Retrieval Systems for Structured Data

The critical missing  for coupling LLM-powered query interfaces for factual data

Madelon Hulsebos

BIDS Seminar
October 2024

Finding the right data for basic questions or deep analysis is not easy.

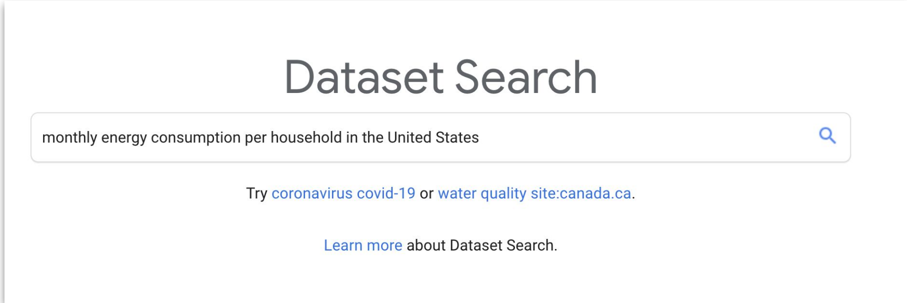
1. How to help data analysts find their data for analytics tasks?
2. How to ground LLM-powered query interfaces in structured data?

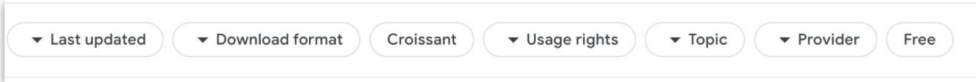
Part 1:
How to help data analysts find their data for analytics tasks?

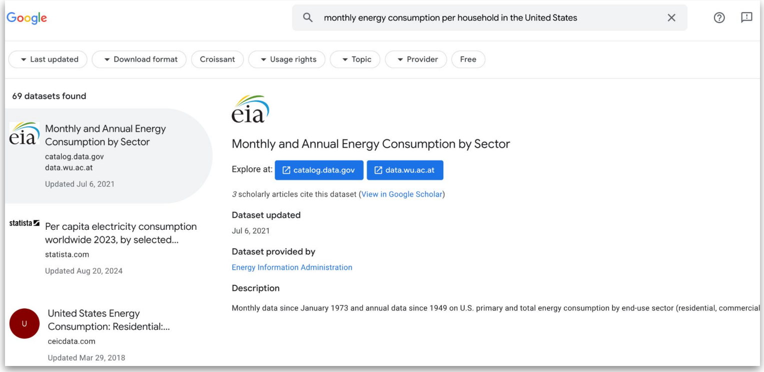
Current dataset search systems for analytics use-cases

Imagine you are looking for a dataset to forecast monthly energy consumption per household until May 2025.

Keyword Search and Metadata Filters





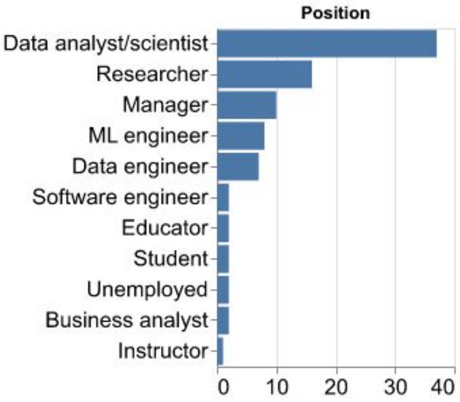


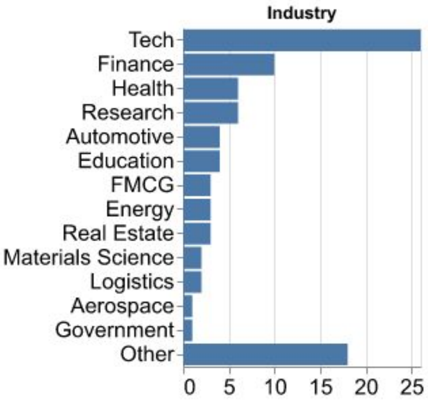
69 datasets… Which one?

BIDS BERKELEY INSTITUTE FOR DATA SCIENCE
EPIC DATA lab

Why is dataset search still so hard in practice?

