Bridging the Semantic Gap: Contrastive Rewards for Multilingual Text-to-SQL with GRPO

Ashish Kattamuri
Independent Researcher
Denver, USA
ashishkattamuri@gmail.com

Ishita Prasad
University of Michigan
Seattle, USA
Isprasad@umich.edu

Meetu Malhotra
Harrisburg University of Science and Technology
Harrisburg, USA
Fmeetu@my.harrisburgu.edu

Arpita Vats
Linkedin
California, USA
arpita.vats09@gmail.com

Rahul Raja
Linkedin
California, USA
rahul.110392@gmail.com

Albert Lie
Forward Labs AI
New York, USA
albert@forwardlabs.ai

***Abstract*—Current Text-to-SQL methods are evaluated and only focused on executable queries overlooking the semantic alignment challenge both in terms of the semantic meaning of the query and the execution results correctness. Even execution accuracy itself shows significant drops when moving from English to other languages, with an average decline of 6 percentage points across non-English languages. We address these challenges by presenting a new framework that combines Group Relative Policy Optimization (GRPO) within a multilingual contrastive reward signal to enhance both task efficiency and semantic accuracy in Text-to-SQL systems in cross-lingual scenarios. Our method teaches models to obtain better correspondence between SQL generation and user intent by combining a reward signal based on semantic similarity. On the seven language MultiSpider dataset, finetuning the Llama-3-3B model with GRPO improved the execution accuracy up to 87.4% (+26 pp over zero-shot) and semantic accuracy up to 52.29% (+32.86 pp). Adding our contrastive reward signal in the GRPO framework further improved the average semantic accuracy to 59.14% (+6.85 pp, up to +10 pp for Vietnamese). Our experiments showcase that a smaller parameter efficient 3B llama model finetuned with our contrastive reward signal outperforms a much larger zero shot 8B llama model with uplift of 7.43pp on execution accuracy from 81.43% on 8B model to 88.86% on 3B model and cutting close on semantic accuracy with 59.14% on 3B vs 68.57% on 8B just with 3,000 reinforcement learning training examples. These results demonstrate how we can improve the performance of Text-to-SQL systems with contrastive rewards for directed semantic alignment without requiring huge amounts of datasets for training.**

***Index Terms*—Multilingual natural language processing, Text-to-SQL, Semantic parsing, Contrastive learning, Reinforcement learning, Cross-lingual alignment, SQL generation, Neural semantic parsing, Large language models (LLMs), Language-agnostic models, Sequence-to-sequence learning, Natural language interfaces to databases.**

I. Introduction

With the current rise in adoptions and integrations of LLMs and improvements in Text-to-SQL systems, the multilingual aspect of these systems will be crucial as it will enable users across the world to query databases in their native language. While the Text-to-SQL systems enable users to access and run queries on the database without experience in coding SQL, the models' capabilities to support multiple languages are still limited. Current approaches suggest that the models' performance is better in English than other languages, with notable performance gaps observed in multilingual settings. The language models like Llama-3 show good results for multilingual generation but still struggle with cross-lingual consistency [1].

Current approaches for improving multilingual Text-to-SQL systems are limited to reinforcement learning methods like GRPO [11] that focus primarily on mathematical reasoning tasks, while specialized prompting techniques like domain-specific knowledge injection [9] and chain-of-thought prompting [13] lack robust feedback mechanisms for semantic alignment. We address these limitations by proposing a novel method with a multilingual contrastive reward with Group Relative Policy Optimization (GRPO) to address this semantic alignment challenge and improve the execution and semantic accuracy of the Text-to-SQL systems. The new contrastive reward signal is used for training the Text-to-SQL model to score how closely the generated SQL query aligns with the intent of the users natural language query. We trained XLM-RoBERTa encoder [5] to create embeddings in the shared semantic space of the questions and the SQL queries for computing the contrastive reward for the generated SQL query. We ran our experiments on MultiSpider dataset [7] with 7 languages (Vietnamese, Spanish, Japanese, German, English, Chinese, and French) and achieved remarkable improvements with the integration of our contrastive reward signal in the GRPO framework giving us an increase in accuracy by 10 percentage points (pp) for Vietnamese and 6 pp on average on all other languages combined. In this paper we present our novel contrastive reward signal and the scalable architecture using XLM-RoBERTa encoder to compute the contrastive reward for the generated SQL query, and a lightweight training recipe that achieves 8B-level performance with a 3B model

979-8-3315-8704-8/25/$31.00 ©2025 IEEE

using only 3,000 examples, making high-quality multilingual Text-to-SQL more accessible and resource efficient.

