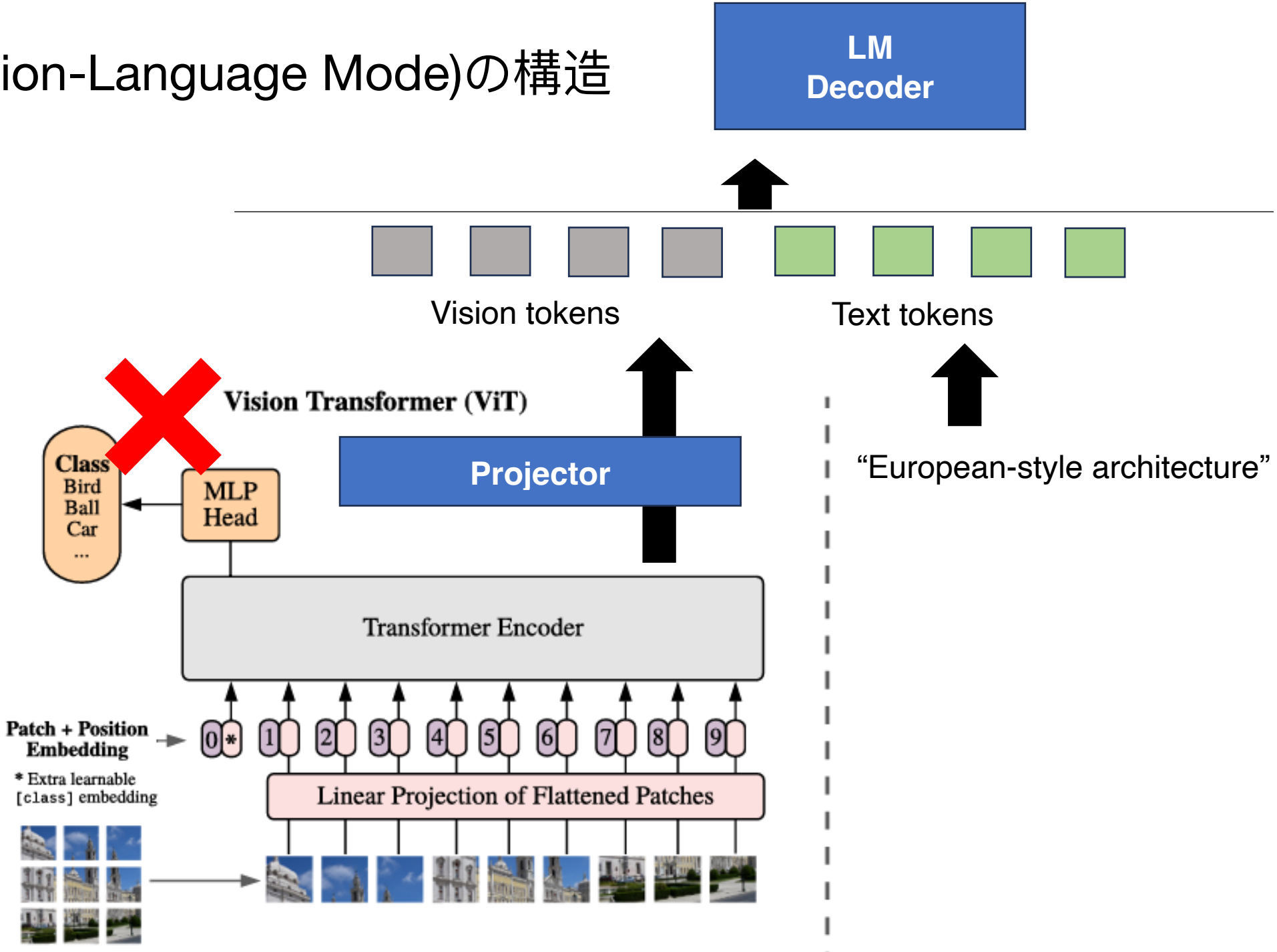
Response

The number of employees for Apple, as shown in the table, is 154,000.

