We present a comprehensive study on chain of thought style prompting for text-to-SQL parsing.

Exploring Chain-of-Thought Style Prompting for Text-to-SQL

Chang-You Tai, Ziru Chen, Tianshu Zhang, Xiang Deng, Huan Sun

The Ohio State University

Chain of Thought Style Prompting Methods for Text-to-SQL Parsing

Model Input (API Doc Format)

SQLite SQL tables, with their properties:
\#
\# allergy_type (allergy, allergytype)
\# has_allergy (stuid, allergy)
\# student (stuid, lname, fname, age, sex, major, advisor, city_code)
\#
Show first name, last name, age for all female students. Their sex is F.

Expected Model Output with Different Prompting Methods

Single-Pass Prompting

A. Chain-of-Thought

\# This query chooses the records from the student table, followed by a WHERE clause that selects records where the sex column is equal to F. It then selects the fname, lname, and age columns

\# Thus, the answer for the question is: Show first name, last name, age for all female students. Their sex is F.
`SELECT fname, lname, age FROM student WHERE sex = 'F'`

C. QDecomp (Ours)

1. Show first name, last name, age for all students.
2. Show first name, last name, age for all female students. Their sex is F.

\# Thus, the answer for the question is: Show first name, last name, age for all female students. Their sex is F.
`SELECT fname, lname, age FROM student WHERE sex = 'F'`

D. QDecomp + InterCOL (Ours)

1. Show first name, last name, age for all students.
SQL table (column): student (fname, lname, age)
2. Show first name, last name, age for all female students. Their sex is F.
SQL table (column): student (sex)

\# Thus, the answer for the question is: Show first name, last name, age for all female students. Their sex is F.
`SELECT fname, lname, age FROM student WHERE sex = 'F'`

Iterative Prompting

B. Least-to-Most Prompting

Problem Reduction:

<Model Input> → LLM → # To answer the question "Show first name, last name, age for all female students. Their sex is F.", we need to know: "Show first name, last name, age for all students."

Problem Solving:

<Model Input>

Q: Show first name, last name, age for all students.

→ LLM → A: `SELECT fname, lname, age FROM student`

<Model Input>

Q: Show first name, last name, age for all students.
A: `SELECT fname, lname, age FROM student`

Q: Show first name, last name, age for all female students. Their sex is F.

→ LLM → A: `SELECT fname, lname, age FROM student WHERE sex = 'F'`

Summary of Experiments and Results

Experimental Setup

- Large Language Model: Codex (01/2023 - 03/2023)
- Datasets: Spider, Spider-Realistic, GeoQuery, IMDB, Yelp
- Evaluation Metrics: Test-Suite Accuracy, Execution Accuracy

Results

- What matters for applying CoT prompting to text-to-SQL parsing?
  - *Iterative prompting (e.g. least-to-most) may not be necessary*
  - *Detailed reasoning steps may lead to error propagation*
- How to test and design new prompting methods for text-to-SQL parsing?
  - *Prompting methods are sensitive to in-context examples selection strategies*
  - *The format and number of in-context examples may not change the relative performance significantly*

Method	Spider Dev					Spider Realistic
	Easy	Medium	Hard	Extra Hard	Overall TS (Overall EX)	Overall TS (Overall EX)
Standard	86.8	65.3	50.3	36.0	63.2 ± 2.51 (68.7 ± 4.08)	51.0 ± 4.29 (62.5 ± 4.01)
Chain-of-Thought	73.9	64.5	44.6	23.4	56.8 ± 5.83 (53.9 ± 7.21)	50.3 ± 4.94 (53.4 ± 9.19)
Least-to-Most	88.1	68.7	52.9	39.5	66.0 ± 2.48 (68.9 ± 3.44)	55.0 ± 2.51 (63.3 ± 2.73)
Least-to-Most (G3)	80.3	64.6	52.8	45.3	63.3 ± 1.95 (73.8 ± 1.72)	-*
QDecomp	**89.8**	71.3	53.1	38.6	67.4 ± 1.89 (70.7 ± 2.80)	55.8 ± 2.01 (**65.8** ± 2.29)
+ InterCOL	89.6	**74.1**	52.4	38.1	68.4 ± 2.05 (69.7 ± 5.82)	**56.5** ± 2.05 (63.3 ± 4.19)
+ InterCOL (G3)	88.7	71.1	**56.8**	**45.7**	**68.8** ± 1.16 (**78.2** ± 1.07)	-*

	GeoQuery	IMDB	Yelp	MacroAvg
Standard	60.99	73.28	45.31	59.86
Least-to-Most	60.99	58.78	36.72	52.16
QDecomp	64.84	**77.86**	48.44	63.71
+ InterCOL	**75.82**	73.28	**49.22**	**66.11**