Towards Optimizing SQL Generation via LLM Routing

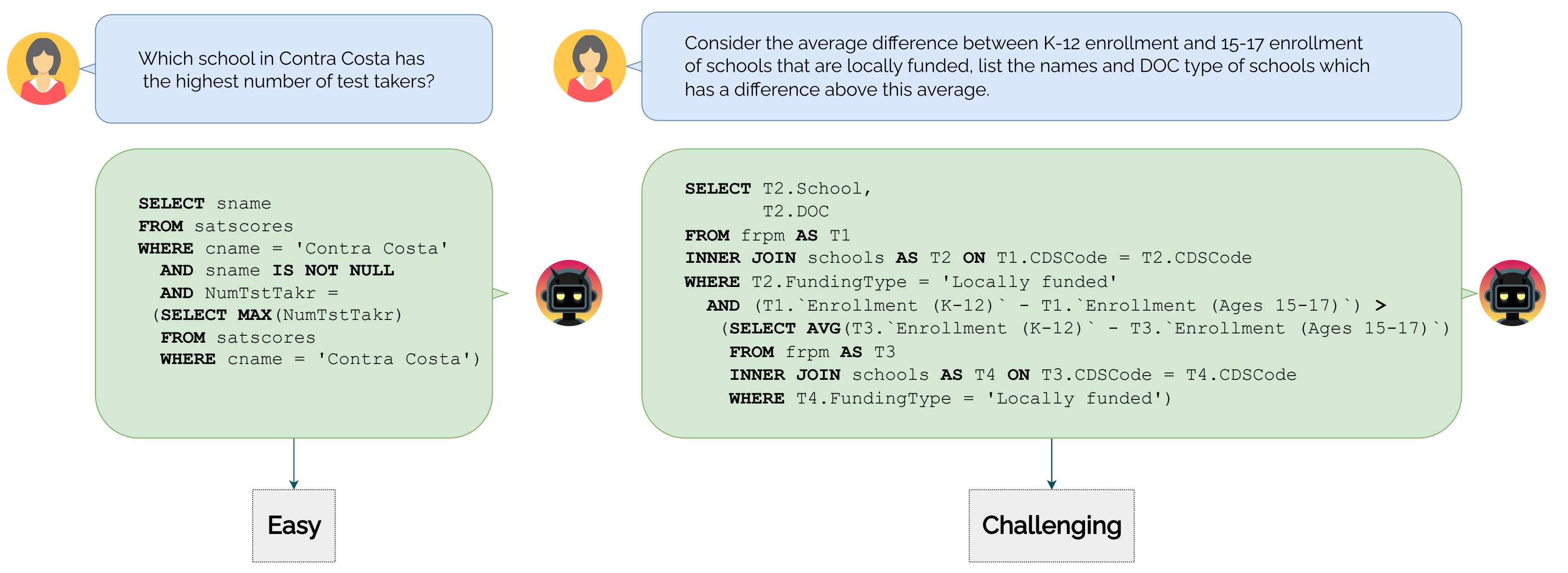
Mohammadhossein Malekpour Nour Shaheen Foutse Khomh Amine Mhedhbi

Polytechnique Montréal

Given an NL query, route to the cheapest LLM capable of generating executable SQL

Introduction

Text-to-SQL allows users to query databases in natural language, making data more accessible to non-experts. However, using powerful LLMs like `GPT-4o` for all queries leads to unnecessary costs and latency, especially for simpler queries.



Which school in Contra Costa has the highest number of test takers?