II. Related Work

The Text-to-SQL task has long been a key area of research in natural language processing with an emphasis mainly on the English language. Efforts like Spider [15] have enabled significant progress in the accurate translation of natural language questions to working SQL queries. This success has led to models achieving high degrees of execution accuracy and syntactic validity in monolingual English settings.

Extending this progress to multilingual contexts, however, presents a new challenge. Early approaches to multilingual Text-to-SQL heavily relied upon translation-driven frameworks, where queries written in languages other than English were first translated into English before the parsing process. While this process might seem straightforward, it is extremely prone to semantic imprecision and misalignment during the process of translation, often generating errors or uneven results. Evaluation on multilingual datasets shows performance drops for non-English language scenarios compared to English, supporting the need for stronger multilingual approaches.

In an effort to respond to these challenges, several approaches have been proposed. Domain-specific knowledge injection techniques [9] and chain-of-thought prompting methods [13] have shown promise in improving Text-to-SQL performance. Such approaches use schema-aware prompts and provide few-shot examples created directly for multilingual datasets. Though these methods support cross-lingual generalization to an extent, they remain inherently bounded by their fixed properties. Without a mechanism that supports learning from model execution or correcting errors, prompt-reliant methods fail to dynamically adjust, often falling short of ensuring semantic accuracy even while attaining syntactic accuracy.

Our proposed method aims to address these limitations by employing reinforcement learning (RL) supplemented by semantic feedback. Rather than relying solely on manually designed prompts, our framework allows models to learn from explicit reward signals that include not only execution outcomes but also create a closer semantic mapping to user goals. This allows for creating an adaptive feedback loop that can steer the model towards producing more accurate and generally applicable SQL for multiple languages. To achieve this goal, we leverage the underlying principles of contrastive learning, which has seen widespread use in multilingual learning of representations as well as zero-shot cross-lingual transfer [3], [8]. In particular, we adapt methods from proven approaches like InfoXLM [3] and LaBSE [8] to build a multilingual contrastive encoder from XLM-RoBERTa [5].

In contrast to past uses of reinforcement learning in semantic parsing using methods like REINFORCE [14] and PPO [10], these methods often suffer from instability when used in combination with large language models. In an attempt to overcome this limitation, Group Relative Policy Optimization (GRPO) [11] has been proposed as an improved and stable option using KL-regularized updates to increase training stability and policy consistency. Originally developed for mathematical reasoning tasks, GRPO provides a more memory-efficient alternative to PPO while maintaining training stability. In this study, we build upon the principles of GRPO by combining it with our contrastive semantic reward, enabling the model to achieve optimal execution accuracy and semantic consistency in an efficient and stable manner.

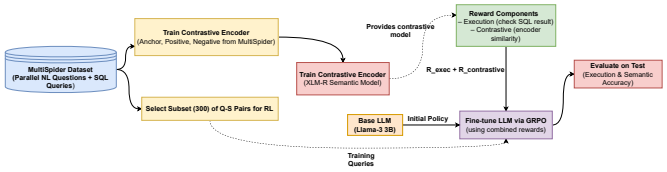

Figure 1. End-to-End Training and Evaluation Pipeline. The process begins with the MultiSpider dataset with parallel NL questions in 7 languages and corresponding SQL. A multilingual contrastive encoder is trained to align semantically equivalent questions across languages. This encoder provides a contrastive semantic reward signal during policy fine-tuning. The Generalized Reward Policy Optimization (GRPO) fine-tuning uses: (1) an execution reward (R_{exec}) giving a binary signal if the SQL yields correct results, and (2) a contrastive reward ($R_{\text{contrastive}}$) based on cosine similarity between question embeddings. These rewards update the LLM policy to produce SQL that is both executable and semantically faithful to the user's intent.

In summary, while it is interesting to see that large foundation models like GPT-3 [2], PaLM [4], and LLaMA [12] exhibit proficiency across a variety of tasks in zero-shot settings, their robustness can be limited in multilingual Text-to-SQL settings. Our study shows that a relatively smaller model, LLaMA-3B, when fine-tuned using semantically aware reinforcement learning, matches or outperforms larger zero-shot counterparts like LLaMA-8B. This highlights both the importance of semantic rewards and the effectiveness of our method in facilitating smaller models to attain a high level of accuracy in multilingual semantic parsing.

III. Methodology

A. Approach Overview

In our approach we fine-tuned a pre-trained Llama-3 3B model with reinforcement learning, using a novel contrastive reward that encourages consistent meaning across languages. Figure 1 illustrates our complete framework, including the novel contrastive reward mechanism that complements traditional execution and syntax rewards.