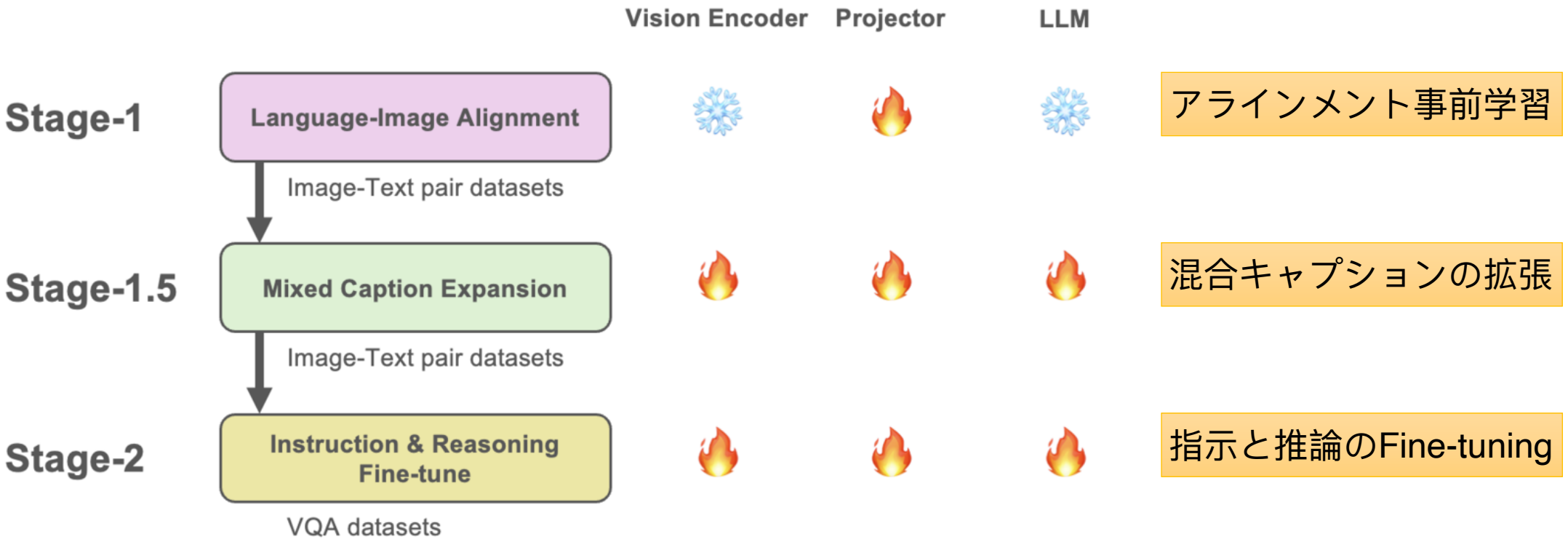
Prompt

what is the employee number of Apple?

VLM (Vision-Language Mode)の構造



VLM training

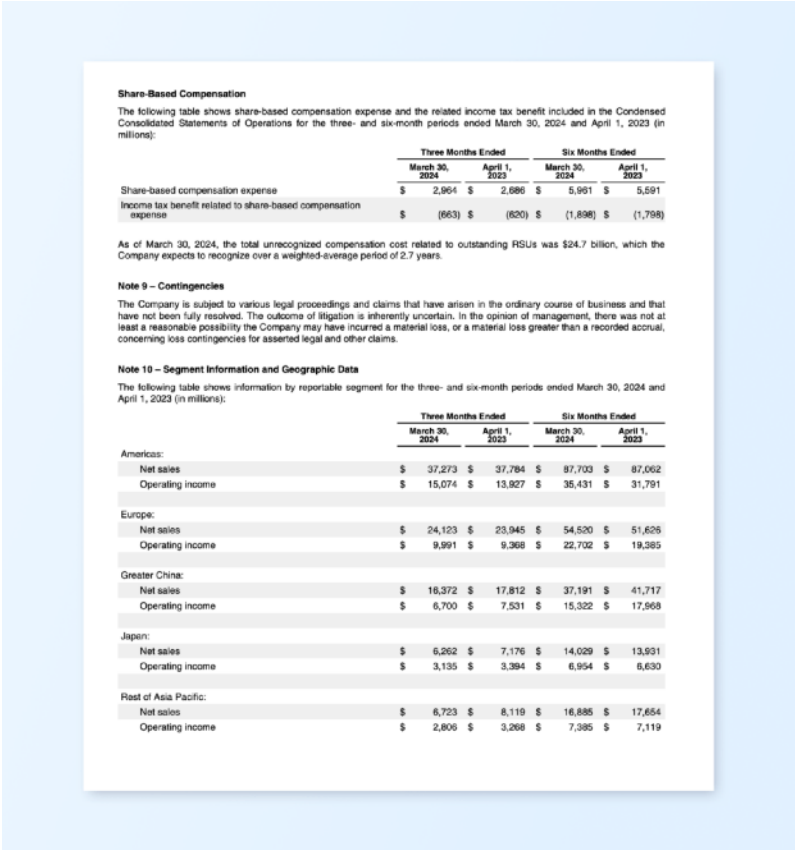


From “Stockmark-2-VL-100B：日本語に特化したドキュメント読解のためのChain-of-Thought視覚言語モデル”

https://stockmark-tech.hatenablog.com/entry/2025/06/03/101007

なぜ VLM使うのか?

- 多くのテーブルデータは 人間のためにデザインされている
  - markdown, JSON, HTML, latex, etc… などのテキストに表現できない
  - 視覚要素 (図、icon, markdown, 色)も含める
- 視覚的に理解するのは自然



Share-Based Compensation

The following table shows share-based compensation expense and the related income tax benefit included in the Condensed Consolidated Statements of Operations for the three- and six-month periods ended March 30, 2024 and April 1, 2023 (in millions):

	Three Months Ended		Six Months Ended	
	March 30, 2024	April 1, 2023	March 30, 2024	April 1, 2023
Share-based compensation expense	$ 2,964	$ 2,686	$ 5,961	$ 5,591
Income tax benefit related to share-based compensation expense	$ (663)	$ (620)	$ (1,898)	$ (1,798)

As of March 30, 2024, the total unrecognized compensation cost related to outstanding RSUs was $24.7 billion, which the Company expects to recognize over a weighted-average period of 2.7 years.

Note 9 – Contingencies

The Company is subject to various legal proceedings and claims that have arisen in the ordinary course of business and that have not been fully resolved. The outcome of litigation is inherently uncertain. In the opinion of management, there was not at least a reasonable possibility the Company may have incurred a material loss, or a material loss greater than a recorded accrual, concerning loss contingencies for asserted legal and other claims.