89 data practitioners recruited through social media & mailing lists

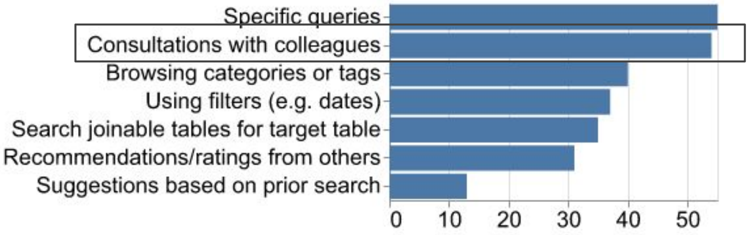




"It took longer than I was expecting": Why is Dataset Search Still So Hard?, Hulsebos, M., Lin, W., Shankar, S., Parameswaran, P., HILDA@SIGMOD 2024.

How do Practitioners Search?

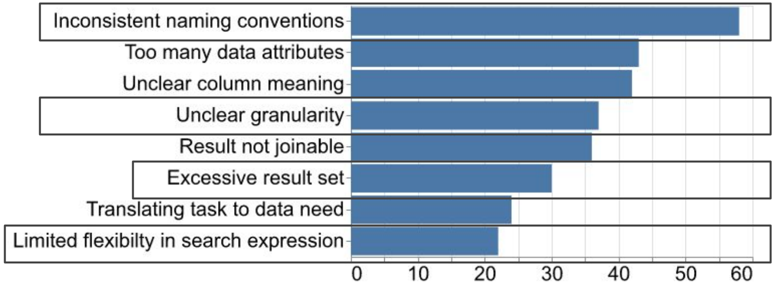
How do you search?



*"…Also, identify **people** who have worked on **similar problems**…"*

*"… I ask more **experienced colleagues which ones are most inherent** to the analysis..."*

Key challenges with existing systems?



*"The biggest challenge I've noticed is **messy variable naming**…."*

*"**Categorical level of detailing** is required..."*

*"There are **too many table results** after the initial search…."*

*"**Not many features to search**/query keywords…"*

BIDS BERKELEY INSTITUTE FOR DATA SCIENCE EPIC DATA lab

"It took longer than I was expecting": Why is Dataset Search Still So Hard?, Hulsebos, M., Lin, W., Shankar, S., Parameswaran, P., HILDA@SIGMOD 2024.

Towards Next-Generation Dataset Search Systems

Task-driven: explicit **data needs unknown to user**, start from the task

Hybrid: search spans **diff representations** of a table; raw metadata + **semantics**

Iterative: beyond keywords+filters; complex process, need for more pruning

Comprehensibility+diversity: result sets hard to **digest and navigate; IR principles?**

"It took longer than I was expecting": Why is Dataset Search Still So Hard?, Hulsebos, M., Lin, W., Shankar, S., Parameswaran, P., HILDA@SIGMOD 2024.

Task-driven Search with Hypothetical Schema Embeddings

(1) Task-driven query

What data is needed to **train a machine learning model** to **forecast demand for medicines across suppliers?**

(2) Hypothetical Schema generation

```
Instruction: generate schema needed for the given task query.
Query: {task-driven query}
```

LLM output: "`hypothetical_schema`"

```
medication table: medication id, medication name, …
sales table: medication id, supplier id, date, quantity sold, …
```

(3) Embed(`hypothetical_schema`)

(4) Retrieve source tables from vector store similar to `hypothetical_schema`

BIDS BERKELEY INSTITUTE FOR DATA SCIENCE EPIC DATA lab

Iterative Search through Query Breakdown

Initial complex search query

A dataset to train a **medicine forecast model**, should **contain a, b, c**, and **span 2 years** with a **date range e to b**.
The fact table should be at **r granularity** and contain **20,000 records**.

"A dataset for **task x**,

<retrieved tables>

Retrieve through Hypothetical Schema Embeddings

Data should contain **a , b , c**

<pruned tables>

prune: schema similarity

Data should span **2 years** with a **date range e to b**.

<pruned tables>

prune: metadata **text-to-sql**

The fact table should be at **r granularity** and **contain 20,000 records**."

<pruned tables>

prune: metadata **text-to-sql**

LLM-Assisted Interface for Dataset Search

Things we want from Dataset Search interfaces:

