

What Table Representation Learning Brings to Data Systems

Madelon Hulsebos

ETH Zürich
5 December 2024

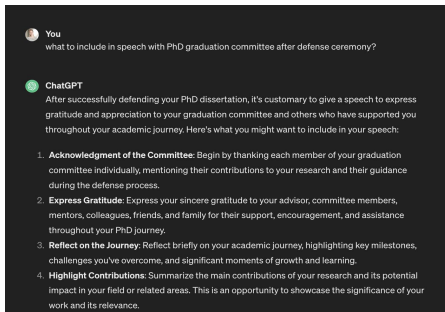
The logo for the Centrum voor Wiskunde en Informatica (CWI) is located in the bottom left corner. It consists of a red parallelogram with the white text "CWI" inside.

The Impressive Capabilities of Transformers

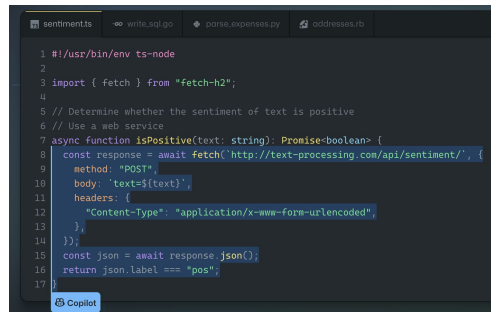
Transformer-architecture leveraged for applications over images, text, code:



Generating funny dog images



Writing graduation speech

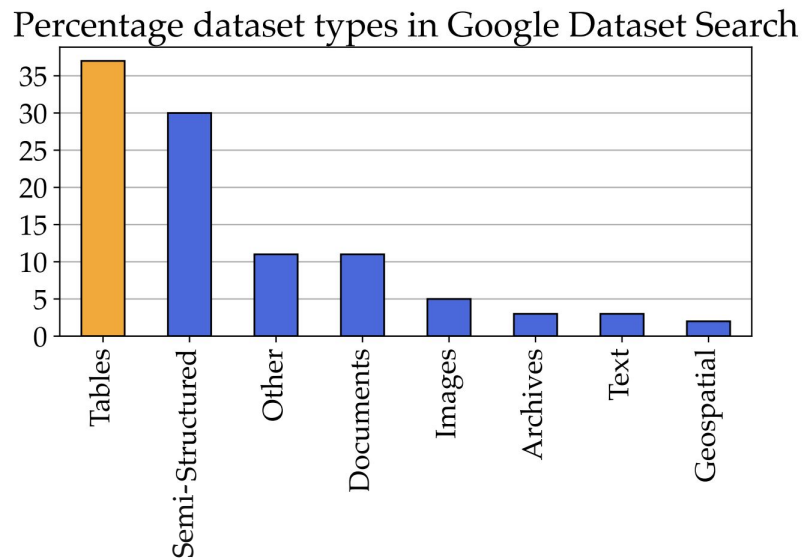
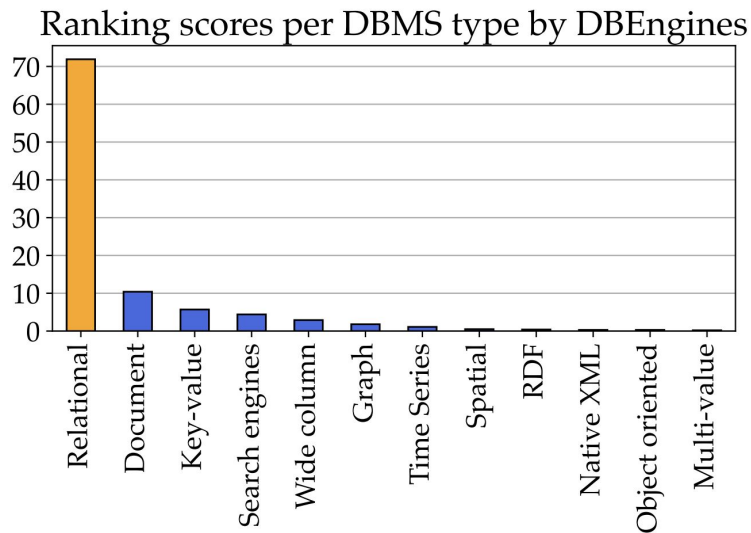


Completing code

What about tables?

We have LLMs... why not analyze docs?

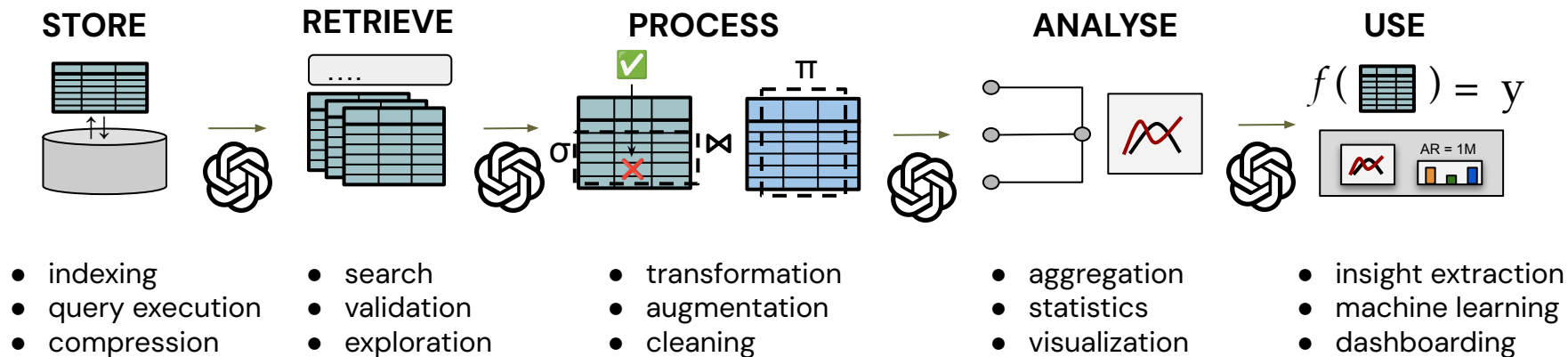
Tables Dominate the Data Landscape



Potential of Table Representation Learning

Available... but also: fresh, structured, domain, data!

