Deep Reinforcement Learning Based Automated Prompt Optimization for Domain Relation Extraction

Anonymous ACL submission

Abstract

Relation extraction (RE) is a fundamental task in information extraction. Still, existing supervised approaches rely heavily on large-scale annotated data, limiting their applicability in domain-specific and low-resource scenarios. Prompt-based methods with large language models (LLMs) provide a parameter-efficient alternative; however, their performance is susceptible to prompt design, which often requires extensive domain expertise and heuristic trial-and-error. We propose REPO, a deep reinforcement learning (DRL)-based automated prompt optimization framework for domain relation extraction. REPO formulates prompt construction as a structured, sequential decision-making problem, optimizing prompt quality through interaction with a black-box LLM. To enable efficient and stable optimization, we introduce a two-stage framework comprising an initial prompt-construction stage that generates semantically grounded candidates and a DRL-based refinement stage that iteratively improves prompts within a constrained, domain-aware action space. We further design a composite evaluation metric that integrates extraction accuracy and semantic consistency to serve as a dense reward signal. Extensive experiments on multiple relation extraction datasets across medical, financial, legal, and news domains demonstrate that REPO consistently outperforms existing prompt-based methods and supervised baselines. Ablation studies further confirm the effectiveness and robustness of the proposed DRL-based prompt optimization strategy.

1 Introduction

Relation extraction as a core task in information extraction plays a critical role in applications such as knowledge graph construction and alignment (Chen et al., 2022; Zhang et al., 2022), question answering (Luo et al., 2018), and knowledge retrieval (Yang, 2020). Deep learning approaches, particularly supervised learning methods, have significantly improved RE performance. However, these methods rely heavily on large-scale, high-quality annotated datasets (Zeng et al., 2014), severely limiting their scalability across domains, especially in specialized fields such as medicine, law, and finance.

The in-context prompting paradigm of large language models (LLMs), which uses natural-language instructions to enable few-shot learning, provides a direct mechanism to reduce this dependency on annotated data. Encouraged by these advances, researchers have begun exploring in-context prompts for RE tasks, achieving encouraging results. Liu et al. (2024) proposes the Self-Prompting framework, which includes synonym and label generation and sentence rewriting to optimize the prompt. Lu et al. (2022) proposed the SURE model to guide the model's objectives by applying entity marking and sentence rewriting strategies. These approaches primarily rely on manually constructed templates, fully utilizing prior knowledge embedded in PLMs (Pre-trained Language Models). Li et al. (2023a) summarizes the semantic relation between head and tail entities to reason multi-step for summarization, and question and answer. While this method reduces task complexity, its label mapping process remains relatively intricate, leaving room for further optimization. More importantly, the manual design of high-quality prompts itself is a labor-intensive process that demands extensive domain expertise and iterative experimentation (Jiang et al., 2020), which fundamentally limits the efficiency and broader applicability of these methods.

Against this background, automated prompt generation have emerged as a key approach, aiming to reduce manual overhead and improve scalability. Luo et al. (2025) introduces TAPO, a multitask-aware prompt optimization framework integrating task-aware metric selection, multi-

metric evaluation, and evolution-based prompt refinement which improves prompt generation and enhances LLM adaptability across diverse tasks. As LLMs are highly sensitive to the surface form of prompts. Slight variations in wording, structure, or example ordering can lead to substantial performance fluctuations, especially in structured prediction tasks. Zhao et al. (2024) propose a method that includes automatic template generation, weighting, grouping, and optimization, effectively addressing template bias. Nevertheless, it requires additional entity-type annotations that lack generalization.

However, the effectiveness of automated prompt generation for relation extraction is constrained by two key challenges. On the one hand, prompt optimization for relation extraction entails a combinatorial and sequential decision process over heterogeneous components, including template structures, lexical realizations, entity descriptions, relation constraints, and example formats. Reliance on unstructured search strategies makes it difficult to efficiently explore this space, leading to slow convergence and high computational cost. On the other hand, existing automated prompts lack explicit modeling of domain-specific relation semantics, which limits their generalization to domain-specific relation extraction scenarios such as medicine and law.

To address these challenges, we propose a structured, controllable two-stage framework for automated prompt optimization in domain relation extraction, employing deep reinforcement learning (DRL) for fine-grained prompt refinement. During the initial prompt construction stage, we leverage large language models' semantic understanding to generate candidate prompts from example instances automatically. In the prompt-optimization stage, DRL is introduced to learn prompt-enhancement strategies that iteratively refine these candidates toward optimal prompts. Furthermore, by analyzing the relation extraction task, we propose a comprehensive prompt quality evaluation metric that integrates supervised metrics with similarity-based metrics, enabling a multi-dimensional assessment of prompt effectiveness. Our main contributions are as follows.