The key insight of our method is that for multilingual Text-to-SQL, it's just not enough to check the execution completeness of the query, but we also need to ensure that the query is semantically aligned with the user question/input and the results actually match the gold SQL results.

B. Generalized Reward Policy Optimization (GRPO)

We employ Group Relative Policy Optimization (GRPO) [11], a model-free reinforcement-learning algorithm designed for language-model fine-tuning. Originally developed for mathematical reasoning tasks in the DeepSeekMath framework, GRPO follows a policy-gradient approach, but its update

rule is crafted to be more stable and memory-efficient than traditional PPO on large language models. At each step the policy is moved only a small distance from a fixed reference policy (the frozen pre-trained model), and an adaptive KL-divergence penalty automatically scales to keep distribution shifts in check. GRPO also evaluates rewards directly on generated samples, so it avoids the noise introduced by training a separate reward model and can incorporate arbitrary reward functions. This flexibility lets us combine execution and multilingual semantic rewards within a single, stable optimization framework. The GRPO objective is designed to achieve maximum expected reward while maintaining proximity to the reference policy as shown below.

$$\max_{\theta} \mathbb{E}_{x\sim\mathcal{D}}[\mathbb{E}_{y\sim\pi_\theta(y|x)}[R(x,y)]] - \beta D_{\text{KL}}(\pi_\theta||\pi_{\text{ref}})$$

where x represents input (NL query + schema), y is the generated SQL, R is our combined reward function, and β controls the KL penalty strength.

C. Contrastive Reward Design

Rather than simply using execution success or failure, we supply the model with a continuous semantic reward during training. We compute the cosine similarity between the meaning embedded into the input query and the meaning embedded into the gold-standard English query. This similarity (–1 to 1) is then used as a reward: the higher when the model's interpretation is closer to the actual intention.

Naturally, this reward doesn't explicitly rely on the SQL output; it's a directional cue that informs the model "Yes, you answered the question correctly" or "No, you may be off the mark". Coupling this with execution-based rewards, the model now receives feedback on both what to understand (question meaning) and what to do (act correctly).

D. Contrastive Encoder Architecture and Training

1) Architecture: Our reward encoder E_C is built on XLM-RoBERTa-base [5], a 12-layer transformer pre-trained on text from 100 languages. We enhance this with a 2-layer projection head (hidden dim 256 + ReLU) to produce 256-dimensional unit embeddings. Following the approach in BERT [6], we use the CLS token's final representation as the sentence embedding, which is then passed through the projection head and L2-normalized.

2) Training Data: We use the MultiSpider dataset [7] to construct training triples. Two questions, an English question $Q^{(en)}$ and a Japanese question $Q^{(ja)}$, are regarded as a positive pair if they have the same SQL. For a negative, we use a question of another SQL (typically one that might lexically or structurally confuse the encoder).

3) Loss Function: We train E_C with a triplet margin loss (margin 0.5), so that an anchor question and its translation (positive) are closer than any anchor and an unrelated question (negative) by at least the margin, as defined in Equation (1):

$$\mathcal{L}_{\text{triplet}} = \max(0, d(a,p) - d(a,n) + \text{margin}) \tag{1}$$

where d is the cosine distance, a is the anchor, p is the positive example, and n is the negative example.

E. Reward Integration Mechanism

At each training step the actor receives four feedback signals. The execution reward R_{exec} is set to 1.0 when the generated query exactly matches the reference answer and 0 otherwise. The syntax reward R_{syntax} is again 1.0 if the query can be executed without error. To encourage correct use of the database schema, we add a schema-matching reward R_{schema}, which grows when the query selects the right tables and columns. Finally, a semantic reward R_{sem} is computed as the cosine similarity produced by our multilingual contrastive encoder.

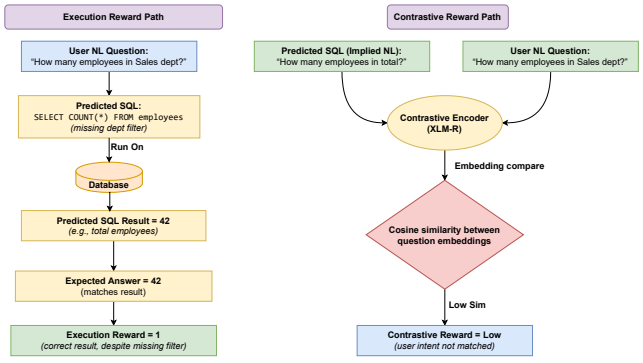

Figure 2. Comparison of Execution Reward vs. Semantic (Contrastive) Reward. While execution rewards only verify if SQL queries run correctly on a database (binary signal), our contrastive rewards provide nuanced feedback about semantic alignment between user intent and generated SQL across languages, capturing subtle differences that execution signals miss. This is particularly important for multilingual queries where translation can introduce semantic ambiguities that binary execution signals cannot address.