High value use-cases, e.g. *data analysis*: many tables, many tasks!



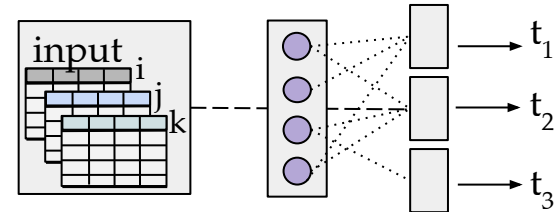
Relational data: rich and challenging

Diverse in dimensions, structure, cleanliness and semantics...

crop rotation : Tabelle												
Nr	ID	seed rate	yield	crop	cultivar	pre crop	pre-pre crop	pre-pre-pre	soil type	precipita	tempera	comment
1	68	91		winter wheat		sugar beets	beans		sandy loam, loe	636	9,6	wb, sg.
2	68	100		winter wheat		sugar beets	rotation fallow		sandy loam, loe	636	9,6	cultivation
3	68	97		winter wheat		sugar beets	fallow land (5,5y)		sandy loam, loe	636	9,6	1993-1996
4	136	95		winter wheat		oats	sugar beets		sandy loam, loe	636	9,6	
5	136	96		winter wheat		potatos	sugar beets		sandy loam, loe	636	9,5	cultivation
6	136	107		winter wheat		sugar beets	maize		sandy loam, loe	636	9,5	1991-1994
7	136	107		winter wheat		sugar beets	summer wheat	maize	sandy loam, loe	636	9,5	
8	136	82		winter wheat		oats	sugar beets	sugar beets	sandy loam, loe	636	9,5	organic
9	136	77		winter wheat		potatos	sugar beets		sandy loam, loe	636	9,5	organic
10	136	85		winter wheat		sugar beets	maize	maize	sandy loam, loe	636	9,5	organic
11	136	84		winter wheat		sugar beets	summer wheat	sugar beets	sandy loam, loe	636	9,5	organic
12	57 371	98		winter wheat	Sperber	sugar beets	winter barley	winter wheat	sandy loam, loe	635		wb, ww
13	57 365	98		winter wheat	Sperber	potatos	sugar beets	summer barley	sandy loam, loe	635		cultivation, weed
14	57 365	105		winter wheat	Sperber	sugar beets	maize	maize	sandy loam, loe	635		1987-1992
15	57 365	97		winter wheat	Sperber	sugar beets	winter wheat	sugar beets	sandy loam, loe	635		
16	39 433	90		winter wheat	Okapi	summer barley			sandy loam, loe	690	8,5	oats, cultivation, weec
17	39 433	100		winter wheat	Okapi	oats			clay, silt	690	8,5	1982-1986
18	39 433	97		winter wheat	Okapi	winter wheat			clay, silt	690	8,5	

	2019			data		
	Profit	Quantity	Sales	Quantity	Sales	Profit
Overall	128.9K	13.3K	1.0M	128.9K	13.3K	1.0M
France	35.1K	3.9K	306.4K	35.1K	3.9K	306.4K
Austria	7.5K	24.6K	24.6K	7.5K	24.6K	24.6K
Belgium	4.2K	202.0	17.3K	4.2K	202.0	17.3K
Denmark	1.3K	86.0	2.8K	1.3K	86.0	2.8K
Zaaland	245.0	15.0	242.7	245.0	15.0	242.7
South Denmark	362.9	40.0	1.3K	362.9	40.0	1.3K
Sonderborg	45.4	6.0	87.5	45.4	6.0	87.5
Odense	280.3	30.0	1.1K	280.3	30.0	1.1K
Estbjerg	37.3	4.0	88.7	37.3	4.0	88.7
Hovedstaden	0.7K	31.0	1.3K	0.7K	31.0	1.3K
Copenhagen	0.7K	31.0	1.3K	0.7K	31.0	1.3K
Frederiksberg	232.1	11.0	1.1K	232.1	11.0	1.1K
Finland	36.0K	2.4K	216.5K	36.0K	2.4K	216.5K
Germany	3.9K	7.2K	109.7K	3.9K	7.2K	109.7K
Ireland	10.6K	1.4K	109.7K	10.6K	1.4K	109.7K
Italy	11.6K	0.6K	23.7K	11.6K	0.6K	23.7K
Netherlands	2.9K	135.0	12.9K	2.9K	135.0	12.9K
Norway	1.0K	74.0	1.8K	1.0K	74.0	1.8K
Portugal	20.6K	1.2K	99.1K	20.6K	1.2K	99.1K
Spain	9.4K	360.0	15.6K	9.4K	360.0	15.6K
Sweden	2.2K	84.0	7.3K	2.2K	84.0	7.3K
Switzerland	36.8K	2.4K	194.0K	36.8K	2.4K	194.0K
United Kingdom	1.3K	48.0	4.0K	1.3K	48.0	4.0K
Wales	1.3K	118.0	6.6K	1.3K	118.0	6.6K
Scotland	35.6K	2.2K	185.4K	35.6K	2.2K	185.4K

Goal TRL: map tables to some consistent input.
Learn some representation that helps detect
patterns relevant to given task(s).



