# REWRITE WHAT MATTERS: ADAPTIVE MULTILINGUAL QUERY REWRITING FOR REASONING VIA AGENTIC REINFORCEMENT LEARNING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

In multilingual scenarios, queries with equivalent semantics but in different languages could guide the model into different reasoning trajectories, leading to performance disparities. To mitigate this gap, previous studies typically apply a one-size-fits-all query rewriting strategy, such as translation, which overlooks the fact that different scenarios require diverse types of semantic transformations. In this paper, we propose mRewriter-R1, an agentic multilingual query rewriting framework with reinforcement learning. Unlike single-turn rewriting, mRewriter-R1 formulates multilingual query rewriting as a multi-turn sequential decision-making process, where the model dynamically performs multi-aspect optimization through adaptive operator selection. Experimental results demonstrate that mRewriter-R1 outperforms all strong multilingual rewriting baselines on different large reasoning backbones. Further analyses show that the learned policy can adaptively decide on rewriting operators according to query characteristics, exhibiting strong generalization ability across diverse reasoning tasks, and plug-and-play compatibility with heterogeneous reasoning language models[1].

## 1 INTRODUCTION

Large Language Models (LLMs) have demonstrated strong capabilities on reasoning. However, their performance has been shown to vary significantly across languages (Huang et al., 2025; Shi et al., 2023). This disparity is caused not only by discrepancies in various language resources but also by a high sensitivity to query understanding. In multilingual settings, queries that are semantically equivalent but differ in syntactic structure and expression across languages could guide the model into different reasoning trajectories, thereby generating inconsistent results (Qi et al., 2025; Huang et al., 2024). Therefore, adequately understanding multilingual queries via essential rewriting can align them with the correct reasoning trajectories so as to leverage the intrinsic reasoning capabilities of LLMs to improve performance for downstream tasks.

Existing multilingual query rewriting (MQR) methods primarily rely on empirical manners by applying a one-size-fits-all rewriting strategy to all queries, such as full-sentence translation or word-level translation (Kang et al., 2025; Lu et al., 2024; Yoo et al., 2025). Such strategies cannot consistently achieve satisfactory performance for all queries in diverse scenarios and requirements. As illustrated in Figure 1, prior methods are one-step for a single-aspect, which performs rewriting only once and fails to consider other features that may affect reasoning, such as syntactic structure (Li et al., 2024). For instance, translation helps to understand entities in some queries written in low-resource languages but hurts others that instead only require rewriting the expressions. Thus, an ideal approach is to adopt the appropriate MQR strategies at the query-level and a critical research question is *whether it is possible for LLMs to dynamically learn to employ multiple MQR strategies in an adaptive way.*

To address this issue, the query needs to interact with LLMs continuously, leveraging the model's reasoning process for the query at the current state to determine the most appropriate query rewriting strategy. A promising approach is to employ agentic reinforcement learning (RL) to achieve

[1]Our code is available at `https://anonymous.4open.science/r/mRewriter-R1-F684`



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

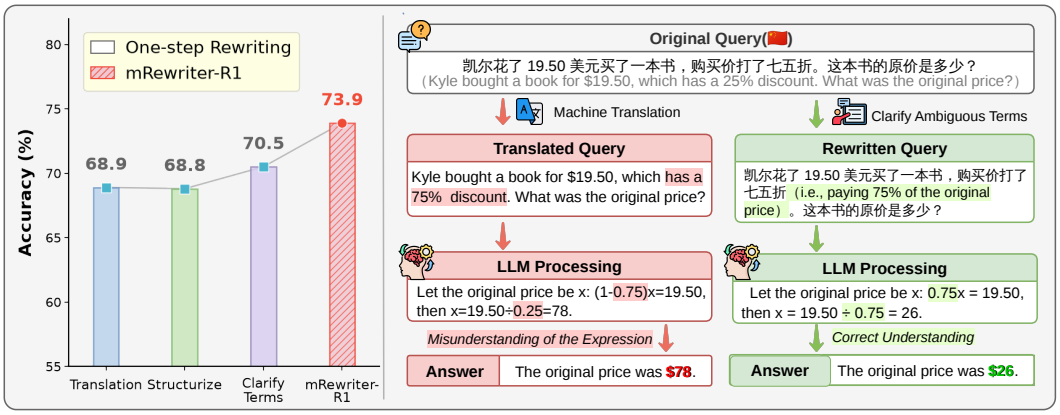



Figure 1: Illustration of the impact of query variations for multilingual reasoning.

autonomous query rewriting in dynamic environments (Jin et al., 2025). However, unlike the well-defined decision of retrieval, MQR under agentic RL involves a much more open-ended action space, making stable RL substantially more challenging. It also remains unclear how to effectively integrate MQR into the agentic workflow with RL in rewriting-and-reasoning scenarios.

In this paper, we propose mRewriter-R1, an agentic query rewriting framework with RL to solve multilingual reasoning tasks. Instead of relying on static and single-turn approaches, we implement a sequential decision-making paradigm where the LLM acts as both a planner and a rewriter. At each iteration, mRewriter-R1 first generates a plan based on previous turns and then performs query rewriting accordingly. Through multiple iterations, it can learn to autonomously plan the rewriting strategy and rewrite the query, improving the accuracy of the models for the multilingual reasoning.

Considering the three essential aspects of a query (*i.e., how to express, how to organize, and what is truly intended*) (Faustini et al., 2023), a single-aspect rewriting strategy is insufficient to simultaneously address the corresponding challenges: **eliminating expression discrepancies across languages**, **constructing well-structured logical forms**, and **understanding the underlying intent**. To alleviate this, we design a multi-aspect query rewriting mechanism driven by specialized operators in a planning space. Specifically, the MQR space is categorized into three aspects: (1) Language aspect, which aims to align queries with high-resource languages via translation-based methods; (2) Structural aspect, which aims to transform complex natural language into well-structured logical expressions; and (3) Semantic aspect, which disambiguates the input and adapts it into a representation that maximizes the utilization of LLMs. Within each aspect, various multilingual rewriting strategies are encapsulated into discrete operators. This formulation constrains the inherently open-ended rewriting decision space to a finite and structured action space, thereby facilitating stable and tractable agentic RL. Subsequently, the model learns to compositionally invoke these operators, achieving combinatorial query optimization tailored to each input. Experimental results on three multilingual benchmarks demonstrate that our mRewriter-R1 framework can improve the performance of multilingual reasoning for various LLMs, consistently outperforming the static application of any single operator across all aspects.

*Our contributions are three-fold:* **(1)** We propose a multi-turn Agentic RL framework that reformulates MQR into a sequential decision-making process across multiple aspects, breaking away from empirical rewriting manners. **(2)** The query rewriter trained by our framework mRewriter-R1 can be deployed as a plug-and-play module for the other LLMs, which is easily adapted with various backbone models and compatible with prompt-based methods such as Chain-of-Thought (CoT). **(3)** Experiments on different types of multilingual reasoning benchmarks demonstrate that mRewriter-R1 achieves better multilingual reasoning performance compared to other baseline methods.

## 2 METHODS

As shown in Figure 2, we present the formal framework of mRewriter-R1, an adaptive method designed to systematically rewrite multilingual queries across language, structural, and semantic

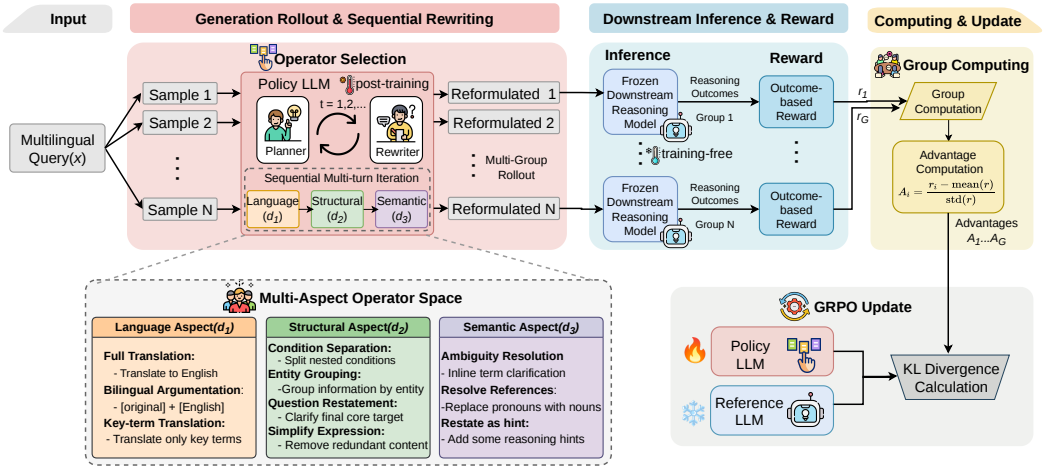



Figure 2: The workflow of mRewriter-R1, where the policy LLM is trainable while the reference LLM is frozen. The action follows a multi-turn iteration in a sequential decision-making process with multi-aspect operator space.

aspects. The framework follows a multi-turn decision-making manner and is optimized via outcome-based rewards to dynamically employ rewriting operators across multiple aspects.

### 2.1 Agentic Workflow for Multi-Turn Decision-Making

We formulate MQR as a multi-turn decision-making process, where a single policy model $\pi_\theta$ plays two roles at each step. The model first acts as a planner to diagnose the current query and determine how it should be improved, and then acts as a rewriter to produce the updated query. Unlike conventional translation-based preprocessing, our framework allows the model to dynamically determine how the query should be rewritten through multi-aspect rollouts rather than relying on a single static transformation, optimizing multilingual queries in an adaptive and interpretable manner. Formally, this action is serialized with tags:

$$p_t = (d_t, o_t), \quad o_t \in \mathcal{O}_{d_t},$$

where $o_t$ is the selected operator from the operator set $\mathcal{O}_{d_t}$ associated with aspect $d_t$.