D1 **LLM Elicitation through Proactive Guidance**

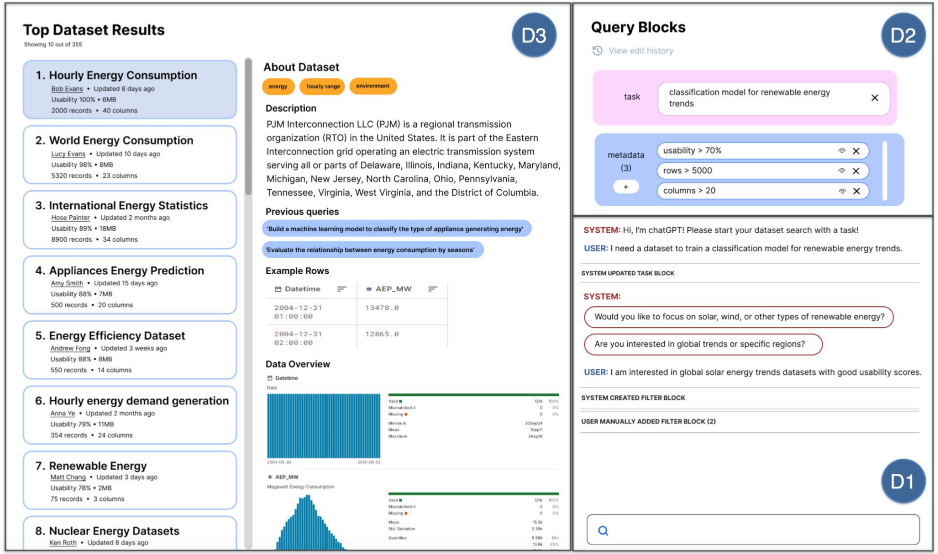
Purpose: Prompt users to share more information about their needs, which will be reflected in the query blocks & search interface.

D2 **Dynamic Query Decomposition**

Purpose: Allow users to see how the LLM is dynamically updating and refining the search space, providing transparency into the search process.

D3 **Allowing Users to Compare Datasets Efficiently**

Purpose: Facilitate high-level exploration of datasets by organizing them into topics and enable users to delve into metadata details of individual datasets as they iteratively build and refine their queries.



BIDS BERKELEY INSTITUTE FOR DATA SCIENCE

EPIC DATA lab

Part 2:
How to ground LLM-powered query interfaces in structured data?

Asking LLMs complex questions

What is the highest eligible free rate for K-12 students in the schools in Alameda County?

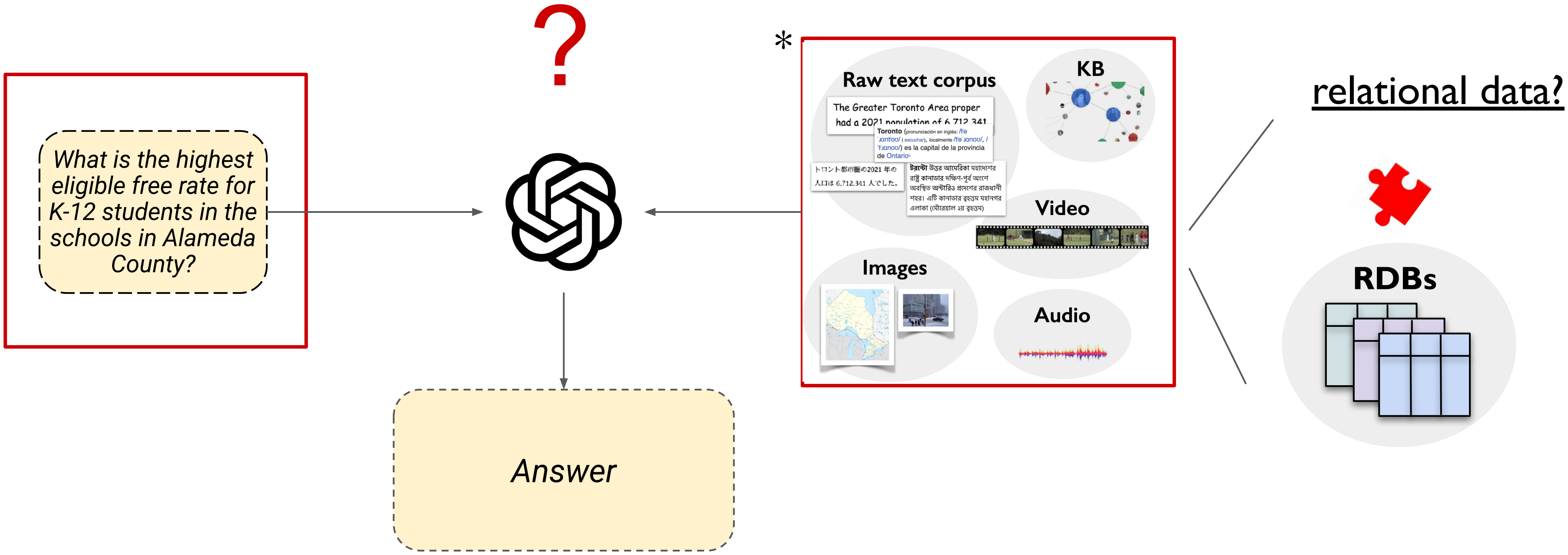
→

→

*"…. To determine the highest free rate specifically in Alameda County schools, **you'd generally need data from specific school districts or schools in the area**, as this rate can vary widely depending on the socio-economic demographics of each district. …" **

*Generated with chatgpt on 6 October 2024

We need “specific” data to ground LLMs



*Retrieval-based Language Models and Applications, Asai, A. et al, Tutorial ACL, 2023.