Note 10 – Segment Information and Geographic Data

The following table shows information by reportable segment for the three- and six-month periods ended March 30, 2024 and April 1, 2023 (in millions):

	Three Months Ended		Six Months Ended	
	March 30, 2024	April 1, 2023	March 30, 2024	April 1, 2023
Americas:				
Net sales	$ 37,273	$ 37,784	$ 87,703	$ 87,062
Operating income	$ 15,074	$ 13,927	$ 35,431	$ 31,791
Europe:				
Net sales	$ 24,123	$ 23,945	$ 54,520	$ 51,626
Operating income	$ 9,991	$ 9,368	$ 22,702	$ 19,385
Greater China:				
Net sales	$ 16,372	$ 17,812	$ 37,191	$ 41,717
Operating income	$ 6,700	$ 7,531	$ 15,322	$ 17,968
Japan:				
Net sales	$ 6,262	$ 7,176	$ 14,029	$ 13,931
Operating income	$ 3,135	$ 3,394	$ 6,954	$ 6,630
Rest of Asia Pacific:				
Net sales	$ 6,723	$ 8,119	$ 16,885	$ 17,654
Operating income	$ 2,806	$ 3,268	$ 7,385	$ 7,119

FEATURE COMPARISON

vertex42

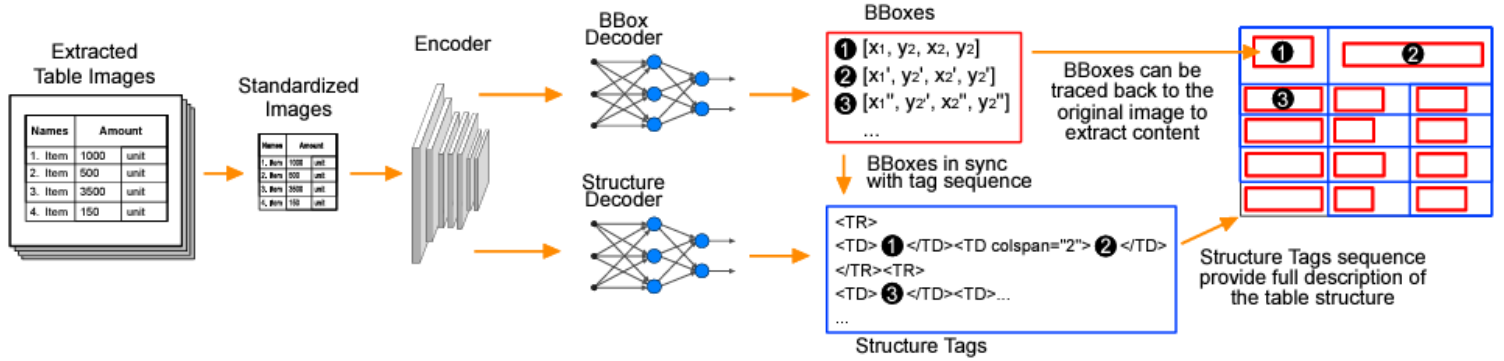
FEATURES	PRODUCT NAME 1	PRODUCT NAME 2	PRODUCT NAME 3	PRODUCT NAME 4	PRODUCT NAME 5	PRODUCT NAME 6
PRICE	FREE	$25-$30	$45-$60	$199	FREE	$500
Unicode ✓ or × symbols	✓	×	✓		×	✓
Custom Icon Sets	○	●	●	●		●
Up / Down Arrow Icons	▼	▬	▲		▼	▲
Check / X Icons	✖	!	✓	✓		✓
Flag Icons	⚑	⚑	⚑	⚑	⚑	
Ratings						
Unicode ★ symbol	★★★★	★	★★	★★★		★★★★★
Diamond ◆ symbol	◆	◆◆	◆◆◆	◆◆	◆	◆◆◆
Icon Set: Star (0,1,2)	☆	☆	☆	☆		☆
Icon Set: Quarter Circle (0,1,2,3,4)	○	◔	◑	◕	○	●
Icon Set: Boxes (0,1,2,3,4)	▯	▯	▯	▯		▯
Icon Set: Bars (0,1,2,3,4)	▯	▯	▯	▯	▯	▯
Numeric & Text Specifications						
Data Bars	30	25	15	25	5	90
Custom number formats	128 GB	64 GB	256 GB	512 GB	32 GB	1024 GB
Basic Numeric Entry	4.23	1.23	5	6.2	1.54	3.52
Yes / No / na	No	Yes	No	Yes	na	Yes
Numbers with Different Units	2.3 TB	470 GB	125 GB	64 GB	512 MB	128 GB

VLM時代の前のモデル

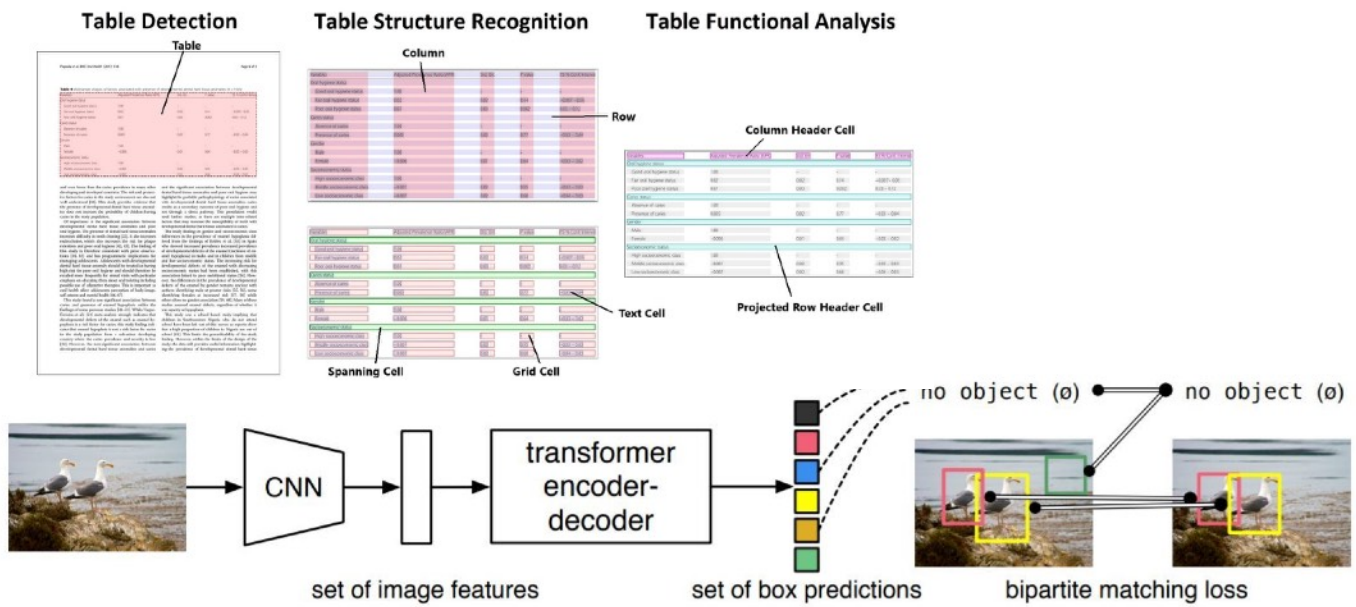
- Table detection
- TableFormer [CVPR22]
  - Bounding box detection, structure generation