Figure 2 illustrates the key difference between traditional execution-based rewards and our contrastive semantic rewards. While execution rewards provide only a binary signal based on query correctness, contrastive rewards offer continuous feedback about semantic alignment, capturing subtle but important distinctions in meaning across languages.

In the current study we place full weight on R_{exec} and treat R_{sem} as a secondary signal that nudges the model toward better cross-lingual meaning preservation. An interesting direction for future work is to explore alternative weighting strategies—for example, increasing the influence of the semantic term—to learn whether a different balance among these rewards can further improve multilingual performance.

F. Training Process

1) Dataset and Preprocessing: We use MultiSpider [7], which provides parallel queries across seven languages (Vietnamese, Spanish, Japanese, German, English, Chinese, and French). We use 3,000 training examples and evaluate on 50 examples per language from the dev set.

2) Model Configuration: We use the Llama-3 3B model with parameter-efficient fine-tuning via LoRA adapters (rank=16) targeting the attention layers. This approach significantly reduces memory requirements while maintaining performance.

3) Training Loop: Algorithm 1 outlines our complete training procedure. The model generates SQL candidates for each input, receives combined rewards, and updates its parameters via GRPO while maintaining KL constraints. The reward design integrates multiple complementary signals. The syntax reward ensures that generated SQL queries are well-formed by granting credit if they can be parsed and executed without errors. The schema-match reward measures whether the query correctly references the intended schema elements, rewarding valid table and column usage while penalizing irrelevant or nonexistent references. The execution reward captures semantic fidelity by comparing the execution results of the candidate query with those of the ground-truth query, assigning the highest score when the outputs match. Together, these rewards guide the model toward producing syntactically valid, schema-aware, and semantically faithful SQL.

Algorithm 1 Contrastive GRPO for Multilingual Text-to-SQL

Require: Base LLM π_{ref}, contrastive encoder E_C, dataset $\mathcal{D}$

Initialize policy $\pi_\theta \leftarrow \pi_{\text{ref}}$ with LoRA adapters
for iteration $= 1$ to max_iterations **do**
  Sample batch $(Q_L, S, D) \sim \mathcal{D}$ (query, gold SQL, schema)
  Generate SQL candidates $\hat{S} \sim \pi_\theta(\cdot|Q_L, D)$
  Compute rewards:
    $R_{\text{exec}} \leftarrow \text{ExecutionReward}(\hat{S}, D)$
    $R_{\text{syntax}} \leftarrow \text{SyntaxReward}(\hat{S})$
    $R_{\text{schema}} \leftarrow \text{SchemaMatchReward}(\hat{S}, D)$
    $R_{\text{sem}} \leftarrow \text{CosineSimilarity}(E_C(Q_L), E_C(Q_{\text{ref}}))$
    $R_{\text{total}} \leftarrow \text{CombineRewards}(R_{\text{exec}}, R_{\text{syntax}}, R_{\text{schema}}, R_{\text{sem}})$
  Update π_θ using GRPO with R_{total} and KL penalty to π_{ref}
end for
return Fine-tuned policy π_θ

4) Evaluation Metrics: We assess performance using two complementary metrics:

- **Execution Accuracy (ExecAcc)**: Percentage of generated SQL queries that execute without errors.
- **Semantic Accuracy (SemAcc)**: Percentage of generated SQL queries that are semantically equivalent to the gold standard, determined by comparing query results across multiple database states.

