Evaluating Ambiguous Questions in Semantic Parsing

Simone Papicchio, Paolo Papotti, Luca Cagliero

International Workshop on Databases and Machine Learning – 13th May 2024

Semantic Parsing

Please translate in SQL query:
"Give me all the employees with salary above 2k"

for the schema
Emp(name, age, salary)

"Select name
From Emp
Where salary>2000"

- Text to SQL: example of *NL text to code*
- LLMs do very well… according to results on public benchmarks

Spider: Semantic Parsing and Text-to-SQL Challenge

- Manually annotated corpus [EMNLP **2018**]
  5.7k (NL Question, SQL query) on 200 databases

Which countries in Europe have at least 3 car manufacturers?

```
SELECT T1.country_name
FROM countries AS T1 JOIN continents
AS T2 ON T1.continent = T2.cont_id
JOIN car_makers AS T3 ON
T1.country_id = T3.country
WHERE T2.continent = 'Europe'
GROUP BY T1.country_name
HAVING COUNT(*) >= 3
```

Rank	Model	Test
1 Nov 2, 2023	MiniSeek *Anonymous* Code and paper coming soon	**91.2**
1 Aug 20, 2023	DAIL-SQL + GPT-4 + Self-Consistency *Alibaba Group* (Gao and Wang et al.,'2023) code	**86.6**
2 Aug 9, 2023	DAIL-SQL + GPT-4 *Alibaba Group* (Gao and Wang et al.,'2023) code	**86.2**
3 October 17, 2023	DPG-SQL + GPT-4 + Self-Correction *Anonymous* Code and paper coming soon	85.6

https://yale-lily.github.io/spider

Can we adopt these models?

- Solutions are validated on **public** benchmark
- Risks:
  - **Overfit** – systems optimized for queries in this dataset
  - **Contamination** - examples are on the Web
  - **Assumptions** – clear and complete questions

Today
13:30

Assumptions in Benchmarks vs Reality

Which countries in Europe have at least 3 car manufacturers?

```
SELECT T1.country_name
FROM countries AS T1 JOIN continents
AS T2 ON T1.continent = T2.cont_id
JOIN car_makers AS T3 ON
T1.country_id = T3.country
WHERE T2.continent = 'Europe'
GROUP BY T1.country_name
HAVING COUNT(*) >= 3
```

AbaloneId	Sex	Length	Diameter	Height
1	F	0.40	0.32	0.13
2	M	0.39	0.32	0.11
3	M	0.32	0.26	0.09

What is the *size* of the Abalone fish with Id 1?

"customers use compact and informal language to interact with our systems"
[Microsoft - Floratou et al, CIDR 2024]

Related Work

- Analysis: 45% questions attribute ambiguity [Wang et al, EMLNLP 2023]

- Ambiguity detection in Semantic Parsing:
  - fine tuning encoder [Veltri et al, ICDE 2023],
  - add documentation, data examples with GPT4 [Huang et al, TRL 2023]
  - GPT4 high agreement with humans [Floratou et al, CIDR 2024]

  → Revise the workflow of NL2SQL?

- Top k solutions [Bhaskar et al, EMNLP 2023]

What is a good answer?

L, D, H data-ambiguous wrt label “size”

AbaloneId	Sex	Length	Diameter	Height
1	F	0.40	0.32	0.13
2	M	0.39	0.32	0.11
3	M	0.32	0.26	0.09

What is the *size* of the Abalone fish with Id 1?

Ranking:

SELECT Length FROM ….

SELECT Diameter FROM ….