Today...

- 1 The power of table and column **semantics**
- 2 What we need to **make TRL work**
- 3 Towards **end-to-end data analysis** tools

The power of table and column semantics

Essential understanding
of a table comes
through its columns.

Semantic column types: what and why?

name	salary	country
name	salary	cntr

Looks easy, but....

- Undescriptive header?
- Messy values?
- Diverse data types?

Semantic column types dictate operations *sensible* to perform on them:

name	salary	cntr



naam	status	land

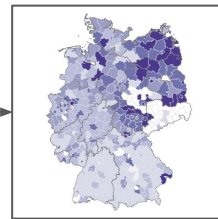
Join tables on "name" and "country" columns

name
Xi
carl
sara

name
Xi
Carl
Sara

Capitalize "name" columns

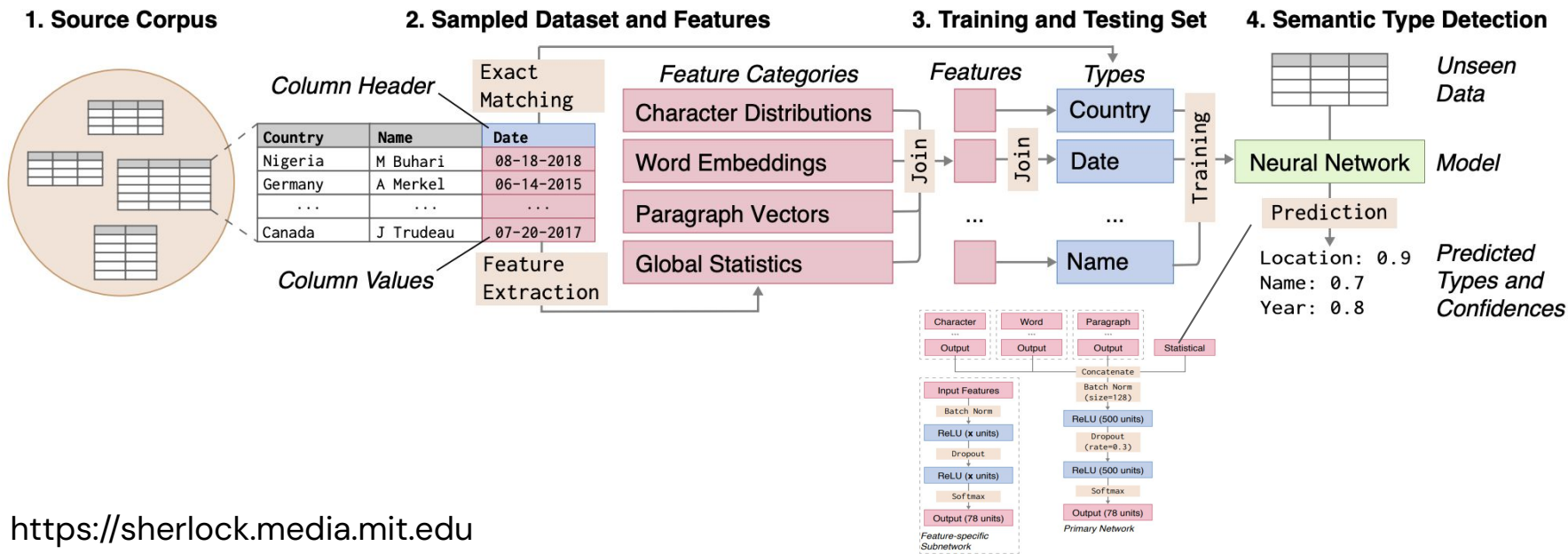
name	salary	cntr



Plot "country" data

Sherlock: Column Type Detection with DL

Prior: string matching (header/values) w/ **regex** or dict: robust? scale? accurate?



<https://sherlock.media.mit.edu>

Don't we have LLMs now?

“Table-tuned” LLM (but not for semantic type detection) [1]:

Zero-Shot		Few-Shot	
GPT-3.5	+table-tune	GPT-3.5	+table-tune
0.332	0.449	0.528	0.538

Sherlock model: ~0.88 F1.

LLM (GPT-3.5) w/ more examples and specific context [2]:

	F_1 -score	Precision	Recall
DoDuo-VizNet*	0.876	89.4%	87.2%
Sherlock*	0.954	96.2%	94.6%
TaBERT	0.321	32.6%	32.0%
DoDuo-Wiki	0.440	59.2%	45.4%
CHORUS	0.891	91.2%	88.8%

Sure, GPT-x might do better..
but w/ **billions** of params vs **thousands**!

Representation Learning (LM trained on type detection) [3]:

Method	$\bar{F}_1$	P	R
Sherlock (only entity mention) [17]	78.47	88.40	70.55
TURL + fine-tuning (only entity mention)	88.86	90.54	87.23
TURL + fine-tuning w/o table metadata	94.75	94.95	94.56
only table metadata	90.24	89.91	90.58

[1] TableGPT: Table-tuned gpt for diverse table tasks. P. Li et al, VLDB, 2024

[2] CHORUS: Foundation Models for Unified Data Discovery and Exploration. Kayali, et al. VLDB, 2024.

[3] TURL: Table understanding through representation learning. Xiang Deng, et al., ACM SIGMOD Record, 2022.

Semantics for optimizing data systems

Example: column semantics $\rightarrow$ correlations.

