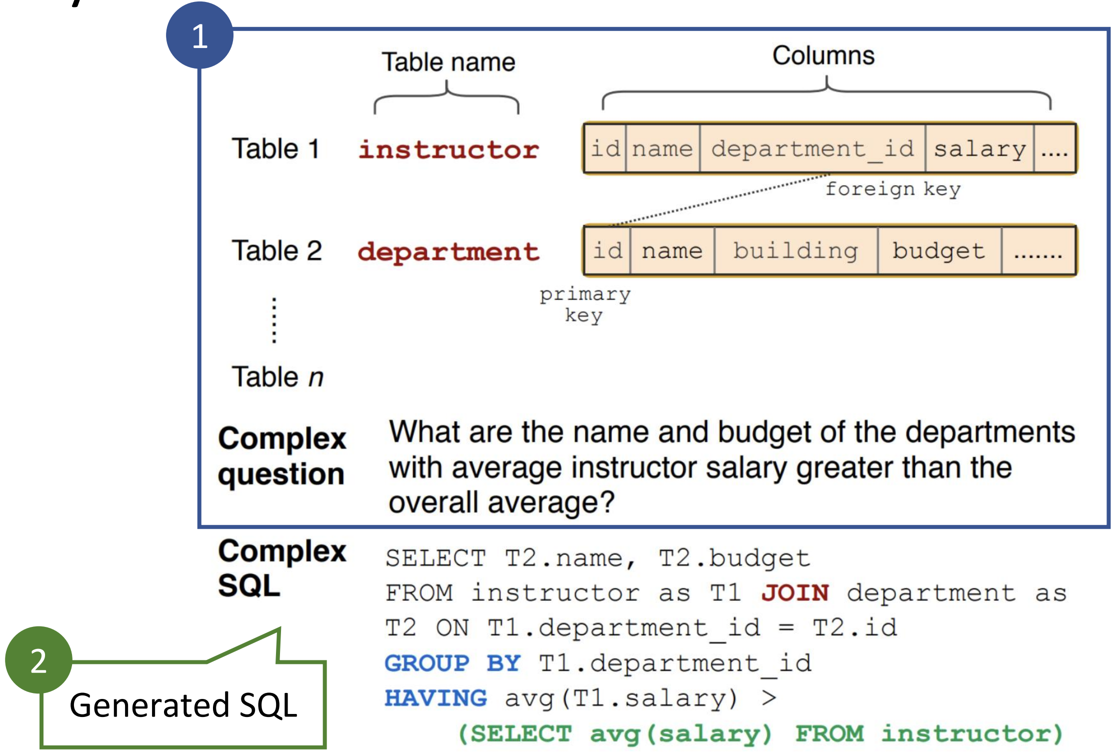
41st IEEE International Conference
on Data Engineering
HONG KONG SAR, CHINA | MAY 19 – 23, 2025

Grounding Natural Language to SQL Translation with Data-Based Self-Explanations

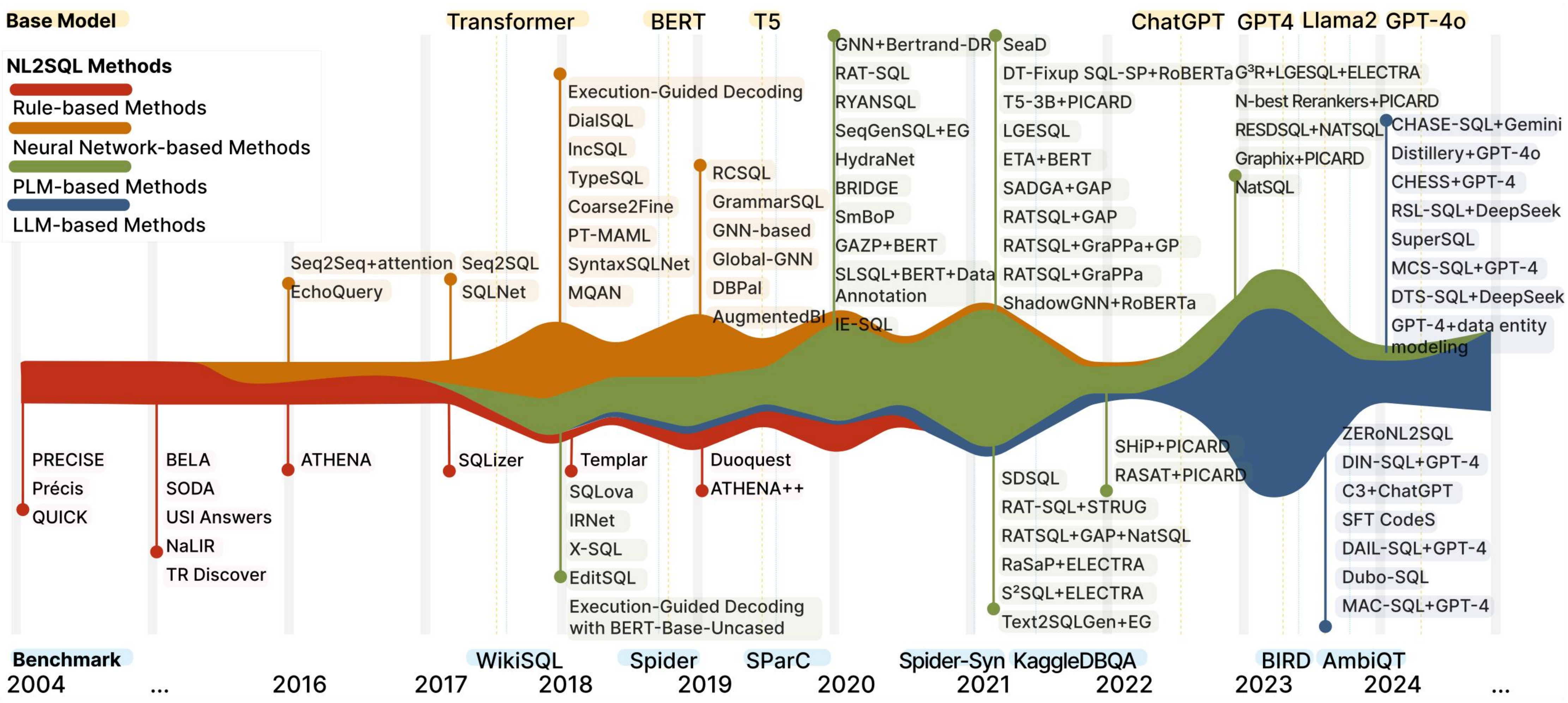
Yuankai Fan, Tonghui Ren, Can Huang,
Zhenying He, X.Sean Wang

Natural Language to SQL (NL2SQL)

- Writing SQLs to query databases is NOT easy for non-SQL-savvies
- Natural Language to SQL (**NL2SQL**) comes in rescue:
  1. Given a natural language query and the database scheme
  2. Generate the corresponding SQL query