1

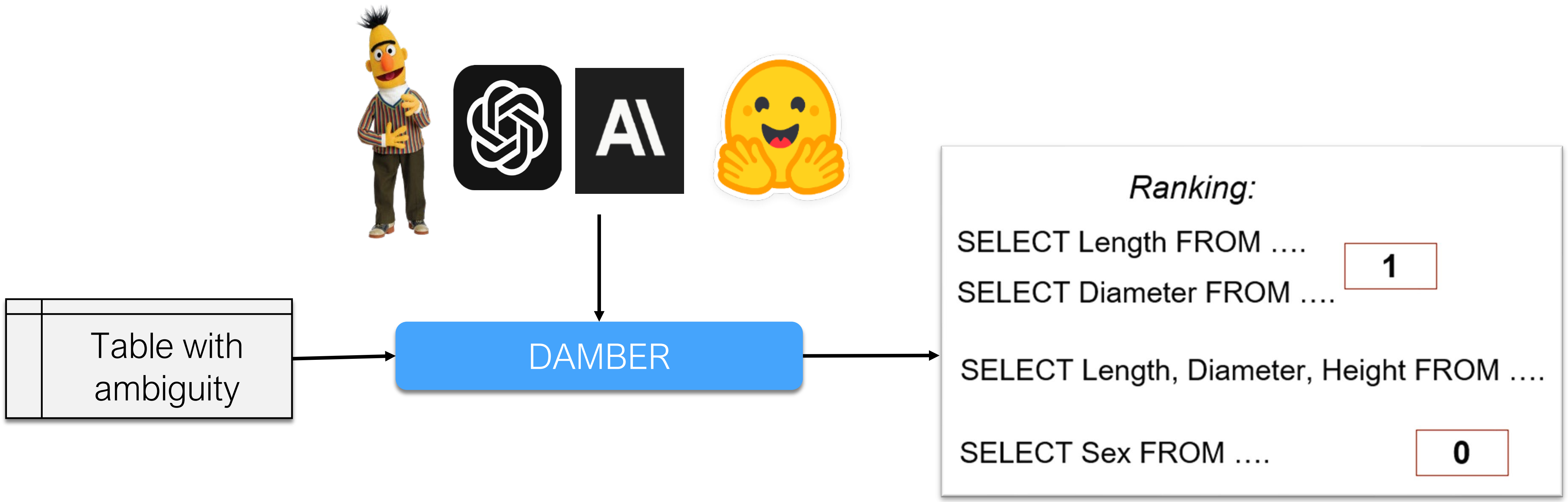
SELECT Length, Diameter, Height FROM ….

SELECT AbaloneId FROM ….

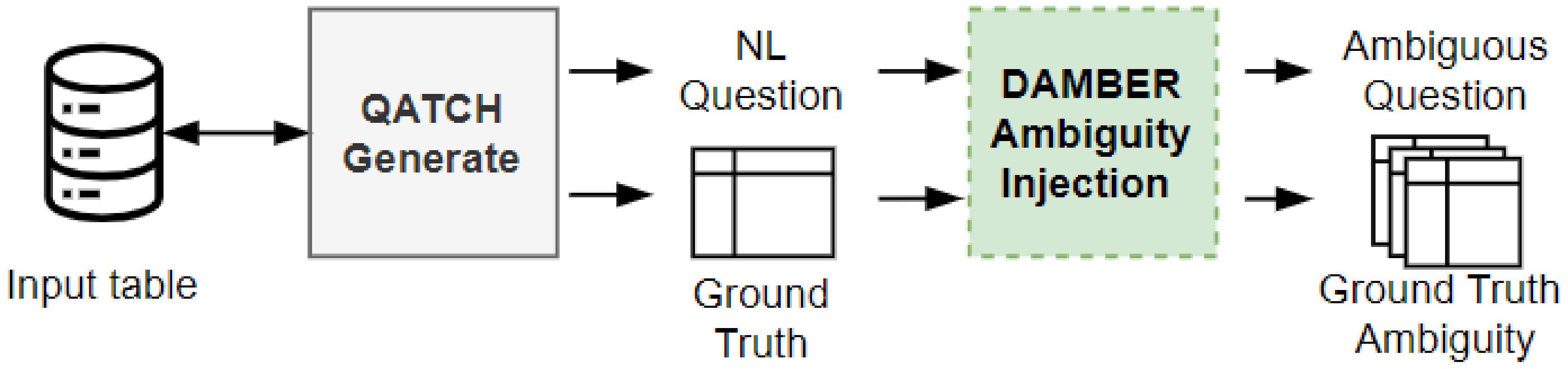
0

Benchmarking models on tables with attribute ambiguity

- Given a table *D* with attributes *A1, .., An* data-ambiguous wrt label *L*
  - Rank existing LLMs on *D* for Semantic Parsing



DAMBER: **D**ata-**AMB**iguity test**ER**



- DAMBER built on top of QATCH (**Q**uery-**A**ided **T**RL **Ch**ecklist)