単独タスクのためのモデルが多い

- TATR [CVPR22]
  - Based on object detection transformer (DETR)



TableFormer

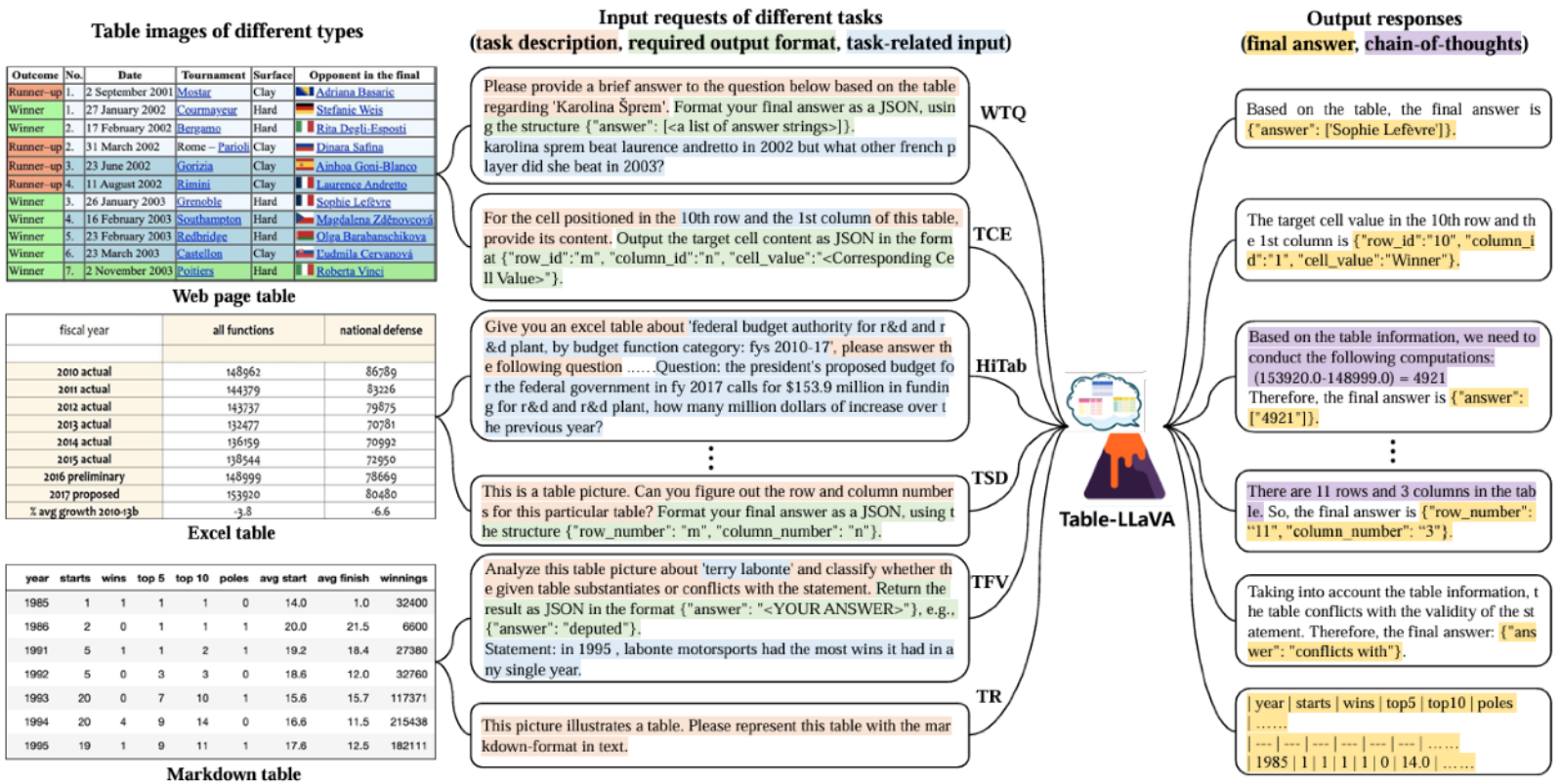


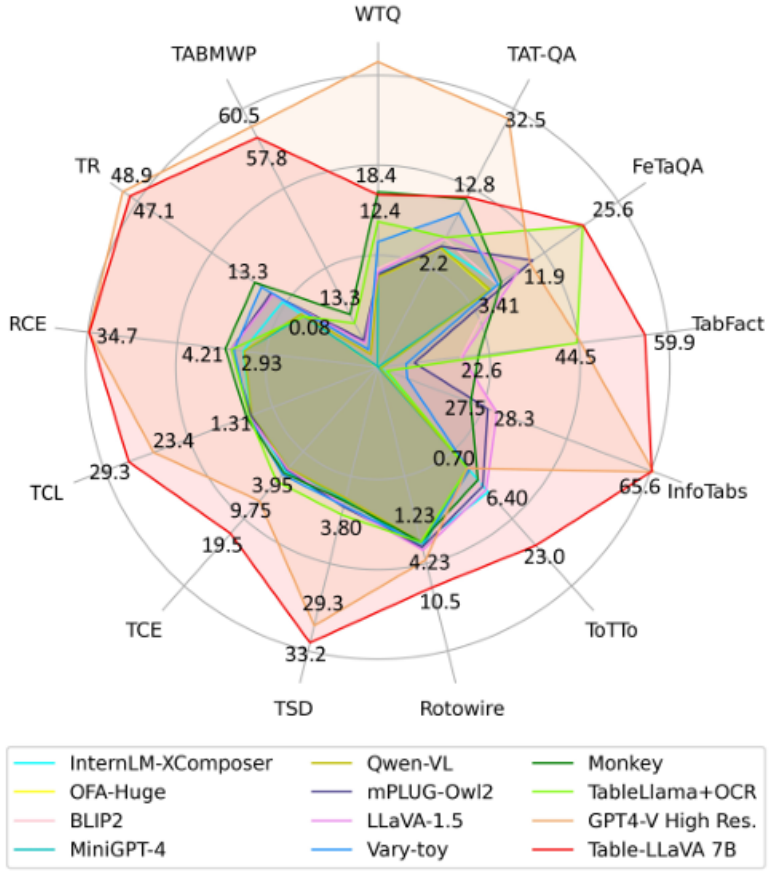
TATR(DETR)

Tableformer: https://arxiv.org/abs/2203.01017