Followed by the action of operator selection, the policy model performs both planning and rewriting in a unified generation process, and the transition of the query from current state to the next is defined by extracting the rewritten query from the generated action:

$$a_t \sim \pi_\theta(\cdot \mid x_t, d_t), \quad x_{t+1} = \mathcal{R}(a_t),$$

where $x_t$ denotes the current state of queries, and $\mathcal{R}$ denotes the rewriting operator that transforms the query into the next state.

The final rewritten query $\tilde{x}$ is generated before the downstream reasoning process starts, and is fed into a frozen downstream reasoning model after all rewriting steps are completed:

$$\hat{y} = M(\tilde{x}), \quad \tilde{x} = x_{T+1},$$

where $M$ denotes the downstream LLMs.

### 2.2 Multi-Aspect Rewriting Operators

As observed in the previous investigation, the failures in multilingual reasoning do not originate from a single factor. In many cases, reasoning errors are not caused by language mismatch alone, but also by implicit structural relations and ambiguous semantic expressions (Qi et al., 2025). Therefore, instead of treating rewriting as a one-step translation problem, we formulate it as a multi-aspect optimization process. However, allowing arbitrary rewriting within a single action space increases the complexity of the optimization process and makes learning more challenging. To alleviate this,

we organize the action space into aspect-specific operator sets $\mathcal{O}_{d_t}$. At each step, given the current optimization aspect $d_t$, the planner chooses an operator, and then chooses an operator $o_t \in \mathcal{O}_{d_t}$. The theoretical derivation details of the Constrained Operator are in Appendix D. Formally, the rewriting policy can be viewed as a hierarchical decision process:

$$\pi_\theta(a_t \mid x_t, d_t) = \pi_\theta(o_t \mid x_t, d_t) \cdot \pi_\theta(x_{t+1} \mid x_t, d_t, o_t).$$

The first stage determines *how* the query should be modified, while the second stage generates the rewritten query conditioned on the selected operator. By decomposing reformulation into aspect-specific operator spaces, our framework transforms query rewriting from unconstrained text generation into structured sequential optimization. This not only improves interpretability, but also stabilizes RL by reducing the complexity of the action space.

The rewriting aspects are divided into three levels and executed sequentially $\{d_1, d_2, d_3\} = \{\texttt{language}, \texttt{structural}, \texttt{semantics}\}$. This sequential processing manner is designed with the consideration that cross-lingual alignment provides the foundational linguistic space for subsequent structural and semantic optimization, while semantic grounding is most effective after the query structure has been clarified. The specific rewrite spaces of the three aspects are as follows.

**Language-Aspect.** The first aspect focuses on cross-lingual alignment. Instead of enforcing full translation for all queries, the model adaptively determines whether the original language should be preserved, translated, or partially rewritten. Representative operators include full translation, bilingual augmentation, and key-term translation. This aspect aims to reduce language distribution mismatch while preserving language-specific information that may benefit reasoning.

**Structural-Aspect.** The second aspect focuses on improving the structural organization of the query. Some multilingual reasoning failures arise from important relationships being expressed implicitly or scattered across the query. Therefore, structural-aspect operators reorganize information into reasoning-friendly forms, such as explicitly separating conditions, grouping information by entities, or restating the target question. These operators improve the compositional clarity of the query without changing its semantics.

**Semantic-Aspect.** The final aspect focuses on semantic grounding and ambiguity resolution. Even after language and structure are improved, multilingual queries may still contain underspecified references, culturally dependent expressions, or implicit computational targets. Semantic-aspect operators therefore clarify ambiguous references, normalize expressions, and convert implicit reasoning requirements into explicit executable forms while preserving the original intent.

### 2.3 Training Objectives and Protocol

We train the rewriting policy model $\pi_\theta$ using the Group Relative Policy Optimization (GRPO) (Shao et al., 2024), where the objective is not to optimize the downstream model, but to optimize the query rewriting process itself. Specifically, the policy learns to generate rewritten queries that maximize the final reasoning performance of a frozen downstream model using outcome-based reward.

**Outcome-Based Reward.** Following recent RL frameworks (Jin et al., 2025), we adopt a simple yet effective outcome-based reward without process supervision. The rewriting policy is rewarded solely according to the final reasoning performance of the downstream model. This design encourages the rewriting policy to discover query transformations that improve downstream reasoning performance, even without explicit supervision on intermediate trajectories. The reward is defined as:

$$r(x, \tilde{x}) = \mathrm{Acc}(M(\tilde{x}), y^*),$$

where $y^*$ denotes the ground-truth answer and $\mathrm{Acc}(\cdot)$ denotes accuracy after answer normalization.

**Policy Optimization.** For each query, we sample a group of rewriting trajectories $\{\tau_1, \tau_2, \ldots, \tau_G\}$. The relative advantage and the clipped objective of the policy are computed as:

$$A_i = \frac{r_i - \mathrm{mean}(\{r_j\}_{j=1}^G)}{\mathrm{std}(\{r_j\}_{j=1}^G)}.$$

$$\mathcal{L}(\theta) = \mathbb{E}\left[\frac{1}{G}\sum_{i=1}^{G} \min\Big(\rho_i(\theta)A_i,\ \mathrm{clip}\big(\rho_i(\theta), 1-\epsilon, 1+\epsilon\big)A_i\Big) - \beta D_{\mathrm{KL}}(\pi_\theta \| \pi_{\mathrm{ref}})\right],$$

| MGSM | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Method | En | Bn | De | Es | Fr | Ja | Ru | Sw | Th | Zh | Te | Avg. |
| Direct | 87.2 | 60.0 | 73.2 | 77.2 | 74.8 | 64.8 | 72.0 | 18.0 | 76.0 | 76.0 | 37.6 | 65.2 |
| Translate | 86.0 | 63.2 | 78.4 | 84.0 | 77.2 | 75.2 | 84.4 | 20.8 | 74.0 | 79.2 | 33.2 | 68.7 |
| Selective | 88.0 | 59.6 | 74.8 | 80.8 | 74.8 | 67.2 | 72.0 | 20.8 | 76.4 | 76.8 | 37.2 | 66.2 |
| DIP | 85.2 | 65.4 | 72.8 | 78.8 | 74.6 | 70.2 | 78.6 | 24.0 | 74.2 | 76.6 | 34.2 | 66.8 |
| CSICL | 91.2 | 62.0 | 78.4 | 83.6 | 76.8 | 70.4 | 83.2 | 20.8 | 77.6 | 82.4 | **40.4** | 69.7 |
| *mRewriter-R1* | **92.8**† | **74.4**† | **83.2**† | **84.8**† | **84.4**† | **78.8**† | **86.8**† | **24.8**† | **79.2**† | **84.0**† | 40.0 | **73.9**† |
| **XCOPA** | | | | | | | | | | | | |
| Method | Et | Ht | Id | It | Qu | Sw | Ta | Th | Tr | Vi | Zh | Avg. |
| Direct | 60.4 | 60.2 | 90.4 | 91.4 | 52.8 | 55.2 | 59.6 | 81.0 | 82.8 | 84.8 | 90.8 | 73.6 |
| Translate | 68.4 | 65.8 | 93.2 | 92.8 | 50.0 | 53.6 | 63.8 | 82.6 | 85.8 | 89.2 | 90.2 | 75.9 |
| Selective | 64.0 | 60.2 | 90.8 | 92.2 | 53.2 | 56.0 | 60.8 | 82.8 | 83.6 | 85.8 | **91.4** | 74.6 |
| DIP | 64.6 | 59.4 | 84.8 | 90.2 | 49.8 | 52.0 | 62.0 | 78.6 | 80.2 | 82.4 | 88.0 | 72.0 |
| CSICL | 60.6 | 60.0 | 85.6 | 90.0 | 53.2 | **56.8** | 56.8 | 77.8 | 79.8 | 83.2 | 87.8 | 72.0 |
| *mRewriter-R1* | **70.6**† | **66.6**† | **93.8**† | **93.8**† | **54.6**† | **56.8** | **64.2**† | **84.6**† | **86.6**† | **90.6**† | **91.4**† | **77.6**† |
| **BELEBELE** | | | | | | | | | | | | |
| Method | Ar | Zh | En | Fr | De | Hi | Ja | Ru | Sw | Th | | Avg. |
| Direct | 78.8 | 88.0 | **91.6** | 83.6 | 83.2 | 60.8 | 80.0 | 83.2 | 35.6 | 71.2 | | 75.6 |
| Translate | 79.2 | 87.6 | 90.0 | 88.0 | 84.0 | 52.8 | 80.4 | **89.6** | 36.0 | 66.0 | | 75.4 |
| Selective | 78.8 | 86.8 | 91.2 | 84.0 | 82.4 | 58.4 | 78.0 | 84.4 | 37.2 | 70.4 | | 75.2 |
| DIP | 81.6 | 82.8 | 90.6 | 86.2 | 82.8 | 56.0 | 80.6 | 83.6 | 36.8 | 70.4 | | 75.1 |
| CSICL | 80.0 | 88.4 | 90.8 | 88.0 | 84.8 | 62.4 | 82.4 | 84.4 | 38.8 | 75.6 | | 77.6 |
| *mRewriter-R1* | **86.4**† | **90.0**† | 91.2† | **90.0** | **87.8**† | **65.6**† | **84.8**† | 87.4† | **40.2**† | **78.2**† | | **80.2**† |

Table 1: Accuracy (%) on three benchmarks across different languages based on Qwen2.5-7B-Instruct. † denotes significant improvements with McNemar test at $p < 0.05$ between our methods with the second best method. The best results in each column within the same benchmark are **bold**, and the second-best is underlined.

where $\rho_i(\theta)$ denotes the policy ratio and $\pi_{\text{ref}}$ denotes the reference policy.