Cardinality estimation

Learned Cardinalities: Estimating Correlated Joins with Deep Learning

Andreas Kipf
Technical University of Munich
kipf@in.tum.de

Thomas Kipf
University of Amsterdam
t.n.kipf@uva.nl

Bernhard Radke
Technical University of Munich
radke@in.tum.de

Viktor Leis
Technical University of Munich
leis@in.tum.de

Peter Boncz
Centrum Wiskunde & Informatica
boncz@cw.nl

Alfons Kemper
Technical University of Munich
kemper@in.tum.de

Can Large Language Models Predict Data Correlations from Column Names?

Immanuel Trummer
Cornell Database Group
Ithaca, NY, USA
itrummer@cornell.edu

Compression

Lightweight Correlation-Aware Table Compression

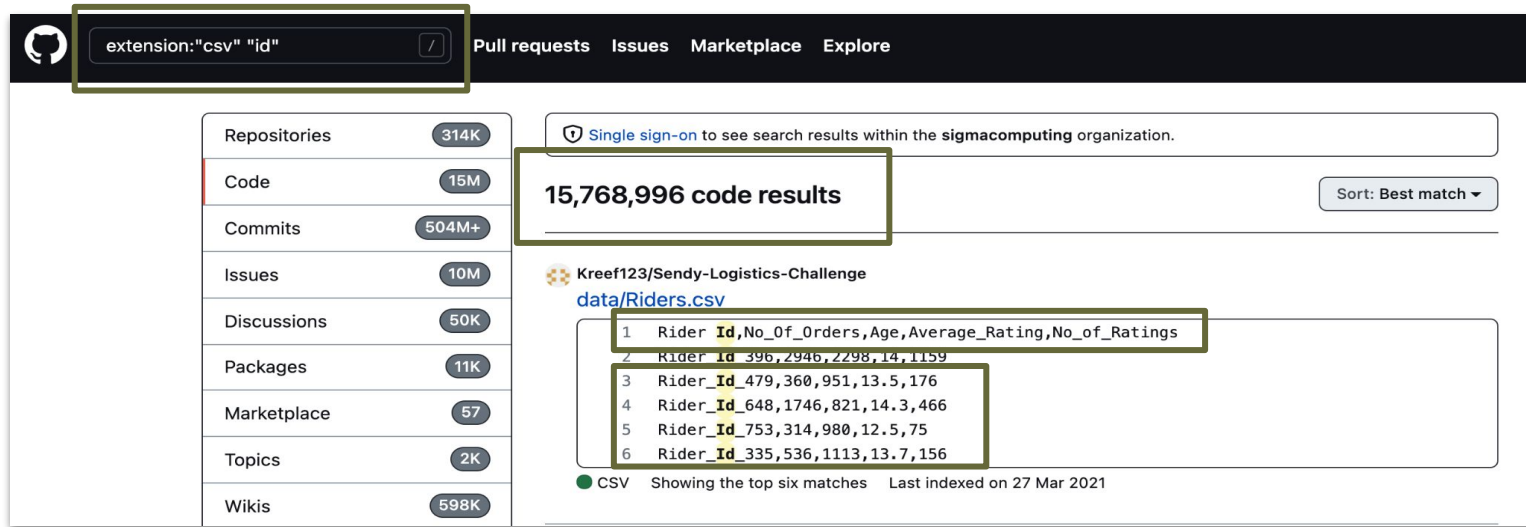
Mihail Stoian, Alexander van Renen, Jan Kobiolka, Ping-Lin Kuo,
Josif Grabocka, Andreas Kipf
{mihail.stoian, alexander.van.renen, jan.kobiolka, ping-lin.kuo,
josif.grabocka, andreas.kipf}@utn.de
University of Technology Nuremberg

What we need to make TRL work

As LLM scaling laws
reach their limits:
it is all about the
“quality” of the data,
and the “tricks” we apply.

What Data Do We Need?

- Web/WikiTables → Web applications. Web tables ≠ DB tables...
- Data tasks on offline tables? GitHub as a data source?



extension:".csv" "id"

Pull requests Issues Marketplace Explore

Single sign-on to see search results within the sigma computing organization.