(1) We propose a two-stage prompt optimization framework that decomposes prompt learning into candidate generation and fine-grained refinement, enabling efficient DRL-based optimization within a constrained search space.

(2) We formulate prompt optimization for relation extraction as a sequential decision-making problem, explicitly encoding domain knowledge through a structured action space and domain-aware state representation.

(3) We conduct extensive experiments on multiple relation extraction datasets across different domains, demonstrating that the proposed method consistently outperforms existing prompt-based and supervised baselines, and validating the effectiveness of the RL-based prompt optimization through ablation studies.

2 Related Works

Relation extraction (RE) aims to identify structured triples ⟨head entity, relation, tail entity⟩ from natural language text. Traditional deep learning approaches to RE can be broadly categorized into pipeline-based methods (Xu et al., 2015), span-based methods (Dixit and Al-Onaizan, 2019; Mandya et al., 2020), sequence-to-sequence (Seq2Seq) methods (Nayak and Ng, 2020; Zeng et al., 2020), and machine reading comprehension (MRC)-based methods (Li et al., 2019; Zhao et al., 2021). With the emergence of large language models (LLMs), recent work has explored leveraging their strong language understanding and reasoning capabilities for relation extraction.

One line of research enhances RE by integrating LLMs with auxiliary modules or external reasoning frameworks. For example, Li et al. (2023b) combine LLMs with a natural language inference module to generate relational triplets, achieving improved performance on document-level RE tasks. While effective, these approaches typically require additional model components and customized training pipelines, increasing architectural complexity and computational overhead and limiting scalability.

Another line of work focuses on fine-tuning LLMs for relation extraction. Wadhwa et al. (2023) train the Flan-T5 model using Chain-of-Thought (CoT) style explanations to enhance relational reasoning and extraction accuracy. Despite their strong empirical performance, fine-tuning-based methods generally rely on large-scale, high-quality annotated datasets and incur substantial computational costs, making them less practical in low-resource or domain-specific scenarios.

More recently, prompt-based relation extraction has emerged as a parameter-efficient alterna-

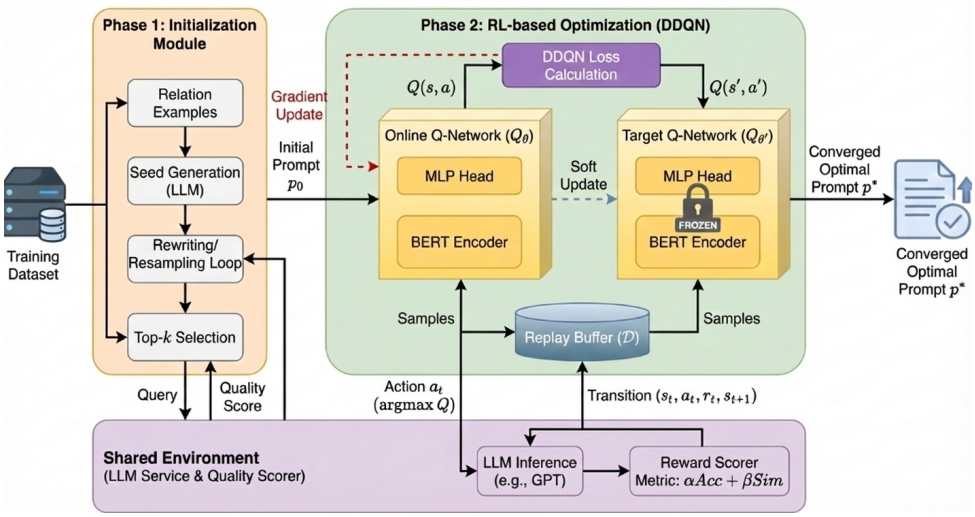

Figure 1: Relationship extraction prompts construction process

tive, in which LLMs perform RE via carefully designed prompts without updating model parameters (Mann et al., 2020). Wan et al. (2023) improve entity–relation alignment by retrieving task-relevant examples and incorporating logical reasoning demonstrations into prompts. Gutierrez et al. (2022) introduce a k-nearest neighbors (kNN) retrieval module to select representative in-context examples and systematically construct effective prompts, improving GPT-3s performance on bioinformatics information extraction tasks. Zhao et al. (2024) further explore prompt template design by varying the number of demonstrations and relation types, achieving notable improvements in entityrelation extraction.

