Data Ambiguity Strikes Back
How Documentation Improves GPT's Text-to-SQL

Zachary Huang, Pavan Kalyan Damalapati, Eugene Wu
Columbia University

Text-to-SQL

Query

Who is the best salesperson?

```
SELECT Name, Total_Sale
FROM Sales
GROUP BY Name
ORDER BY Total_Sales DESC
LIMIT 1;
```

Data

Sales

Name	Product	Sale Date	Total Sale	...
Alice	0	12/11/2013	$500	...
Bob	1023	5/7/2014	$1000	...
Robert	1023	5/7/2014	1000 USD	...
...	...	...	...	...

Text-to-SQL

Query Ambiguity is well known

Query

Who is the best salesperson?

How is “best” defined?

What’s the output schema?

How to handle tie?

Data

Sales

Name	Product	Sale Date	Total Sale	...
Alice	0	12/11/2013	$500	...
Bob	1023	5/7/2014	$1000	...
Robert	1023	5/7/2014	1000 USD	...
...	...	...	...	...

Text-to-SQL

But Data can also be Ambiguous!

Query

Who is the best salesperson?

How is “best” defined?

What’s the output schema?

How to handle tie?

Data

Sales

Name	Product	Sale Date	Total Sale	...
Alice	0	12/11/2013	$500	...
Bob	1023	5/7/2014	$1000	...
Robert	1023	5/7/2014	1000 USD	...
...	...	...	...	...

Customer or Salesperson?

Missing value?

Aggregated by Name, Product?

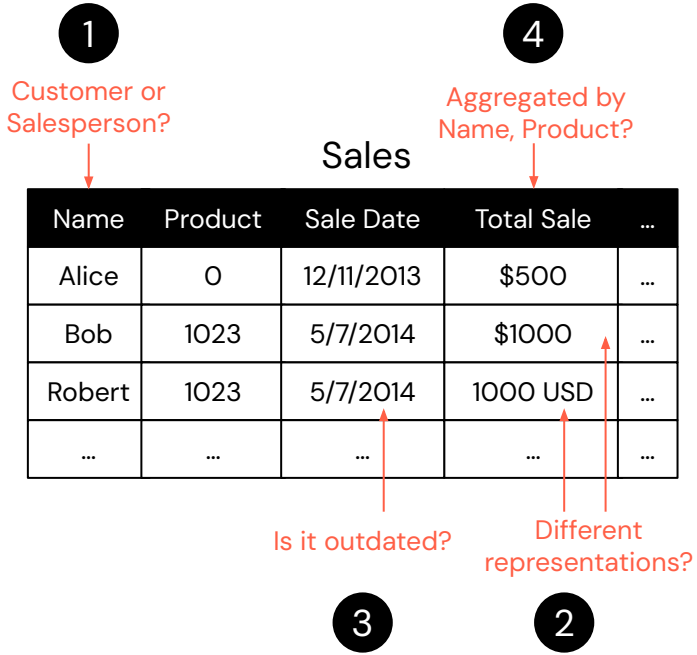
Duplicate?

Is it outdated?

Different representations?

To Address Data Ambiguity

Documentation, also for GPT



Currently 4 types

1. **Name Description**
   "Name" is for salesperson
2. **Value Consistency**
   "Total Sale" is represented by regex of "\$\d+(\.\d{2})?" or "\d+(\.\d{2})?\sUSD"
3. **Data Coverage**
   This table covers all sales record between 2013-2014
4. **Data Granularity**
   Each row is an aggregated total sale for each salesperson and product

... More for future works

Experiment

How documentation improves GPT accuracy

- **Data:** KaggleDBQA, a real-world benchmark with Query & Data Ambiguity
- **Documentation:** Name Description provided. Manually construct the rest.
- **Model:** GPT-4 + standard chain-of-thought

Table: Sales, with attributes Name, Product, Sale Date, Total Sale

Column Name Description:
- Name: salesperson name

...
Query: Who is the best salesperson?
Steps: Go through each the table and descriptions.
Reason about how to construct the SQL to answer query.

Table =Schema

Documentation

Query

Chain-of-thought

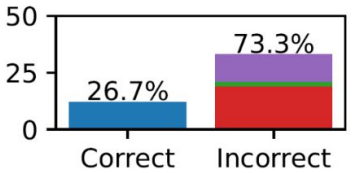
To identify the "best" salesperson ...
SELECT Name, SUM(Total Sale) ...