15,768,996 code results

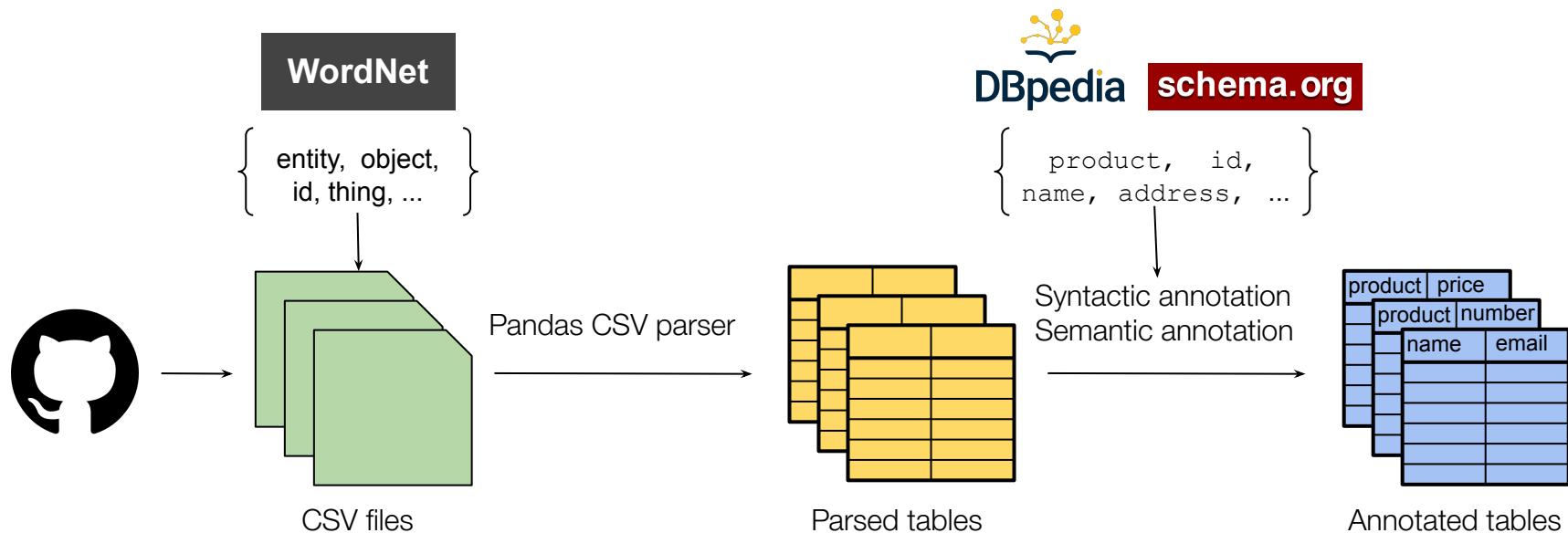
Sort: Best match

Kreef123/Sendy-Logistics-Challenge
data/Riders.csv

	Rider_Id	No_Of_Orders	Age	Average_Rating	No_of_Ratings
1	Rider_Id	No_Of_Orders	Age	Average_Rating	No_of_Ratings
2	396	2946	2298	14	1159
3	479	360	951	13.5	176
4	648	1746	821	14.3	466
5	753	314	980	12.5	75
6	335	536	1113	13.7	156

CSV Showing the top six matches Last indexed on 27 Mar 2021

GitTables: a new large corpus with tables



<https://gittables.github.io>

Using GitTables

- >1M tables and 800K CSV files.
- Wider+taller, and lots of IDs; more representative.
- Useful for *semantic column type detection* and *schema completion*:

Header prefix

Suggested completion

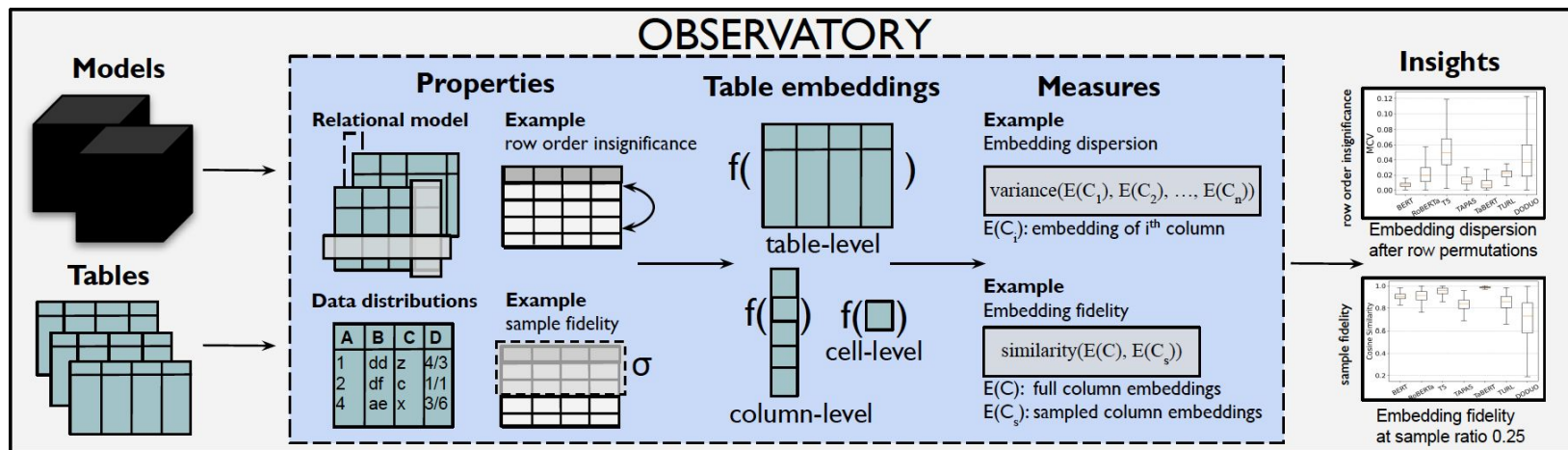
payment_id, customer_id	review_id, product_id, product_parent, product_title, ...
id, company	ReceivablePaymentHeader, ReceivablePayment, Status, Customer, BankEntity, ...
id, name, location	phone, email, uid, active, ad_organization_id, ...

Used for join discovery, CSV parsing, KG enhancement, retrieval eval, etc.
Other corpora to bridge the “realism gap” e.g. **SchemaPile**, **BIRD**, **Spider**.

Do 'tableLM' tricks capture relational properties?

Tables $\neq$ natural language

Studying neural table embeddings through **Codd's relational model**.