TATR: https://github.com/microsoft/table-transformer

汎用モデル: Table-Llava (ACL24)

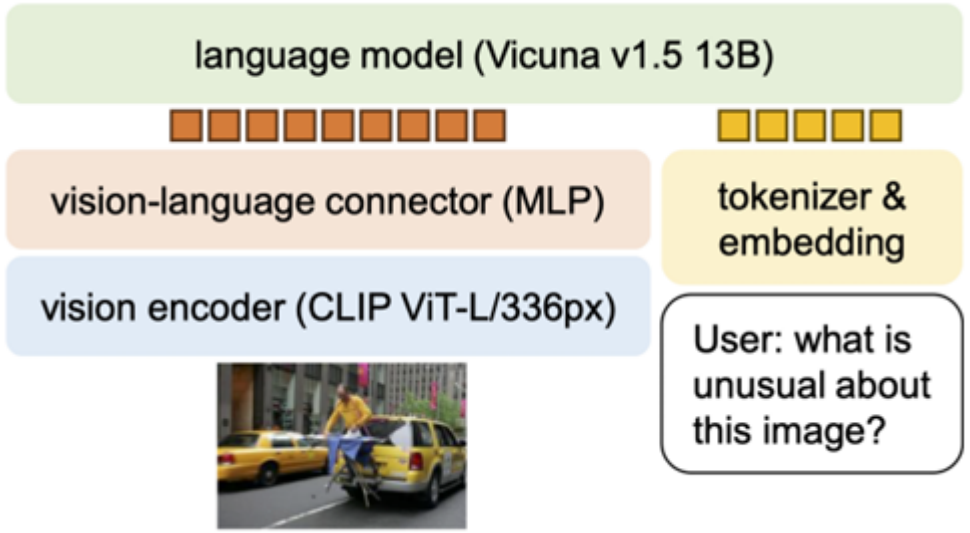
- たくさんタスクデータで学習してきたVLMモデル





https://arxiv.org/abs/2406.08100

Table-Llava (ACL24)