Although prompt-based methods substantially reduce annotation requirements and deployment costs, they still face several fundamental challenges. Prior studies have shown that LLMs are highly sensitive to prompt formulations, and even semantically similar prompts can yield significantly different performance. Moreover, existing prompt construction strategies are largely heuristic, lacking unified optimization objectives or principled search mechanisms. Most approaches rely heavily on manual design choices, domain expertise, and extensive trial-and-error, particularly in domain-specific RE tasks. As a result, prompt quality is difficult to control, performance stability is hard to guarantee, and adapting prompts across domains or datasets remains costly and inefficient.

In contrast to prior prompt-based RE methods that treat prompt design as a static or heuristic process, our work formulates prompt construction as a structured and sequential decision-making problem. By explicitly incorporating domain knowledge into a constrained action space and optimizing promptsvia a two-stage reinforcement learning framework, our approach provides a principled,d scalable solution for robust prompt optimization in relation extraction.

3 Method

To address the aforementioned issues, we propose Reinforcement Learning-based Automated Prompt Optimization for Domain Relation Extraction(REPO). We formulate automated prompt construction for relation extraction as a structured optimization problem, consisting of two sequential stages: (i) an initial prompt generation stage for producing semantically reasonable seed prompts, and (ii) a reinforcement-learning-based prompt optimization stage for controlled and performance-driven refinement. We also propose a composite metric termed *Relation Extraction Prompt Quality Score* (REPQS). REPQS measures prompt quality from both prediction accuracy and semantic alignment perspectives, providing a more robust signal than conventional task metrics alone. An overview of the framework is illustrated in Figure 1.

3.1 Reward Signal and Evaluation Metric

To ensure that prompt optimization aligns with downstream task objectives, we design a comprehensive evaluation metric, REPQS, that serves as both an evaluation criterion and a reinforcement learning reward signal. This metric jointly considers extraction accuracy and semantic consistency,

enabling more precise and stable optimization of prompts for relation extraction.

Formally, given a prompt p, an input text x, and the LLM output $y = \text{LLM}(p \mid x)$, while y^* denotes the ground-truth relational triple. $F_1(\cdot)$ is the standard F1 score, measuring the overlap between predicted and ground-truth relation triples in terms of precision and recall; $Sim(\cdot)$ quantifies the semantic similarity between predicted and ground-truth triples; α and β are weighting coefficients that balance task accuracy and semantic consistency. The REPQS score of p is defined as:

$$\text{REPQS}(p) = \alpha \cdot F_1(y, y^*) + \beta \cdot Sim(y, y^*) \quad (1)$$

F1-based Evaluation. For the $F_1(\cdot)$ component, relation triples $\langle e_1, r, e_2 \rangle$ are extracted from the LLM output and compared against annotated ground-truth triples. Precision and recall are computed based on exact matching of entities and relations, and the F1 score is calculated accordingly.

Semantic Similarity Evaluation. To capture partial correctness and semantic proximity, the $Sim(\cdot)$ component measures similarity at the embedding level. Specifically, we employ a pre-trained language model to obtain vector representations for entities and relations in both predicted and ground-truth triples. For each predicted entity (or relation), cosine similarities with all ground-truth entities (or relations) are computed, and the maximum similarity score is selected. The final similarity score is obtained by averaging the entity-level and relation-level similarities and normalizing the result to $[0, 1]$.

By combining exact matching and semantic similarity, REPQS provides a dense, informative reward signal, enabling stable prompt optimization even under limited supervision and mitigating the sparsity issues associated with pure accuracy-based rewards.

3.2 Stage 1: Initial Prompt Construction.

To provide a stable, semantically grounded initialization for subsequent reinforcement learning, we design a heuristic algorithm (Algorithm 1) to construct an initial prompt set automatically. This stage aims to generate a compact, diverse, and high-quality search space rather than exhaustively exploring free-form prompt variations.

The algorithm first constructs the initial seed set p_{seed} by selecting the Top-k% prompts via REPQS (Line 4). It then enters an iterative optimization loop (Lines 5 to 21) until the maximum number of rounds, max_round, is reached or convergence is achieved. In each round, a large language model (LLM) rewrites the current seed prompts to generate semantically equivalent variants with more optimal structures, which are subsequently merged with the original seed set. Based on REPQS, the Top-k% prompts are filtered to maintain quality and reduce the search space, and the optimal prompt $best\_prompt$ is updated.