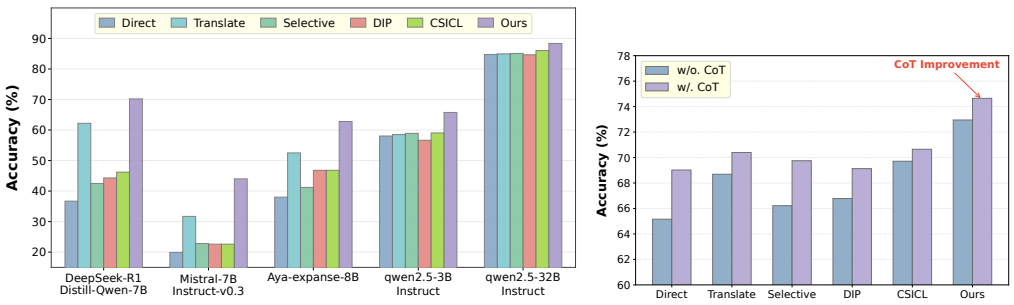
**Prompting and Structured Generation.** To stabilize policy learning and enforce interpretable reformulation behaviors, we adopt a structured prompting format during training. The prompt explicitly specifies the current optimization aspect, the allowed operator set, and the required output structure. Importantly, the model is instructed to optimize the query without directly solving the problem. We additionally constrain the reformulation process to preserve all original facts, numbers, and constraints, preventing the policy from introducing hallucinated information during optimization. The full prompting templates and implementation details are provided in Appendix G.

# 3 Experiments

## 3.1 Experimental Setup

**Models.** We conduct the experiments using Qwen2.5-7B-Instruct (Yang et al., 2024) as both the rewriting policy model and the downstream reasoning model. To further evaluate the plug-and-play property of mRewriter-R1, we additionally use the rewriting model together with five downstream models: Qwen2.5-3B/32B-Instruct , DeepSeek-R1-Distill-Qwen-7B (Guo et al., 2025), Mistral-7B-Instruct-v0.3 (Jiang et al., 2024) and Aya-expanse-8B (Dang et al., 2024). In these experiments, the rewriting policy remains unchanged while only the downstream reasoning backbone is replaced.

**Benchmarks and Evaluation.** This paper reports the accuracy to assess the ability of the methods for all benchmarks. We evaluate mRewriter-R1 on three multilingual reasoning benchmarks covering different capabilities. **MGSM** (Shi et al., 2023) evaluates multilingual mathematical reasoning using grade-school math word problems translated into multiple typologically languages. **BELEBELE** (Bandarkar et al., 2024) is a massively multilingual machine reading comprehension benchmark spanning 122 language variants, designed to evaluate multilingual understanding and passage-level reasoning. **XCOPA** (Ponti et al., 2020) evaluates cross-lingual causal commonsense




(a) (b)

Figure 3: Performance of plug-and-play and CoT generalization across backbones. Our rewriter is trained based on `Qwen2.5-7B-Instruct`. Detailed results are in Appendix E.

reasoning by requiring models to infer plausible causes or effects across multiple languages. Details are in Appendix A.

**Baselines.** We compare our method against several representative baselines. (1) **Translate** (Ghosh et al., 2025) (2) **Selective** (Kang et al., 2025) detects understanding failures and injects an English translation only for inputs likely to be misunderstood. (3) **DIP** (Lu et al., 2024) inserts bilingual dictionary-based English glosses into the original multilingual query to support English-centric reasoning. (4) **CSICL** (Yoo et al., 2025) uses gradual code-switching demonstrations to guide the model from the target language toward English reasoning.

### 3.2 Main Results

**Overall Performance.** As shown in Table 1, our method consistently outperforms several strong baselines across various multilingual reasoning benchmarks. Notably, existing MQR approaches apply a fixed rewriting strategy to all inputs, their effectiveness varies substantially across languages and instances. For example, translation-based methods improve performance on some languages but simultaneously degrade others, suggesting that no single rewriting strategy universally fits all multilingual queries. In contrast, mRewriter-R1 alleviates these issues by adaptively selecting suitable rewriting strategies for different multilingual inputs, leading to more robust reasoning performance. In addition, we observe similar improvements on XCOPA and BELEBELE, where our method consistently achieves the best overall performance among all compared approaches. These results suggest that the proposed framework generalizes beyond mathematical reasoning to broader multilingual understanding and commonsense reasoning tasks.

**Plug-and-Play Generalization across Backbone Models.** As shown in Figure 3a, to evaluate the generality of the learned rewriting policy, we apply the query rewriter trained by mRewriter-R1 to different downstream backbone models. Experimental results show that our rewriter can function as a plug-and-play module that is easily adapted to off-the-shelf LLMs, improving multilingual reasoning performance across various model families and parameter scales. Across different model scales within the same family, our method substantially enhances multilingual reasoning capabilities in smaller models (3B). More importantly, as the model size increases (32B), our method continues to yield further improvements even when the models already exhibit strong multilingual reasoning capabilities, while other baselines begin to suffer from performance degradation. Moreover, such transferability is particularly important for practical multilingual deployment, where retraining or jointly optimizing large reasoning models is often prohibitively expensive.

**Compatibility with Prompting Strategies.** As shown in Figure 3b, we further investigate whether mRewriter-R1 can cooperate with existing reasoning-augmented prompting methods such as CoT. The results show that our framework is complementary rather than competitive to existing prompting paradigms. While CoT primarily improves reasoning decomposition in the generation stage, mRewriter-R1 focuses on optimizing multilingual query understanding before reasoning begins. Therefore, the two approaches enhance multilingual reasoning from orthogonal perspectives and can be naturally integrated together. It further demonstrates the flexibility of our framework and its potential as a general multilingual reasoning enhancement module.

| Language | Structural | Semantic | Accuracy |
|---|---|---|---|
| ✓ | ✓ | ✓ | **73.93**$_{\text{(ours)}}$ |
| ✗ | ✓ | ✓ | 71.45$_{-2.48}$ |
| ✓ | ✗ | ✓ | 72.11$_{-1.82}$ |
| ✓ | ✓ | ✗ | 73.09$_{-0.84}$ |
| ✓ | ✗ | ✗ | 71.71$_{-2.22}$ |
| ✗ | ✓ | ✗ | 71.09$_{-2.84}$ |
| ✗ | ✗ | ✓ | 70.44$_{-3.49}$ |
| w/o. operators in agentic RL | | | |
| ✗ | ✗ | ✗ | 64.92$_{-9.01}$ |

Table 2: Ablation studies on different aspects. We report the degradation in red to denote the disparity in accuracy compared to our method.

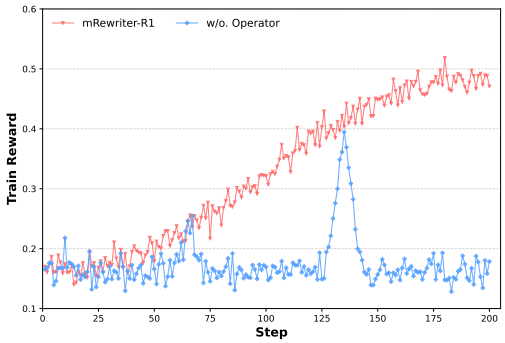



Figure 4: Training rewards for mRewriter-R1 and the agent RL that does not use operators.

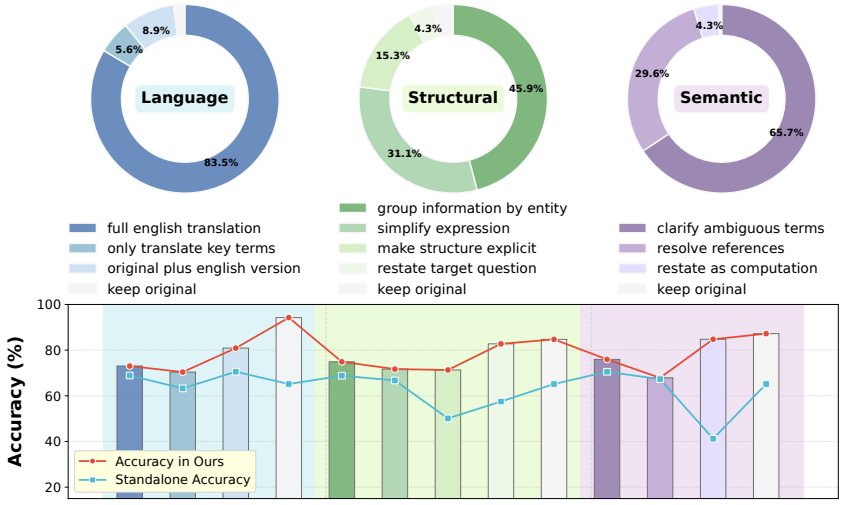



Figure 5: Operator distribution and reasoning performance across operators in each aspect. The donut charts (top) show the usage frequency of rewriting operators selected by the policy LLM. The lower plot compares accuracy between mRewriter-R1 (red line), where operators are adaptively selected conditioned on the query, and standalone one-step rewriting of individual operators (blue line).

### 3.3 Ablation Studies

As shown in Table 2, we conduct ablation studies to investigate the contribution of the operators at each aspect in our framework. The results show that removing any aspect consistently degrades performance, demonstrating that all aspects contribute positively to the overall framework. In particular, removing the language-aspect operator leads to the largest performance drop, indicating that it plays the most critical role in correcting multilingual reasoning failures. In contrast, removing the semantic-aspect operator results in a relatively smaller degradation, implying that this aspect mainly serves as an auxiliary refinement component.

In addition, we further evaluate variants that retain only one aspect at a time. Although each individual aspect outperforms several baseline methods, all single-aspect variants remain significantly worse than the complete framework. The language-aspect achieves the strongest standalone performance, consistent with the previous ablation results. However, combining all aspects yields substantially larger improvements, confirming that the proposed multi-aspect framework enables complementary interactions across different fixing strategies. Notably, we further investigate a setting in which the operator process is removed, allowing the model to freely rewrite queries. This modification causes the rewriting module to become ineffective. As evidenced by Figure 4, the open-ended rewriting leads to training collapse, making the underlying rewriting difficult for RL.