Llava architecture

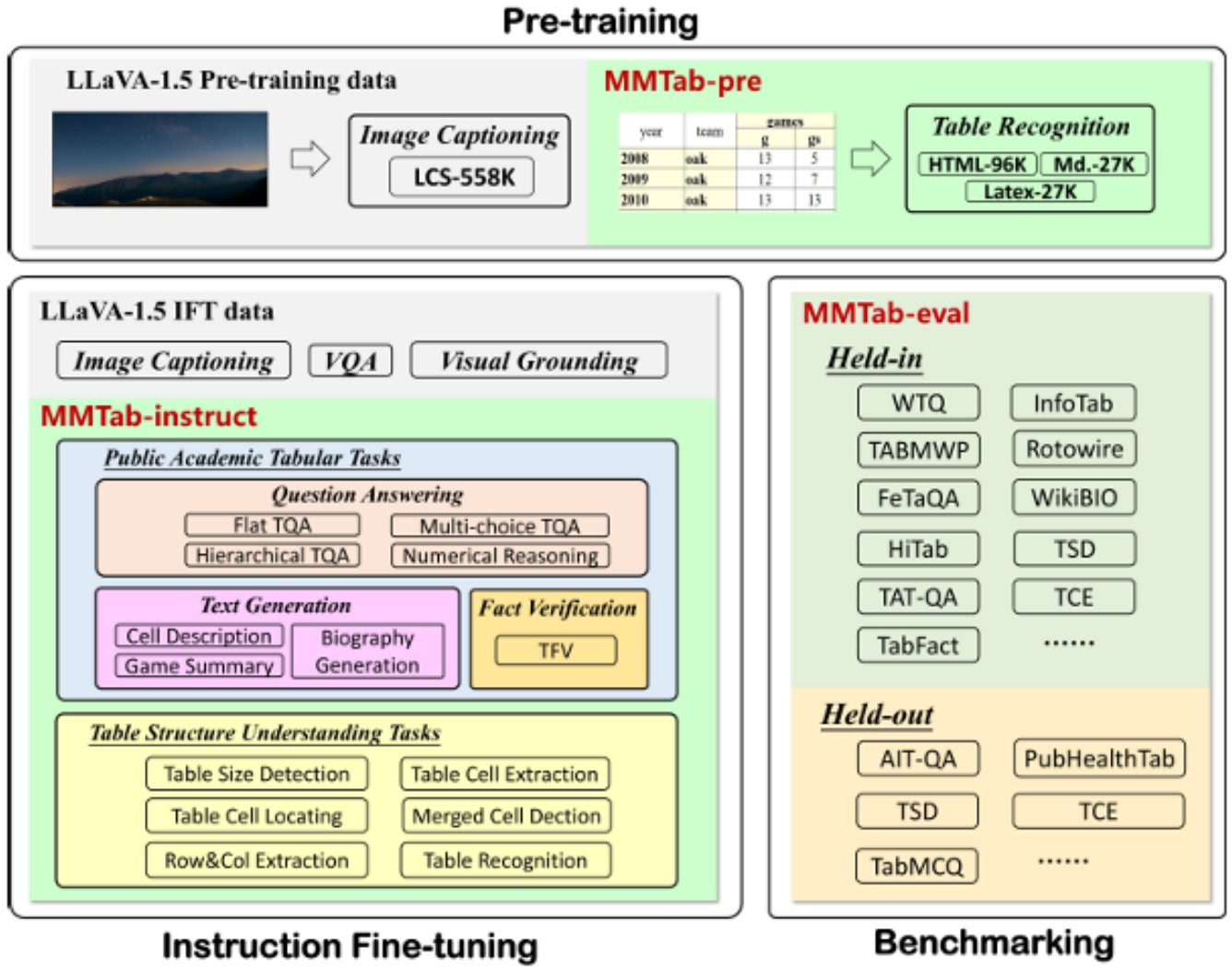


Table-Llava training

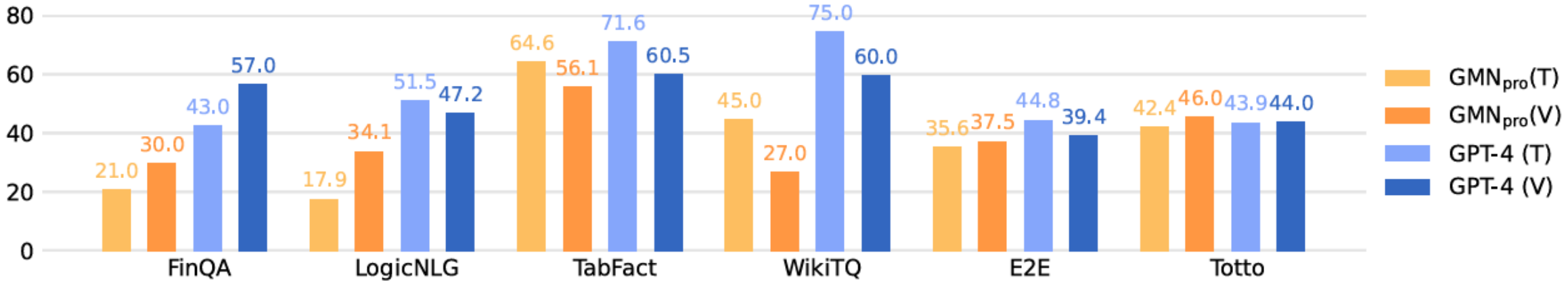
Table VLMの発展

	Vision encoder	LM decoder backbone	Task
Tableformer [CVPR22]	Resnet-18(CNN) + transformer encoder (2 layers)	Scratch transformer decoder (4 layers)	Table detection & structure recognition
TableVLM [ACL23]	ResNeXt101 (CNN) + transformer encoder (12 layers)	Scratch transformer decoder (4 layers)	Table detection & structure recognition
PixT3 [ACL24]	Pix2Strct (ViT)	Pix2Strct	Table-to-text
TablePedia [arxiv24]	ViT-L (ViT) Swin-B (ViT)	Vicuna-7B	Table detection & structure recognition TableQA
Table-Llava [ACL24]	CLIP (ViT)	Vicuna-v1.5-7B	Table-to-text TableQA Table structure recognition

Vision Encoder も LM decoder も 大量データ事前学習済みモデルにベース

Table as Texts or Images (ACL24)

- 同じデータをテキストで入れると精度が高いのか?　画像で精度高いのか?



Text (vanilla-T) vs Image (vanilla-V)

- 結論
  - FinQA (rich document) 以外に テキストの方が良い
  - テキスト、画像両方入れると、常に良いわけではない
  - Closed model > open model

Datasets	T+V	T	V	Metric
WikiTQ	80.0	75.0	60.0	Acc
TabFact	64.0	71.6	60.5	
LogicNLG	48.0	51.5	54.1	
FinQA	61.0	43.0	57.0	
ToTTo	42.4	43.9	44.0	ROUGE-L
E2E	42.6	44.8	39.4	

Table 19: GPT-4's performance when we pass the text representation (T), image representation (V) and both representation (T+V) to the model.