Algorithm 1 Automatic construction of prompt

Require: Supply relation extraction dataset R, number of iterations max_round, initial prompt p

1: previous_score $\leftarrow$ 0, best_score $\leftarrow$ -1, best_prompt $\leftarrow$ None
2: $r \leftarrow$ random_sample(R)
3: $I \leftarrow \text{LLM}(r)$
4: $p_{\text{seed}} \leftarrow \{p_i \mid p_i \in I, \text{Rank}(\text{REQPS}(p_i)) \leq \text{len}(I) \times k\%\}$
5: **while** round $<$ max_round **do**
6: $\quad I \leftarrow \text{re\_write}(p_{\text{seed}})$
7: $\quad I \leftarrow p_{\text{seed}} \cup I$
8: $\quad$ update_best_prompt(I)
9: $\quad I \leftarrow \{p_i \mid p_i \in I, \text{Rank}(\text{REQPS}(p_i)) \leq \text{len}(I) \times k\%\}$
10: $\quad$ **if** round $\bmod 5 = 0$ **then**
11: $\quad\quad$ resample(I)
12: $\quad$ **end if**
13: $\quad$ current_score $\leftarrow$ mean $(\sum \text{REQPS}(p_i))$, $p_i \in I$
14: $\quad$ delta $\leftarrow$ current_score $-$ previous_score
15: $\quad$ **if** abs(delta) $<$ 0.05 **and** holds for three consecutive iterations **then**
16: $\quad\quad$ **break**
17: $\quad$ **end if**
18: $\quad p_{\text{seed}} \leftarrow I$
19: $\quad$ previous_score $\leftarrow$ current_score
20: $\quad$ round $\leftarrow$ round $+ 1$
21: **end while**
22: **return** best_prompt

To preserve semantic diversity and avoid premature convergence, the algorithm resamples the filtered prompt set every five rounds. Convergence is determined by calculating the difference δ between the average REPQS scores of the current and previous rounds. If the absolute value of delta is less than 0.05 and this condition holds for three consecutive rounds, the algorithm terminates early. During the iteration, the seed set p_{seed} and the his-

torical average score are continuously updated.

Upon completing the loop, the algorithm outputs the optimal prompt $best_prompt$ with the highest score throughout the optimization process. This stage ultimately yields a compact, diverse, and task-aligned prompt set, significantly reducing the search space and laying a solid foundation for stable, efficient reinforcement learning-based optimization in the subsequent stage.

3.3 Stage 2: Reinforcement Learning-based Prompt Optimization.

With the seed prompts obtained in Stage 1, we further refine prompts through reinforcement learning to better adapt them to domain-specific RE tasks. Instead of unconstrained natural-language editing, we cast prompt optimization as a sequential decision-making problem over a structured, interpretable action space, enabling controlled exploration while preserving semantic validity.

MDP formulation. Prompt optimization is formulated as a Markov Decision Process (MDP) defined by (S, A, R). Since the transition dynamics are unknown and the LLM operates as a black box, a model-free reinforcement learning approach is adopted.

The state $s_t \in S$ represents the semantic state of the current prompt at step t. Concretely, the prompt text is encoded by a BERT encoder, and the hidden representations from the final layer are aggregated as a contextualized semantic embedding. This representation jointly captures prompt structure, entity-related cues, and relation-oriented semantics, which are critical for guiding effective prompt edits.

Action space. The action space in REPO is intentionally constrained to a set of predefined, task-relevant prompt editing operations, such as refining relation descriptions and adjusting entity role specifications. This design choice does not aim to guarantee theoretical near-optimality, but rather to ensure semantic validity and stable optimization trajectories. In contrast to unconstrained prompt rewriting, the structured action space reduces semantic drift and enables more reliable exploration during RL-based prompt optimization. The action space A consists of 11 predefined prompt-editing operations (Table 1), designed according to both relation extraction characteristics and domain-specific prior knowledge. These actions are grouped into seven categories:

i. *Original prompt retention*, which allows the agent to keep the current prompt unchanged to prevent over-optimization when a locally optimal prompt has been reached;

ii. *Structure rewriting*, which modifies sentence patterns to improve linguistic diversity and robustness to varied expression styles;

iii. *Vocabulary regularization*, which aligns relation-related expressions with canonical label definitions, reducing semantic ambiguity across different surface forms;

iv. *Entity information enhancement*, including adding entity types, descriptive attributes, and positional emphasis, to strengthen the models discrimination of entity roles;

v. *Relation information enhancement*, which supplements relation semantics and explicitly clarifies relation directionality (head vs. tail entity);

vi. *Example augmentation*, incorporating a small number of inputoutput pairs or sentence templates in a few-shot manner to provide task demonstrations;

vii. *Output format refinement*, which enforces structured and label-consistent outputs.