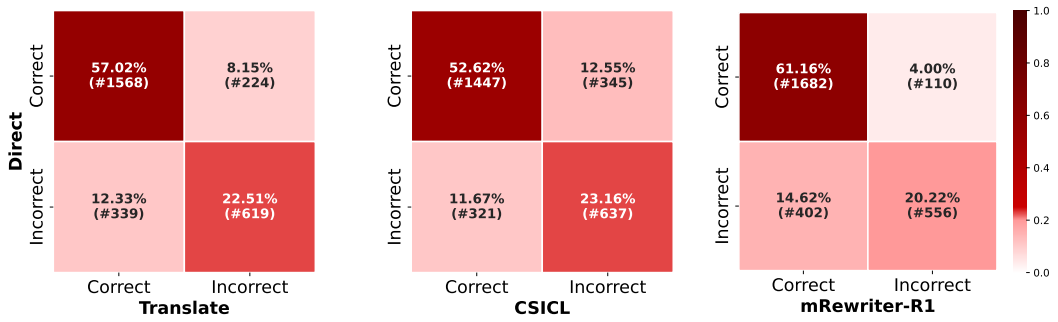



Figure 6: Correctness transitions from Direct to rewriting methods on 2,750 MGSM test instances.

## 4 Analysis of Operator Distribution

To better understand the behavior of the learned rewriting policy, we analyze both the operator selection distribution and the corresponding reasoning performance across different rewriting aspects. Figure 5 presents the usage frequency of operators together with two types of performance comparisons: (1) the downstream reasoning accuracy on queries where an operator is selected by our policy, and (2) the performance obtained when a single operator is uniformly applied to all queries.

**Comparison of Diverse Operators.** First, each rewriting aspect exhibits a dominant operator with substantially higher usage frequency, e.g., *full_english_translation* in the language aspect, indicating that the policy gradually converges toward several broadly reliable rewriting strategies during RL optimization. Second, several low-frequency operators still achieve high performance on their assigned queries, indicating that the policy learns selective and precise operator-selection behaviors rather than uniformly applying aggressive rewriting strategies.

**Comparison with One-Step.** The accuracy on policy-selected queries is consistently higher than the standalone performance of the same operator. These results suggest that rewriting operators are highly context-dependent, and that the key advantage of mRewriter-R1 lies not merely in the operators themselves, but in learning when each operator should be applied.

**Relationship between Distribution and Performance.** A comprehensive analysis between distributional behavior and performance shows that operators with stronger standalone performance generally tend to be selected more frequently. For instance, *clarify_ambiguous_terms* achieves relatively strong standalone performance and is also frequently selected by the policy, indicating that the learned policy can identify operators with broadly transferable utility for multilingual reasoning.

## 5 Discussion

**Query-Level Risk Analysis.** Beyond aggregate accuracy, we perform a paired query-level analysis to examine both the benefits and potential risks introduced by rewriting. As shown in Figure 6, mRewriter-R1 corrects 402 queries that are incorrectly answered by Direct, while degrading 110 initially correct predictions. Compared with the baseline methods, our rewriting strategy minimizes the fraction of queries that are negatively affected, with the gains from successful corrections outweighing the risk of performance degradation. In contrast, the other two methods introduce more than twice as many incorrect corrections as ours. CSICL also struggles to preserve queries (close to 10%) that are already correct and consistent.

**Semantic-Preservation.** To further investigate the explainability of our framework, we provide additional case studies and analyses from MGSM to illustrate the sequential rewriting behavior in Appendix F. The case studies illustrate the actual sequential operator trajectories generated by the learned rewriting policy, while the analyses evaluate the preservation of critical semantic constraints. They show that mRewriter-R1 learns compositional rewriting behaviors across multiple aspects while preserving the semantics of the original queries. We further compute similarity to quantitatively assess semantic preservation. The results show that, among the 2,750 instances, more than 95% of the rewritten queries retain semantics close to those of the original queries, with a cosine similarity

| Metric | Direct | Translate | Selective | DIP | CSICL | mRewriter-R1 |
|---|---|---|---|---|---|---|
| Input Tokens | **172.17** | 189.17 | 174.17 | 213.80 | 174.17 | 193.42 |
| Output Tokens | 507.25 | 512.00 | 512.00 | 439.13 | 241.42 | **217.76** |
| Total Tokens | 679.42 | 701.17 | 686.17 | 652.93 | 415.59 | **411.18** |
| Time (s) | 0.7194 | 0.7291 | 0.7281 | 0.8106 | 0.7233 | **0.5390** |

Table 3: Comparison of token consumption and inference latency per sample across different methods. The values are the average over all samples.

above 0.9. Instead of uniformly applying aggressive rewriting strategies, the policy incrementally restructures multilingual queries through adaptive operator routing, producing representations that are more aligned with downstream models.

**Inference Latency.** Table 3 compares token consumption and inference latency across different methods. Although mRewriter-R1 introduces additional rewriting steps and slightly increases the input token, it produces fewer output tokens than several prompting-based baselines. We hypothesize that the rewritten queries are structurally clearer and semantically less ambiguous, allowing downstream reasoning models to generate more concise reasoning trajectories without unnecessary exploration. In particular, operators such as *simplify_expression* can reduce verbose or redundant descriptions before reasoning begins, which helps improve generation efficiency. As a result, mRewriter-R1 achieves the lowest average inference latency despite performing multi-stage query rewriting. Furthermore, we also provide the non-batched per-query inference latency for each query with mRewriter-R1. The median per-query latency is 4.188 s, close to the mean of 4.655 s, while 90% and 95% of queries are completed within 7.034 s and 8.252 s, respectively. This suggests that the rewriting latency remains relatively stable for most queries, and only a small number of instances exhibit longer tail latency.

## 6 Related Work

Existing studies on multilingual reasoning attempt to influence the reasoning trajectory of language models mainly from three perspectives: (1) query transformation methods, which modify the input query through strategies such as translation or code-switching (Kang et al., 2025; Lu et al., 2024; Yoo et al., 2025); (2) prompt-based guidance methods, which improve multilingual reasoning through prompting strategies such as CoT (Koo & Kim, 2025; Tran et al., 2025; Qi et al., 2025; Kang & Kim, 2025); (3) training-time alignment methods, which enhance cross-lingual reasoning consistency by updating model parameters through multilingual instruction tuning or RL (She et al., 2024; Zhu et al., 2024; Li et al., 2025a; Hwang et al., 2025; Fan et al., 2025). Among these directions, query transformation directly intervenes at the input level to reduce cross-lingual misunderstanding and is also compatible with the other branches (Lu et al., 2024). Existing query transformation methods adopt empirical strategies, applying the same rewriting operation to all multilingual queries (Ghosh et al., 2025). Although Kang et al. (2025) introduce the dynamic mechanism, it relies on intermediate feedback, making it difficult to complete query rewriting before reasoning begins. In contrast, we propose a learnable agentic MQR framework that dynamically plans and applies multi-aspect rewriting operators before reasoning begins. Moreover, it is more challenging than monolingual settings (Li et al., 2025b; Ye et al., 2025; Nguyen et al., 2025; Zhu et al., 2025), since the rewriting space becomes much more diverse when handling a larger number of languages. To address this, we decouple the complex rewriting space into multi-aspect following a sequential decision-making paradigm through multi-turn iterations, ensuring stable and effective optimization in rewriting-and-reasoning manners.

## 7 Conclusion

In this paper, we propose mRewriter-R1, an adaptive MQR framework with RL for multilingual reasoning. By formulating MQR as a multi-turn sequential decision-making process, our method dynamically performs language-, structural-, and semantic-aspect rewriting through adaptive operator selection. Experimental results and further analyses on multilingual reasoning demonstrate that the learned policy can adaptively employ different rewriting operators according to query characteristics, exhibiting competitive performance, strong generalization ability, and plug-and-play compatibility with heterogeneous backbones.

## AI USE STATEMENT

We did not use generative AI tools for any aspect of this work, including research, methodology development, experimentation, and analysis. The authors take full responsibility for the content of this paper.

## REFERENCES

Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabsa. The belebele benchmark: a parallel reading comprehension dataset in 122 language variants. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 749–775, 2024.

Nuo Chen, Zinan Zheng, Ning Wu, Ming Gong, Dongmei Zhang, and Jia Li. Breaking language barriers in multilingual mathematical reasoning: Insights and observations. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 7001–7016, 2024.

John Dang, Shivalika Singh, Daniel D'souza, Arash Ahmadian, Alejandro Salamanca, Madeline Smith, Aidan Peppin, Sungjin Hong, Manoj Govindassamy, Terrence Zhao, et al. Aya expanse: Combining research breakthroughs for a new multilingual frontier. *arXiv preprint arXiv:2412.04261*, 2024.

Yuchun Fan, Yongyu Mu, Yilin Wang, Lei Huang, Junhao Ruan, Bei Li, Tong Xiao, Shujian Huang, Xiaocheng Feng, and Jingbo Zhu. Slam: Towards efficient multilingual reasoning via selective language alignment. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 9499–9515, 2025.

Pedro Faustini, Zhiyu Chen, Besnik Fetahu, Oleg Rokhlenko, and Shervin Malmasi. Answering unanswered questions through semantic reformulations in spoken QA. In Sunayana Sitaram, Beata Beigman Klebanov, and Jason D Williams (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 5: Industry Track)*, pp. 729–743, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-industry.70. URL https://aclanthology.org/2023.acl-industry.70/.