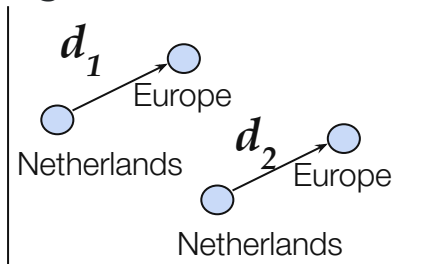


Example Property: Functional Dependencies

Given table with FD: $X=\text{country} \rightarrow Y=\text{continent}$

We argue that:

- FD relations interpretable as *translation* between embeddings $E(\pi X(s))$ and $E(\pi Y(s))$
- Model preserves FD if $d(E(\pi X(s)), E(\pi Y(s))) = d(E(\pi X(t)), E(\pi Y(t)))$ where d preserves magnitude+direction (L1/L2-norm).
- Intuitively:



ID	name	country	continent
1	Kathryn	Netherlands	Europe
2	Oscar	Netherlands	Europe
3	Lee	Canada	North America
4	Roxanne	USA	North America
5	Fern	Netherlands	Europe
6	Raphael	USA	North America
7	Rob	USA	North America
8	Ismail	Canada	North America

Current Architectures Often Fall Short...

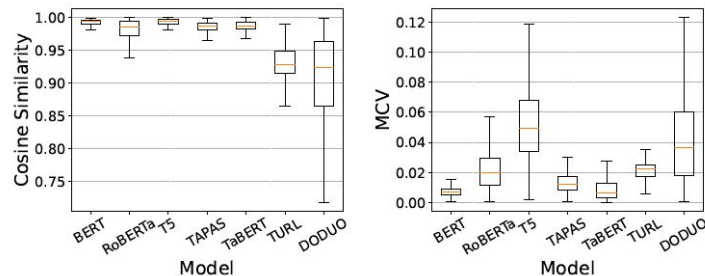
Turns out, most models do not preserve FDs!

RM [4] also has straightforward properties:

A *relation* then consists of a set of tuples, each tuple having the same set of attributes. If the domains are all simple, such a relation has a tabular representation with the following properties.

- (1) There is no duplication of rows (tuples).
- (2) Row order is insignificant.
- (3) Column (attribute) order is insignificant.
- (4) All table entries are atomic values.

Measure by avg cosine similarity of col embeddings across row permutations.

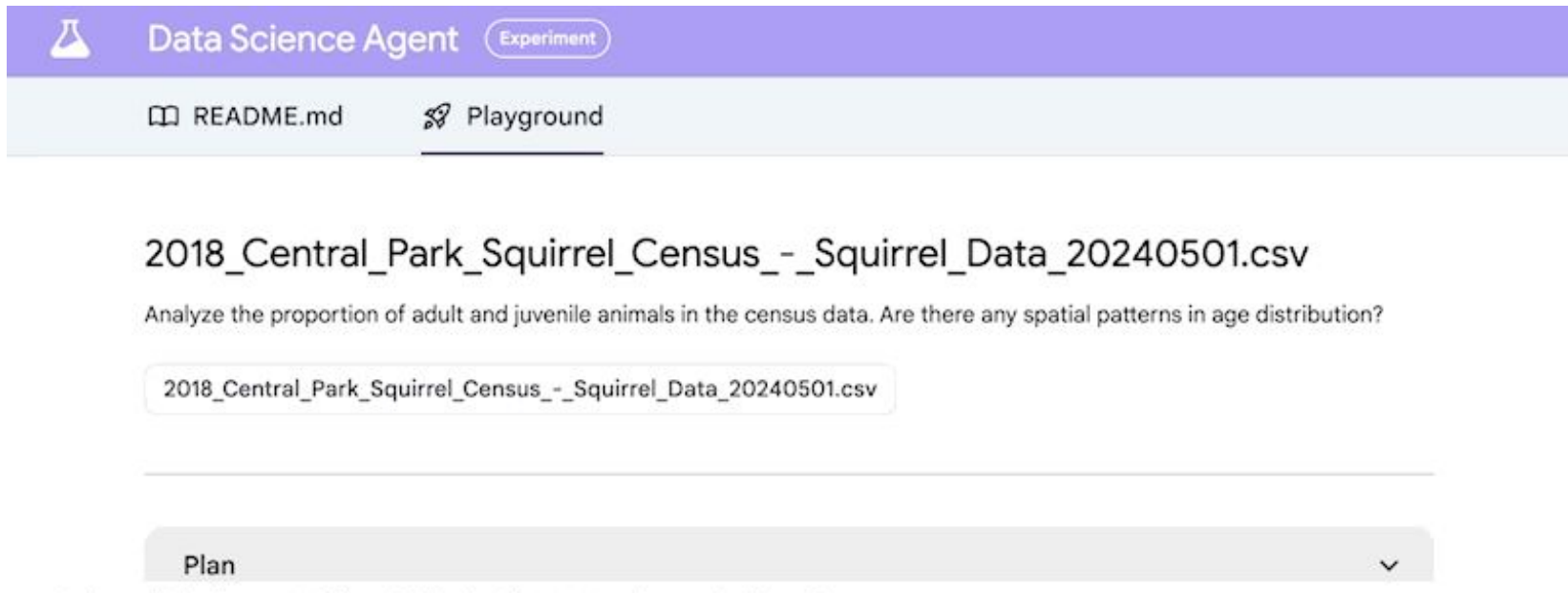


row order robustness

Impact downstream tasks: **row shuffling affects 34%** semantic column types!

