LEVERAGING HARD NEGATIVE SAMPLES: WEAKNESS-DRIVEN POLICY OPTIMIZATION FOR GENERATIVE INFORMATION EXTRACTION

Anonymous authors
Paper under double-blind review

ABSTRACT

While Large Language Models (LLMs) show remarkable capabilities in Generative Information Extraction, their performance remains constrained by the uncontrollable nature of pre-defined label generation. Existing research, centred on prompt engineering or supervised fine-tuning, offers limited gains due to the neglect of hard negative samples. However, leveraging these samples introduces challenges, including deficient error-type awareness, inadequate training focus, and weak dynamic adaptability. To address these issues, we propose a novel framework titled Weakness-driven Policy Optimization (WPO). It first collects error samples that reveal LLM weaknesses, followed by pre-defined atomic error types. Then, WPO uses an LLM to mine composite error types to gain an error distribution. Subsequently, the framework constructs a training set via error-centric data synthesis and error-type-based data augmentation. Finally, WPO employs a dynamic weight allocation mechanism that strategically assigns weights to original training, synthetic, and augmented samples, fine-tuning the LLM using Group Relative Policy Optimization (GRPO). Experimental results on the Qwen3 series LLMs show that WPO achieves a performance gain of up to 1.12% in F1 and 5.31% in precision on Qwen3-8B in Relation Extraction compared to the state-of-the-art (SoTA) baseline. Furthermore, extensive analysis validates the cross-task generalization and cross-scenario robustness of our method.

1 INTRODUCTION

Information Extraction (IE) is a foundational capability of LLMs that focuses on extracting structured information from plain text, including Relation Extraction (RE) (Thanh et al., 2025) and Event Extraction (EE) (Huang et al., 2025). As LLM scales increase, adaptation and generalization capabilities have evolved, driving a paradigm shift toward Generative IE (GIE). Under this paradigm, IE is treated as a sequence generation task, where the LLM decodes target tokens to form the final structured output (Zhang et al., 2025).

Though scaling up models improves IE performance, it incurs high computational costs. To balance efficiency and efficacy, we propose enhancing LLMs for GIE by exploiting samples that reveal model weaknesses. Related research can be broadly grouped into two directions: **Model Weakness Identification**. One line of work attributes errors to pre-defined weakness types (Cheng et al., 2024; Kamoi et al., 2024). Another explores automatic error discovery, such as clustering-based methods like EvalTree (Zeng et al., 2025). While the former is constrained by incomplete predefined categories, the latter may fail to generalize common error patterns. **Weakness-Driven Model Enhancement.** Work in this category strengthens models through training, using methods such as RISE (Xu et al., 2025a) and SwS (Liang et al., 2025). A key limitation is that such negative sample training often remains insensitive to specific error types.

In this paper, we focus on Leveraging Hard Negative Samples to improve GIE. Current methods encounter three main challenges: *1) Deficient Error-Type Awareness*: Binary feedback offers limited granularity for model refinement, failing to distinguish critical error types in GIE such as span errors versus label errors; *2) Inadequate Training Focus*: Conventional training lacks attention to dominant or systematic error patterns, with no targeted correction for model-specific weaknesses; *3) Weak*

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.

Dynamic Adaptability: Statically constructed training datasets cannot adapt to the evolving and model-dependent error profiles of different LLMs.

To address these challenges, we propose a novel Weakness-driven Policy Optimization (WPO) framework that leverages hard negative samples to optimize the LLM weaknesses in GIE. Specifically, for 1) Deficient Error-Type Awareness, establish a hierarchical error evaluation mechanism. Define atomic errors first, then use LLMs to mine composite errors. This enables fine-grained awareness of error samples and accurately characterizes the error types distribution. For 2) Inadequate Training Focus, design a weakness-driven data construction strategy. Configure sample types and scales flexibly via Error-centric Data Synthesis and Error-type-based Data Augmentation based on error distribution. Additionally, introduce a sample weighting mechanism to guide focused training through differentiated weights. For 3) Weak Dynamic Adaptability, implement the weakness-driven data construction strategy. Dynamically mine error type lists based on the specific weaknesses of different LLMs. Generate customized training data to achieve dynamic alignment between data LLM capabilities. Experimental results on the Qwen3-8B in RE show that WPO outperforms the SoTA method, achieving relative performance gains of 1.12% in F1 and 5.31% in P.