5) Baselines and Models: We compare the following configurations:

- **L3B-ZS**: Zero-shot Llama-3 3B model
- **L8B-ZS**: Zero-shot Llama-3 8B model
- **L3B-GRPO-NC**: Llama-3 3B fine-tuned with GRPO (no contrastive reward)
- **L3B-GRPO-C**: Our full approach with contrastive reward

6) Detailed Experimental Setup: To ensure reproducibility of our results, we provide comprehensive details about our training infrastructure and hyperparameters:

a) Infrastructure:: Training was conducted on a single NVIDIA A100 GPU (40GB), with XLM-RoBERTa encoder training taking approximately 1 hour and GRPO fine-tuning taking 8-10 hours for 3000 training steps.

b) XLM-RoBERTa Encoder Training:: The contrastive encoder was trained for 2 epochs with a batch size of 96, learning rate of 2e-5, weight decay of 0.01, dropout of 0.1 in the projection head, and a warmup of 500 steps, using the AdamW optimizer. The triplet margin loss used a margin value of 0.5.

c) GRPO Fine-tuning:: For the policy fine-tuning phase, we used LoRA adaptation with rank=16 on the query, key, value matrices and output projection matrices of the self-attention layers. GRPO fine-tuning ran for 10 epochs over 3000 training steps, with a learning rate of 5e-6, batch size of 16, warmup of 500 steps, gradient clipping at 1.0, and a KL penalty coefficient $\beta = 0.02$. The reward weighting combined execution, syntax, schema, and semantic rewards with weights 1.0, 0.5, 0.5, and 0.2 respectively. These values were chosen heuristically to prioritize execution accuracy, while still encouraging syntactic validity, schema alignment, and semantic similarity. The work presented in this paper did not conduct an exhaustive hyperparameter search for optimal values; instead, the selected weights provided stable training across development runs. A more systematic study of adaptive or learned weight assignment is left for future work.

This methodology combines parameter efficient fine-tuning of a Llama-3 3B language model with our contrastive reward mechanism to improve cross-lingual semantic understanding. In the next section, we present the results of our experimental evaluation.

IV. Results and Discussion

We evaluate our approach on the MultiSpider development set across seven languages. Table I and Figure 3 present the core findings.

Baseline Performance: The Llama-3 3B model (L3B-ZS) achieved an average ExecAcc of 61.43% and SemAcc of 19.43% in zero shot setting. The Llama-3 8B model (L8B-ZS) achieves an average ExecAcc of 81.43% and SemAcc of 68.57% with better performance over L3B-ZS as expected.

Impact of GRPO Fine-tuning (No Contrastive Reward): The performance of the finetuned Llama-3B (L3B-GRPO-NC) with vanilla GRPO fine-tuning achieved an average ExecAcc of 87.43% and SemAcc of 52.29%.

Table I
TEXT-TO-SQL PERFORMANCE ON MULTISPIDER (DEV SET). SCORES ARE EXECACC / SEMACC (%). L3B-ZS = LLAMA-3B ZERO-SHOT; L8B-ZS = LLAMA-8B ZERO-SHOT; L3B-GRPO-NC = L3B + GRPO (NO CONTRASTIVE); L3B-GRPO-C = L3B + GRPO (WITH CONTRASTIVE - OURS).

Language	**L3B-ZS**		**L8B-ZS**		**L3B-GRPO-NC**		**L3B-GRPO-C (Ours)**	
	Exec	Sem	Exec	Sem	Exec	Sem	Exec	Sem
Vietnamese (vi)	62.0	24.0	82.0	70.0	90.0	50.0	90.0	60.0
Spanish (es)	54.0	18.0	78.0	66.0	88.0	52.0	88.0	60.0
Japanese (ja)	66.0	16.0	82.0	70.0	88.0	50.0	88.0	58.0
German (de)	58.0	18.0	82.0	72.0	88.0	56.0	90.0	60.0
English (en)	66.0	20.0	86.0	70.0	86.0	54.0	88.0	56.0
Chinese (zh)	62.0	16.0	78.0	64.0	88.0	56.0	90.0	64.0
French (fr)	62.0	24.0	82.0	68.0	84.0	48.0	88.0	56.0
Average	**61.43**	**19.43**	**81.43**	**68.57**	**87.43**	**52.29**	**88.86**	**59.14**

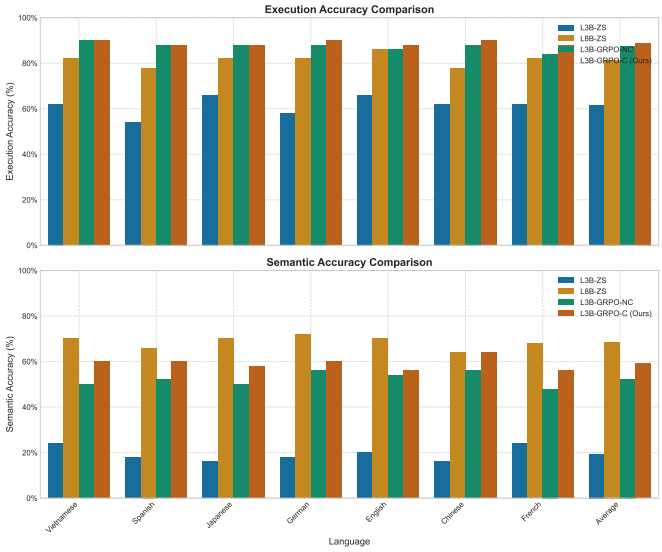

Figure 3. Execution Accuracy (top) and Semantic Accuracy (bottom) across different models and languages. Our L3B-GRPO-C model (dark blue) consistently outperforms the baseline L3B model and approaches or exceeds the larger L8B model, particularly in Execution Accuracy.