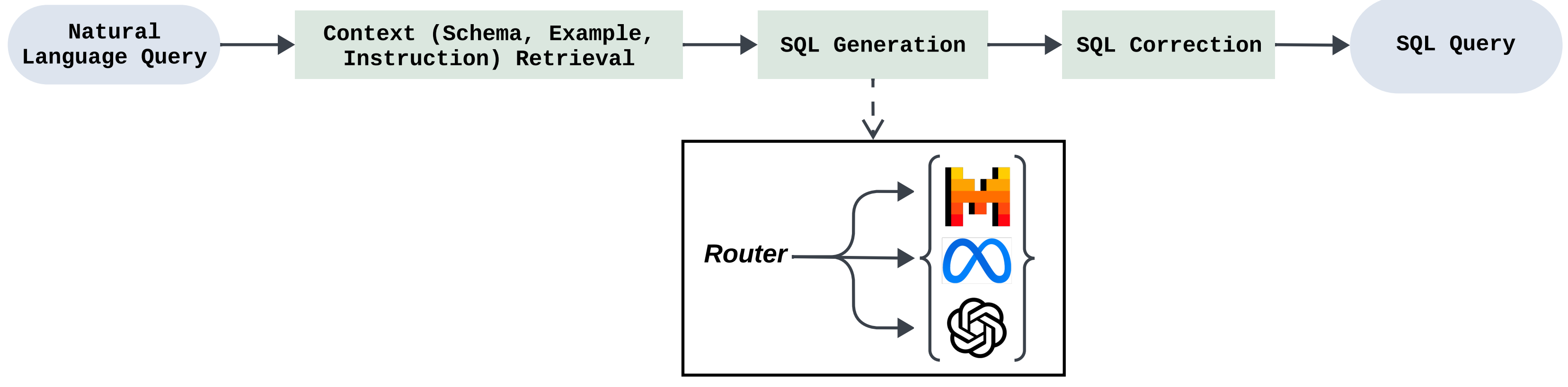
```
SELECT sname
FROM satscores
WHERE cname = 'Contra Costa'
  AND sname IS NOT NULL
  AND NumTstTakr =
 (SELECT MAX(NumTstTakr)
  FROM satscores
  WHERE cname = 'Contra Costa')
```

Consider the average difference between K-12 enrollment and 15-17 enrollment of schools that are locally funded, list the names and DOC type of schools which has a difference above this average.

```
SELECT T2.School,
       T2.DOC
FROM frpm AS T1
INNER JOIN schools AS T2 ON T1.CDSCode = T2.CDSCode
WHERE T2.FundingType = 'Locally funded'
  AND (T1.`Enrollment (K-12)` - T1.`Enrollment (Ages 15-17)`) >
   (SELECT AVG(T3.`Enrollment (K-12)` - T3.`Enrollment (Ages 15-17)`)
    FROM frpm AS T3
    INNER JOIN schools AS T4 ON T3.CDSCode = T4.CDSCode
    WHERE T4.FundingType = 'Locally funded')
```

Problem statement

Given a set of N models $\{\mathcal{M}_0, \mathcal{M}_1, \ldots, \mathcal{M}_{N-1}\}$ with varying SQL generation capabilities and costs, the goal is to select the weakest model $\mathcal{M}_i$ that can generate accurate SQL for a query Q, balancing accuracy, cost, and latency. We use a dataset H of past queries and SQL outputs to create an N-ary routing function that assigns a query to the best model $\mathcal{M}_l$ (where $l \in [0, N-1]$), or determines that no model can generate SQL ($l = N$).



Datasets & Metrics

Dataset: BIRD →1534 dev queries and 9428 training queries

Metric: Execution Accuracy (EX) measures the proportion of queries in the evaluation set S where the predicted SQL's output matches the ground-truth relation, with $0 \leq EX(S) \leq 1$.

$$EX = \frac{1}{N}\sum_{n=1}^{N} \mathbf{1}(V_n, \hat{V}_n) \quad \text{where} \quad \mathbf{1}(V_n, \hat{V}_n) = \begin{cases} 1 & \text{if } V_n = \hat{V}_n \\ 0 & \text{if } V_n \neq \hat{V}_n \end{cases}$$

Failure Analysis

The analysis shows a clear gap in model capabilities, with high overlap in failed queries. This suggests that routing primarily reduces costs, without improving EX beyond the strongest model.