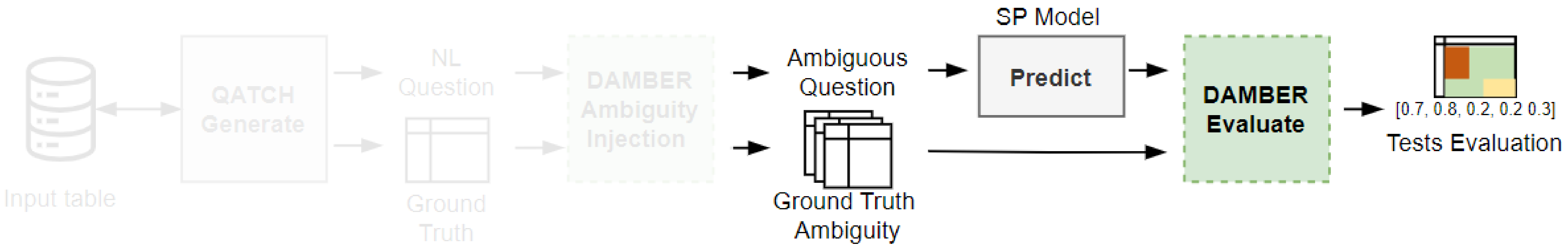
Prior to ambiguity injection

Table name	Abalone
SQL category	Project
Query	SELECT "Length" FROM "abalone"
Question	Show all "Length" in the table abalone

After ambiguity injection

Table name	Abalone
SQL category	Project
Target Queries	SELECT "Length" FROM "abalone" SELECT "Diameter" FROM "abalone" SELECT "Height" FROM "abalone"
Ambiguous Question	Show all "distance" in the table abalone

DAMBER: **D**ata-**AMB**iguity test**ER**



- QATCH metrics computed on data outputs
- Measure model prediction against the **best** matching target query

Model Predictions	
Model 1	SELECT "distance" FROM abalone
Model 2	SELECT * FROM abalone
Model 3	SELECT "Length" FROM abalone

Model Evaluation

	Cell precision	**Cell recall**	**Tuple cardinality**	**Tuple constraint**	**Tuple order**
Model 1 evaluation	0.0	0.0	0.0	0.0	-
Model 2 evaluation	1/5	1.0	1.0	0.0	-
Model 3 evaluation	1.0	1.0	1.0	1.0	-

Experiments setting

- corpus: 13 tables (UCI repository and WebTables),
  10 annotators identify ambiguous attributes+label for each pair
  E.g., “weight” and “height” as ambiguous → label “measure”
  1321 attribute pairs, 252 ambiguous pairs

- three TRL models: RESDSQL, GAP, UNIFIEDSKG
  two LLMs: CHATGPT 3.5 Turbo and Code-LLAMA

Results for Semantic Parsing

Model	Cell precision	Cell recall	Tuple cardinality	Tuple constraint	Tuple order
ChatGPT 3.5 (LLM)	**0.76**	**0.78**	**0.80**	**0.63**	0.83
LLama-code (LLM)	0.52	0.54	0.58	0.39	**0.86**
ResdSQL (TRL)	0.37	0.38	0.42	0.31	0.46
UnifiedSKG (TRL)	0.36	0.37	0.39	0.31	0.65
GAP (TRL)	0.24	0.24	0.26	0.21	0.27

avg over 13 tables and all tests

- ChatGPT avg results range from 0.98 in WDC_631 to 0.60 in Abalone
  "length", "diameter", and "height" vs label "distance“
- ChatGPT returns all relevant attributes when faced with uncertainty: higher recall than precision, struggle with aggregate queries

Evaluating Ambiguous Questions in Semantic Parsing

- Semantic Parsing is a mature technology… under assumptions common in benchmarks
- Attribute Ambiguity affects the quality of the results