TableVQA-Bench

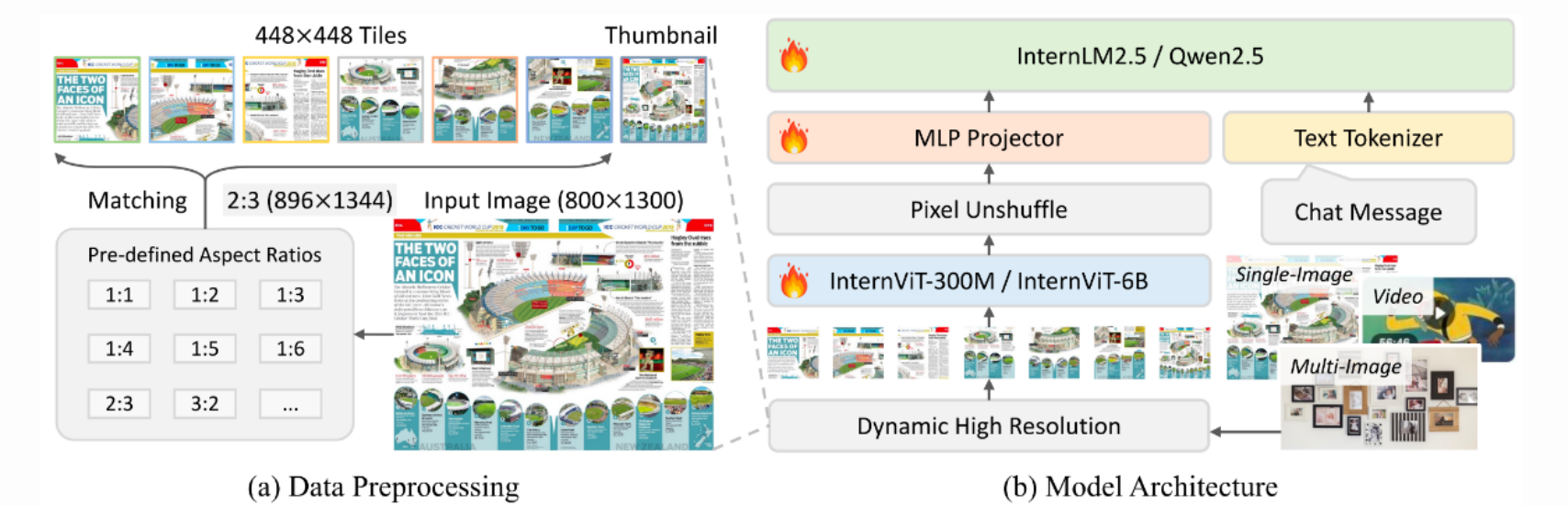
- 結論
  - VLMはまだ発展する必要
  - VLMテキスト -> LLM QAの方が良い

Input Modality	Model	VWTQ	VWTQ-Syn	VTabFact	FinTabNetQA	Avg.
Multi-modal Large Language Models (MLLMs)						
Vision	GPT-4V [1]	**42.5**	**52.0**	**68.0**	**79.6**	**54.5**
	Gemini-ProV [23]	26.7	33.2	55.6	60.8	38.3
	SPHINX-MoE-1k	27.2	33.6	61.6	36.0	35.5
	SPHINX-v2-1k	25.3	28.0	66.8	31.2	33.7
	QWEN-VL-Chat [3]	19.0	23.2	60.4	29.6	28.4
	QWEN-VL [3]	17.2	21.2	52.0	34.0	26.5
	SPHINX-MoE	15.3	16.8	58.8	2.8	20.7
	SPHINX-v1-1k [14]	13.2	17.2	58.0	3.2	19.7
	mPLUG-Owl2 [26]	10.7	14.4	56.8	2.8	17.7
	LLaVA-1.5 [16]	12.4	12.4	55.6	0.8	17.7
	CogVLM-1k [24]	9.7	11.6	52.0	4.8	16.3
	SPHINX-v1 [14]	7.1	9.6	55.2	1.2	14.5
	CogAgent-VQA [7]	0.3	0.8	58.4	22.8	13.8
	InstructBLIP [6]	5.9	6.4	50.4	0.4	12.5
	BLIP-2 [13]	5.2	5.6	51.6	0.4	12.2
	CogVLM [24]	0.8	0.8	40.8	1.2	7.5
	CogAgent-VQA* [7]	37.2	41.2	58.4	22.8	39.0
Table Structure Reconstruction + Large Language Models (LLMs)						
Vision	GPT-4V [1] → GPT-4 [2]	**45.2**	**55.6**	**78.0**	**95.2**	**60.7**
	Gemini-ProV → Gemini-Pro [23]	34.8	40.4	71.0	75.6	48.6
Large Language Models (LLMs)						
Text	GPT-4 [2]	**68.1**	**69.6**	**80.0**	**98.8**	**75.5**
	Gemini-Pro [23]	56.4	61.2	69.6	96.4	66.1
	GPT-3.5	50.5	54.4	68.0	93.2	61.2
	Vicuna-13B [5]	32.8	39.2	57.6	84.8	46.7
	Vicuna-7B [5]	21.5	34.4	54.0	68.8	37.0