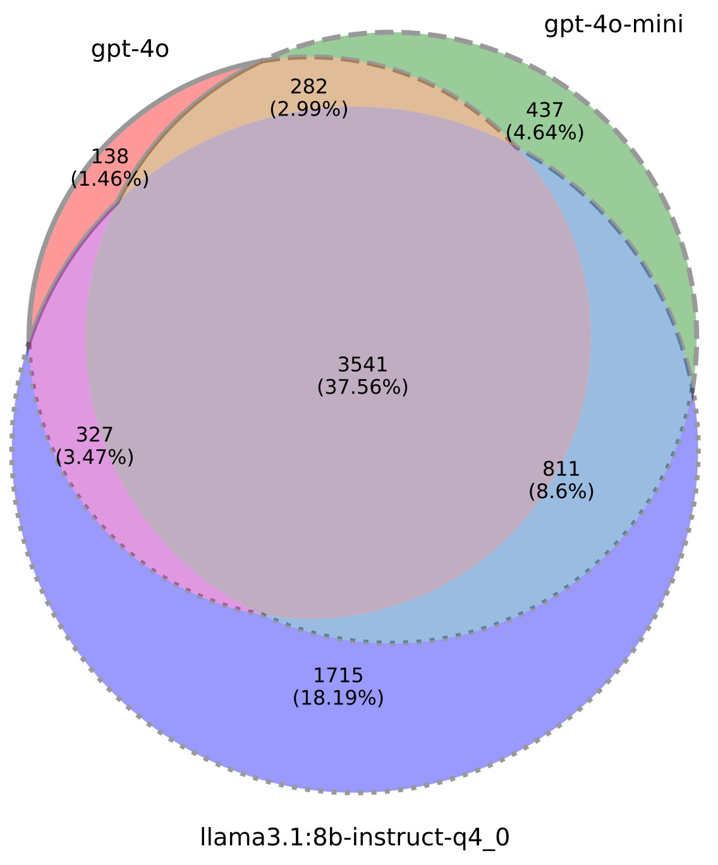


Figure 1. Failure Cases Dist.