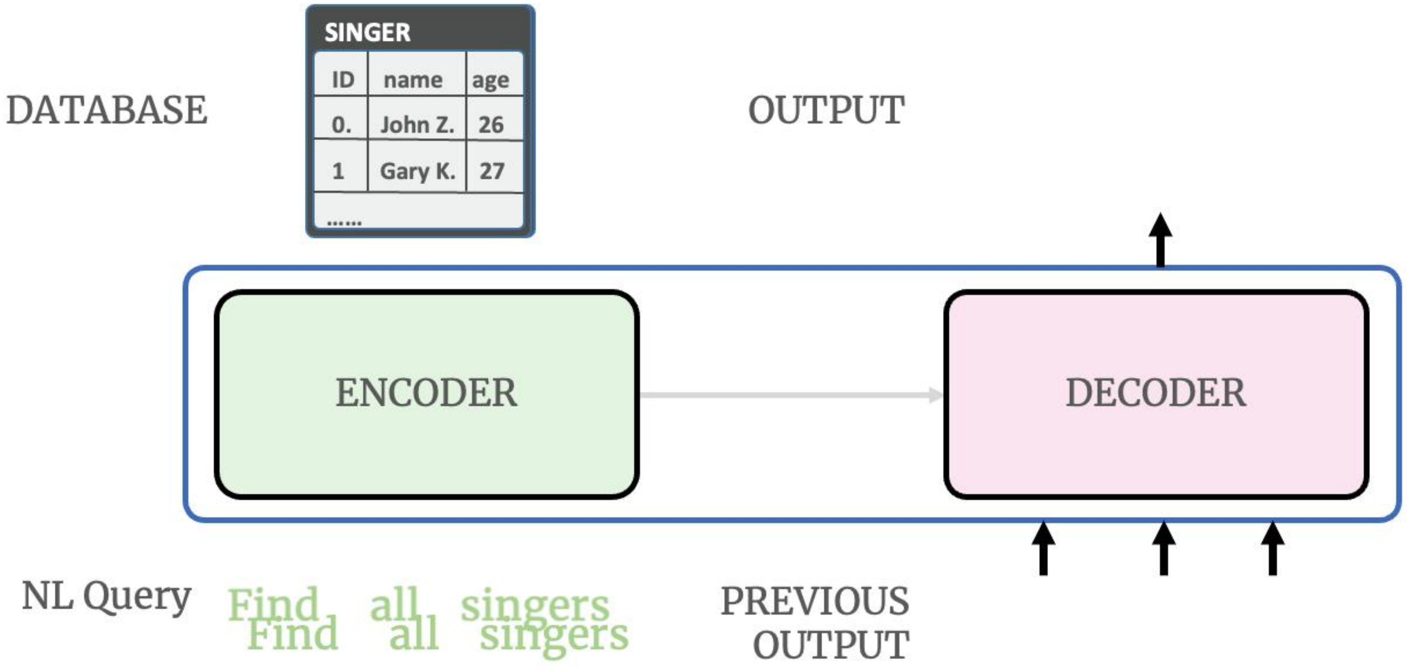
NL2SQL Evolution



Xinyu Liu et al. A Survey of NL2SQL with Large Language Models: Where are we, and where are we going? CoRR (2024)

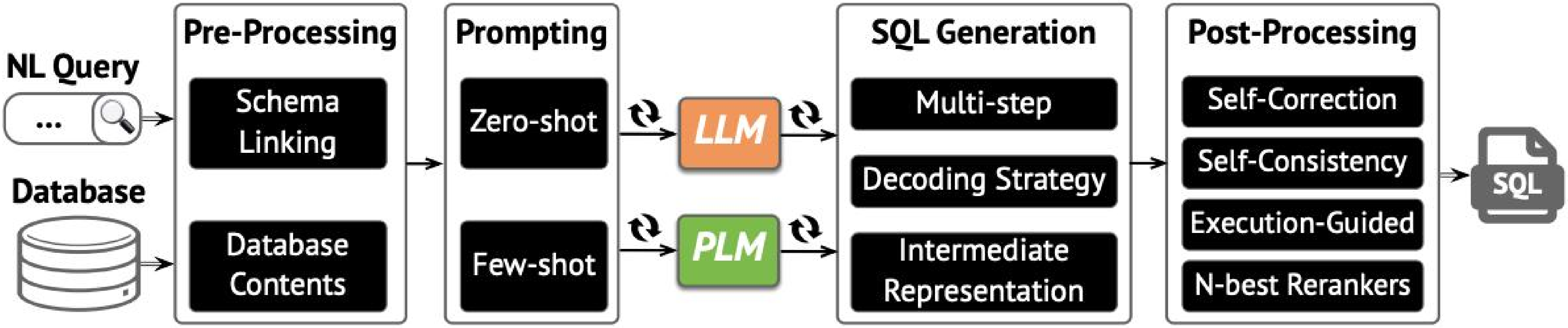
Pre-LLM Era Approach - Seq2Seq

- Consider as machine translation problem
- Based on **Sequence-to-sequence** Encoder-Decoder framework



Post-LLM Era Approach – LLMs

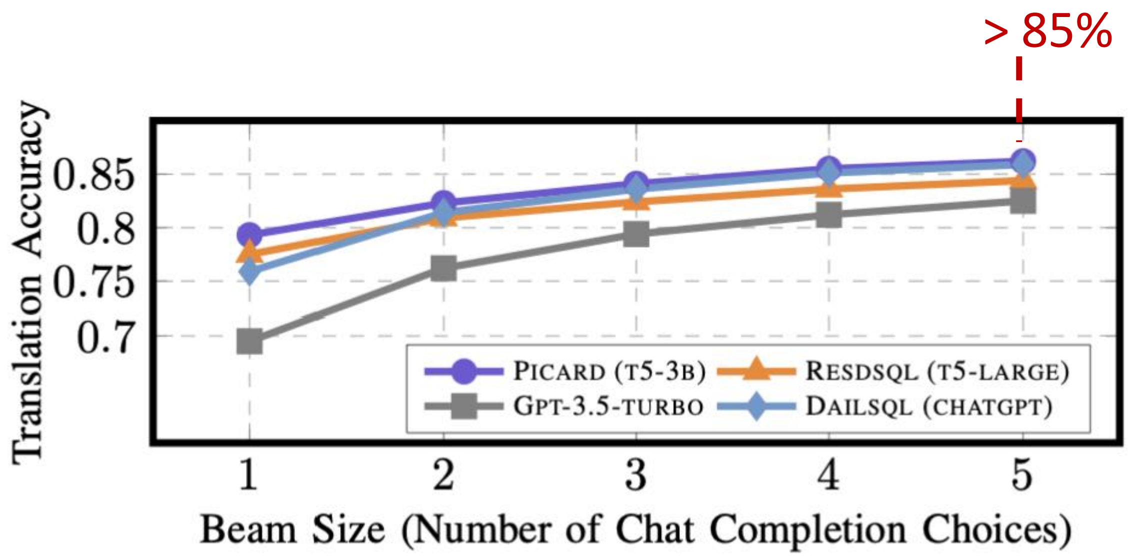
- Build NL2SQL system with **LLMs**
  1. Pre-processing
  2. SQL Generation
  3. Post-Processing



Boyan Li, et al, The Dawn of Natural Language to SQL: Are We Fully Ready? [Experiment, Analysis & Benchmark]. Proc. VLDB Endow.