Experiment

How documentation improves GPT accuracy

- **Data:** KaggleDBQA, a real-world benchmark with Query & Data Ambiguity
- **Documentation:** Name Description provided. Manually construct the rest.
- **Model:** GPT-4 + standard chain-of-thought
- **Error types**: Output Schema, Fuzzy Predicate, Other

Schema + Query
(e.g., Langchain)

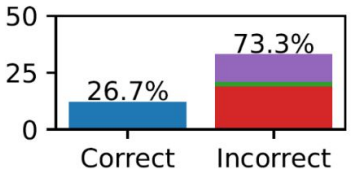


Experiment

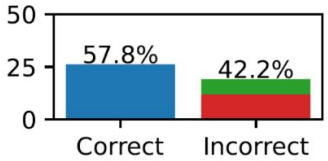
How documentation improves GPT accuracy

- **Data:** KaggleDBQA, a real-world benchmark with Query & Data Ambiguity
- **Documentation:** Name Description provided. Manually construct the rest.
- **Model:** GPT-4 + standard chain-of-thought
- **Error types**: Output Schema, Fuzzy Predicate, Other

Schema + Query
(e.g., Langchain)



+ Query Disambiguation

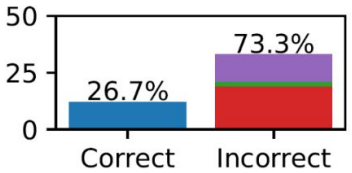


Experiment

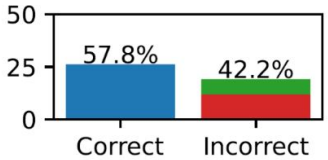
How documentation improves GPT accuracy

- **Data:** KaggleDBQA, a real-world benchmark with Query & Data Ambiguity
- **Documentation:** Name Description provided. Manually construct the rest.
- **Model:** GPT-4 + standard chain-of-thought
- **Error types**: Output Schema, Fuzzy Predicate, Other

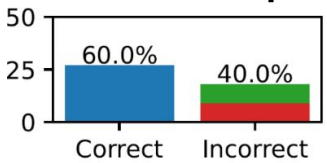
Schema + Query
(e.g., Langchain)