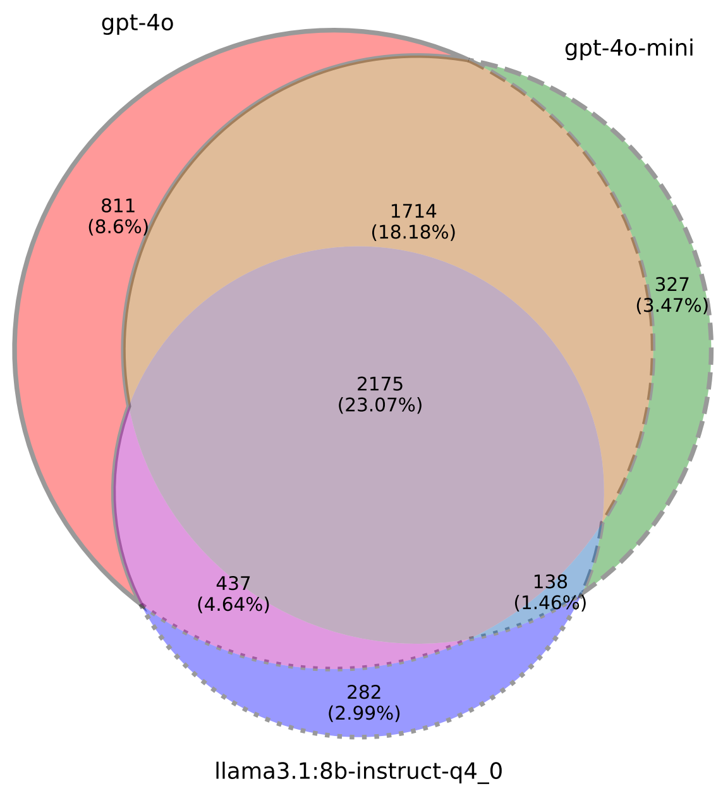
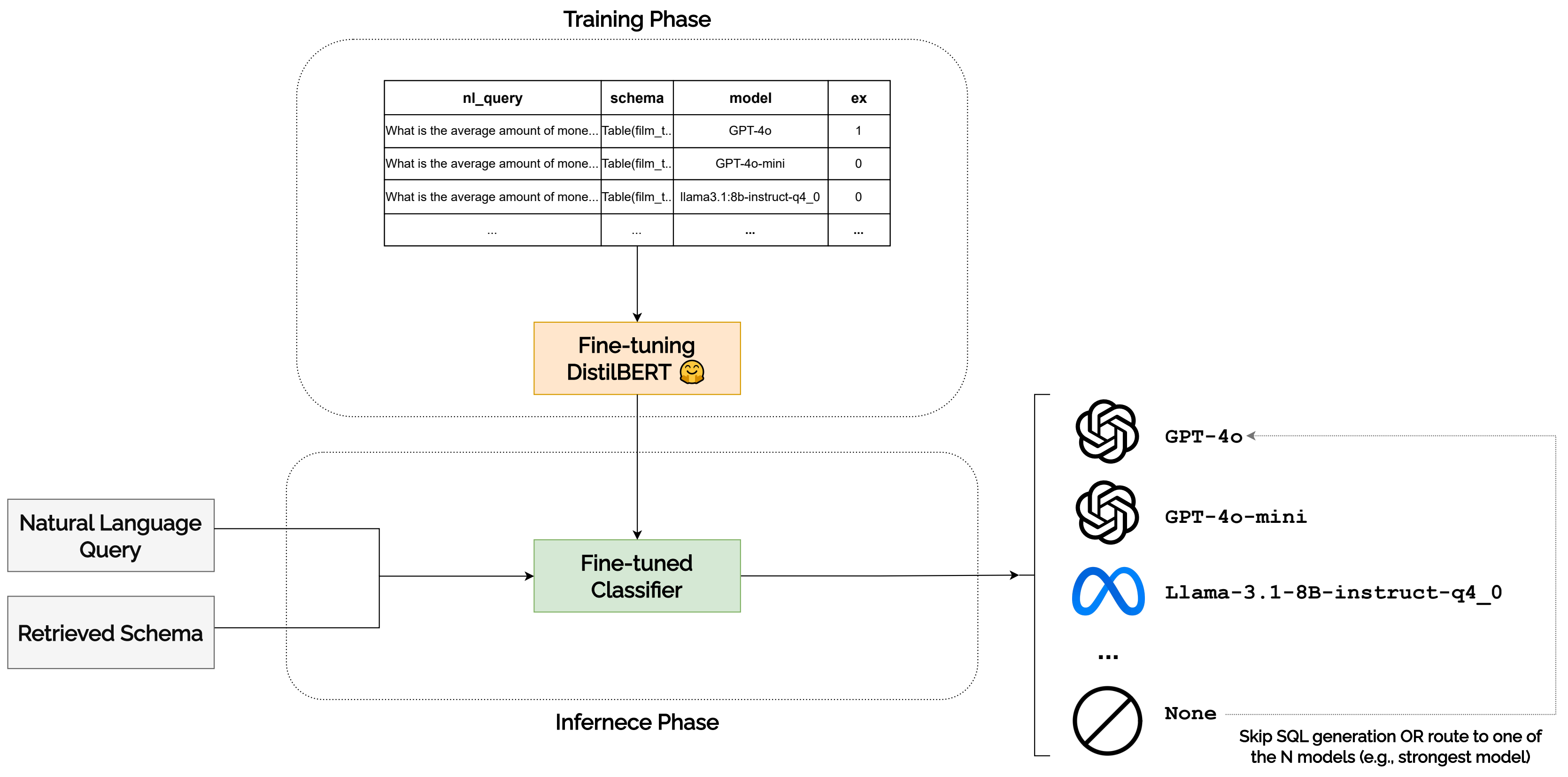


Figure 2. Correct Cases Dist.