By constraining the optimization process to these semantically meaningful operations, the action space balances expressiveness and tractability, while maintaining interpretability and reducing the risk of semantic drift commonly observed in automatic prompt generation.

Category	ID	Specific Action
Original Prompt	0	Avoid over-optimization
Structure	1	Change sentence structure
Vocabulary	2	Regularize and correct keywords
Entity Information	3	Add entity type
	4	Add entity description
	5	Enhance entity position
Relation Information	6	Add relation description
	7	Enhance relation direction
Examples	8	Add input/output pairs
	9	Add sentence templates
Output Format	10	Enhance output format

Table 1: Prompt optimization action space

Reward design. Given a prompt p and an input text x, the concatenated input $[p : x]$ is fed into a black-box LLM to produce output y. We adopt REPQS as the basic reward signal. However, due to variations in input difficulty and stochasticity in

LLM inference, single-instance rewards can be unstable.

To address this issue, we define the reinforcement learning reward at the prompt level as the mean REPQS over a dataset $\mathcal{T}(x, y)$:

$$\text{M-REPQS}(p) = \frac{1}{n}\sum_{i=1}^{n} R_p(x_i), \qquad (2)$$

where $R_p(x_i)$ denotes the REPQS score of prompt p on input x_i. This averaged reward reduces variance and provides a more reliable optimization signal for policy learning.

Optimization algorithm. We adopt the Double Deep Q-Network (DDQN) algorithm to learn the optimal prompt-editing policy and alleviate Q-value overestimation. The BERT-encoded prompt representation is fed into a task-specific multilayer perceptron (MLP) to estimate action-value functions for all candidate actions.

To improve training stability and sample efficiency, an experience replay buffer stores transitions (s_t, a_t, r_t, s_{t+1}) collected during interaction with the environment. During training, mini-batches are randomly sampled from the buffer to break temporal correlations. Action selection follows an ϵ-greedy strategy with linear decay, encouraging exploration in early stages and exploitation of high-value actions in later stages. The online and target networks are periodically synchronized to further stabilize learning.

Through iterative interaction and reward-guided updates, the agent progressively learns to apply practical prompt-editing actions, yielding optimized prompts that demonstrate strong performance and robustness in low-resource relation extraction settings.

4 Experiments

4.1 Datasets

We evaluate the proposed method on four relation extraction datasets from different domains, including medicine, finance, law, and corporate news. For each dataset, we select representative relation types to construct a focused evaluation setting.

(1) **CMeIE**[1]: A Chinese medical relation extraction dataset. We select samples with the relations *etiology*, *drug treatment*, and *clinical manifestation*, resulting in 864 instances after filtering.

(2) **FinCUGE**[2]: A financial news relation extraction dataset. Only samples annotated with the relations *cooperation* and *ownership* are retained, yielding 1,219 instances.

(3) **LexEval**[3]: A Chinese legal case dataset. We select four relation types: *drug trafficking*, *human trafficking*, *illegal harboring*, and *possession*, resulting in 497 instances.

(4) **LCN** [4]: A domain-specific dataset collected from Cailian Press. After data cleaning and manual annotation, we retain samples with the relations *supplier*, *production*, and *composition*, resulting in 1,953 instances.

4.2 Baseline Methods

We compare our method with a range of representative prompt-based and supervised relation extraction approaches:

(1) **APE** (Zhou et al., 2023): an automatic prompt generation framework that produces multiple candidate prompts from examples and iteratively selects and rewrites prompts based on performance scores.

(2) **OPRO** (Yang et al., 2023):a prompt optimization approach that formulates prompt refinement as a natural language optimization task, where the large language model iteratively generates and evaluates new prompts.

(3) **SPO** (Xiang et al., 2025): a prompt optimization framework that improves prompts via self-supervised comparison of model outputs without relying on annotated ground-truth labels.

(4) **CasRel** (Wei et al., 2020): a supervised neural relation extraction model based on cascade and residual learning, which serves as a representative end-to-end baseline.