+ Query Disambiguation



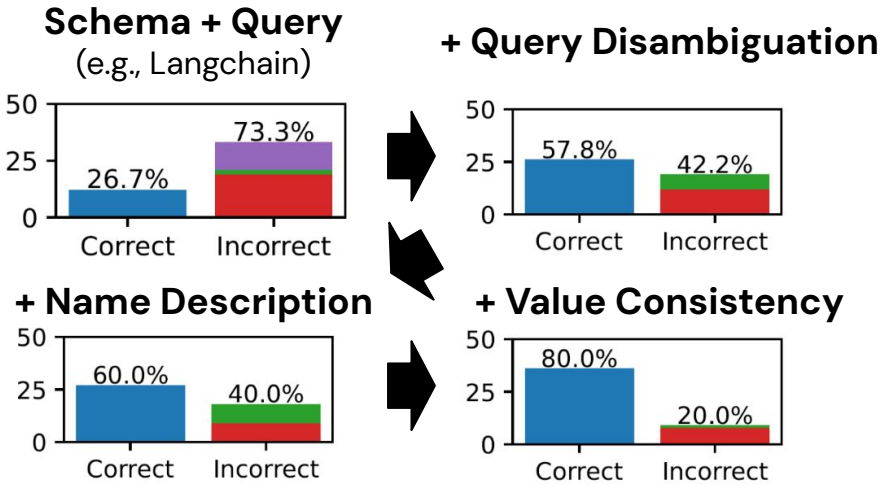
+ Name Description



Experiment

How documentation improves GPT accuracy

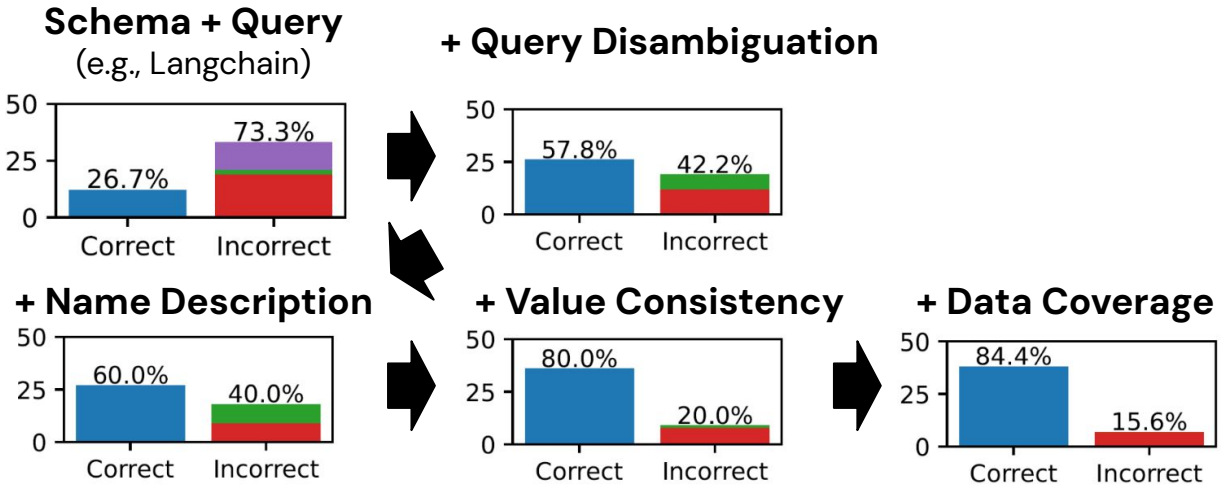
- **Data:** KaggleDBQA, a real-world benchmark with Query & Data Ambiguity
- **Documentation:** Name Description provided. Manually construct the rest.
- **Model:** GPT-4 + standard chain-of-thought
- **Error types**: Output Schema, Fuzzy Predicate, Other



Experiment

How documentation improves GPT accuracy

- **Data:** KaggleDBQA, a real-world benchmark with Query & Data Ambiguity
- **Documentation:** Name Description provided. Manually construct the rest.
- **Model:** GPT-4 + standard chain-of-thought
- **Error types**: Output Schema, Fuzzy Predicate, Other

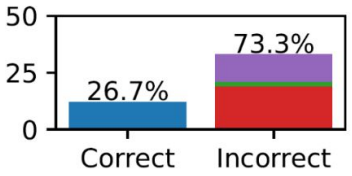


Experiment

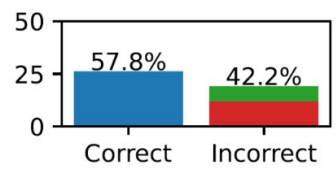
How documentation improves GPT accuracy

- **Data:** KaggleDBQA, a real-world benchmark with Query & Data Ambiguity
- **Documentation:** Name Description provided. Manually construct the rest.
- **Model:** GPT-4 + standard chain-of-thought
- **Error types**: Output Schema, Fuzzy Predicate, Other

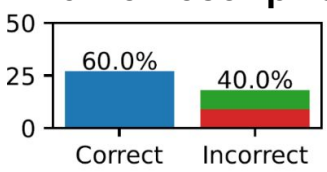
Schema + Query
(e.g., Langchain)



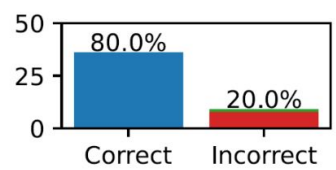
+ Query Disambiguation



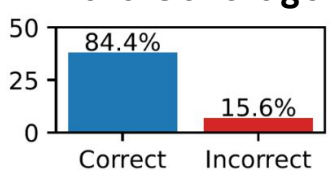
+ Name Description



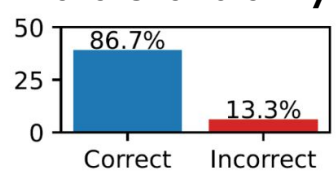
+ Value Consistency



+ Data Coverage



+ Data Granularity



Conclusion and Future Work

Conclusion:

- Data ambiguities are prevalent but understudied for Text-to-SQL.
- Documentation effectively improves accuracy by 28.9%.

Open questions:

1. How to systematically provide the documentation?
2. Other data ambiguities (e.g., missing values, duplications…)?

We are actively developing semi-automated tools for this.
Follow us at Columbia University for updates!