Efficacy of Contrastive Rewards (Our Full Model): Including the multilingual contrastive reward (L3B-GRPO-C) in the GRPO training loop achieved an average ExecAcc of 88.86% and SemAcc of 59.14% showcasing the efficacy of our approach. This results shows that contrastive reward is effective in improving semantic understanding of the model especially in the Vietnamese language where we see a +10pp improvement in semantic accuracy.

Overall Improvement and Efficiency: Our final model (L3B-GRPO-C) shows a total improvement of +27.43pp in ExecAcc and +39.71pp in SemAcc over the L3B-ZS baseline. This substantial gain is achieved using only 3000 GRPO training examples, demonstrating the sample efficiency of our approach.

Comparison with Llama-8B Zero-Shot: Remarkably, our fine-tuned Llama-3 3B model (L3B-GRPO-C) outperforms the zero-shot Llama-3 8B model in average ExecAcc (88.86% vs. 81.43%, a +7.43pp difference). While the L8B-ZS showed a better average SemAcc (68.57% vs. 59.14%), our L3B-GRPO-C model significantly closes this gap, reducing the L8B-ZS advantage to 9.43pp. This indicates that our targeted fine-tuning approach using contrastive rewards can enables smaller models to achieve highly competitive, and in some cases better performance compared to larger models in a zero-shot setting, offering a more resource-efficient path to high performance.

A. Qualitative Analysis

To illustrate how our contrastive reward mechanism corrects semantic errors that execution accuracy alone would miss, we present a representative example from our Vietnamese experiments shown in Figure 4:

Example: Vietnamese Query Translation

Original NLQ (Vietnamese):
Hiển thị tên của tất cả các diễn viên đã tham gia vào nhiều hơn 3 bộ phim.

English Translation:
Show the names of all actors who have participated in more than 3 films.

L3B-GRPO-NC (No Contrastive):
```
SELECT actor.name FROM actor JOIN casting ON
actor.id = casting.actorid GROUP BY actor.id
HAVING COUNT(*) >= 3;
```

L3B-GRPO-C (With Contrastive):
```
SELECT actor.name FROM actor JOIN casting ON
actor.id = casting.actorid GROUP BY actor.id,
actor.name HAVING COUNT(DISTINCT cast-
ing.movieid) > 3;
```

Gold SQL:
```
SELECT actor.name FROM actor JOIN casting ON
actor.id = casting.actorid GROUP BY actor.id,
actor.name HAVING COUNT(DISTINCT cast-
ing.movieid) > 3;
```

Figure 4. Example query where contrastive reward improves semantic accuracy while maintaining execution accuracy. The model without contrastive rewards uses $\geq$ instead of $>$ as specified in the question and doesn't count distinct movies.

In this example, both models generate SQL queries that execute successfully and happen to return the same result set on the test database state (passing execution accuracy). However, the query from the model without contrastive rewards (L3B-GRPO-NC) has two semantic issues: (1) it uses >= instead of > as specified in the question, and (2) it counts all casting entries rather than distinct movies, which could give incorrect results if an actor appears multiple times in the same film. The contrastive reward guides the model (L3B-GRPO-C) toward capturing these semantic nuances correctly, producing SQL that precisely matches the user's intent. This example demonstrates how execution accuracy alone can be misleading and the role of contrastive reward in capturing subtle semantic nuances.

V. Ablation Studies

A. Impact of Contrastive Rewards

Table II isolates the contribution of the contrastive reward component by comparing L3B-GRPO-C with L3B-GRPO-NC. The data clearly shows that the contrastive reward consistently improves Semantic Accuracy across all seven languages, with an average gain of +6.85pp. The most significant gains are seen in Vietnamese (+10pp), Spanish (+8pp), Japanese (+8pp), Chinese (+8pp), and French (+8pp), demonstrating the reward's broad utility in enhancing semantic fidelity.