Why we need RAG over *structured data*

A pattern in practice: "everyone cares about structured data".

Structured data serve **high-value** insights! Up-to-date, domain-specific, facts…

Queries & RAG pipeline

“Which urban Japanese prefecture is not associated with thorny trees?” [table lookup]

“Shane Hall ran a total of 190 races between the year of 1995 - 2008” [aggregate & compare]

“What is the highest eligible free rate for K-12 students in the schools in Alameda County” [aggregate]

- interpret query
- enrich/transform query

how to retrieve tables?

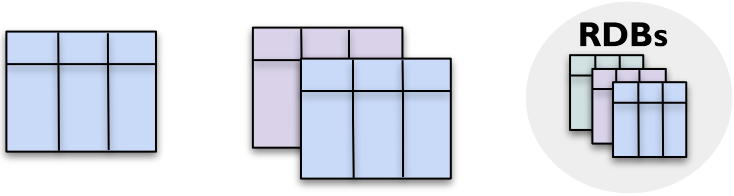
- read table with LLM
- generation & SQL execution

Retrieval is difficult, but crucial…

"... keep in mind that a good RAG system is really **hard to build**.
If your **retrieval system is mediocre**,
the **retrieval can easily distract LLMs** to backfire…
There is still a long way to go." **-** Wenhu Chen (Univ of Waterloo)

Important grounds to explore…

Retrieval/generation complexity depends on query



- What "task" does the **query intend** to do?
- How should we **process** the table(s), relational DBs?
- **What should we embed** of the table(s) and metadata, and **how**?
- Given query and embedded corpus, **how to retrieve relevant table**?
- **Which data source** to retrieve from, and when?
- (How) should methods, models, systems **generalize across tasks, datasets**?

Methods for table retrieval

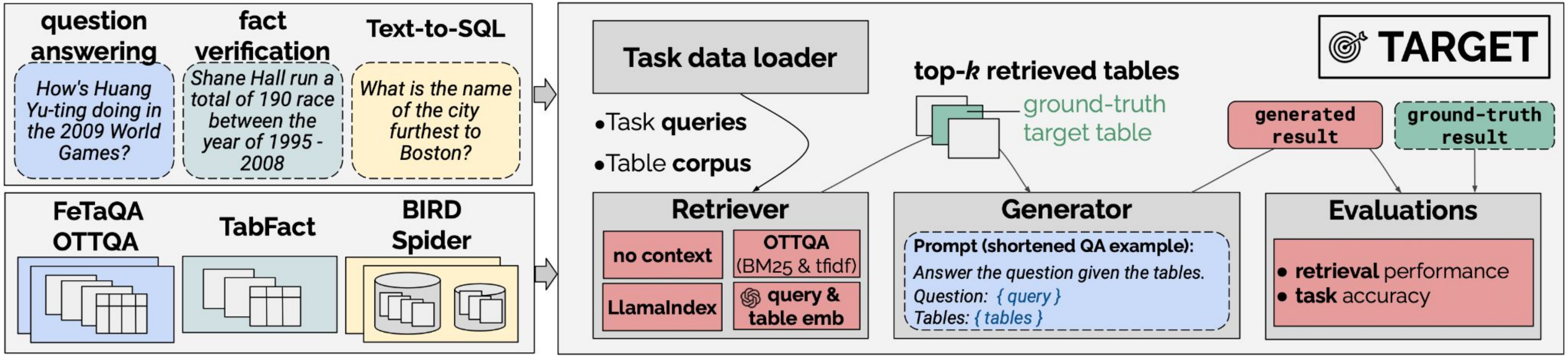
① Embedding of tables in corpus, and input query

- BM25 / TF-IDF (sparse lexical representations)
- Generate summary/metadata → embed summary + table
- "Naive" embedding of table (header / header+rows) and query

② Similarity search (e.g. cosine similarity) to identify top-*k* relevant tables

But how effective are these? How robust across datasets and tasks? No one really knows!