In summary, our contributions are as follows:

- We propose a novel Weakness-driven Policy Optimization (WPO) framework that leverages hard negative samples to optimize the LLM weaknesses in GIE. Besides, we establish an error evaluation mechanism to improve error awareness.
- We introduce two modules: dataset construction via error, which dynamically builds training data based on LLM errors; weakness-driven GRPO training, which leverages the weighting mechanism to enhance training focus on model weaknesses.
- We conduct extensive experiments on three public datasets and one private dataset. The results indicate that WPO achieves SoTA performance not only in general but also in complex scenarios. Moreover, extensive analysis validates the cross-task generalization and cross-scenario robustness of WPO.

2 Related Work

Model weakness identification. To efficiently analyze the massive errors produced by LLMs daily, automatic identification of model weaknesses is crucial for reducing human effort (Zeng et al., 2024; Deshpande et al., 2025). Existing approaches can be categorized into two paradigms: First, attribution to pre-defined error types, where errors are assigned to known categories. For instance, AUTODETECT (Cheng et al., 2024) introduces a multi-agent framework based on a hierarchical taxonomy. ReaLMistake (Kamoi et al., 2024) investigates LLM-as-judge capabilities using prompting strategies across four error types. Further advancing this line, MisAttributionLLM (Xu et al., 2025b) is trained on a dataset with 6 primary and 15 secondary pre-defined categories, enabling more accurate error attribution. Second, automatic error discovery, which identifies novel, recurring error patterns without prior categorization. For example, EAP (Sun et al., 2025) introduces a framework for dynamic error classification. Moreover, EvalTree builds a capability tree by clustering error samples and summarizing patterns with LLMs. However, pre-defined taxonomies may be incomplete, while purely automatic discovery approaches may fail to capture common error patterns systematically.

Weakness-Driven model enhancement. With these errors identified, prompt-based methods offer a straightforward approach. They typically utilize the error cases or their solutions to guide LLM responses during inference (Tong et al., 2024). However, for enhancing the intrinsic capabilities of the model itself, current research predominantly focuses on training-based methods (An et al., 2023). Such methods typically involve synthesizing training data based on the identified error types. For example, RISE injects errors and applies self-editing before using DPO to train on those specific cases, while SwS analyzes RL errors to build targeted data that systematically addresses model weaknesses. However, a key limitation remains: such training with negative samples often lacks sensitivity to different error types.

3 OVERVIEW

3.1 TASK FORMULATION AND PRELIMINARY

Definition 1: Relation extraction (RE). Given an input text $C = [c_1, c_2, ..., c_n]$ and a set of pre-defined relations $\mathcal{R} = \{r_1, r_2, ..., r_l\}$, where n and l denote the numbers of tokens in C and relations in $\mathcal{R}$, respectively, the purpose of relation extraction task is to obtain a set of triplets $\mathcal{T} = \{(sb_i, r_i, ob_i)\}_{i=1}^{m}$, where sb_i, ob_i represent the subjects and objects connected by r_i and m denotes the total number of extracted triplets.

Definition 2: Event extraction (EE). Given a text $C_e = [c_1, c_2, ..., c_{n_e}]$ and the pre-defined event type set $\mathcal{E} = \{e_1, e_2, ..., e_{l_e}\}$, event extraction task aims to detect the mention event trigger tr and arguments $\mathcal{A} = \{a_1, a_2, ..., a_{m_e}\}$ from C_e and the event type $e \in \mathcal{E}$ (e.g., `LOCATION`, etc).

Definition 3: Aspect-based sentiment analysis (ABSA). Given an input text $C_a = [c_1, c_2, ..., c_{n_a}]$ and a set of pre-defined sentiment polarities $\mathcal{S} = \{s_1, s_2, ..., s_{l_a}\}$ (e.g., $\{$`POSITIVE`, `NEGATIVE`, `NEUTRAL`$\}$), where n_a and l_a is the number of tokens in C_a sentiment polarities in $\mathcal{S}$, respectively, aspect-based sentiment analysis task aims to extract a set of aspect-sentiment triplets $\mathcal{Y} = \{(at_j, o_j, s_j)\}_{j=1}^{k}$ from C_a. k is the number of aspect-sentiment triplets and at_i, o_i denote the aspect term and opinion term in C_a.