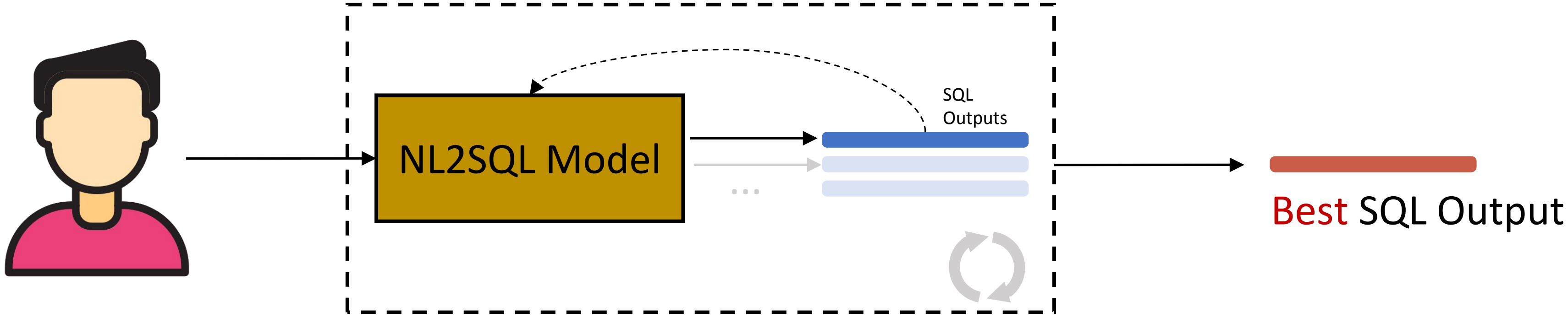
Unsatisfied Results

- Mainstream approaches (either Seq2seq models or LLMs) typically implement NL2SQL translation in **an end-to-end fashion**
- These models may benefit from broader exploration options over successive attempts



Idea: Feedback Loop in NL2SQL

- Can we build a **self-provided feedback loop** along with NL2SQL, to improve end-to-end performance?



Is SQL2NL Back-Translation Sufficient?

- Use **SQL2NL** to establish an **NL-to-SQL-to-NL** translation lifecycle?
- NOT reliable !!!
  - Fase Positive feedback 2 for NL Question 1

aid	flno	origin	destination
9	2	Los Angeles	Tokyo
3	7	Los Angeles	Sydney
3	13	Los Angeles	Chicago
10	68	Chicago	New York
9	76	Chicago	Los Angeles
7	33	Los Angeles	Honolulu
5	34	Los Angeles	Honolulu
1	99	Los Angeles	Washington D.C.
2	346	Los Angeles	Dallas
6	387	Los Angeles	Boston

aid	name	distance
1	Boeing 747-400	8430
2	Boeing 737-800	3383
3	Airbus A340-300	7120
4	British Aerospace Jetstream 41	1502
5	Embraer ERJ-145	1530
6	SAAB 340	2128
7	Piper Archer III	520
8	Tupolev 154	4103
9	Lockheed L1011	6900
10	Boeing 757-300	4010

NL-SQL pair

1 **[Question]:**
Show all flight numbers with aircraft Airbus A340-300.

[Ground-truth SQL]
```
SELECT T1.flno FROM Flight AS T1
JOIN Aircraft AS T2 ON T1.aid = T2.aid
WHERE T2.name = 'Airbus A340-300'
```

Model Output

[Translated SQL]
```
SELECT count ✗ FROM Flight AS T1
JOIN Aircraft AS T2 ON T1.aid = T2.aid
WHERE T2.name = 'Airbus A340-300'
```

count(*)
2

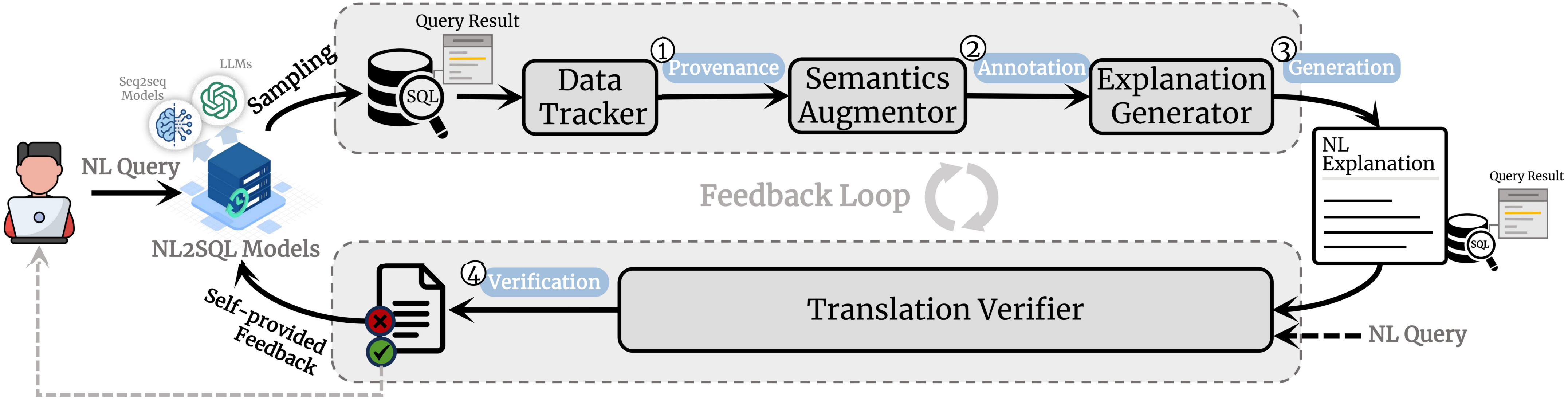
[Query Result]
<execution>

2 **[SQL2NL-generated Explanation]**
Find the number of flights that are operated by aircraft with the name Airbus A340-300.

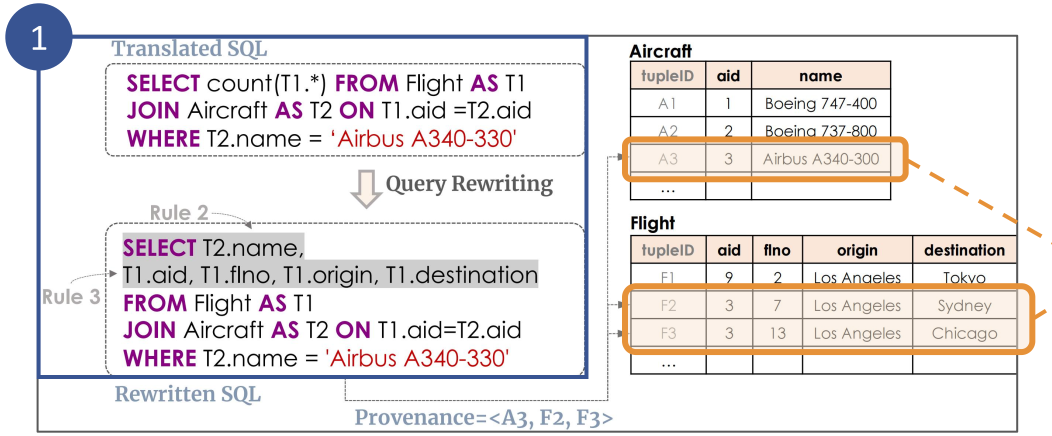
[Feedback] (Incorrect)
The translated SQL above is correct. ✗