- Keep exploring the impact of other types of ambiguity

Evaluating Ambiguous Questions in Semantic Parsing

Simone Papicchio
Politecnico di Torino
Turin, Italy
simone.papicchio@polito.it

Paolo Papotti
EURECOM
Sophia Antipolis, France
paolo.papotti@eurecom.fr

Luca Cagliero
Politecnico di Torino
Turin, Italy
luca.cagliero@polito.it

***Abstract*—Tabular Representation Learning and Large Language Models have recently achieved promising results in solving the Semantic Parsing (SP) task. Given a question posed in natural language on a relational table, the goal is to return to the end-users executable SQL declarations. However, models struggle to produce the correct output when questions are ambiguously defined w.r.t. the table schema. Assessing the robustness to data-ambiguity can be particularly time-consuming as entails seeking ambiguous patterns on a large number of queries with varying complexity. To automate this process, we propose *Data-Ambiguity Tester*, a pipeline for data-ambiguity testing tailored to SP. It first automatically generates non-ambiguous natural language questions and SQL queries of varying complexity. Then, it injects ambiguous patterns, extracted from a human-annotated set of relational tables, in the natural language questions. Finally, it quantifies the level of ambiguity using customized performance metrics. Results show strengths and limitations of existing models in coping with ambiguity between questions and tabular data.**

***Index Terms*—Tabular Representation Learning, Semantic Parsing, Text2SQL, Data-Ambiguity, NL2SQL, Large Language Models.**

I. Introduction

State-of-the-art models for table understanding are pretrained

TABLE I
TOY EXAMPLE EXTRACTED FROM THE ABALONE DATASET [13]

AbaloneId	Sex	Length	Diameter	Height
1	F	0.40	0.32	0.13
2	M	0.39	0.32	0.11
3	M	0.32	0.26	0.09

language question that contains free text that ambiguously refers to more than one attribute of the relation schema. Therefore, the Text2SQL process should generate many valid SQL declarations, each corresponding to a different combination of attributes. In this work, we focus on studying how existing models handle ambiguous relationships between the text of the NL question and the relational schema.

Figure 1 shows the main steps in DAMBER[1] (*Data-AMBiguity testER*), a new pipeline for ambiguous test generation and evaluation tailored to SP on tabular data. DAMBER relies on QATCH [15], a recently proposed testing benchmark for TRL models, to initially generate a large set of Text2SQL

https://github.com/spapicchio/QATCH

SQL Category	Cell precision	Cell recall	Tuple cardinality	Tuple constraint	Tuple order
`Project`	0.76	0.89	0.95	0.61	-
`Order By`	0.80	0.82	0.93	0.75	0.83
`Distinct`	0.85	0.87	0.93	0.82	-
`SIMPLE-AGG AVG-MAX-MIN`	0.74	0.76	0.96	0.72	-
`SIMPLE-AGG COUNT-DISTINCT`	0.88	0.88	**1.00**	0.88	-

Evaluate on output data

1. Benchmark multiple tasks: QA output is data

2. Data comparison enables accurate metrics for SP: execute correct SQL and generated SQL on D, compare data outputs

		Cell precision	Cell recall	Tuple cardinality	Tuple constraint	Tuple order
Target **Prediction**	SELECT DISTINCT "emailisfree" FROM "fraud" SELECT "emailisfree", "income" FROM "fraud"	0.5	1.0	0.2	0.0	-
Target **Prediction**	SELECT "emailisfree" FROM "fraud" ORDERBY ASC SELECT "emailisfree" FROM "fraud" ORDERBY DESC	1.0	1.0	1.0	1.0	0.0
Target **Prediction**	SELECT * FROM "fraud" SELECT "emailisfree" FROM "fraud"	1.0	0.10	1.0	0.0	-

References

- Papicchio et al, QATCH: Benchmarking SQL-centric tasks with Table Representation Learning Models on Your Data. NeurIPS 2023
- Saeed et al, Querying Large Language Models with SQL. EDBT 2024
- Saeed and Papotti, You Are My Type! Type Embeddings for Pre-trained Language Models. EMNLP 2022
- Thorne et al, From Natural Language Processing to Neural Databases. Proc. VLDB Endow. 2021
- Veltri et al, Data Ambiguity Profiling for the Generation of Training Examples. ICDE 2023
- Yu et al, Spider: A Large-Scale Human-Labeled Dataset for Complex and Cross-Domain Semantic Parsing and Text-to-SQL Task. EMNLP 2018
- Badaro et al. Transformers for Tabular Data Representation: A Survey of Models and Applications TACL 2023

User Input:

NL Question

SQL Query

Storage:

Documents

Question answering (QA)

Relations

Table QA

Semantic Parsing

Table Retrieval

Fact Checking

Query Execution

[Badaro et al, 2023]