Definition 4: Atomic error type. The most irreducible error in IE tasks serves as the foundation for fine-grained error diagnosis. For example, the subject span error is an atomic error type in RE. Let $E_{atom} = \{ea_1, ea_2, \ldots, ea_p\}$ denote the exhaustive set of p pre-defined atomic error types.

Definition 5: Composite error type. A high-level error is formed by the combination or co-occurrence of multiple atomic errors. It offers a holistic view of model weaknesses by mining atomic error types. For example, in RE, the entity-pair span error is a composite error type consisting of subject and object span errors. Formally, a composite error e_{comp} is a non-empty subset of atomic errors:

$$e_{comp} \subseteq E_{atom}, \quad \text{s.t.} \quad |e_{comp}| \geq 2. \tag{1}$$

Definition 6: Hard negative sample. It is designed to expose model weaknesses and trigger incorrect outputs or hallucinations by probing the boundaries of its capabilities.

3.2 FRAMEWORK

As shown in Figure 1, our WPO comprises two key modules: (1) Dataset Construction via Error and (2) Weakness-Driven GRPO Training. The first module operates in three stages: A. Error Sample Collection: acquiring error samples that expose model weakness. B. Error Type Distribution: Utilizing LLMs to identify and rank error types based on pre-defined atomic error types. C. Data Construction: obtaining RL training data through error-centric data synthesis (covering instance-level, scenario-level) and error-type-based data augmentation. In the second module, we tune the initial SFT model using GRPO. By strategically weighting original training, synthetic, and augmented samples, we target identified weaknesses to further enhance model performance.

4 METHODS

4.1 DATASET CONSTRUCTION VIA ERROR

To address the weaknesses revealed by the model, we leverage samples where the model outputs errors. We propose a novel, error-centric data construction framework comprising three stages: Error Sample Collection, Error Type Distribution, and Data Construction. Although demonstrated via RE, this framework is generalizable to various IE tasks.

Error samples collection. To obtain samples that expose model weaknesses, we first perform SFT on a base LLM using half of the training set to establish instruction-following capabilities. Then, we evaluate this LLM on the remaining training set to get samples with incorrect outputs. Besides, we pre-defined atomic error types for IE tasks across multiple granularities. For RE, these include triplet-level (i.e., triplet not found), span-level (i.e., subject/object boundary errors), and label-level (i.e., subject/relation/object type errors).

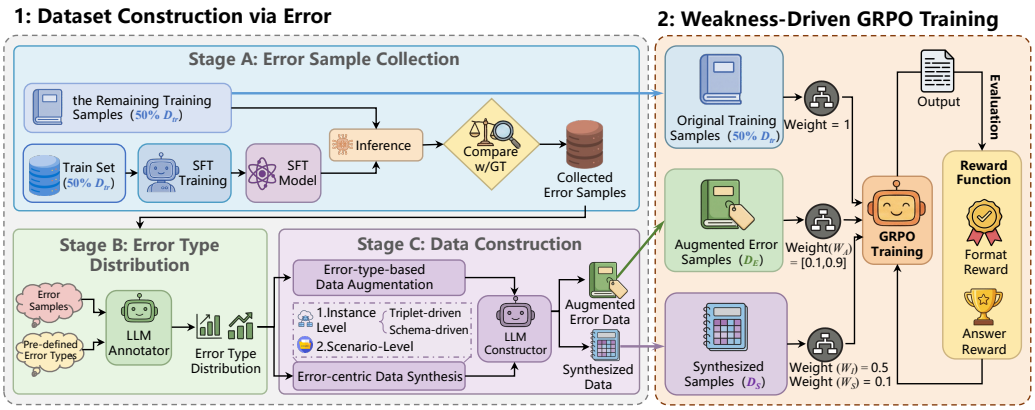


Figure 1: Overview of WPO. **Left:** Dataset Construction via Error: A) collects error samples from the initial SFT model; B) LLMs annotates a ranked error-type distribution; C) builds synthesized samples D_c (instance-level and scenario-level) and augmented samples D_a accordingly. **Right:** Weakness-Driven GRPO Training over D_{tr} (remaining original samples, weight 1.0), D_a ($W_A \in [0.1, 0.9]$, by error-type ranking), and D_c ($W_I = 0.5$ instance-level, $W_S = 0.1$ scenario-level), with a composite reward (span-level F1 + format consistency).