Figure 5 illustrates the language-specific improvements achieved by our contrastive reward mechanism, clearly showing the substantial semantic accuracy gains across all seven languages, with particularly notable improvements for Vietnamese and other non-English languages that typically challenge large language models.

Table II
Ablation: Semantic Accuracy (%) Improvement from Contrastive Rewards (L3B-GRPO-C vs. L3B-GRPO-NC).

Language	L3B-GRPO-NC	L3B-GRPO-C	Δ SemAcc
French (fr)	48.0	56.0	+8.0
Vietnamese (vi)	50.0	60.0	+10.0
Chinese (zh)	56.0	64.0	+8.0
Japanese (ja)	50.0	58.0	+8.0
English (en)	54.0	56.0	+2.0
Spanish (es)	52.0	60.0	+8.0
German (de)	56.0	60.0	+4.0
Average	**52.29**	**59.14**	**+6.85**

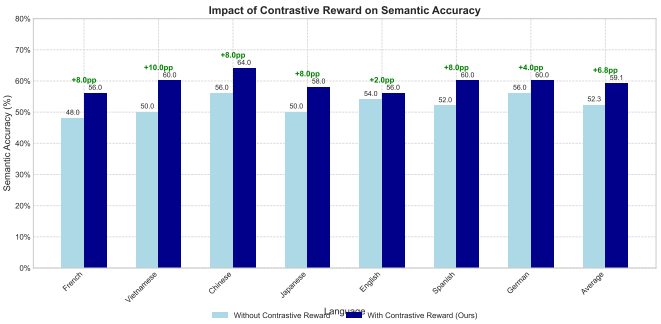

Figure 5. Visual comparison of semantic accuracy with and without contrastive rewards across languages. Green annotations show the percentage point improvements achieved with our contrastive reward approach.

B. Encoder Choice for Contrastive Reward

To validate the choice of XLM-RoBERTa-base as our primary multilingual contrastive encoder, we conducted an ablation study where it was replaced by a smaller, less potent multilingual model, MiniLM-L6-v2 [**?**]. The MiniLM model was trained using the same contrastive learning setup. Our initial experiments during encoder development showed that XLM-RoBERTa achieved significantly higher cross-lingual semantic similarity scores on held-out NLQ pairs compared to MiniLM. Specifically, XLM-RoBERTa attained approximately 0.9 average cosine similarity for positive pairs, while MiniLM only reached about 0.5.

When this MiniLM-based contrastive reward was integrated into the GRPO fine-tuning of Llama-3 3B, the resulting improvement in average Semantic Accuracy (over the L3B-GRPO-NC baseline) was only +1.5pp. This is substantially lower than the +6.85pp average SemAcc gain achieved when using the XLM-RoBERTa-based contrastive reward. This disparity underscores the importance of a strong multilingual encoder capable of capturing nuanced cross-lingual semantic equivalences to provide an effective reward signal. XLM-RoBERTa's extensive multilingual pretraining and larger capacity appear crucial for this task.

VI. Conclusion

This research provided a novel concept for improving multilingual Text-to-SQL systems by merging a multilingual contrastive reward signal with Generalized Reward Policy

Optimization (GRPO). Our method leverages a fine-tuned XLM-RoBERTa encoder to provide dense semantic similarity rewards, along with a Llama-3 3B model to build SQL queries that are both executable and semantically associated with user intent across seven different languages from the MultiSpider dataset. The efficiency of this method is shown by various experimentations. Substantial improvements above zero-shot performance are possible with standard GRPO fine-tuning alone, especially in execution accuracy. The model's semantic knowledge is further improved by integrating our contrastive reward system, with particularly evident improvements for languages that are typically difficult for large language models to recognize. Our method takes 3000 training samples to outperform larger models in prominent metrics. This effectiveness makes it possible to use equivalent approaches in even more languages and fields with limited comparable data.

ACKNOWLEDGMENT

In preparing this manuscript, we used grammar correction and rephrasing tools to improve clarity and readability of certain technical sections. All intellectual contributions, experimental design, results, and conclusions remain our original work.

REFERENCES

[1] K. Ahuja, C. Alberti, K. Lee, A. Bapna, A. Siddhant, L. Martin, Y. Wang, J. Michael, O. Firat, M. Krikun, "MegaVerse: Scaling up multilingual language models," in *Proc. 2023 Conf. Empirical Methods in Natural Language Processing*, pp. 4532–4544, 2023.

[2] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, "Language Models are Few-Shot Learners," *Adv. Neural Inf. Process. Syst.*, vol. 33, pp. 1877–1901, 2020.