4.3 Experimental Settings

For each dataset, we split the data into training, validation, and test sets at 1:1:8. The training and validation sets are used for prompt construction and optimization, while the test set is reserved for final evaluation. We set the hyperparameter in REPO score as $\alpha = 5$, $\beta = 1$.

We report **Precision (P)**, **Recall (R)**, and **F1-score** as evaluation metrics. Precision measures the proportion of predicted relations that are correct, recall measures the proportion of gold rela-

[1] https://github.com/Robin-WZQ/CBLUE_CMeIE_model

[2] https://github.com/Macielyoung/FinCUGE_Instruction

[3] https://github.com/CSHaitao/LexEval

[4] https://github.com/dddong2-star/REPO

tions that are successfully identified, and F1-score is the harmonic mean of precision and recall. All reported results are computed on the test sets.

To further explore the potential value of prompt optimization in improving model performance, we also fine tune REPO (REPO-FT) applied to the main large language model Qwen2.5-7B-Instruct-1M by using the LoRA fine-tuning framework(Hu et al., 2022), and compare the performance of the fine-tuned model with the GPT-4o model.

5 Experimental Results

5.1 Main Results

Table 2 summarizes results on four domain-specific relation extraction datasets. REPO consistently achieves the highest F1 scores, demonstrating effectiveness and robustness in low-resource and cross-domain settings. The performance gains are further supported by the ablation study in Table 3, which shows a consistent drop in F1 when the prompt optimization component is removed.

REPO attains F1 scores of 0.72 on LexEval, 0.60 on FinCUGE, 0.58 on CMeIE, and 0.56 on LCN, outperforming all prompt-based baselines: APE, OPRO, and SPO. Compared to these, REPO improves F1 by roughly $4\% - 8\%$, with the most significant gains on FinCUGE and LCN. OPRO and SPO sometimes yield higher recall but have consistently lower precision, resulting in lower F1 scores. REPO more evenly balances precision and recall across datasets, indicating that its structured reinforcement-learning prompt optimization reduces false negatives without adding noise.

Compared with the supervised baseline, CasRel and REPO demonstrate competitive or superior performance under limited-annotation conditions. CasRel achieves substantial precision on LexEval, where data is abundant, and relations are less diverse. Still, its performance degrades on FinCUGE, CMeIE, and LCN due to scarce training data and more varied relation expressions. In contrast, REPO, which does not rely on task-specific supervised training, exhibits more stable performance, highlighting its advantage when labeled data are expensive or difficult to obtain.

We evaluate model fine-tuning by comparing REPO to its fine-tuned variant, REPO-FT, on Qwen2.5-7B-Instruct-1M using LoRA. REPO-FT achieves F1 improvements of 0.08 on LexEval, 0.05 on FinCUGE, and 0.07 on CMeIE, with a slight decrease on LCN. On average, fine-tuning increases F1 by 4.7%, confirming that task-aware parameter adaptation enhances RL-based prompt optimization.

Across all four data sets spanning the medical, financial, legal, and news domains, REPO consistently improves performance, indicating strong cross-domain generalization. These results suggest that learning reusable, constrained prompt-editing strategies via RL-based prompt optimization enhances the stability and adaptability of prompt-based relation extraction methods.

5.2 Ablation Study

To further analyze the sources of the performance gains observed in the main experiments, we conduct ablation studies to disentangle the contributions of the two key components in the REPO framework: the initial prompt construction stage and the reinforcement learning-based prompt optimization stage. Specifically, we compare the full REPO model with two ablated variants: (i) **Only Init**, which removes the RL-based optimization stage and retains only the initial prompt construction, and (ii) **Only RL**, which removes the initial prompt construction stage and performs prompt optimization solely through RL. All other components and experimental settings are kept identical.

Table 3 reports the F1 scores of the full model and the ablated variants across four datasets. The full REPO model consistently achieves the best performance on all datasets, outperforming both ablated variants. Compared to **Only Init** and **Only RL**, the full model yields absolute F1 improvements of 0.02/0.02 on LexEval, 0.04/0.07 on FinCUGE, 0.03/0.05 on CMeIE, and 0.04/0.05 on LCN, respectively. These results indicate that the two stages are complementary: the initial prompt construction provides a strong and stable starting point, while RL-based optimization further refines prompts toward higher-quality solutions.