Error type distribution. Based on the collected error sample set and pre-defined atomic error types, we utilise an LLM (e.g. GPT-5.1[1]) for error mining. The LLM analyzes samples via prompting to identify either single atomic errors or composite error types (combinations of two or more atomic error types). This process yields a dynamic list of error types along with their quantity distribution and ranking. It is worth noting that the number of error types is not fixed.

Data construction. Based on the ranked error types and error samples obtained, we design two data construction strategies: **ERROR-CENTRIC DATA SYNTHESIS** and **ERROR-TYPE-BASED DATA AUGMENTATION**.

For **DATA SYNTHESIS**, we perform synthesis at two levels: Instance-level and Scenario-level.

1) *At the instance-level*, we have two synthesis paths: **a) Triplet-driven**, synthesizes samples by extracting error triplets from error instances. The core idea is to add entity descriptions (e.g., subject background information) to increase knowledge and compensate for model weaknesses. Specifically, based on the golden answers of error instances, we use LLMs to generate sentences to synthesize the source text. **b) Schema-driven**, synthesizes samples by extracting error triplet schemas from error instances. The core idea is to increase the diversity of triplets within a schema and mitigate schema distribution imbalance, thereby addressing the lack of training focus. Specifically, based on the triplet schemas of the golden answers from error samples, we first utilize an LLM to generate new triplets and then construct sentences based on those triplets to synthesize the source text.

2) *At the scenario-level*, we synthesize samples based on complex triplet scenarios. The core idea is to improve the long-tail distribution of complex triplet samples, thereby addressing the lack of training focus. Specifically, based on the complex scenarios to which the error samples belong, we first use an LLM to generate specific complex triplets (e.g., overlapping triplets, multiple triplets, and cross-sentence triplets). Then, we prompt the LLM to generate the corresponding source text via these synthesized complex triplets.

For **DATA AUGMENTATION**, we adopt the remaining half of the training set and then rewrite the golden answers to obtain error triplets of specific error types. The core idea is to strengthen the model's perception of error types by constructing error samples and labelling error types. Specifically, we first perform error rewrite on the golden answers of error samples and then attach error types. The augmented error sample format is: "*Input:* text + error type + error result; *Output:* golden answer".

[1]See Appendix B for GPT-5.1 computational cost.

Finally, we proportionally sample the four types of constructed samples based on the error type distribution, while filtering out Out-of-Capacity Hard Samples (i.e., samples with 0 successes out of N), to form the weakness-driven GRPO training dataset. It is worth noting that our data construction is determined by the distribution of obtained error types. This is a dynamic construction process that adapts to the varying weaknesses of different LLMs.

4.2 WEAKNESS-DRIVEN GRPO TRAINING

In this stage, the initial SFT model obtained from the Error Sample Collection stage is tuned by a GRPO algorithm (Guo et al., 2025). The GRPO training data consists of: 1) *the remaining Original Training Samples*: these represent the highest quality and serve as the foundational support for training and are assigned a static weight of 1.0. 2) *Synthetic Samples*: Instance-level and scenario-level samples are assigned weights of $W_I = 0.5$ and $W_S = 0.1$, respectively, aiming to mitigate the impact of potential data noise introduced by the LLM generation. 3) *Augmented Samples*: For error samples with annotated error types, we implement a dynamic weight distribution ranging from $[0.1, 0.9]$ based on error-type ranking; samples with the highest-quantity error types receive a weight of $W_A = 0.9$, while those with the lowest receive $W_A = 0.1$. During the training, these weights serve as adjustment factors applied directly to the reward function. By multiplying them by $R(o_i)$, we achieve differentiated learning across data of varying quality and types.

As shown in Figure 1, generated candidate outputs are sent to the "evaluation" step and scored by a Reward Function. To guide the LLM towards optimal performance, a composite reward is designed to assess both answer accuracy and format adherence. First, a span-level F1 reward R_{F1}, quantifies the match between the extracted entities and the ground truth. It is computed as the micro-F1 score, linearly scaled to the range $[0, 1]$, providing a direct incentive for precision and recall. Second, a structural consistency reward R_{schema}, evaluates adherence to the pre-defined output format and entity type constraints. Outputs with any schema violations (e.g., malformed structures or unpermitted entity types) receive $R_{\text{schema}} = 0$, whereas compliant outputs receive $R_{\text{schema}} = 1$. The total reward $R(o_i)$ is a weighted combination:

$$R(o_i) = \lambda_{F1} R_{F1}(o_i) + \lambda_{\text{schema}} R_{\text{schema}}(o_i), \tag{2}$$

where o_i is candidate outputs. In our experiments, the λ_{F1} and λ_{schema} are set to 10 and 1, respectively, to prioritize entity correctness while strongly enforcing structural validity.