CycleSQL: Self-Provided Feedback in NL2SQL

1. Data Provenance
2. Semantics Enrichment
3. Explanation Generation
4. Translation Verification

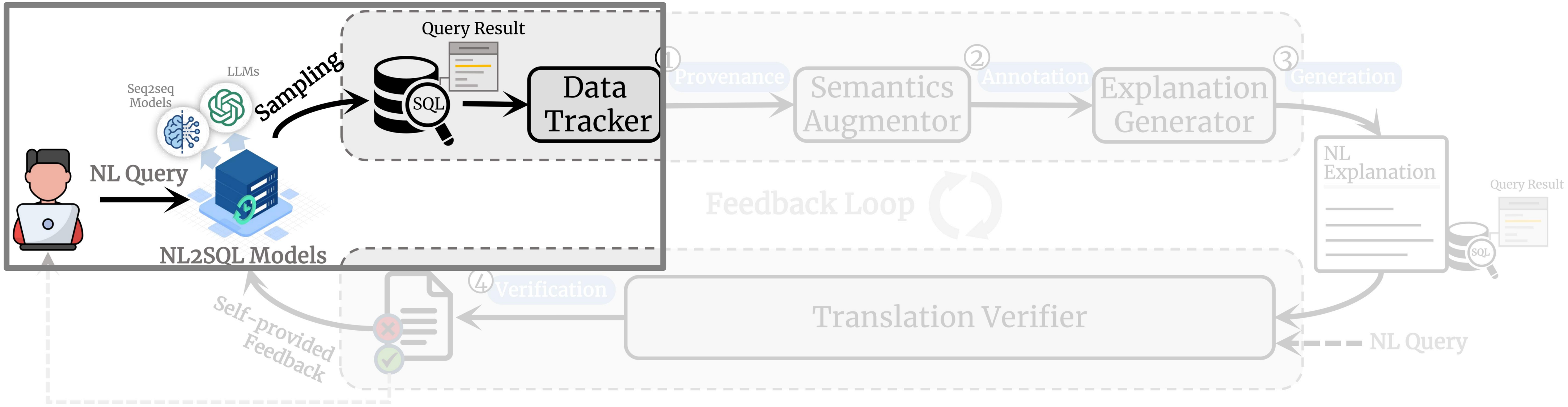


Data Provenance

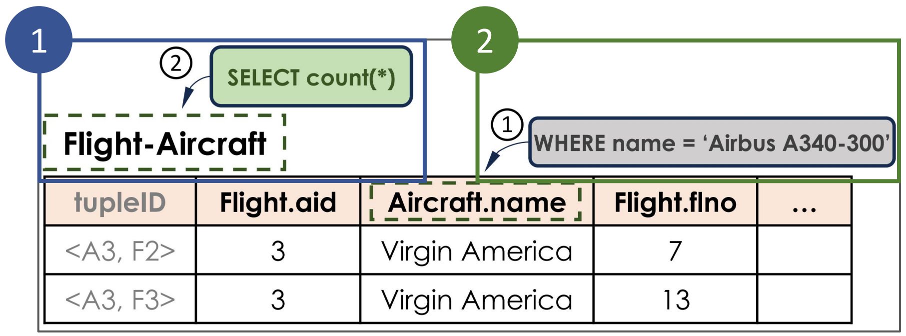


Provenance Tuples

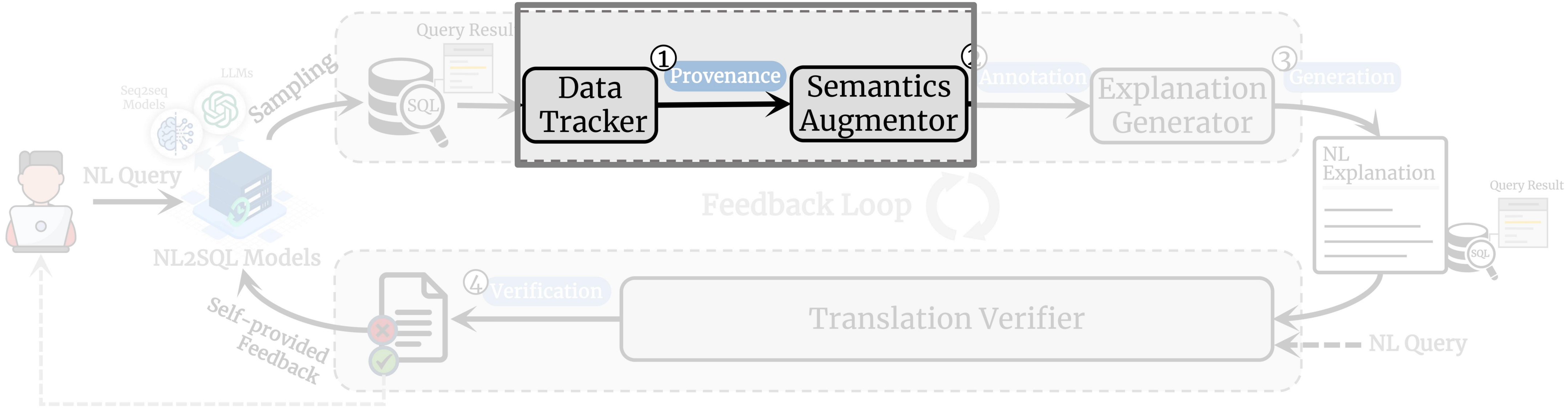
Capture provenance information using **query rewriting** 1



Semantics Enrichment



Integrate **operation-level semantics** 1 2 of the SQL queries to better reflect user query intent