[3] Z. Chi, L. Dong, F. Wei, W. Wang, X.-L. Mao, H. Huang, "InfoXLM: An Information-Theoretic Framework for Cross-Lingual Language Model Pre-Training," in *Proc. 2021 Conf. North American Chapter Assoc. Comput. Linguistics: Human Language Technologies*, pp. 3576–3588, 2021.

[4] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, I. Misra, K. M. Ganchev, S. Meier-Hellstern, D. Mielke, Y. Ni, A. Riquelme, A. Zoph, T. Fedus, W. Wang, D. Koner, K. Anil, Y. Cui, M. Dai, M. D. Tran, R. Hutchins, A. Mayes, J. Rajani, S. Singh, X. Tan, J. Yu, Z. Zheng, A. Kumar, H. S. Ahmed, B. Alaa, M. Chen, S. Chen, S. D. Chi, J. Clark, M. Coenen, A. Elibol, S. Ghosh, N. Goldie, L. Jernite, L. Johnson, S. Khan, M. Khare, C. Krikun, P. Lazaridou, H. Lee, M. Lester, K. Li, X. Li, H. Liu, P. M. Dai, N. Mishra, A. M. Rabe, D. Ram, A. Ramesh, D. G. Ritchie, N. Roberts, K. Shamsian, J. Slama, K. Smith, A. Sorokin, H. Wang, K. Wang, R. Wang, J. Wei, Z. Wu, Y. Xue, S. Zha, J. Zhang, C. Zhou, "PaLM: Scaling Language Modeling with Pathways," *arXiv preprint arXiv:2204.02311*, 2022.

[5] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, "Unsupervised Cross-Lingual Representation Learning at Scale," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, pp. 8440–8451, 2020.

[6] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding," *arXiv preprint arXiv:1810.04805*, 2019.

[7] L. Dou, Y. Gao, M. Pan, D. Wang, W. Che, D. Zhan, J.-G. Lou, "MultiSpider: Towards Benchmarking Multilingual Text-to-SQL Semantic Parsing," in *Proc. AAAI Conf. Artificial Intelligence*, vol. 37, no. 11, pp. 12636–12644, 2023.

[8] F. Feng, Y. Yang, D. Cer, N. Arivazhagan, W. Wang, "Language-Agnostic BERT Sentence Embedding," in *Proc. 60th Annu. Meeting Assoc. Comput. Linguistics (Vol. 1: Long Papers)*, pp. 878–891, 2022.

[9] H. Chang, S. Joty, X. Wang, "Enhancing Text-to-SQL Capabilities of Large Language Models via Domain Database Knowledge Injection," in *Proc. 2024 Conf. Language Resources and Evaluation*, pp. 6234–6244, 2024.

[10] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, O. Klimov, "Proximal Policy Optimization Algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[11] Z. Shao, P. Wang, Q. Zhu, R. Xu, J. Song, X. Bi, H. Zhang, M. Zhang, Y. K. Li, Y. Wu, D. Guo, "DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models," *arXiv preprint arXiv:2402.03300*, 2024.

[12] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, R. Biligiri, D. Black, G. Chen, D. D'Oliveira, M. E. Sajjad, R. Goyal, T. Goyal, R. Hesslow, D. Kharitonov, A. Khetan, M. Lhoest, S. Malik, Y. Miao, G. Muennighoff, T. Mu, M. Poulton, M. Scales, C. Stock, R. Subramanian, T. Wang, S. Xie, M. Yazdanbakhsh, A. Zettlemoyer, L. Zha, D. Lavril, G. Leclerc, S. W. Walker, "Llama 2: Open Foundation and Fine-Tuned Chat Models," *arXiv preprint arXiv:2307.09288*, 2023.

[13] P. Pourreza, D. Rafiei, "Exploring Chain of Thought Style Prompting for Text-to-SQL," in *Proc. 2023 Conf. Empirical Methods in Natural Language Processing*, pp. 5264–5277, 2023.

[14] R. J. Williams, "Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning," *Machine Learning*, vol. 8, no. 3, pp. 229–256, 1992.

[15] T. Yu, R. Zhang, K. Yang, M. Yasunaga, D. Wang, Z. Li, J. Ma, I. Li, Q. Yao, S. Roman, D. S. Zhang, C. Xiong, R. Socher, "Spider: A Large-Scale Human-Labeled Dataset for Complex and Cross-Domain Semantic Parsing and Text-to-SQL Task," in *Proc. 2018 Conf. Empirical Methods in Natural Language Processing*, pp. 3911–3921, 2018.