Finally, the computed rewards guide the GRPO policy update. The average reward of the samples within the group is utilized as a baseline to compute the relative advantage. Then, the advantage A_i for o_i is defined as:

$$A_i = R(o_i) - \frac{1}{G} \sum_{j=1}^{G} R(o_j), \tag{3}$$

where the advantage A_i is used for policy updates. The policy parameters θ are optimized by maximizing the following objective function:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{q,\{o_i\}\sim\pi_{\theta_{\text{old}}}} \left[\frac{1}{G} \sum_{i=1}^{G} \left(\min\left(r_i A_i, \text{clip}\left(r_i, 1-\varepsilon, 1+\varepsilon\right) A_i \right) - \beta \mathbb{D}_{\text{KL}}\left(\pi_\theta \| \pi_{\text{ref}}\right) \right) \right], \tag{4}$$

where $r_i = \pi_\theta(o_i \mid q) / \pi_{\theta_{\text{old}}}(o_i \mid q)$ is the probability ratio between the current policy π_θ and the old policy $\pi_{\theta_{\text{old}}}$, and ε is a clipping hyperparameter. The $\beta \mathbb{D}_{\text{KL}}$ represents KL divergence regularization, penalizing deviations of the current policy π_θ from a reference policy π_{ref}.

5 EXPERIMENTS

5.1 EXPERIMENTAL SETUP

Datasets. We evaluate our method on CMeIE-V2 (Zhang et al., 2022), which involves three complex scenarios, i.e., overlapped triplets, multiple triplets, and cross-sentence triplets. To assess the robustness of our method, we further conduct experiments on the other two general GIE datasets, namely ACE05 (Walker et al., 2006), and 14Lap (Xu et al., 2020). The statistical details of the

Table 1: Statistics of the three public and one private datasets. "Hard" denotes negative sample. As the CMeIE-V2 test labels remain unreleased, the original validation set serves as our test set.

Dataset	Task	Langage	Domain	Public	Training		Hard	Test
					SFT	RL		
CMeIE-V2	RE	Chinese	Medical	✓	7,170	7,169	3,000	3,585
ACE05	EE	English	General	✓	1,650	1,692	800	293
14Lap	ABSA	English	General	✓	450	456	200	328
PMD	Dialogue	Chinese	Medical	✗	1,429	1,204	133	387

datasets are provided in Table 1. To validate effectiveness across tasks and reduce data contamination risks (Jiang et al., 2024), we collaborate with an online hospital to construct a private medical dialogue dataset (PMD), with all personal identification information removed for privacy protection.

Baselines. Our framework is compared with several LLMs performing well in GIE. 1) Two SoTA ***Closed-source LLMs***: GPT-4o (Achiam et al., 2023) and Qwen-Max (Yang et al., 2025). 2) Four ***Open-source LLMs***: InternLM3-8B (Cai et al., 2024), LLaMA3.1-8B (Dubey et al., 2024), GLM4-9B (GLM et al., 2024), Qwen3-4B, and Qwen3-8B (Yang et al., 2025). 3) Three ***Instruction-Tuned*** strategies for two LLMs. 4) Four ***Effectiveness of RL*** strategies for two LLMs: DPO (Rafailov et al., 2023), KTO (Ethayarajh et al., 2024), ORPO (Hong et al., 2024), and GRPO (Guo et al., 2025).

Hyperparameters. We conduct experiments on both Qwen3-4B and Qwen3-8B. Across both the SFT and RL stages, we utilize LoRA (Hu et al., 2022) with a rank $r = 8$ and a scaling factor $\alpha = 16$. For SFT, we train for 6 epochs with a batch size of 32 and a learning rate of 1×10^{-4}. In the RL stage, we evaluate GRPO, DPO, KTO, and ORPO methods. These are trained for 4 epochs with a batch size of 8. The learning rate is set to 5×10^{-6}, except for GRPO, which uses 5×10^{-5} with 6 rollouts per prompt. For inference, we set the temperature to 0.3 and the top-p sampling parameter to 0.9 to balance output diversity and coherence. All experiments are conducted on a workstation running Ubuntu 20.04.6 LTS, equipped with two Intel(R) Xeon(R) Platinum 8336C CPUs, four NVIDIA A800 GPUs, and 1.0 TiB of RAM.