Towards end-to-end data analysis systems

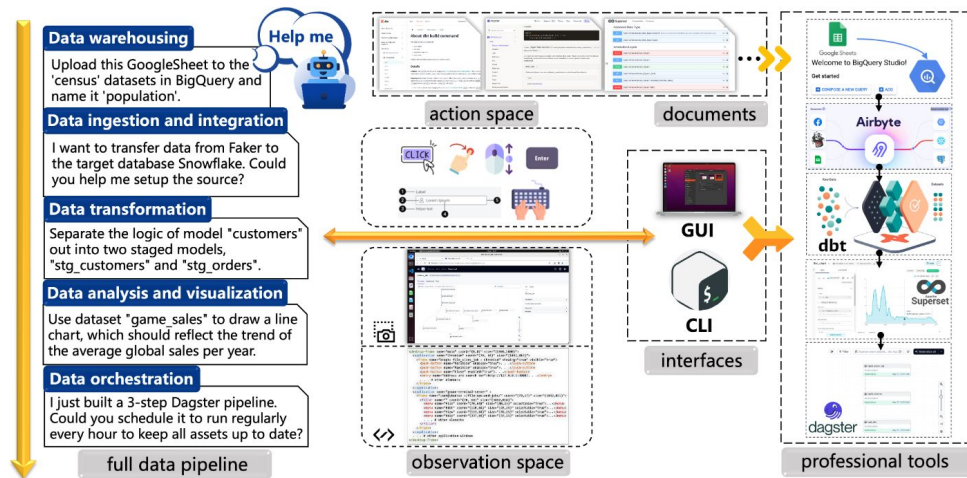
End-to-end DS goes far beyond “automl”



Typically lots of text-to-code (e.g. SQL) involved!

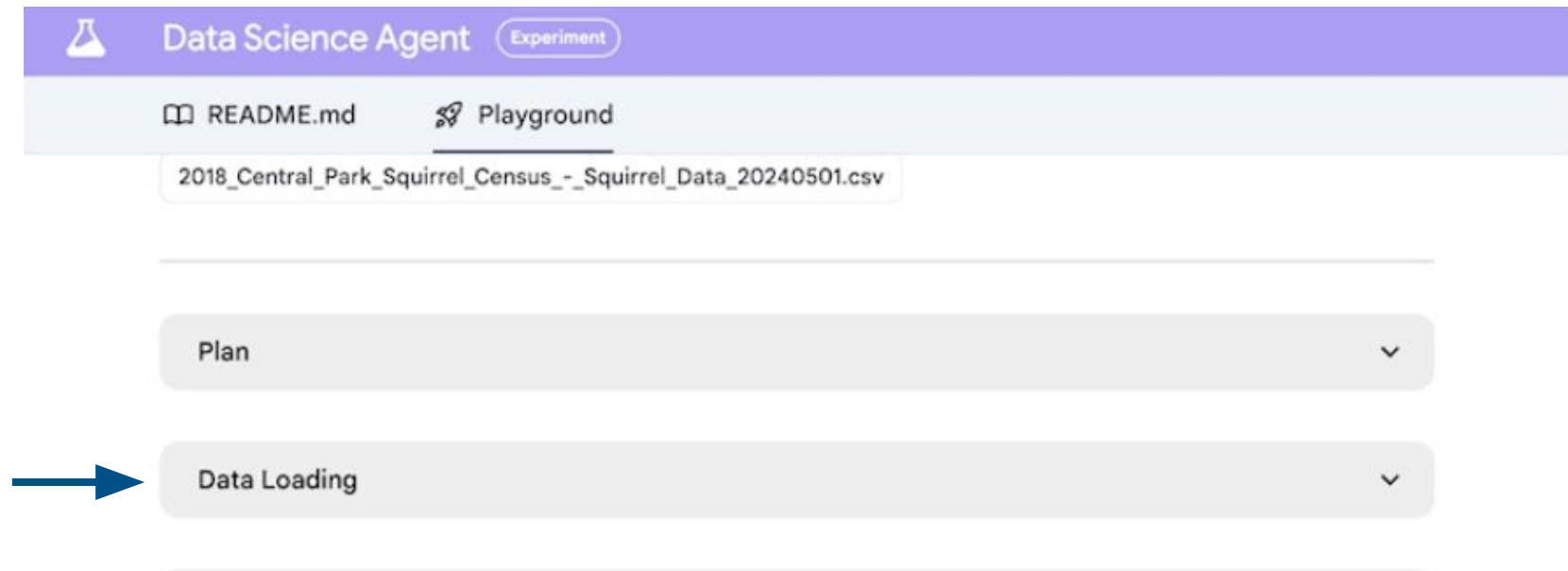
From DS codegen to DS GUI agents

Spider2-V Framework Infrastructure



What's the right level of abstraction for data agents to operate on, anyway?
Should systems even still focus on human usage?

Sounds cool, how do we get there?



Who or what is doing data analysis, it will need the right data first.