Akash Ghosh, Debayan Datta, Sriparna Saha, and Chirag Agarwal. A survey of multilingual reasoning in language models. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 8920–8936, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.474. URL https://aclanthology.org/2025.findings-emnlp.474/.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Kaiyu Huang, Fengran Mo, Xinyu Zhang, Hongliang Li, You Li, Yuanchi Zhang, Weijian Yi, Yulong Mao, Jinchen Liu, Yuzhuang Xu, Jinan Xu, Jian-Yun Nie, and Yang Liu. A survey on large language models with multilingualism: Recent advances and new frontiers, 2025. URL https://arxiv.org/abs/2405.10936.

Zixian Huang, Wenhao Zhu, Gong Cheng, Lei Li, and Fei Yuan. Mindmerger: efficiently boosting llm reasoning in non-english languages. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA, 2024. Curran Associates Inc. ISBN 9798331314385.

Jaedong Hwang, Kumar Tanmay, Seok-Jin Lee, Ayush Agrawal, Hamid Palangi, Kumar Ayush, Ila Fiete, and Paul Pu Liang. Learn globally, speak locally: Bridging the gaps in multilingual reasoning. *arXiv preprint arXiv:2507.05418*, 2025.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning, 2025. URL https://arxiv.org/abs/2503.09516.

Deokhyung Kang, Seonjeong Hwang, Daehui Kim, Hyounghun Kim, and Gary Geunbae Lee. Why do multilingual reasoning gaps emerge in reasoning language models? *arXiv preprint arXiv:2510.27269*, 2025.

Eojin Kang and Juae Kim. When language shapes thought: Cross-lingual transfer of factual knowledge in question answering. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pp. 4868–4873, 2025.

Hamin Koo and Jaehyung Kim. Extracting and emulsifying cultural explanation to improve multilingual capability of llms. *arXiv e-prints*, pp. arXiv–2503, 2025.

Bryan Li, Tamer Alkhouli, Daniele Bonadiman, Nikolaos Pappas, and Saab Mansour. Eliciting better multilingual structured reasoning from LLMs through code. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5154–5169, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.281. URL https://aclanthology.org/2024.acl-long.281/.

Nianqi Li, Zujie Liang, Siyu Yuan, Jiaqing Liang, Feng Wei, and Yanghua Xiao. Multilingpot: Boosting mathematical reasoning in llms through multilingual program integration. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 19794–19811, 2025a.

Rui Li, Liyang He, Qi Liu, Zheng Zhang, Heng Yu, Yuyang Ye, Linbo Zhu, and Yu Su. Unirag: Unified query understanding method for retrieval augmented generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14163–14178, 2025b.

Hongyuan Lu, Zixuan Li, and Wai Lam. Dictionary insertion prompting for multilingual reasoning on multilingual large language models. *arXiv preprint arXiv:2411.01141*, 2024.

Duy A Nguyen, Rishi Kesav Mohan, Shimeng Yang, Pritom Saha Akash, and Kevin Chen-Chuan Chang. Minielm: A lightweight and adaptive query rewriting framework for e-commerce search optimization. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 6952–6964, 2025.

Edoardo Maria Ponti, Goran Glavaš, Olga Majewska, Qianchu Liu, Ivan Vulić, and Anna Korhonen. XCOPA: A multilingual dataset for causal commonsense reasoning. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2362–2376, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.185. URL https://aclanthology.org/2020.emnlp-main.185/.

Rui Qi, Zhibo Man, Yufeng Chen, Fengran Mo, Jinan Xu, and Kaiyu Huang. Sot: Structured-of-thought prompting guides multilingual reasoning in large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 11024–11039, 2025.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Shuaijie She, Wei Zou, Shujian Huang, Wenhao Zhu, Xiang Liu, Xiang Geng, and Jiajun Chen. Mapo: Advancing multilingual reasoning through multilingual-alignment-as-preference optimization. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10015–10027, 2024.

Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. Language models are multilingual chain-of-thought reasoners. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=fR3wGCk-IXp.

Khanh-Tung Tran, Barry O'Sullivan, and Hoang D Nguyen. Scaling test-time compute for low-resource languages: Multilingual reasoning in llms. *arXiv e-prints*, pp. arXiv–2504, 2025.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

Xiaopeng Ye, Chen Xu, Chaoliang Zhang, Zhaocheng Du, Jun Xu, Gang Wang, and Zhenhua Dong. Q-prm: Adaptive query rewriting for retrieval-augmented generation via step-level process supervision. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 15113–15128, 2025.

Haneul Yoo, Jiho Jin, Kyunghyun Cho, and Alice Oh. Code-switching in-context learning for cross-lingual transfer of large language models. *arXiv preprint arXiv:2510.05678*, 2025.

Changtai Zhu, Siyin Wang, Ruijun Feng, Kai Song, and Xipeng Qiu. Convsearch-r1: Enhancing query reformulation for conversational search with reasoning via reinforcement learning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 26558–26575, 2025.

Wenhao Zhu, Shujian Huang, Fei Yuan, Shuaijie She, Jiajun Chen, and Alexandra Birch. Question translation training for better multilingual reasoning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 8411–8423, 2024.

## A Dataset Statistics

We provide detailed statistics of the datasets used in our experiments in Table 4, with the corresponding language coverage. Our evaluation covers three representative multilingual reasoning benchmarks: MGSM (Shi et al., 2023) for mathematical reasoning, BELEBELE (Bandarkar et al., 2024) for multilingual reading comprehension requires knowledge for reasoning, and XCOPA (Ponti et al., 2020) for commonsense causal reasoning. For MGSM, we construct the training split from the multilingual version of GSM8K (Chen et al., 2024), while keeping the standard MGSM-style evaluation setting with 250 test examples per language. To better reflect different language-resource conditions, we assign variable training sizes across languages, using fewer examples for high-resource languages and more examples for low-resource languages, as low-resource languages benefit more from rewriting and therefore require more supervision for effective RL policy learning. For BELEBELE and XCOPA, we use only a limited number of training examples per language, as summarized in Table 4. Overall, the training sets remain relatively small across all datasets, which supports our goal of evaluating whether the proposed rewriting policy can be learned without relying on large-scale task-specific supervision. To prevent randomness, we report the performance based on the average results of three inference runs.

## B Results of Linguistic Groups

For the typology-aware analysis in Table 5, we group the 11 MGSM languages along three complementary dimensions: word order, writing system, and resource level. Word order describes the dominant ordering of subject, verb, and object, and is categorized into SVO, SOV, and flexible-order groups. Writing system distinguishes languages primarily written in Latin scripts from those using

| Dataset | Training Set | | | Test Set | |
|---|---|---|---|---|---|
| | # Lang | # / Lang | **Total** | # / Lang | **Total** |
| MGSM | En, Bn, De, Es, Fr, Ja, Ru, Sw, Th, Zh (10) | *V** | 10,300 | 250 | 2,500 |
| BELEBELE | Ar, Zh, En, Fr, De, Hi, Ja, Ru, Sw, Th (10) | 650 | 6,500 | 250 | 2,500 |
| XCOPA | Et, Ht, Id, It, Qu, Sw, Ta, Th, Tr, Vi, Zh (11) | 100 | 1,100 | 500 | 5,500 |

***Note:** MGSM uses variable training sizes (*V**) based on language resource level: en(700), bn(1200), de(850), es(800), fr(900), ja(1100), ru(800), sw(2200), th(850), zh(900).

Table 4: Number of training and evaluation instances across datasets and languages used in each dataset.

| Language | Word Order | Writing System | Resource Level |
|---|---|---|---|
| English | SVO | Latin | High |
| Bengali | SOV | Non-Latin | Medium |
| German | Flexible | Latin | High |
| Spanish | SVO | Latin | High |
| French | SVO | Latin | High |
| Japanese | SOV | Non-Latin | High |
| Russian | Flexible | Non-Latin | High |
| Swahili | SVO | Latin | Low |
| Thai | SVO | Non-Latin | Medium |
| Chinese | SVO | Non-Latin | High |
| Telugu | SOV | Non-Latin | Low |

Table 5: Linguistic grouping of the MGSM languages used in the typology-aware analysis.

| | Word Order | | | Writing System | | Resource Level | | |
|---|---|---|---|---|---|---|---|---|
| | **SVO** | **SOV** | **Flexible** | **Latin** | **Non-Latin** | **High** | **Medium** | **Low** |
| Direct | 68.20 | 54.13 | 72.60 | 66.08 | 64.40 | 75.03 | 68.00 | 27.80 |
| mRewriter-R1 | 75.00 | 64.40 | 85.00 | 74.00 | 73.87 | 84.97 | 76.80 | 32.40 |
| $\Delta$ | +6.80 | +10.27 | +12.40 | +7.92 | +9.47 | +9.94 | +8.80 | +4.60 |

Table 6: Performance of mRewriter-R1 across linguistic groups on MGSM. Languages are grouped by word order, writing system, and resource level. $\Delta$ denotes the absolute accuracy improvement over Direct.

non-Latin scripts. Resource level provides a coarse grouping of languages according to their relative resource availability.

As shown in Table 6, mRewriter-R1consistently improves over Direct across all linguistic groups. The gains remain substantial for SOV and flexible-order languages, as well as for both Latin- and non-Latin-script languages, indicating that the benefits of adaptive rewriting are not limited to a specific syntactic structure or writing system.

In terms of resource level, mRewriter-R1 also improves performance across high-, medium-, and low-resource groups. However, the absolute accuracy and improvement on low-resource languages remain lower than those on high- and medium-resource languages. This suggests that query rewriting can better expose and organize the information already captured by the backbone model, but cannot fully compensate for limited multilingual competence in severely underrepresented languages.

| Method | Am | Bn | Ko | Te | Vi | Yo | Avg. |
|---|---|---|---|---|---|---|---|
| Direct | 21.6 | 62.0 | 78.8 | 34.0 | 69.6 | 16.0 | 47.0 |
| CSICL | 30.8 | 36.4 | 66.8 | 25.6 | 78.8 | 30.8 | 44.9 |
| mRewriter-R1 | **34.8** | **66.8** | **84.8** | **36.0** | **82.4** | **31.6** | **56.1** |

Table 7: Performance on six unseen BELEBELE languages without additional training.

| Rewriter | MGSM | XCOPA | BELEBELE | Avg. |
|---|---|---|---|---|
| Direct | 65.2 | 73.6 | 75.6 | 71.5 |
| CSICL | 69.7 | 72.0 | 77.6 | 73.1 |
| mRewriter-R1 | **71.5** | **75.2** | **78.2** | **75.0** |

Table 8: Performance of cross-task transfer.