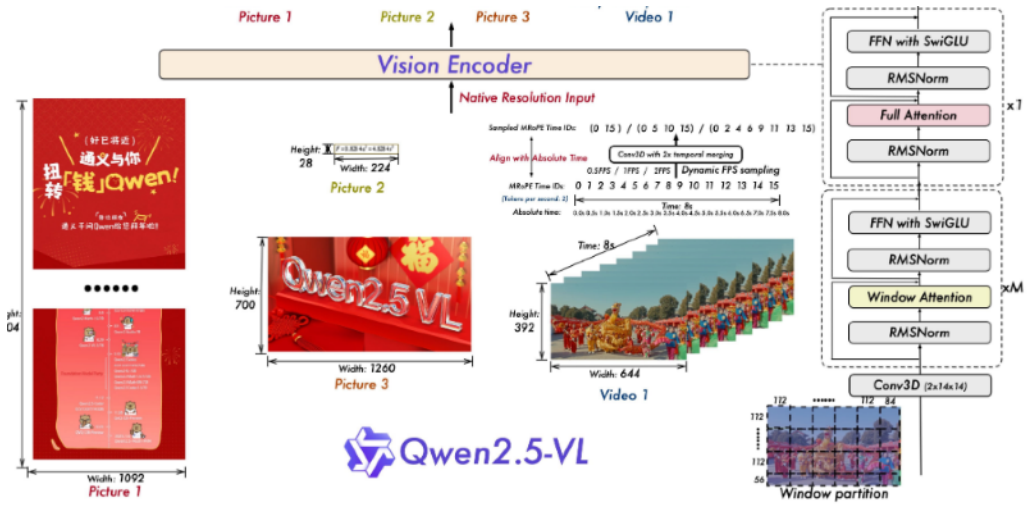
https://arxiv.org/pdf/2404.19205

Openモデルでもどんどん強いVLMが出てくる

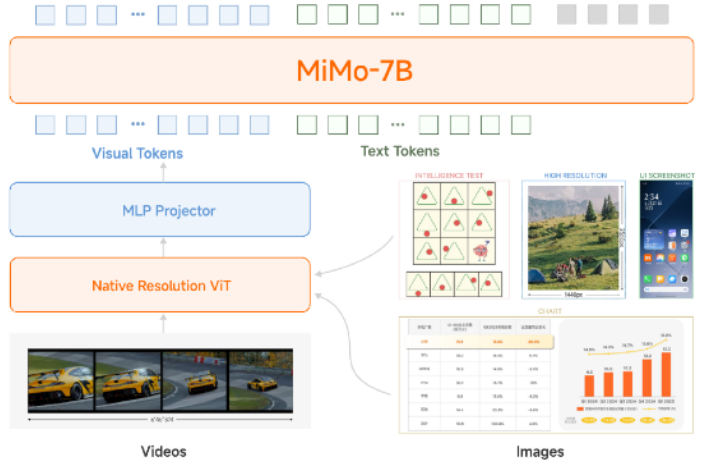
InternVL3
(2025.04)



Qwen2.5-VL
(2025.01)



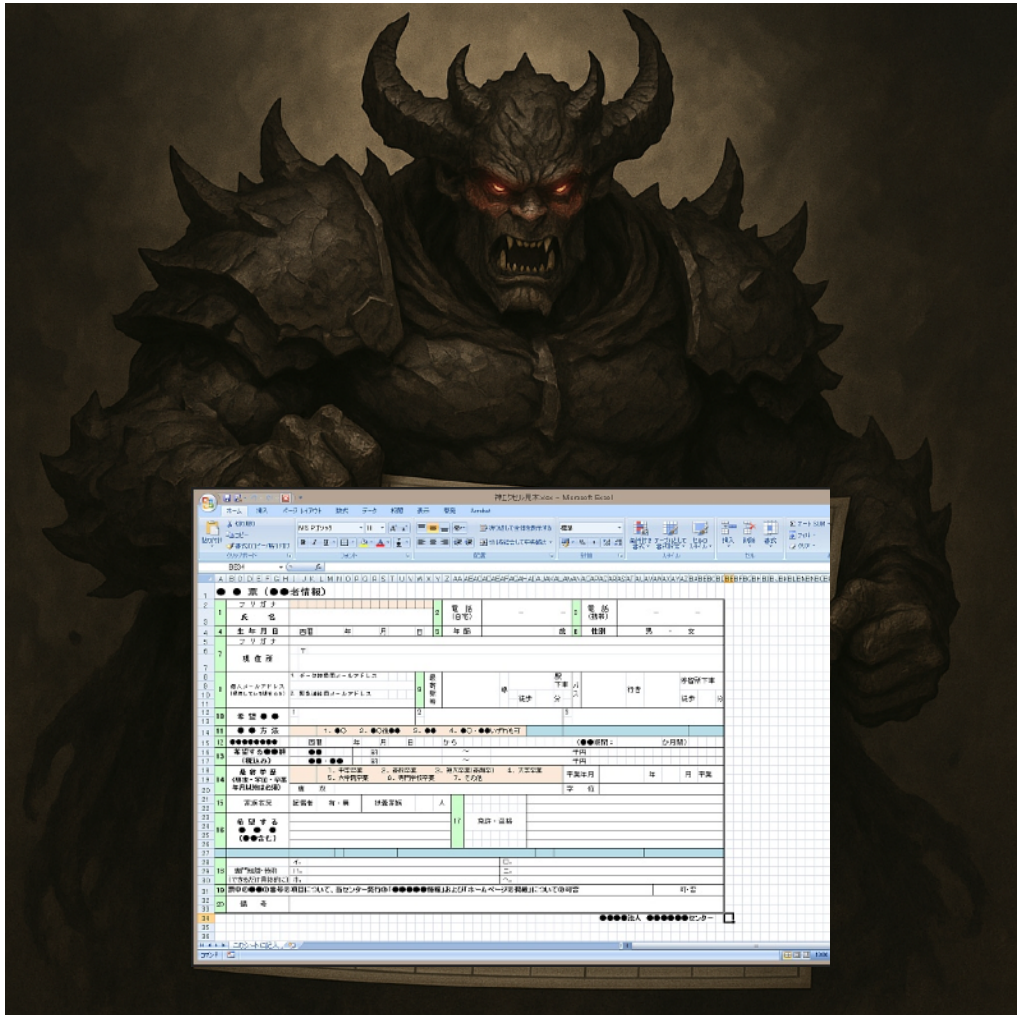
MiMo-VL
(2025.06)



まとめ

- 背景
  - 表形式データはよくある、データベース以外のテーブルデータの処理は課題
  - LLMはテキスト、画像扱いが得意なので、LLMを用いて活用するのは良い
- テーブル LLM
  - 追加学習するために、テーブルの入れ方、データ品質、データの多様性が大事
- テーブル Agent
  - やはり一つのモデルで、一回だけの推論では複雑なタスクを解けないため
  - Agentで タスク分解、ツール活用 (RAG, Python), Planningも工夫する必要
- テーブル VLM
  - 視覚の要素も含めたテーブルは VLMの力が必要
  - まだまだテキストの方が強い、発展する必要

ラスボスは? 。。。。。神エクセル!



- VLMモデルの発展
- Visual Agent Systemの発展
- Table LLM/VLM reasoning技術
- テーブル識別技術
- Document Layout Analysis 技術

が進化して協力すれば撃退できるのか? ? ? ? ?