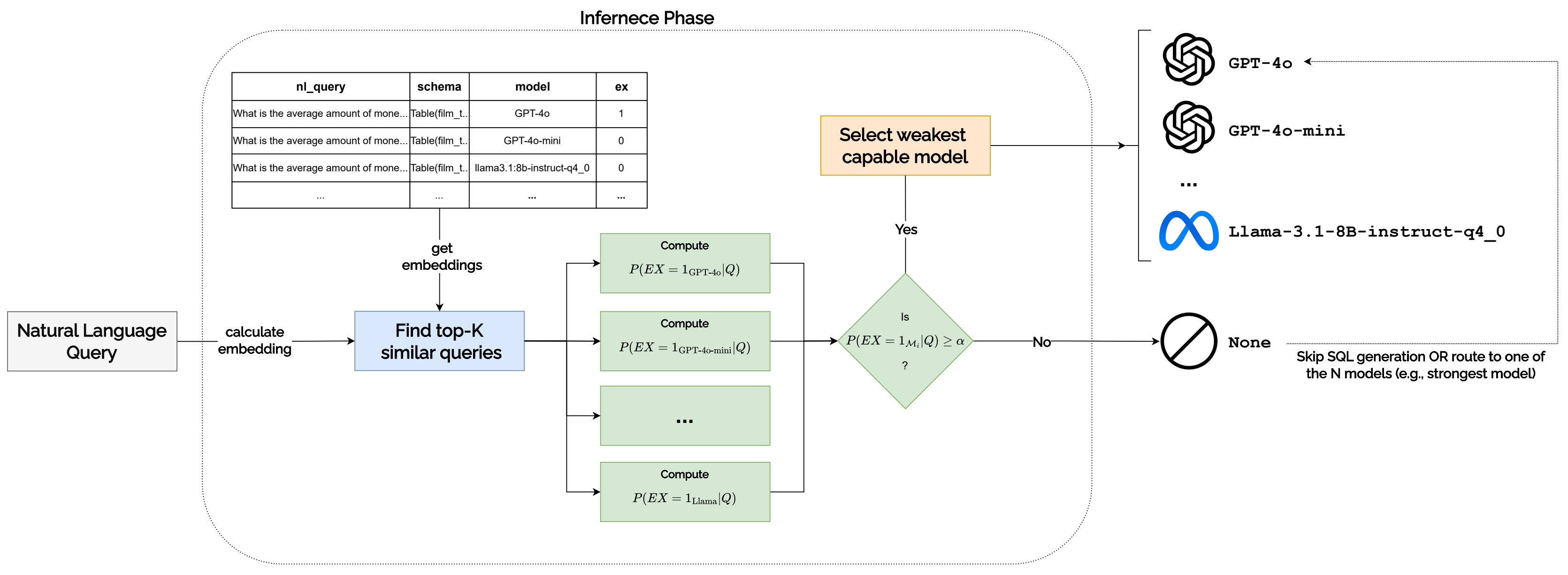
Methodology

Classification-Based Routing (R_{BERT})



nl_query	schema	model	ex
What is the average amount of mone...	Table(film_t..	GPT-4o	1
What is the average amount of mone...	Table(film_t..	GPT-4o-mini	0
What is the average amount of mone...	Table(film_t..	llama3.1:8b-instruct-q4_0	0
...	...	...	...

Score-Based Routing (R_k^α)



Infernece Phase
get embeddings
Natural Language Query
calculate embedding
Find top-K similar queries
Compute $P(EX = 1_{\text{GPT-4o}}|Q)$
Compute $P(EX = 1_{\text{GPT-4o-mini}}|Q)$
...
Compute $P(EX = 1_{\text{Llama}}|Q)$
Is $P(EX = 1_{\mathcal{M}_i}|Q) \geq \alpha$?
Yes
No
Select weakest capable model
GPT-4o
GPT-4o-mini
...
Llama-3.1-8B-instruct-q4_0
None
Skip SQL generation OR route to one of the N models (e.g., strongest model)

nl_query	schema	model	ex
What is the average amount of mone...	Table(film_t..	GPT-4o	1
What is the average amount of mone...	Table(film_t..	GPT-4o-mini	0
What is the average amount of mone...	Table(film_t..	llama3.1:8b-instruct-q4_0	0
...	...	...	...

Results

Gen.	EX%	gpt-4o	4o-mini	Llama	None	Cost Red.
`gpt-4o`	61.02	1534	-	-	-	1x
`4o-mini`	49.22	-	1534	-	-	16.6x
`Llama`	29.34	-	-	1534	-	∞
$R_{25}^{0.7}$	60.14	197	88	5	1243	1.1x
$R_{10}^{0.8}$	59.42	160	127	26	1220	1.1x
$R_{24}^{0.6}$	57.92	324	265	54	890	1.3x
R_{BERT}	55.21	118	311	167	938	1.4x

Accuracy & Cost Trade-Off

In the score-based approach, by adjusting K (similar queries) and α (threshold), we balance EX and cost. Higher values improve EX but rely more on the strongest model, increasing cost.