TARGET: Benchmarking Table Retrieval for Generative Tasks



- Diverse: tasks & datasets
- Extensible: easily add new tasks, new datasets
- Adaptable: eval custom retriever, generator

Ji, X, Parameswaran, A., Hulsebos, M., "TARGET: benchmarking Table Retrieval for Generative Tasks", Table Representation Learning workshop NeurIPS, 2024.

TARGET insights

Method	Question Answering						Fact Verification			Text-to-SQL					
	OTTQA			FeTaQA			TabFact			Spider			BIRD		
	R@10	s	SB	R@10	s	SB	R@10	s	P/R/F1	R@1	s	EX	R@1	s	EX
No context	-	-	0.414	-	-	12.495	-	-	0.578/0.42/0.44	-	-	0	-	-	0
OTT-QA BM25	**0.955**	0.001	0.606	0.082	0.001	1.631	0.338	0.001	0.75/0.26/0.39	0.635	0.001	0.385	0.709	0.001	0.181
w/o table title	0.443	0.001	0.529	0.084	0.001	1.555	0.331	0.001	0.75/0.26/0.38	0.5	0.001	0.376	0.535	0.001	0.164
OTT-QA TF-IDF	0.950	0.001	0.425	0.083	0.001	1.639	0.336	0.001	0.75/0.26/0.38	0.622	0.001	0.474	0.640	0.001	0.227
w/o table title	0.43	0.001	0.593	0.083	0.001	1.527	0.322	0.001	0.75/0.25/0.37	0.492	0.001	0.376	0.491	0.001	0.164
LlamaIndex	0.458	0.354	0.507	0.435	0.396	13.745	**0.827**	0.297	0.73/0.34/0.47	0.735	0.198	0.559	0.937	0.228	0.311
OpenAI embedding	0.950	0.190	0.599	**0.722**	0.200	17.64	0.779	0.189	0.76/0.51/0.61	0.768	0.193	0.602	0.926	0.199	0.317
header only	0.950	0.189	0.61	0.718	0.18	17.66	0.781	0.187	0.75/0.48/0.58	**0.833**	0.175	0.646	**0.958**	0.191	0.323

- BM25/TF-IDF less effective, only with *very* descriptive table name.
- Table rows can “distract” embeddings, *particularly in RDBs as seen in practice.*
- Generating summary/metadata can help, but not all tables easy to LLM-summarize.

Ji, X, Parameswaran, A., Hulsebos, M., “TARGET: benchmarking Table Retrieval for Generative Tasks”, Table Representation Learning workshop NeurIPS, 2024.

Still much to explore…

- What is right input of (meta)data to not "distract" embedding?
- How do we route to proper data source, interpret the task, etc?
- The reality in practice is much harder:
  - How do methods perform on more *challenging tasks & datasets*?
  - Closing semantic gap e(query) and e(table); most public datasets relatively "easy" match between query and tables.
  - Relational databases are large → in-DB schema and table retrieval.

Roadmap for TARGET

Ji, X, Parameswaran, A., Hulsebos, M., "TARGET: benchmarking Table Retrieval for Generative Tasks", Table Representation Learning workshop NeurIPS, 2024.

TARGET is out TODAY!

RAG tables over tables with TARGET!

```python
from target_benchmark.retrievers import AbsCustomEmbeddingRetriever
class YourRetriever(AbsCustomEmbeddingRetriever):
    def __init__(self, **kwargs):
        # load your favorite table retriever!

    def retrieve(self, query: str, dataset_name: str, top_k: int):
        # given a query, retrieve the top-k table id

    def embed_corpus(self, dataset_name: str, corpus: Iterable[Dict]):
        # use retriever, embedding models, etc. to embed the corpus!
```

- Ready to eval table retrieval and e2e generation: **input welcome for v2**
- Data on HF, code on GH, `pip install target_benchmark`
- https://target-benchmark.github.io

Ji, X, Parameswaran, A., Hulsebos, M., "TARGET: benchmarking Table Retrieval for Generative Tasks", Table Representation Learning workshop NeurIPS, 2024.

Key takeaways

- Retrieval (RAG, agents, or dataset discovery) is critical to get insights from data.

- Dataset discovery requires co-worker consultation. Existing interfaces are inflexible. We propose lightweight retrieval w **Hypothetical Schema Embeddings**, and a **flexible LLM-assisted interface** that takes a *task* and *doesn't fix metadata filters*.

- To push table retrieval for grounding LLMs in structured data forward, we introduce 🎯 **TARGET: the first benchmark for RAG for structured data**, and uncover deficits of methods that are robust for unstructured text! More to come…