Evaluation metrics. Following the previous works (Bao et al., 2025), we employ three standard evaluation metrics, i.e., Precision (P), Recall (R), and strict Micro-F1 (F1). Notably, in RE, a triplet is considered correct only if the relation type, along with the types and the boundaries of the head-tail entities are precisely determined. For EE, we decompose events into (trigger, event type, argument, argument role) quadruples, requiring exact matching of all components. For ABSA, a triplet is regarded as correct only when the aspect, opinion, and sentiment all coincide with the ground truth. For PMD, we employ the LLM-as-judge method (Zhu et al., 2025) to evaluate responses across four dimensions: **medical accuracy**, **clinical logic**, **communication**, and **empathy**, on a scale from 1 to 5 and normalized to 0-100. The weights for these four dimensions are 4:4:1:1.

5.2 Main Results

To validate the effectiveness of WPO, we compared various LLMs and training methods on the three datasets. As shown in Table 2, we observed several results: 1) Our method outperforms all baselines, including prominent closed- and open-source LLMs, achieving SoTA performance across three IE datasets. Specifically, on CMeIE-V2, it improves F1 score and precision by 1.12% / 5.31% for Qwen3-8B, by 1.02% / 4.72% for Qwen3-4B, and by 1.06% / 3.94% for LLaMA3.1-8B. 2) Our approach also exceeds various RL baselines, highlighting the effectiveness of our weighting mechanism in addressing the inadequate training focus. 3) By leveraging weakness-driven data construction, our method outperforms SFT trained on static datasets.

5.3 Complex Scenario Analysis

In this section, we analyze our WPO on complex RE in three scenarios, with the results listed in Table 3. **Analysis on overlapping scenario.** We conduct an exploration of complex samples containing various overlapping patterns on the CMeIE-V2 to verify the effectiveness of WPO. From the results, we observe that our method achieves SoTA F1 on SEO and EPO, demonstrating good performance in dealing with overlapped triplets. **Analysis on multiple triplet scenario.** We fur-

Table 2: Comparisons on three GIE tasks. D_{tr} indicates training data. For RL training, the model uses the remaining 50% of the training data and synthetic hard negative samples. All results are the average of three runs, with standard deviations of 0.1-0.2.