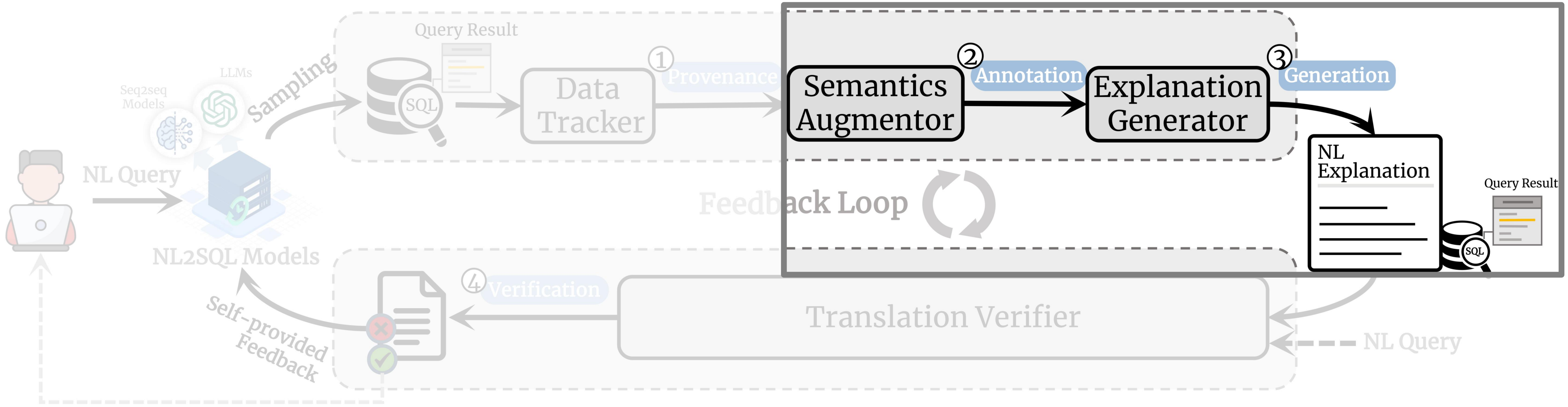


NL Explanation Text Generation

Synthesize **NL explanation** baesd on a rule-based method

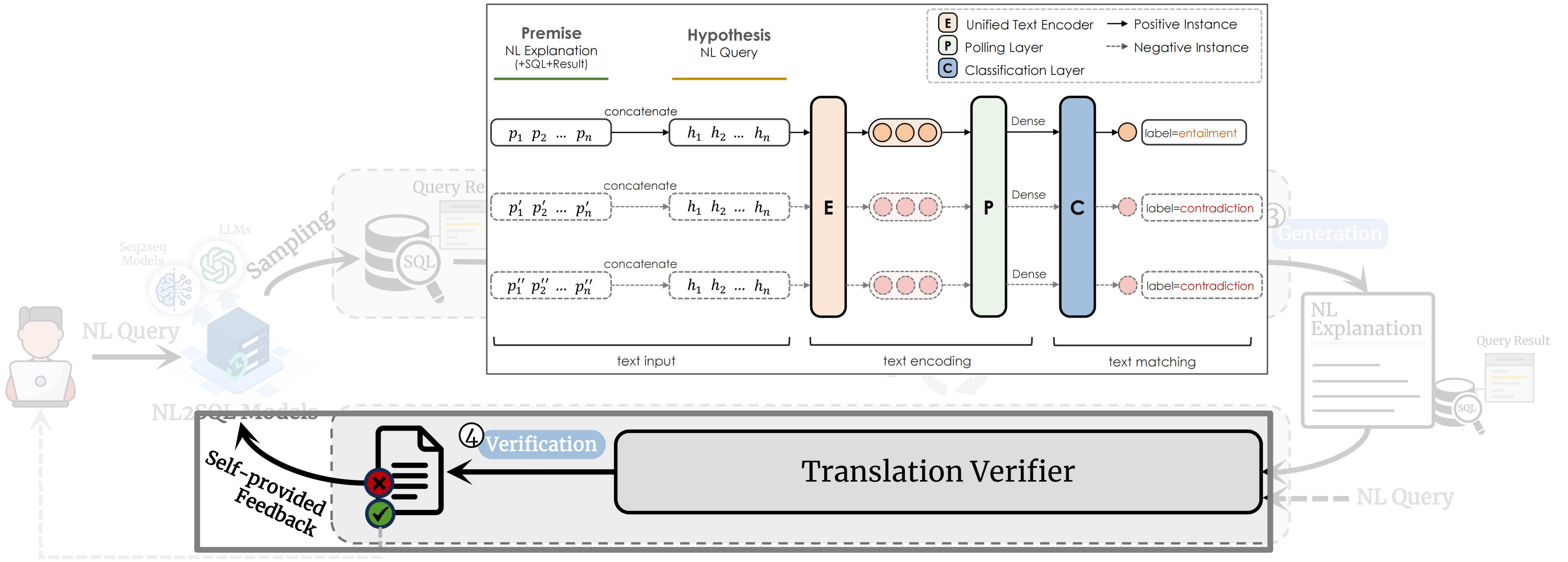
<u>Synthesized Explanation Text Example:</u>

The query returns a result with one column of aggregation type (count) and one row.
For lights with aircraft, named Airbus A340-300, there are 2 flights in total.



NL2SQL Validation

- Formulate the translation validation problem as a **textual entailment task**
- Use textual entailment model to determine if the translation is correct or not



Experimental Settings

- Benchmarks
  - Spider Spider and its variants (Spider-DK/Spider-Syn/Spider-Realistic)
  - ScienceBenchmark

- Baselines
  - **Seq2seq-based**: SmBoP/PICARD/RESDSQL
  - **LLM-based**: GPT-3.5-Turbo/GPT-4/CHESS/DAILSQL

- Metrics
  - Syntactic Accuracy (EM): *exact match* the ground truth
  - **Execution Accuracy (EX)**: *execution* result match
  - Test Suit Accuracy (TS): Similar to EX, but more robust

Overall Results

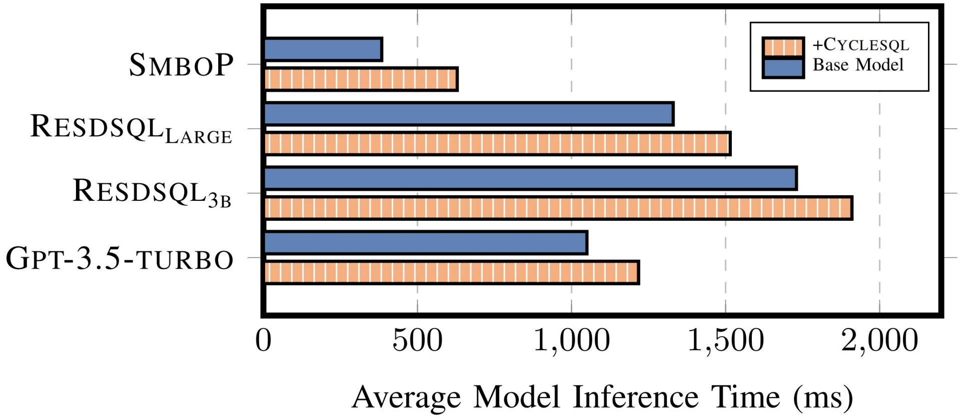
- CycleSQL **consistently** improves over all base models