# C Analysis under Training Constraints

## C.1 Training Data Requirement.

We use only 650 training instances per language for BELEBELE and 100 instances per language for XCOPA. It shows that although training the rewriting policy requires multilingual supervision, our framework can be learned with relatively limited data. To further evaluate data efficiency, we apply the trained rewriter to six unseen BELEBELE languages without any additional training or adaptation. As shown in Table 7, mRewriter-R1 improves the average accuracy from 47.00 to 56.07 and consistently improves over Direct across all six unseen languages. These results indicate that the learned rewriting policy can transfer across languages with limited multilingual supervision.

## C.2 Cross-Task Generalization.

To further examine whether the policy is overly specialized to a particular dataset, we additionally train a cross-task rewriter on the combined training sets without target tasks (e.g., training the mathematical task MGSM on XCOPA and training the causal task on MGSM). The same frozen rewriter is then evaluated on the two benchmarks without task-specific adaptation. The results show that our rewriter consistently improves over Direct and CSICL. This indicates that the learned rewriting behaviors are not tied to a single dataset and can be shared across tasks.

# D Constrained Operator Space for Stable Agentic RL

A key challenge in applying agentic RL to MQR is the inherently open-ended nature of the rewriting space. If the policy is allowed to directly generate an arbitrary rewritten query at each step, the action space grows with the vocabulary and sequence length, making the optimization substantially more difficult than conventional discrete decision-making. More importantly, semantically equivalent queries may correspond to highly diverse surface forms, resulting in a large number of distinct actions that yield similar downstream rewards. This redundancy increases the difficulty of credit assignment and introduces high variance into policy optimization.

To address this issue, we constrain the rewriting space with a finite set of semantically motivated rewriting operators. Specifically, we decompose MQR into three aspects, namely language, structural, and semantic rewriting. Each aspect ($d \in \mathcal{D}$) is associated with a finite operator set ($\mathcal{O}_d$), where each operator corresponds to a predefined class of rewriting behaviors. The overall operator space is therefore given by $\mathcal{O}| = \sum_{d\in\mathcal{D}} |\mathcal{O}_d|$. At each iteration (t), the policy first identifies the current rewriting aspect ($d_t$), and then selects an operator ($o_t \in \mathcal{O}_{d_t}$). The action is represented as

$$p_t = (d_t, o_t), \qquad o_t \in \mathcal{O}_{d_t}.$$

Conditioned on the selected operator, the policy generates the corresponding rewritten query ($x_{t+1}$). Thus, instead of directly learning a mapping from the current query to an unconstrained textual output $x_t \to x_{t+1}$, we reformulate rewriting as a structured decision process:

$$x_t \to d_t \to o_t \to x_{t+1}.$$

This decomposition separates what aspect should be improved from how that aspect should be rewritten, while retaining the flexibility of language generation within each operator. The resulting

policy can be factorized as

$$\pi_\theta(a_t \mid x_t) = \pi_\theta(d_t \mid x_t)\pi_\theta(o_t \mid x_t, d_t)\pi_\theta(x_{t+1} \mid x_t, d_t, o_t).$$

Equivalently, under a fixed aspect ($d_t$, in this work, the rewriting aspects are divided into three levels and executed sequentially $\{d_1, d_2, d_3\} = \{\texttt{language}, \texttt{structural}, \texttt{semantics}\}$), the operator-selection space is reduced from an unconstrained textual generation space ($\mathcal{X}$) to a finite set ($\mathcal{O}_{d_t}$). This converts the high-dimensional rewriting problem into a structured and compositional decision process.

Consider an unconstrained rewriting policy that directly generates a sequence of ($L$) tokens from a vocabulary of size ($V$). Even ignoring syntactic and semantic constraints, the number of possible rewriting actions is on the order of $|\mathcal{A}\text{open}| \approx \sum_{l=1}^{L} V^l$, which grows exponentially with the output length. Suppose that an operator ($o$) corresponds to multiple textual realizations:

$$\mathcal{A}(o) = a_1, a_2, \ldots, a_K.$$

When directly optimizing the policy in the open-ended action space,

$$\nabla_\theta J = \mathbb{E}_a \left[ A(a) \nabla_\theta \log \pi_\theta(a|x) \right],$$

different ($a_i$) may correspond to the same rewriting intent while exhibiting substantially different reward variance: $\mathrm{Var}[R(a) \mid o] \gg 0$. This redundancy introduces noise into the policy gradient and makes credit assignment more difficult.

In contrast, our operator-constrained policy first selects from a finite action space with $\sum_{d \in \mathcal{D}} |\mathcal{O}_d|$. The actual textual realization is subsequently conditioned on the selected operator. Therefore, the policy does not need to explore the entire space of possible textual rewrites when learning which rewriting behavior is beneficial.

This reduction is particularly important for outcome-based RL, where the reward is primarily determined by the final downstream reasoning result. In the unconstrained setting, many distinct textual trajectories can lead to the same reward, making the reward signal sparse with respect to the individual rewriting decisions. Let ($a$) denote a textual rewriting action and ($o$) its corresponding rewriting operator. Since many actions may belong to the same operator class,

$$a_1, a_2, \ldots, a_K \in \mathcal{A}(o),$$

the operator representation aggregates multiple semantically related actions into a common decision. The policy therefore receives a more structured learning signal at the operator level:

$$Q(x, o) = \mathbb{E}_{a \sim \pi_\theta(\cdot|x,o)}[R(a)].$$

This aggregation reduces the effective decision granularity and facilitates credit assignment across rewriting trajectories $\mathrm{Var}[Q(x, o)] \leq \mathrm{Var}[R(a)]$. More formally, the policy-gradient objective can be written as

$$\nabla_\theta J = \mathbb{E}\tau \left[ \sum_{t=1}^{T} A_t \nabla_\theta \log \pi_\theta(d_t, o_t \mid x_t) \right], \qquad \pi_\theta(d_t, o_t \mid x_t) = \pi_\theta(d_t \mid x_t)\pi_\theta(o_t \mid x_t, d_t).$$

where ($A_t$) denotes the advantage associated with the rewriting decision. Under our hierarchical factorization,

$$\log \pi_\theta(d_t \mid x_t) + \log \pi_\theta(o_t \mid x_t, d_t) + \log \pi_\theta(x_{t+1} \mid x_t, d_t, o_t).$$

The first two terms operate over finite categorical distributions rather than an unrestricted textual action space. Consequently, the high-level policy learns which rewriting behavior to invoke, while the conditional generation module focuses on realizing that behavior. This separation makes the

optimization more structured and reduces the burden of learning both the rewriting strategy and its textual realization simultaneously. Importantly, restricting the action space does not imply restricting the final rewriting space to a small set of fixed transformations. Since operators can be sequentially composed, the effective rewriting space remains combinatorial. For a trajectory of $(T)$ rewriting steps, the number of possible operator sequences is $|\mathcal{O}|^T$, while each operator can still generate diverse textual realizations. Hence, the proposed formulation provides a favorable trade-off between optimization tractability and rewriting expressiveness. The final rewriting trajectory can be represented as

$$x_0 \xrightarrow{(d_1,o_1)} x_1 \xrightarrow{(d_2,o_2)} \cdots \xrightarrow{(d_T,o_T)} x_T.$$

The policy therefore learns not only which operator to apply, but also when to invoke each operator and how to compose multiple operators for a particular query. This enables input-adaptive query optimization while avoiding direct exploration over the full open-ended rewriting space.

Overall, the operator-constrained formulation transforms multilingual query rewriting from an unconstrained sequence-generation problem into a structured sequential decision-making problem with a finite and compositional action space. This design provides a controllable interface between query rewriting and agentic RL, facilitating more stable policy optimization while preserving the flexibility required for diverse multilingual queries.

## E Full Results of Generalization Experiments

Table 9 presents the full multilingual evaluation results across different downstream backbone models, corresponding to the averaged results shown in Figure 3a in the main paper. We report per-language performance on MGSM to provide a more fine-grained comparison across different model families and rewriting methods. The results further show that the improvements brought by mRewriter-R1 remain consistent across different backbone models, demonstrating its plug-and-play generalization capability without backbone-specific retraining.

Table 10 presents the full multilingual evaluation results under the CoT prompting setting, corresponding to the averaged results shown in Figure 3b in the main paper. We report per-language performance on MGSM to provide a more fine-grained comparison across different prompting and rewriting methods. The results further show that the improvements brought by mRewriter-R1 remain consistent across most languages when combined with CoT prompting.

## F Case Study

**Adaptive Rewriting Trajectories.** Figure 7 and Figure 8 present two representative multilingual reasoning examples to illustrate how mRewriter-R1 performs multi-aspect query rewriting through sequential operator selection.

In the first case, the original Chinese math problem contains multiple entity relationships and potentially ambiguous references. The policy first applies *full_english_translation* to align the query with the dominant reasoning language space, then uses *group_information_by_entity* to explicitly separate relational constraints into structured statements. Finally, the semantic-aspect operator *resolve_references* further identifies and corrects the inconsistent entity reference by replacing repeated mentions with explicit names. This sequential rewriting process transforms the original densely coupled description into a clearer reasoning-friendly representation.