The performance gains vary across datasets, consistent with the main experimental findings. The most notable improvements are observed on FinCUGE, which contains diverse relation expressions and limited annotated data, indicating that RL-based prompt optimization is particularly effective in challenging low-resource settings. On LexEval, where baseline performance is relatively strong, the improvements are smaller but consistent, suggesting stable rather than dataset-specific gains. Overall, the ablation results demonstrate that the initial prompt construction stage effec-

Table 2: Experimental Results of Different Models on Four Datasets

Method	Model	Metric	Dataset			
			Lexeval	FinCUGE	CMeIE	LCN
APE	**GPT-4o**	P	0.52	0.47	0.33	0.42
		R	0.70	0.49	0.59	0.46
		F1	0.60	0.48	0.42	0.44
OPRO	**GPT-4o**	P	0.61	0.49	0.46	0.44
		R	0.84	0.60	0.69	0.74
		F1	0.71	0.54	0.55	0.51
SPO	**GPT-4o**	P	0.38	0.46	0.43	0.42
		R	0.74	0.78	0.70	0.50
		F1	0.50	0.57	0.53	0.46
CasRel	**BERT**	P	0.89	0.36	0.42	0.54
		R	0.50	0.55	0.48	0.23
		F1	0.64	0.44	0.45	0.32
REPO	**GPT-4o**	P	0.59	0.55	0.48	0.51
		R	0.93	0.75	0.72	0.62
		F1	**0.72**	**0.60**	**0.58**	**0.56**
REPO-FT	**Qwen**	P	0.71	0.59	0.59	0.40
		R	0.91	0.73	0.73	0.70
		F1	0.80	0.65	0.65	0.51

Note: P=Precision, R=Recall, F1=F1-score; **bold** denotes the best results on each dataset.

Table 3: Comparison of F1-Scores between Ablation Experiment and Full Model

Dataset	Only RL	Only Init	Full
LexEval	0.70	0.70	0.72
FinCUGE	0.56	0.53	0.60
CMeIE	0.55	0.53	0.58
LCN	0.52	0.51	0.56

tively constrains the search space, while the RL-based optimization stage further refines prompts, and their combination enables robust relation extraction across different domains.

6 Conclusions and Future Work

This paper proposes REPO to improve performance on relation extraction tasks. We have designed a two-stage framework that combines heuristic initialization with DRL optimization. This design significantly reduces the adequate search space, improves optimization efficiency, and mitigates the instability commonly observed in unconstrained prompt search. Compared with representative relation extraction methods, REPO outperforms the baselines across four datasets, fully demonstrating its strong robustness and generalization in low-resource and cross-domain scenarios. In addition, the LoRA-based fine-tuning experiment (REPO-FT) further verifies the framework's potential to enhance the task adaptability of large language models. Through ablation experiments, we also confirm that the RL-based prompt-optimization component significantly improves performance on the relation extraction task.

In future work, we will design a prompt-compression and batch-evaluation strategy to reduce the number of tokens per interaction by removing redundant expressions and standardizing prompt structures, thereby lowering computational overhead. Additionally, we will migrate the core components of the basic framework to multilingual pre-trained models and expand the action space to support adjustments to grammatical structures across different languages.

7 Limitations

Despite its effectiveness, REPO has several limitations. First, the framework requires iterative interactions with large language models during prompt rewriting and evaluation, leading to increased computational cost and token consumption compared to static prompt-based methods. Second, although domain knowledge is encoded through a structured action space, adapting this design to new tasks or domains may require additional manual effort. Finally, our experiments are limited to Chinese relation extraction datasets, and the generalization of REPO to multilingual or cross-lingual settings remains to be explored.

References

Xiang Chen, Ningyu Zhang, Lei Li, Shumin Deng, Chuanqi Tan, Changliang Xu, Fei Huang, Luo Si, and Huajun Chen. 2022. Hybrid transformer with multi-level fusion for multimodal knowledge graph completion. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, pages 904–915.

Kalpit Dixit and Yaser Al-Onaizan. 2019. Span-level model for relation extraction.

Bernal Jimenez Gutierrez, Nikolas McNeal, Clay Washington, You Chen, Lang Li, Huan Sun, and Yu Su. 2022. Thinking about gpt-3 in-context learning for biomedical ie? think again. *arXiv preprint arXiv:2203.08410*.

Edward J. Hu, Yelong Shen, Phil Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Swabha Wang, Lu Wang, Weizhu Chen, and Denny Zhou. 2022. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*. Accepted to ICLR 2022.

Zhengbao Jiang, Frank F Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Guozheng Li, Peng Wang, and Wenjun Ke. 2023a. Revisiting large language models as zero-shot relation extractors. *arXiv preprint arXiv:2310.05028*.