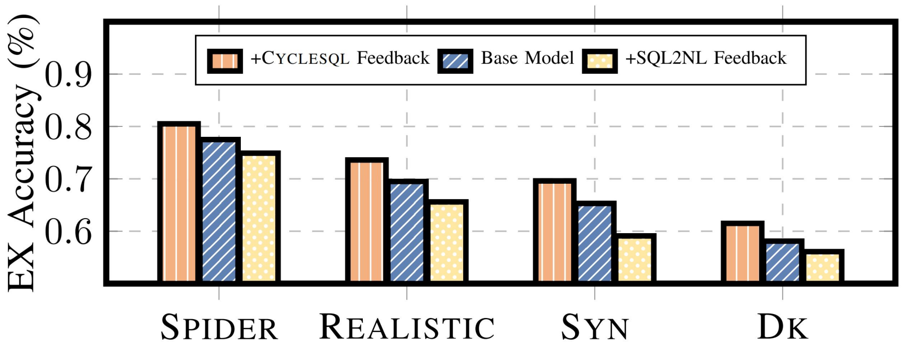
Models		Spider					Spider Variants								ScienceBenchmark		
		Validation			Test		Realistic			Syn			Dk		OncoMx	Cordis	Sdss
		EM	EX	TS	EM	EX	EM	EX	TS	EM	EX	TS	EM	EX	EX	EX	EX
Seq2seq-based Models																	
SmBoP	Base	**74.5**	77.8	72.5	69.5	71.1	**60.2**	61.4	56.1	**60.2**	64.9	58.3	**54.0**	59.1	16.2	**16.0**	7.0
	+CycleSQL	74.1	**78.8**	**73.1**	-	-	59.8	**62.2**	**56.7**	59.1	**65.5**	**58.6**	53.3	**59.8**	**17.2**	**16.0**	**8.0**
PICARD$_{3B}$	Base	75.9	79.3	**72.8**	72.4	75.1	**68.9**	71.7	**64.6**	65.6	69.8	63.2	49.7	63.2	32.3	**26.0**	**8.0**
	+CycleSQL	**76.1**	**79.9**	**72.8**	-	-	68.7	**71.9**	**64.6**	**65.9**	**70.8**	**63.6**	**50.3**	**63.6**	**33.3**	24.0	**8.0**
RESDSQL$_{Large}$	Base	74.0	77.5	71.6	66.8	73.5	67.3	69.5	61.6	58.8	65.3	59.6	50.1	58.1	33.0	29.0	4.0
	+CycleSQL	**75.1**	**80.5**	**74.0**	**68.9**	**77.1**	**69.1**	**73.6**	**64.2**	**61.6**	**69.6**	**63.2**	**50.5**	**61.5**	**35.0**	**32.0**	**8.0**
RESDSQL$_{3B}$	Base	76.0	79.4	73.5	70.2	78.4	68.9	72.2	64.8	62.9	69.5	63.2	52.0	59.3	34.1	28.0	6.0
	+CycleSQL	**76.8**	**82.0**	**76.0**	**72.5**	**81.6**	**71.3**	**76.5**	**67.5**	**65.5**	**73.3**	**66.6**	**52.5**	**61.3**	**37.4**	**31.0**	**8.0**
LLM-based Models																	
Gpt-3.5-turbo (5-shots)	Base	43.8	72.8	61.7	44.7	69.4	39.4	65.4	52.8	34.0	61.0	49.8	39.1	60.6	34.3	30.0	10.0
	+CycleSQL	**49.2**	**77.8**	**66.2**	**50.6**	**75.1**	**46.5**	**71.1**	**56.7**	**40.0**	**66.7**	**55.4**	**43.7**	**65.6**	**43.4**	**34.0**	**12.0**
Gpt-4 (5-shots)	Base	51.9	77.9	71.2	-	-	46.7	68.6	55.3	45.2	75.0	64.4	53.7	**68.5**	51.5	40.0	14.0
	+CycleSQL	**58.7**	**79.8**	**73.1**	-	-	**49.0**	**70.6**	**56.9**	**46.2**	**76.0**	**66.3**	**57.4**	**68.5**	**56.6**	**43.0**	**15.0**
Chess	Base	23.4	41.1	37.5	-	-	21.2	39.7	36.6	17.4	37.0	32.5	19.5	35.6	64.6	46.0	**40.0**
	+CycleSQL	**25.6**	**42.3**	**38.2**	-	-	**23.1**	**41.1**	**37.9**	**19.3**	**38.1**	**33.0**	**20.2**	**36.1**	**65.7**	**52.0**	**40.0**
DailSQL$_{3.5}$	Base	65.1	81.3	74.1													
	+CycleSQL	**67.2**	**81.8**	**74.3**													

Break-Down Results

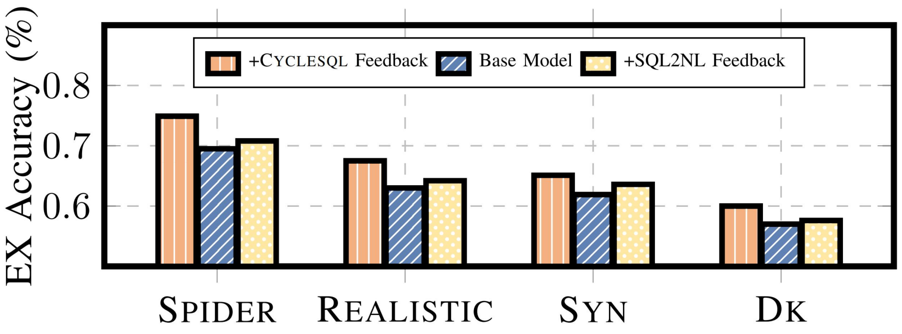
Model		Easy	Medium	Hard	Extra Hard
SmBoP	Base	90.7	82.7	70.7	52.4
	+CycleSQL	90.7	**84.1**($\uparrow$1.4)	69.5($\downarrow$1.2)	53.0($\uparrow$0.6)
PICARD$_{3B}$	Base	95.6	85.4	67.8	50.6
	+CycleSQL	95.6	86.1($\uparrow$0.7)	69.5($\uparrow$1.7)	50.6
RESDSQL$_{LARGE}$	Base	92.3	83.4	66.1	51.2
	+CycleSQL	93.5($\uparrow$1.2)	86.1($\uparrow$0.7)	**73.0**($\uparrow$**6.9**)	53.6($\uparrow$2.4)
RESDSQL$_{3B}$	Base	94.0	85.7	65.5	55.4
	+CycleSQL	94.0	**89.0**($\uparrow$**3.3**)	**74.7**($\uparrow$**9.2**)	53.0($\downarrow$0.4)
GPT-3.5-turbo (5-shots)	Base	84.3	78.5	65.5	48.2
	+CycleSQL	86.3($\uparrow$2.0)	**83.0**($\uparrow$**4.5**)	**73.0**($\uparrow$**7.5**)	**56.0**($\uparrow$**7.8**)
GPT-4 (5-shots)	Base	90.3	84.3	63.8	56.6
	+CycleSQL	90.7($\uparrow$0.4)	85.4($\uparrow$1.1)	**66.7**($\uparrow$**2.9**)	**59.6**($\uparrow$**3.0**)
CHESS	Base	70.2	25.3	39.7	19.3
	+CycleSQL	**71.6**($\uparrow$**1.4**)	25.6($\uparrow$**0.3**)	**41.1**($\uparrow$**1.4**)	**21.6**($\uparrow$**2.3**)
DAILSQL$_{3.5}$	Base	91.1	86.5	77.0	57.2
	+CycleSQL	91.1	**86.8**($\uparrow$**0.3**)	**77.6**($\uparrow$**0.6**)	**59.0**($\uparrow$**1.8**)

	SmBoP (+CycleSQL)	PICARD (+CycleSQL)	RESDSQL$_{LARGE}$ (+CycleSQL)	RESDSQL$_{3B}$ (+CycleSQL)	GPT-3.5-turbo (+CycleSQL)
Average Iterations	2.44	3.78	1.90	1.88	1.87





(a) Execution accuracy results on RESDSQL$_{LARGE}$



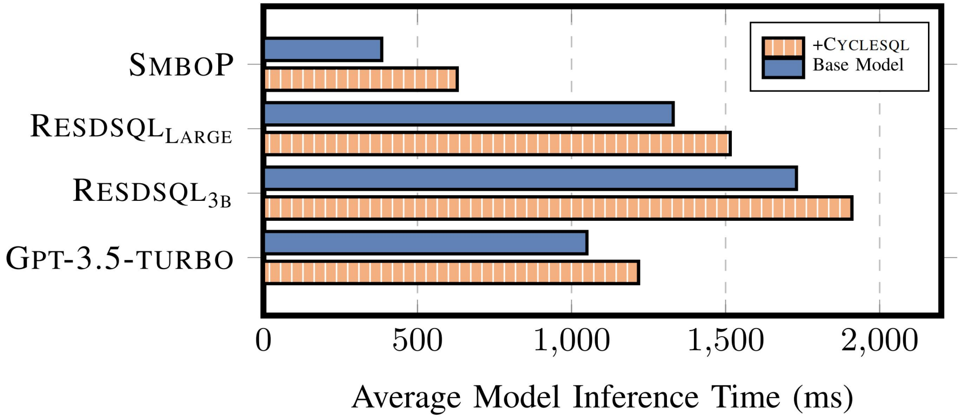
(b) Execution accuracy results on GPT-3.5-turbo