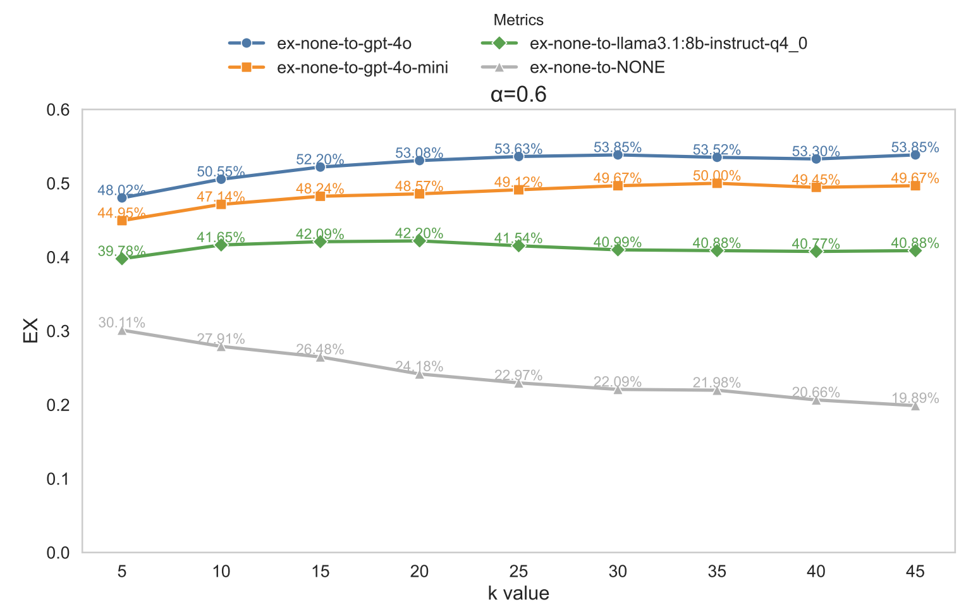


Figure 3. EX vs. K

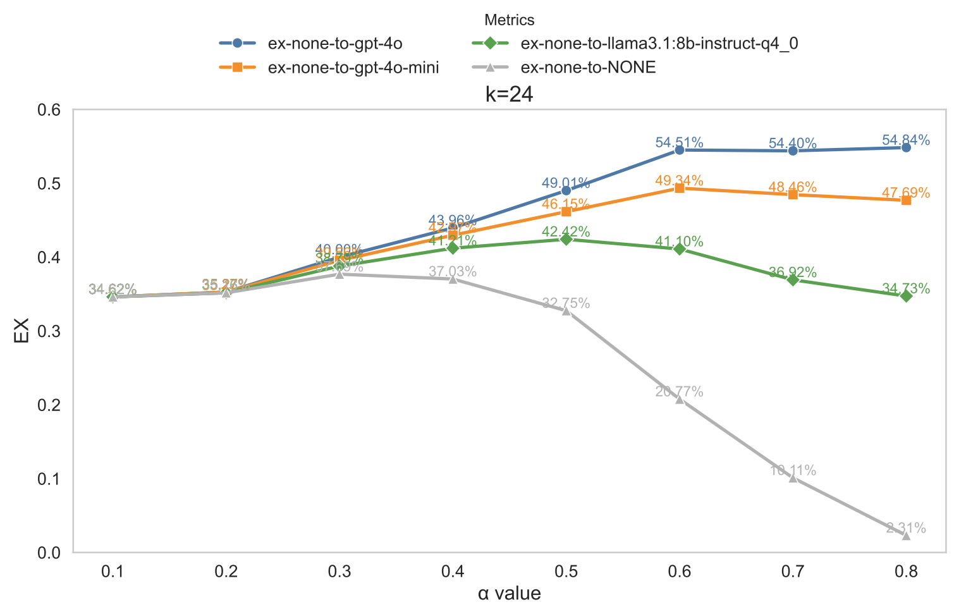


Figure 4. EX vs. α

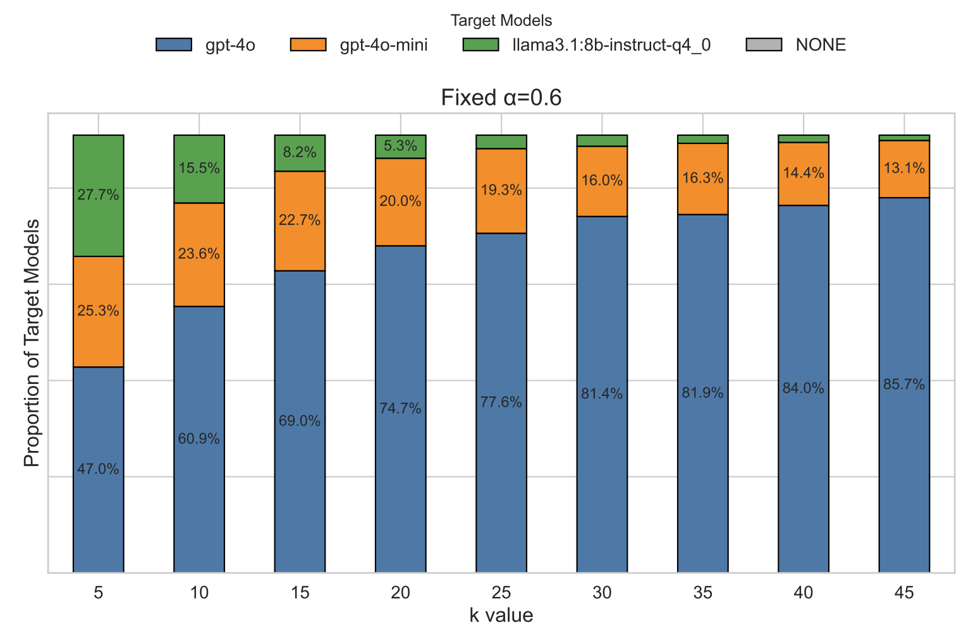


Figure 5. Dist. vs. K (None→`gpt-4o`)