Method		CMeIE-V2			ACE05			14Lap		
		P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)
Close-source LLMs										
GPT-4o (Achiam et al., 2023)		27.38	22.71	24.83	6.28	5.56	5.89	40.69	35.12	37.70
Qwen-Max (Yang et al., 2025)		32.94	22.74	26.90	9.95	18.01	12.82	43.64	46.95	45.24
Open-source LLMs										
InternLM3-8B (Cai et al., 2024)		10.62	7.61	8.87	2.11	4.60	2.90	20.29	39.37	26.78
LLaMA3.1-8B (Dubey et al., 2024)		8.76	9.51	9.12	1.68	3.26	2.22	27.28	38.08	31.79
GLM4-9B (GLM et al., 2024)		7.11	9.31	8.06	6.29	8.42	7.20	27.55	36.97	31.57
Qwen3-8B (Yang et al., 2025)		21.71	20.94	21.32	6.79	11.49	8.53	22.53	39.19	28.61
Qwen3-4B (Yang et al., 2025)		12.57	13.81	13.16	5.37	1.53	2.38	12.63	38.08	18.99
Instruction-Tuned Baselines										
Qwen3-8B	w/ SFT 50% D_{tr}	48.16	50.38	49.24	46.32	44.64	45.46	57.34	53.42	55.31
	w/ SFT 50% D_{tr} + SFT 50% D_{tr}	51.14	51.76	51.45	49.32	48.66	48.99	59.61	56.75	58.14
	w/ SFT 100% D_{tr}	51.42	52.58	52.00	51.06	50.77	50.91	62.20	57.49	59.75
Qwen3-4B	w/ SFT 50% D_{tr}	48.24	49.95	49.08	45.23	42.72	43.94	59.11	51.57	55.08
	w/ SFT 50% D_{tr} + SFT 50% D_{tr}	48.75	52.94	50.76	48.44	47.70	48.07	58.72	56.01	57.33
	w/ SFT 100% D_{tr}	50.67	51.93	51.29	49.90	50.19	50.05	62.29	55.27	58.57
LLaMA3.1-8B	w/ SFT 50% D_{tr}	48.31	48.01	48.16	48.69	46.17	47.39	61.84	54.53	57.96
	w/ SFT 50% D_{tr} + SFT 50% D_{tr}	51.80	49.71	50.74	50.60	48.28	49.41	64.61	58.04	61.15
	w/ SFT 100% D_{tr}	52.61	49.72	51.12	51.14	51.53	51.34	66.05	58.96	62.30
Effectiveness of RL										
Qwen3-8B	w/ SFT + RL *with* DPO	51.47	52.30	51.88	48.59	49.43	49.00	61.63	57.30	59.39
	w/ SFT + RL *with* KTO	52.92	51.94	52.42	49.36	51.34	50.33	61.86	57.86	59.79
	w/ SFT + RL *with* ORPO	51.63	52.61	52.11	49.25	50.57	49.91	62.25	57.30	59.67
WPO (Ours)		**58.23**	49.55	**53.54**	**51.42**	52.11	**51.76**	**64.44**	56.93	**60.45**
Qwen3-4B	w/ SFT + RL *with* DPO	52.46	50.62	51.52	45.25	52.87	48.76	59.46	57.49	58.46
	w/ SFT + RL *with* KTO	53.26	50.90	52.05	47.66	52.68	50.05	62.82	55.27	58.80
	w/ SFT + RL *with* ORPO	52.32	51.38	51.85	49.15	49.62	49.38	62.74	55.08	58.66
WPO (Ours)		**57.98**	48.93	**53.07**	**50.28**	51.92	**51.08**	**63.09**	56.56	**59.65**
LLaMA3.1-8B	w/ SFT + RL *with* DPO	51.72	50.40	51.05	51.90	47.60	49.66	66.88	58.10	62.18
	w/ SFT + RL *with* KTO	52.41	50.76	51.57	52.83	49.27	50.99	65.93	59.18	62.37
	w/ SFT + RL *with* ORPO	51.80	50.85	51.32	50.90	49.99	50.44	65.91	59.08	62.31
WPO (Ours)		**56.35**	49.36	**52.63**	**53.17**	51.34	**52.24**	**68.62**	58.97	**63.43**

Table 3: Comparisons (F1 %) across three complex scenarios on CMeIE-V2 using Qwen3-8B. SEO = single entity overlap, EPO = entity pair overlap, Single = Single-sentence, Cross = Cross-sentence.

Method	Overlapping triplets			Multiple triplets in a sample					Cross-sentence triplets	
	Normal	SEO	EPO	1	2	3	4	≥ 5	Single	Cross
Qwen3-4B										
SFT + DPO	45.81	53.31	50.36	45.90	50.35	50.33	**54.47**	55.09	50.89	52.20
SFT + KTO	46.31	52.95	51.93	46.29	50.65	48.48	52.66	55.44	50.75	52.05
SFT + ORPO	45.85	52.26	54.45	45.81	49.64	47.96	52.05	54.80	49.75	52.05
WPO (Ours)	**47.05**	**54.13**	**55.34**	**47.31**	**50.82**	**50.91**	53.43	**56.53**	**51.45**	**53.71**
Qwen3-8B										
SFT + DPO	45.96	53.36	49.19	46.06	50.42	50.73	53.38	55.39	51.08	52.07
SFT + KTO	45.90	53.10	43.74	46.00	**50.73**	50.94	54.16	54.36	50.44	52.53
SFT + ORPO	46.71	53.27	44.09	46.94	50.24	50.00	53.66	55.36	50.68	53.04
WPO (Ours)	**48.55**	**54.01**	**51.17**	**48.74**	49.87	**52.25**	**54.70**	**56.01**	**51.31**	**54.62**

ther analyze complex samples containing multiple triplets ($N = 1, 2, 3, 4, ...$) in a sample. We observe WPO achieves the best performance in most cases, demonstrating strong robustness to increasing relational complexity. **Analysis on cross-sentence triplet scenario.** We further explore the capabilities of WPO in cross-sentence triplet scenario. Our findings indicate that WPO also outperforms all baseline models in both single-sentence and cross-sentence scenarios. This demonstrates that scenario-level synthesis within our error-centric data synthesis effectively boosts model performance in complex triplet extraction.