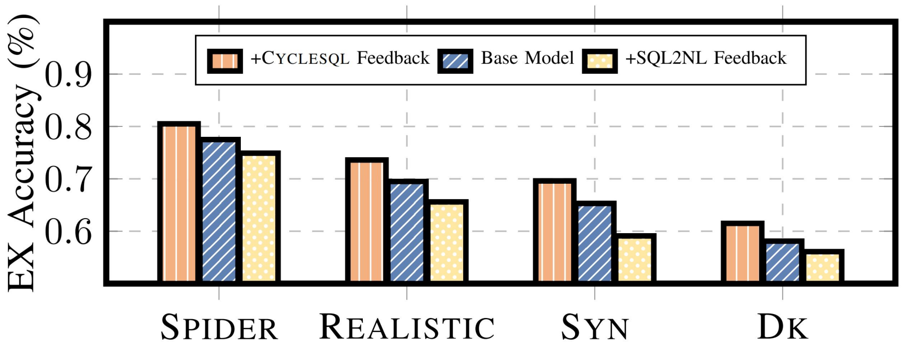
Break-Down Results

Model		Easy	Medium	Hard	Extra Hard
SMBoP	Base	90.7	82.7	70.7	52.4
	+CYCLESQL	90.7	**84.1**(↑1.4)	69.5(↓1.2)	53.0(↑0.6)
PICARD$_{3B}$	Base	95.6	85.4	67.8	50.6
	+CYCLESQL	95.6	86.1(↑0.7)	69.5(↑1.7)	50.6
	Base	92.3	83.4	66.1	51.2
GPT-3.5-TURBO (5-SHOTS)	Base	84.3	78.5	65.5	48.2
	+CYCLESQL	86.3(↑2.0)	**83.0**(↑4.5)	**73.0**(↑7.5)	**56.0**(↑7.8)
GPT-4 (5-SHOTS)	Base	90.3	84.3	63.8	56.6
	+CYCLESQL	90.7(↑0.4)	85.4(↑1.1)	**66.7**(↑2.9)	**59.6**(↑3.0)
CHESS	Base	70.2	25.3	39.7	19.3
	+CYCLESQL	**71.6**(↑1.4)	25.6(↑0.3)	**41.1**(↑1.4)	**21.6**(↑2.3)
DAILSQL$_{3.5}$	Base	91.1	86.5	77.0	57.2
	+CYCLESQL	91.1	**86.8**(↑0.3)	**77.6**(↑0.6)	**59.0**(↑1.8)

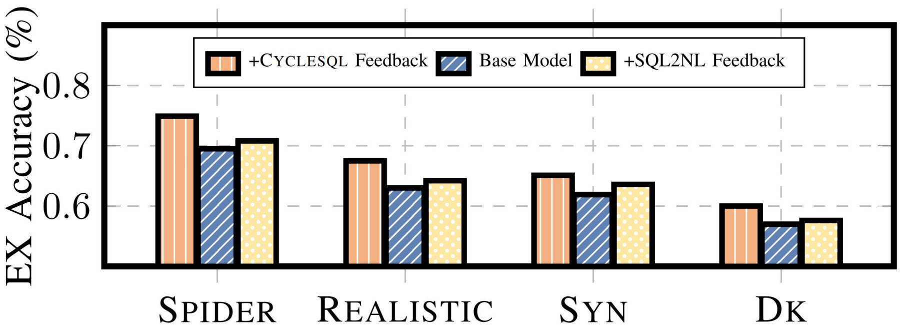
Significant Improvement over Extra-Hard-Queries on LLM Models

Average Iterations	SMBOP (+CYCLESQL)	PICARD (+CYCLESQL)	RESDSQL$_{LARGE}$ (+CYCLESQL)	RESDSQL$_{3B}$ (+CYCLESQL)	GPT-3.5-TURBO (+CYCLESQL)
	2.44	3.78	1.90	1.88	1.87





(a) Execution accuracy results on RESDSQL$_{LARGE}$



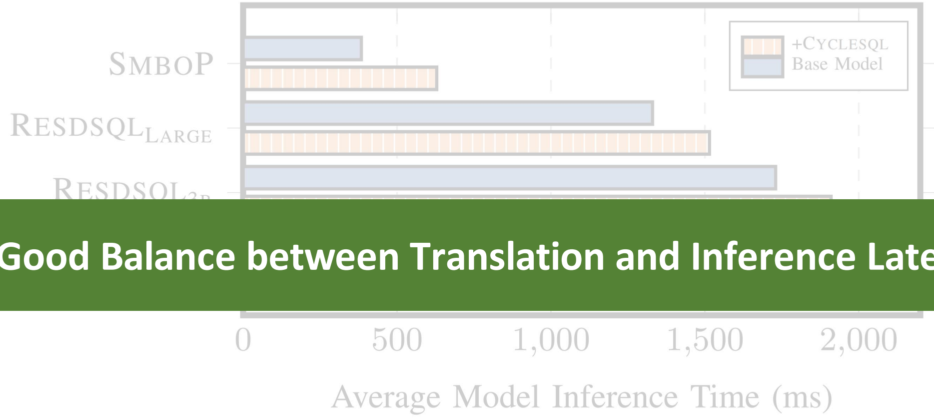
(b) Execution accuracy results on GPT-3.5-TURBO

Break-Down Results

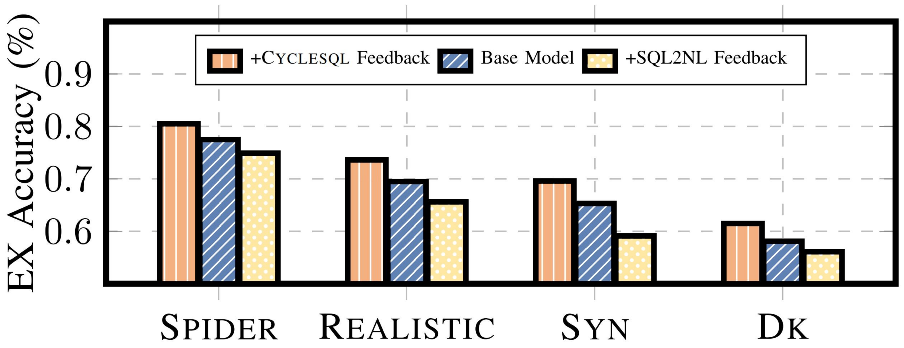
Model		Easy	Medium	Hard	Extra Hard
SMBoP	Base	90.7	82.7	70.7	52.4
	+CYCLESQL	90.7	**84.1**(↑1.4)	69.5(↓1.2)	53.0(↑0.6)
PICARD$_{3B}$	Base	95.6	85.4	67.8	50.6
	+CYCLESQL	95.6	86.1(↑0.7)	69.5(↑1.7)	50.6
	Base	92.3	83.4	66.1	51.2
GPT-3.5-TURBO (5-SHOTS)	Base	84.3	78.5	65.5	48.2
	+CYCLESQL	86.3(↑2.0)	**83.0**(↑4.5)	**73.0**(↑7.5)	**56.0**(↑7.8)
GPT-4 (5-SHOTS)	Base	90.3	84.3	63.8	56.6
	+CYCLESQL	90.7(↑0.4)	85.4(↑1.1)	**66.7**(↑2.9)	**59.6**(↑3.0)
CHESS	Base	70.2	25.3	39.7	19.3
	+CYCLESQL	**71.6**(↑1.4)	25.6(↑0.3)	**41.1**(↑1.4)	**21.6**(↑2.3)
DAILSQL$_{3.5}$	Base	91.1	86.5	77.0	57.2
	+CYCLESQL	91.1	**86.8**(↑0.3)	**77.6**(↑0.6)	**59.0**(↑1.8)