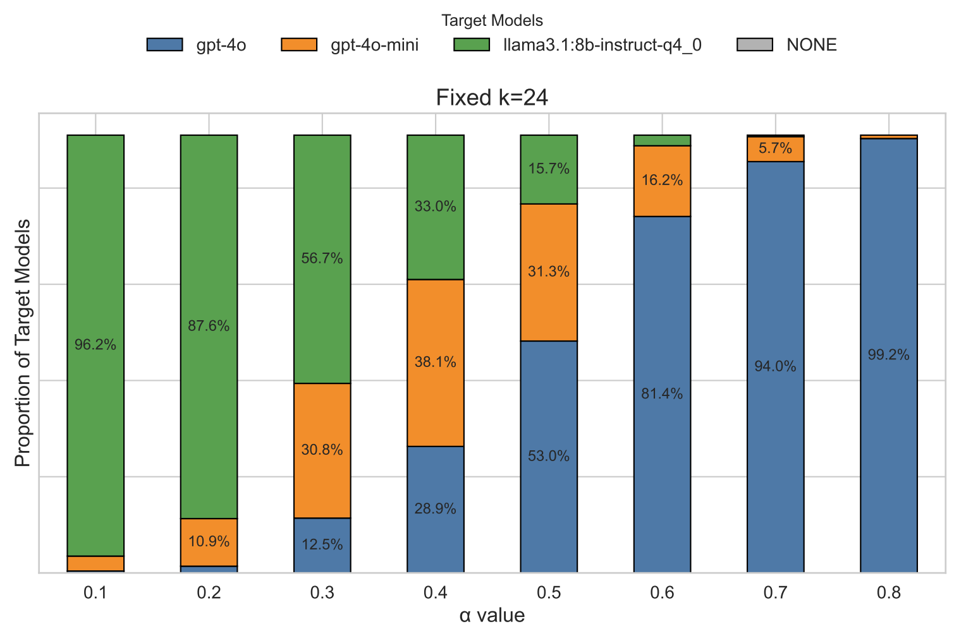


Figure 6. Dist. vs. α (None→`gpt-4o`)

github.com/dsg-polymtl   TRL @ NeurIPS 2024 – arXiv:2411.04319   {firstname.lastname}@polymtl.ca