In the second case, the policy follows a different rewriting trajectory. After language-aspect translation, the structural operator *simplify_expression* compresses verbose narrative descriptions into more concise reasoning steps while preserving all numerical constraints. The semantic-aspect rewriting then resolves implicit references such as “she” and “each” into explicit entities. Compared with the first example, this case demonstrates that the learned policy does not rely on a fixed rewriting template, but adaptively selects different operator combinations according to the characteristics of each query.

| Method | En | Bn | De | Es | Fr | Ja | Ru | Sw | Th | Zh | Te | Avg. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | `DeepSeek-R1-Distill-Qwen-7B` | | | | | | |
| Direct | 34.0 | 10.8 | 50.8 | 58.4 | 52.0 | 49.6 | 56.0 | 3.2 | 15.6 | 67.6 | 6.0 | 36.7 |
| Translate | **87.6** | 58.8 | 74.8 | 75.6 | 72.0 | 66.0 | 74.4 | 4.4 | 63.2 | 76.4 | 32.0 | 62.3 |
| Selective | 49.6 | 22.0 | 55.6 | 63.6 | 57.6 | 52.8 | 58.8 | 4.4 | 25.6 | 67.2 | 11.6 | 42.6 |
| DIP | 66.0 | 27.8 | 50.2 | 57.2 | 49.8 | 52.4 | 55.6 | 6.2 | 35.8 | 66.6 | 20.8 | 44.4 |
| CSICL | 67.2 | 38.4 | 52.0 | 58.0 | 50.8 | 52.8 | 57.2 | 6.4 | 36.0 | 68.0 | 21.6 | 46.2 |
| *mRewriter-R1* | 79.2 | **72.8**† | **80.8**† | **81.6**† | **78.4**† | **75.6**† | **83.6**† | **19.6**† | **77.2**† | **82.4**† | **41.6**† | **70.3**† |
| | | | | | | `Mistral-7B-Instruct-v0.3` | | | | | | |
| Direct | 48.4 | 8.8 | 31.2 | 31.2 | 30.8 | 9.2 | 26.4 | 6.4 | 8.8 | 14.8 | 4.4 | 20.0 |
| Translate | 53.6 | 28.4 | **47.2** | 42.8 | 38.8 | 22.8 | 41.2 | 8.4 | 28.4 | 31.6 | 6.0 | 31.7 |
| Selective | 50.4 | 13.2 | 31.6 | 36.4 | 32.8 | 13.6 | 28.0 | 6.8 | 14.8 | 19.6 | 4.4 | 22.9 |
| DIP | 49.8 | 18.4 | 34.8 | 38.0 | 26.2 | 11.2 | 30.6 | 10.8 | 16.8 | 20.2 | 5.0 | 23.8 |
| CSICL | 50.4 | 16.0 | 32.8 | 37.2 | 27.6 | 10.0 | 30.8 | 8.4 | 14.8 | 16.8 | 5.6 | 22.8 |
| *mRewriter-R1* | **56.8**† | **44.0**† | 45.6 | **49.6**† | **51.2**† | **43.2**† | **53.6**† | **12.0** | **50.8**† | **48.0**† | **30.4**† | **44.1**† |
| | | | | | | `Aya-expanse-8b` | | | | | | |
| Direct | 58.0 | 14.8 | 56.0 | 66.0 | 59.6 | 37.2 | 58.8 | 2.0 | 8.8 | 53.2 | 3.6 | 38.0 |
| Translate | 75.6 | 40.0 | 67.2 | **72.8** | 64.4 | **65.2** | 71.2 | 9.6 | 34.4 | 66.0 | 12.0 | 52.6 |
| Selective | 63.6 | 21.2 | 59.6 | 67.2 | 61.2 | 44.0 | 61.6 | 3.6 | 13.2 | 54.0 | 5.2 | 41.3 |
| DIP | 74.8 | 23.6 | 60.2 | 66.8 | 60.8 | 55.0 | 66.6 | 5.0 | 18.2 | 63.8 | 8.8 | 45.8 |
| CSICL | 76.0 | 24.4 | 66.4 | 67.2 | 62.4 | 55.6 | 68.4 | 5.2 | 16.4 | 64.8 | 9.2 | 46.9 |
| *mRewriter-R1* | **77.2**† | **65.2**† | **70.8**† | 71.2 | **75.2** | 61.6 | **75.2**† | **18.4**† | **67.2**† | **70.4**† | **39.6**† | **62.9**† |
| | | | | | | `Qwen2.5-3B-Instruct` | | | | | | |
| Direct | 79.6 | 51.2 | 66.4 | 71.6 | 72.8 | 60.8 | 74.4 | 10.8 | 62.8 | 70.0 | 18.0 | 58.0 |
| Translate | 78.4 | 53.2 | 70.8 | 70.8 | 72.4 | 61.6 | 70.4 | 10.8 | 63.6 | 69.2 | 22.4 | 58.5 |
| Selective | 79.6 | 52.0 | 73.2 | 71.6 | 72.4 | 61.6 | 75.6 | 11.2 | 63.6 | 68.8 | 18.4 | 58.9 |
| DIP | 76.4 | 52.8 | 68.6 | 70.6 | 69.8 | 56.6 | 72.4 | 10.0 | 60.2 | 68.6 | 17.2 | 56.7 |
| CSICL | 79.2 | 52.4 | 72.0 | 74.8 | 74.4 | 56.8 | 72.8 | 10.4 | 69.2 | 70.0 | 17.6 | 59.1 |
| *mRewriter-R1* | **81.2**† | **66.2**† | **74.4**† | **78.4**† | **78.2**† | **66.8**† | **78.2**† | **19.8**† | **70.6**† | **71.2** | **38.4**† | **65.8**† |
| | | | | | | `Qwen2.5-32B-Instruct` | | | | | | |
| Direct | 92.8 | 86.8 | 89.2 | 90.4 | 86.8 | 84.8 | 90.8 | 67.2 | 89.6 | 86.8 | 67.2 | 84.8 |
| Translate | 93.4 | 86.2 | 88.4 | 89.6 | 86.8 | 84.6 | 91.6 | 69.2 | 89.8 | 88.4 | 66.4 | 84.9 |
| Selective | 91.6 | 87.2 | 90.2 | 91.6 | 85.4 | 86.2 | 89.8 | 68.6 | 88.2 | 88.4 | 68.8 | 85.1 |
| DIP | 94.2 | 85.4 | 88.4 | 89.8 | 87.4 | 85.8 | 90.6 | 65.4 | 90.2 | 87.8 | 66.2 | 84.7 |
| CSICL | 94.8 | 87.2 | 90.0 | 90.8 | 88.0 | 86.0 | 91.2 | 68.8 | 90.0 | 88.4 | 71.2 | 86.0 |
| *mRewriter-R1* | **95.8**† | **88.4**† | **91.8**† | **92.4**† | **89.2**† | **88.4**† | **93.8**† | **75.2**† | **90.4** | **89.6**† | **71.6** | **87.9**† |

Table 9: Accuracy (%) on MGSM using various backbones. † denotes significant improvements with McNemar test at $p < 0.05$ between our methods with the second best method. The best result in each column is shown in **bold**, and the second-best is underlined.

| Method | En | Bn | De | Es | Fr | Ja | Ru | Sw | Th | Zh | Te | Avg. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Direct | 86.4 | 66.0 | 77.2 | 80.0 | 76.8 | 74.8 | 80.8 | 22.4 | 76.4 | 81.2 | 37.2 | 69.0 |
| Translate | 84.0 | 68.0 | 77.2 | 80.4 | 82.0 | 76.4 | 84.8 | 24.4 | 82.4 | 78.0 | 36.8 | 70.4 |
| Selective | 86.4 | 65.6 | 78.0 | 81.2 | 78.0 | 74.8 | 82.0 | 24.0 | 76.8 | 81.6 | 38.8 | 69.8 |
| DIP | 87.8 | 65.2 | 76.0 | 80.2 | 77.8 | 73.4 | 80.8 | 24.6 | 77.4 | 79.6 | 37.6 | 69.1 |
| CSICL | 89.6 | 66.0 | 80.0 | 84.0 | 80.4 | 76.4 | 84.4 | 21.2 | 76.0 | 81.2 | 38.0 | 70.7 |
| *mRewriter-R1* | **92.8**† | **74.0**† | **86.8**† | **86.4**† | **83.2**† | **77.2**† | **86.4**† | **26.8**† | **83.6**† | **83.6**† | **40.4**† | **74.7**† |

Table 10: Accuracy (%) on MGSM using `Qwen2.5-7B-Instruct` under the CoT prompting setting. † denotes significant improvements with McNemar test at $p < 0.05$ between our methods with the second best method. The best result in each column is shown in **bold**, and the second-best is underlined.

**Semantic-Preservation.** We further evaluate whether the rewriting process preserves critical semantic constraints on MGSM. Specifically, we compare each original query with its rewritten query in terms of numerical values, currency markers, and percentage constraints. The results are summarized in Table 11. These results show that mRewriter-R1preserves most critical surface constraints, although rewriting is not perfectly lossless.

We also provide two representative examples to qualitatively examine semantic preservation beyond surface-level token matching. As shown in Table 12, Case 3 verifies the preservation of numerical relations and arithmetic constraints, while Case 4 examines temporal conditions and relational

| Metric | Setting | Value |
|---|---|---|
| Critical-number token recall | How many of the key numbers from the original problem can be found after rewriting it? | 93.28% |
| Queries with no numeric loss | How many samples showed no digital loss? | 89.07% |
| Currency marker preservation | If the original problem contains currency information, is the currency constraint retained after rewriting? | 97.42% |
| Percentage marker preservation | When the original problem involves percentages, should the percentage constraint be retained after rewriting? | 99.71% |

Table 11: Semantic-preservation analysis on the MGSM test set.

constraints. In both cases, the entities, key values, constraints, and original question intent are retained after rewriting.

**Overall.** Overall, these examples show that mRewriter-R1 not only performs adaptive multi-aspect rewriting, but also preserves the critical semantics of the original query in most cases. Instead of uniformly applying aggressive rewriting strategies, the policy incrementally restructures multilingual queries through adaptive operator routing, producing representations that are more aligned with downstream reasoning models while maintaining the key semantic constraints of the original query in most cases.