Junpeng Li, Zixia Jia, and Zilong Zheng. 2023b. Semi-automatic data enhancement for document-level relation extraction with distant supervision from large language models. *arXiv preprint arXiv:2311.07314*.

Xiaoya Li, Fan Yin, Zijun Sun, Xiayu Li, Arianna Yuan, Duo Chai, Mingxin Zhou, and Jiwei Li. 2019. Entity-relation extraction as multi-turn question answering. *arXiv preprint arXiv:1905.05529*.

Siyi Liu, Yang Li, Jiang Li, Shan Yang, and Yunshi Lan. 2024. Unleashing the power of large language models in zero-shot relation extraction via self-prompting. *arXiv preprint arXiv:2410.01154*.

Keming Lu, I-Hung Hsu, Wenxuan Zhou, Mingyu Derek Ma, and Muhao Chen. 2022. Summarization as indirect supervision for relation extraction. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6575–6594.

Kangqi Luo, Fengli Lin, Xusheng Luo, and Kenny Zhu. 2018. Knowledge base question answering via encoding of complex query graphs. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 2185–2194.

Wenxin Luo, Weirui Wang, Xiaopeng Li, Weibo Zhou, Pengyue Jia, and Xiangyu Zhao. 2025. Tapo: Task-referenced adaptation for prompt optimization. *arXiv preprint arXiv:2501.06689*.

Angrosh Mandya, Danushka Bollegala, and Frans Coenen. 2020. Graph convolution over multiple dependency sub-graphs for relation extraction. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6424–6435. International Committee on Computational Linguistics.

Ben Mann, N Ryder, M Subbiah, J Kaplan, P Dhariwal, A Neelakantan, P Shyam, G Sastry, A Askell, S Agarwal, and 1 others. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 1:3.

Tapas Nayak and Hwee Tou Ng. 2020. Effective modeling of encoder-decoder architecture for joint entity and relation extraction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8528–8535.

Somin Wadhwa, Silvio Amir, and Byron C Wallace. 2023. Revisiting relation extraction in the era of large language models. In *Proceedings of the conference. association for computational linguistics. meeting*, volume 2023, page 15566.

Zhen Wan, Fei Cheng, Zhuoyuan Mao, Qianying Liu, Haiyue Song, Jiwei Li, and Sadao Kurohashi. 2023. Gpt-re: In-context learning for relation extraction using large language models. *arXiv preprint arXiv:2305.02105*.

Zhepei Wei, Jianlin Su, Yue Wang, Yuan Tian, and Yi Chang. 2020. A novel cascade binary tagging framework for relational triple extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1476–1488.

Jinyu Xiang, Jiayi Zhang, Zhaoyang Yu, Fengwei Teng, Jinhao Tu, Xinbing Liang, Sirui Hong, Chenglin Wu, and Yuyu Luo. 2025. Self-supervised prompt optimization. *Preprint*, arXiv:2502.06855.

Yan Xu, Lili Mou, Ge Li, Yunchuan Chen, Hao Peng, and Zhi Jin. 2015. Classifying relations via long short term memory networks along shortest dependency paths. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1785–1794.

C. Yang, X. Wang, Y. Lu, and 1 others. 2023. Large language models as optimizers. In *ICLR*.

Zuoxi Yang. 2020. Biomedical information retrieval incorporating knowledge graph for explainable precision medicine. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2486–2486.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. 2014. Relation classification via convolutional deep neural network. In *Proceedings of COLING 2014, the 25th international conference on computational linguistics: technical papers*, pages 2335–2344.

Daojian Zeng, Haoran Zhang, and Qianying Liu. 2020. Copymtl: Copy mechanism for joint extraction of entities and relations with multi-task learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 9507–9514.

Rui Zhang, Bayu Distiawan Trisedya, Miao Li, Yong Jiang, and Jianzhong Qi. 2022. A benchmark and comprehensive survey on knowledge graph entity alignment via representation learning. *The VLDB Journal*, 31(5):1143–1168.

Tianyang Zhao, Zhao Yan, Yunbo Cao, and Zhoujun Li. 2021. Asking effective and diverse questions: A machine reading comprehension based framework for joint entity-relation extraction. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 3948–3954.

Xiaoyan Zhao, Min Yang, Qiang Qu, and Ruifeng Xu. 2024. Few-shot relation extraction with automatically generated prompts. *IEEE Transactions on Neural Networks and Learning Systems*, 36(3):4971–4983.

Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, and 1 others. 2023. Large language models are human-level prompt engineers. In *ICLR*.