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

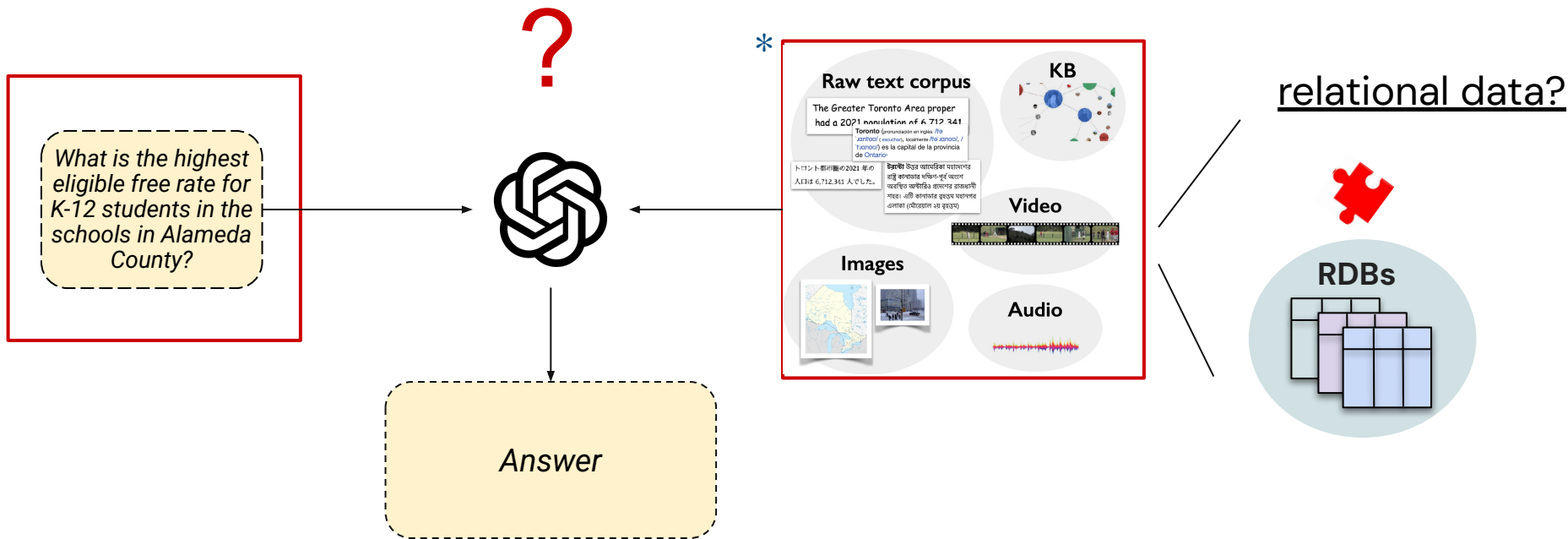
Asking LLMs complex questions

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. ...” *

We need “specific” data to ground LLMs



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]



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...**

There is still a long way to go.” – Wenhui Chen (Univ of Waterloo)

Methods for table retrieval

① Embed tables in corpus

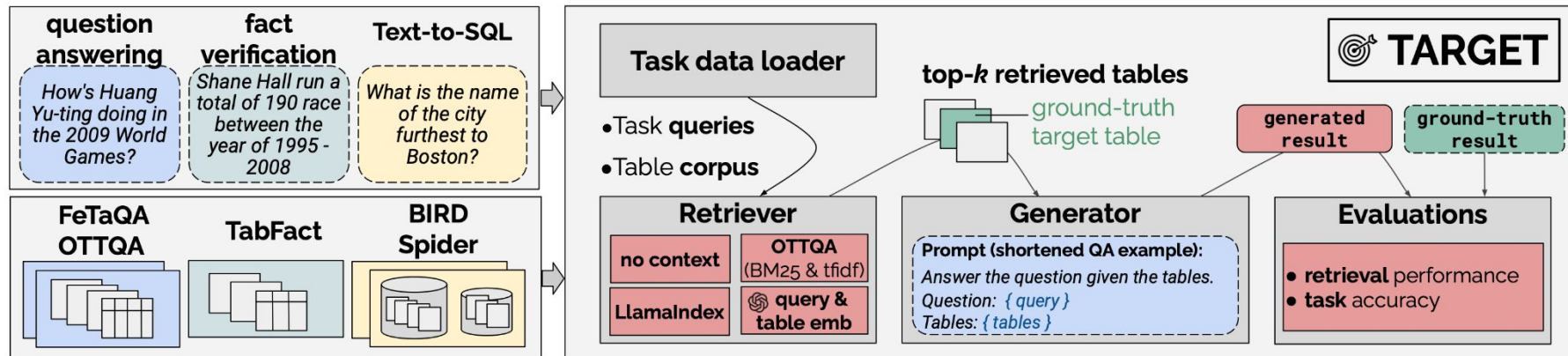
- BM25 / TF-IDF (sparse lexical representations)
- Generate summary/metadata → embed summary + table
- “Naive” embedding of table (header / header+rows) and query

② Embed query

③ 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



<https://target-benchmark.github.io> (pip install target_benchmark)

TARGET insights

Method	Question Answering						Fact Verification			Text-to-SQL					
	OTTQA			FeTaQA			TabFact			Spider		EX		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
<i>w/o table title</i>	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
<i>w/o table title</i>	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
<i>header only</i>	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 than for text**, only works with 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**.

Still much to explore...

- What is the 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 $e(\text{query})$ and $e(\text{table})$; most public datasets relatively “easy” match between query and tables.
 - Relational databases are large → in-DB schema and table retrieval.



Roadmap for TARGET

Take aways...

- **Tables are everywhere**, serving **high-value use-cases** in e.g. gov, health, finance.
- **Tables $\neq$ natural language**: tables come with **specific properties** (e.g. relational).
- Capabilities of **“foundation” models should extend to tables & relational DBs**.
- For this, we need the **right data**, and the **right “tricks”**.
- For any data analysis system, human or agentic; **retrieval is key** (e.g. tables, context).

madelonhulsebos.com, madelon@cw.nl, @madelonhulsebos  

Hulsebos, M., Hu, K., Bakker M., et al. "Sherlock: A deep learning approach to semantic data type detection." ACM SIGKDD 2019.

Hulsebos, M., Demiralp, C., Groth, P. "GitTables: A large-scale corpus of relational tables." SIGMOD 2023.

Cong, T., Hulsebos M., Groth, P., Jagadish, H. "Observatory: Characterizing embeddings of relational tables." VLDB 2024.

Ji, X., Parameswaran, A., Hulsebos., M. "TARGET: Benchmarking Table Retrieval for Generative Tasks." TRL @ NeurIPS 2024.