## G Prompt Template

Table 13 and 14 show the prompt templates used for the query rewriter and downstream reasoning models. The rewriter performs three sequential rewriting stages targeting language-aspect, structural-aspect, and semantic-aspect, respectively. The rewritten query is then fed into task-specific downstream prompts for MGSM, XCOPA, and BELEBELE, enabling a unified query optimization framework across different multilingual reasoning tasks.

**Case 1: Entity Correction and Reference Disambiguation**

**Question:** 艾米莉有4 个孩子，名叫艾米、杰克逊、科里和詹姆斯。艾米比杰克逊大5 岁，比科里小2 岁。如果詹姆斯10 岁，并且比科里小1 岁，杰克逊几岁？

**Round 1**

<think>The query is in Chinese and includes numerical information. A full English translation is needed to facilitate downstream reasoning.</think>

<plan>**dimension:** language **operation:** full_english_translation</plan>

<rewrite>Emily has 4 children named Emily, Jackson, Cory, and James. Emily is 5 years older than Jackson, and 2 years younger than Cory. If James is 10 years old and 1 year younger than Cory, how old is Jackson? </rewrite>

↓ ***Structural reformulation: group information by entity.***

**Round 2**

<think>The current query needs to be reorganized to make the relationships and conditions clearer.</think>

<plan>**dimension:** structural_reformulation **operation:** group_information_by_entity</plan>

<rewrite>Emily has 4 children named Emily, Jackson, Cory, and James.

- Emily is 5 years older than Jackson.
- Emily is 2 years younger than Cory.
- James is 10 years old and 1 year younger than Cory.

How old is Jackson?</rewrite>

↓ ***Semantic grounding: resolve ambiguous references.***

**Round 3**

<think>The current query incorrectly uses “Emily” for both the parent and one of the children, causing ambiguity. Terms like “Cory” and “James” refer to specific children in context.</think>

<plan>**dimension:** semantic_grounding **operation:** resolve_references</plan>

<rewrite>Emily has 4 children named Amy, Jackson, Cory, and James.

- Amy is 5 years older than Jackson.
- Amy is 2 years younger than Cory.
- James is 10 years old.
- James is 1 year younger than Cory.

How old is Jackson?</rewrite>

<final>Emily has 4 children named Amy, Jackson, Cory, and James. Amy is 5 years older than Jackson. Amy is 2 years younger than Cory. James is 10 years old. James is 1 year younger than Cory. How old is Jackson?</final>

Figure 7: Case study for query rewriting on a math word problem. Green highlights indicate corrected/disambiguated references.

**Case 2: Expression Simplification and Reference Resolution**

**Question:** 玛丽是一位充满热情的园丁。昨天，她从自己最喜欢的苗圃收到了18 盆盆栽植物。在她大乡村住宅的40 个窗台中，每个窗台已经有2 盆盆栽植物。怀着慷慨之情，她决定明天从每个窗台上拿1 盆盆栽植物送给朋友和家人。玛丽还剩下多少盆盆栽植物?

**Round 1**

<think>The query is in Chinese. A full English translation is needed to facilitate downstream reasoning.</think>
<plan>**dimension:** language **operation:** full_english_translation</plan>
<rewrite>Mary is a passionate gardener. Yesterday, she received 18 potted plants from her favorite nursery. In her large rural mansion, there are 40 window sills, and each already has 2 potted plants. With a generous heart, she decided to take 1 potted plant from each window sill tomorrow to give to friends and family. How many potted plants will Mary have left?</rewrite>

↓ ***Structural reformulation: simplify expression.***

**Round 2**

<think>The current query is clear but can be reorganized to make the information more explicit and structured, separating the initial state from the action taken.</think>
<plan>**dimension:** structural_reformulation **operation:** simplify_expression</plan>
<rewrite>Yesterday, Mary received 18 potted plants from her favorite nursery. In her large rural mansion, there are 40 window sills, and each already has 2 potted plants. She decided to take 1 potted plant from each window sill tomorrow.
How many potted plants will she have left after giving some away?</rewrite>

↓ ***Semantic grounding: resolve pronoun references.***

**Round 3**

<think>The query contains references to “she” which refers to Mary and “each” which refers to window sills. These pronouns should be resolved for clarity.</think>
<plan>**dimension:** semantic_grounding **operation:** resolve_references</plan>
<rewrite>Yesterday, Mary received 18 potted plants from her favorite nursery. In Mary’s large rural mansion, there are 40 window sills, and each window sill already has 2 potted plants. Mary decided to take 1 potted plant from each window sill tomorrow.
How many potted plants will Mary have left after giving some away?</rewrite>
<final>Yesterday, Mary received 18 potted plants from her favorite nursery. In Mary’s large rural mansion, there are 40 window sills, and each window sill already has 2 potted plants. Mary decided to take 1 potted plant from each window sill tomorrow. How many potted plants will Mary have left after giving some away?</final>

Figure 8: Case study for query rewriting on a gardening word problem. Green highlights indicate resolved pronoun/reference substitutions.

**Case 3: Numerical Relation Preservation**

| Original Query | Rewritten Query |
|---|---|
| 格蕾丝的体重为125 磅。亚历克斯的体重比格蕾丝体重的四倍少2 磅。他们的总体重是多少磅?<br>（Grace weighs 125 pounds. Alex weighs 2 pounds less than four times Grace's weight. What is their total weight in pounds?） | Grace's weight is 125 pounds. Alex's weight is 2 pounds less than four times Grace's weight. What is their total weight in pounds?<br>Rewrite the relationships as computation hints using only given facts:<br>• Grace's weight = 125 pounds<br>• Alex's weight = 4× Grace's weight −2<br>• Total weight = Grace's weight + Alex's weight |

| Faithfulness Element | Evidence in Rewritten Query | Preserved |
|---|---|---|
| Entities: Grace, Alex | Both entities are retained through translation. | ✓ |
| Number: 125 pounds | "Grace's weight = 125 pounds" | ✓ |
| Multiplier: 4× | "4× Grace's weight" | ✓ |
| Offset: −2 pounds | "4× Grace's weight −2" | ✓ |
| Constraint relation | Alex = 4× Grace −2 is preserved exactly. | ✓ |
| Question intent | The target remains the total combined weight. | ✓ |

**Case 4: Temporal and Relational Preservation**

| Original Query | Rewritten Query |
|---|---|
| 杰瑞有两座水池，均有缺口，都以4 加仑/分钟的速度将其排空。4 分钟前，大水池的水量是小水池的两倍。现在大水池的水量是小水池的四倍。小水池里现在有多少水?<br>（Jerry has two pools, both with leaks, and both are draining at a rate of 4 gallons per minute. Four minutes ago, the volume of water in the large pool was twice that of the small pool. Now, the volume of water in the large pool is four times that of the small pool. How much water is currently in the small pool?） | Jerry has two pools, both with leaks that drain at a rate of 4 gallons per minute. 4 minutes ago, the larger pool had twice the amount of water that the smaller pool had. Now, the larger pool has four times the amount of water that the smaller pool has. How much water is currently in the smaller pool?<br>The larger pool and the smaller pool refer to two distinct pools. |

| Faithfulness Element | Evidence in Rewritten Query | Preserved |
|---|---|---|
| Entity: Jerry | "Jerry" is retained through translation. | ✓ |
| Constraint: two leaking pools | "two pools, both with leaks" | ✓ |
| Drain rate: 4 gallons/min | "4 gallons per minute" | ✓ |
| Temporal condition: 4 minutes ago | "4 minutes ago" | ✓ |
| Past ratio: larger = 2× smaller | "twice the amount ... 4 minutes ago" | ✓ |
| Current ratio: larger = 4× smaller | "four times ... now" | ✓ |
| Question intent | The target remains the current amount in the smaller pool. | ✓ |

Table 12: Qualitative faithfulness analysis of mRewriter-R1. For each example, we compare the original query with the rewritten query and audit whether key entities, numerical values, constraints, and question intent are preserved.

**System Prompt:**
You are a query rewriter. Your task is to improve a question for downstream reasoning across three turns, each targeting a different aspect.
**In the first turn, focus on the LANGUAGE Aspect.**
**Step 1:** Think inside `<think>` and `</think>` about the query's language issue.
**Step 2:** Write a plan inside `<plan>` and `</plan>`:
dimension: language
operation: one of [keep_original, full_english_translation, only_translate_key_terms, original_plus_english_version]
**Step 3:** Rewrite inside `<rewrite>` and `</rewrite>`.
**Rules:**
- Do NOT solve the problem.
- Do NOT change any facts, numbers, or constraints.

---

**In the second turn, focus on the STRUCTURAL Aspect.**
dimension: structural
operation: one of [ make_structure_explicit, group_information_by_entity, simplify_expression, restate_target_question, keep_original ]

---

**In the third turn, focus on the SEMANTIC Aspect.**
dimension: semantic
operation: one of [ resolve_references, clarify_ambiguous_terms, restate_as_computation, keep_original, ]

---

**Question:** {question}

Table 13: Multilingual Reasoning Template for the policy model of mRewriter-R1.

**Downstream Prompt for MGSM:**
Solve the following math problem.
Write your final answer on the last line in the format: `Answer: <number only>`
**Question:** {question}

---

**Downstream Prompt for XCOPA:**
Read the following commonsense reasoning question and select the correct answer.
Write your final answer on the last line in the format: `Answer: Choice 0 or Answer: Choice 1.`
**Question:**{question}

---

**Downstream Prompt for BELEBELE:**
Read the following multilingual reading comprehension question and select the correct answer.
Write your final answer as a number on the last line in the format: `Answer: Choice X` , where X is 1, 2, 3, or 4.
**Question:** {question}

Table 14: Multilingual Reasoning Template for downstream reasoning models.