Significant Improvement over Extra-Hard-Queries on LLM Models

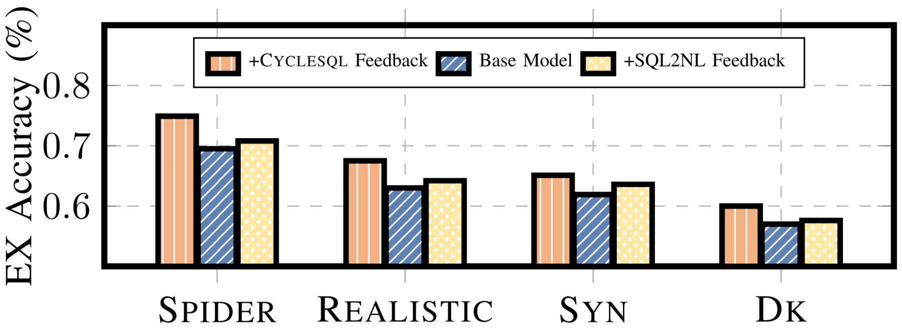
	SMBOP (+CYCLESQL)	PICARD (+CYCLESQL)	RESDSQL$_{LARGE}$ (+CYCLESQL)	RESDSQL$_{3B}$ (+CYCLESQL)	GPT-3.5-TURBO (+CYCLESQL)
Average Iterations	2.44	3.78	1.90	1.88	1.87



Good Balance between Translation and Inference Latency



(a) Execution accuracy results on RESDSQL$_{LARGE}$



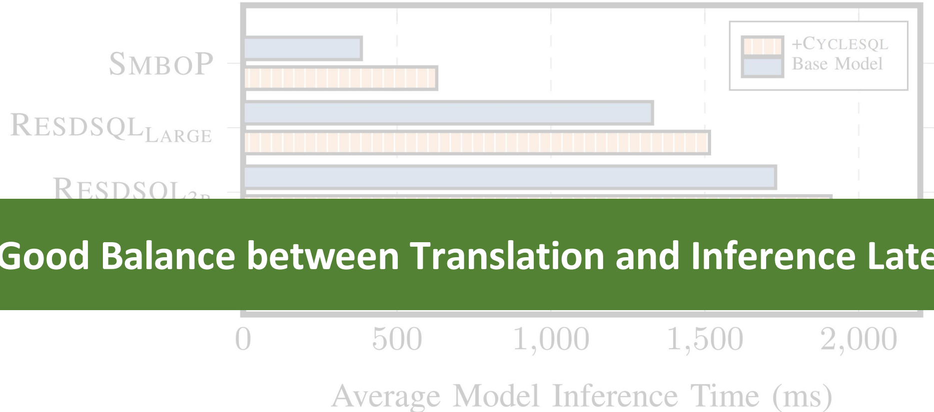
(b) Execution accuracy results on GPT-3.5-TURBO

Break-Down Results

Model		Easy	Medium	Hard	Extra Hard
SMBoP	Base	90.7	82.7	70.7	52.4
	+CycleSQL	90.7	**84.1**(↑1.4)	69.5(↓1.2)	53.0(↑0.6)
PICARD_{3B}	Base	95.6	85.4	67.8	50.6
	+CycleSQL	95.6	86.1(↑0.7)	69.5(↑1.7)	50.6
	Base	92.3	83.4	66.1	51.2
GPT-3.5-turbo (5-shots)	Base	84.3	78.5	65.5	48.2
	+CycleSQL	86.3(↑2.0)	**83.0**(↑4.5)	**73.0**(↑7.5)	**56.0**(↑7.8)
GPT-4 (5-shots)	Base	90.3	84.3	63.8	56.6
	+CycleSQL	90.7(↑0.4)	85.4(↑1.1)	**66.7**(↑2.9)	**59.6**(↑3.0)
CHESS	Base	70.2	25.3	39.7	19.3
	+CycleSQL	**71.6**(↑1.4)	25.6(↑0.3)	**41.1**(↑1.4)	**21.6**(↑2.3)
$\text{DAILSQL}_{3.5}$	Base	91.1	86.5	77.0	57.2
	+CycleSQL	91.1	**86.8**(↑0.3)	**77.6**(↑0.6)	**59.0**(↑1.8)

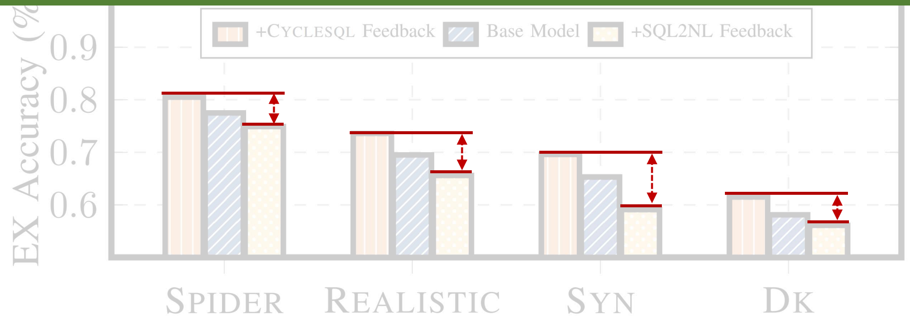
Significant Improvement over Extra-Hard-Queries on LLM Models

	SMBoP (+CycleSQL)	PICARD (+CycleSQL)	$\text{RESDSQL}_{\text{LARGE}}$ (+CycleSQL)	RESDSQL_{3B} (+CycleSQL)	GPT-3.5-turbo (+CycleSQL)
Average Iterations	2.44	3.78	1.90	1.88	1.87

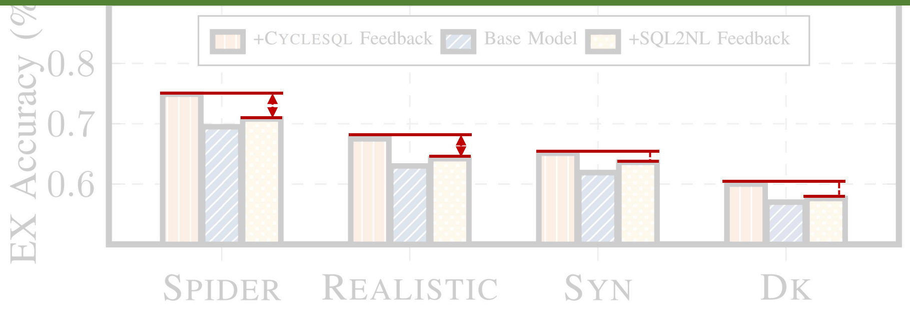


Good Balance between Translation and Inference Latency

CycleSQL Feedback is Better than SQL2NL Feedback!



(a) Execution accuracy results on $\text{RESDSQL}_{\text{LARGE}}$



(b) Execution accuracy results on GPT-3.5-turbo

More Results in the Paper

- Comparison of different translation verifier selection
- Qualitative evaluation results

Conclusion

- **Closing the feedback loop** for NL2SQL brings good benefits
- Good feedback improves not only **accuracy**, but **explanability**

Yuankai Fan <kaimary1221@gmail.com>

https://github.com/Kaimary/CycleSQL