5.4 Ablation Study

Table 4: Ablation study on CMeIE-V2. W denotes the weight. D_c and D_{cR} denote hard negatives synthesized based on the error type distribution and randomly, respectively. All settings use 7,000 samples for first-stage training and 3,000 for second-stage training.

Method	Data	Qwen3-4B			Qwen3-8B		
		P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)
# All but one							
w/o Error type distribution	w/ SFT 50% D_{tr} + GRPO (D_{cR} + 50% D_{tr}) W	56.16	49.49	52.62	58.51	48.49	53.03
w/o weights	w/ SFT 50% D_{tr} + GRPO (D_c + 50% D_{tr})	56.40	48.50	52.15	57.40	48.98	52.86
w/o GRPO	w/ SFT 50% D_{tr} + SFT (D_c + 50% D_{tr}) W	51.09	52.41	51.74	51.41	52.80	52.10
w/o Hard negatives	w/ SFT 50% D_{tr} + GRPO (50% D_{tr}) W	54.06	49.49	51.68	54.11	50.58	52.28
# All but two							
w/o Error type distribution & GRPO	w/ SFT 50% D_{tr} + SFT (D_{cR} + 50% D_{tr}) W	50.99	51.83	51.41	51.08	52.06	51.57
w/o Hard negatives & GRPO	w/ SFT 50% D_{tr} + SFT (50% D_{tr}) W	48.75	52.94	50.76	51.14	51.76	51.45
WPO	w/ SFT 50% D_{tr} + GRPO (D_c + 50% D_{tr}) W	**57.98**	48.93	**53.07**	58.23	49.55	**53.54**

Ablation on proposed method. To evaluate the impact of each component, we perform an ablation analysis on WPO. We systematically remove individual components and assess the impact on the overall performance of the remaining combinations. Additionally, we conduct the analysis to remove two key components, with results listed in Table 4. We conclude that removing any component leads to a drop in performance. Specifically, the ablation of the "Hard negative" has the most negative impact in CMeIE-V2, resulting in a drop of 1.39% on Qwen3-4B, 1.26% on Qwen3-4B, respectively. This demonstrates the efficiency of the hard negative sample.

Table 5: Performance comparison (%) across different error types and synthetic data generation settings on CMeIE-V2.

Method	Qwen3-4B			Qwen3-8B		
	P	R	F1	P	R	F1
100% SFT	50.67	51.93	51.29	51.42	52.58	52.00
Error Types						
Task-specific						
w/ SFT + RL with DPO	52.46	50.62	51.52	51.47	52.30	51.88
w/ SFT + RL with KTO	53.26	50.90	52.05	52.92	51.94	52.42
w/ SFT + RL with ORPO	52.32	51.38	51.85	51.63	52.61	52.11
WPO (ours)	**57.98**	48.93	**53.07**	**58.23**	49.55	**53.54**
General						
w/ SFT + RL with DPO	52.47	50.70	51.57	52.94	50.97	51.94
w/ SFT + RL with KTO	52.83	51.08	51.94	53.31	51.47	52.37
w/ SFT + RL with ORPO	51.76	51.42	51.59	52.77	51.62	52.19
WPO (ours)	**55.37**	50.37	**52.75**	**56.07**	51.32	**53.59**
Synthetic Data						
Generated by GPT-4o						
w/ SFT + RL with DPO	52.46	50.62	51.52	51.47	52.30	51.88
w/ SFT + RL with KTO	53.26	50.90	52.05	52.92	51.94	52.42
w/ SFT + RL with ORPO	52.32	51.38	51.85	51.63	52.61	52.11
WPO (ours)	**57.98**	48.93	**53.07**	**58.23**	49.55	**53.54**
Generated by Qwen3-Next-80B						
w/ SFT + RL with DPO	52.23	50.54	51.37	53.07	51.15	52.09
w/ SFT + RL with KTO	52.84	51.31	52.06	53.31	51.37	52.32
w/ SFT + RL with ORPO	52.41	50.85	51.62	53.07	51.32	52.18
WPO (ours)	**56.54**	49.84	**52.98**	**56.18**	50.83	**53.37**

Ablation on training data. We further evaluated two critical factors in training data construction, namely error type definition and the data synthesis model, with experimental results summarized in Table 5. For error types, compared to task-specific types that achieve the highest Precision and F1, general error types—covering long-context comprehension, reasoning, ambiguity, hallucination, and linguistic variants—likewise bring robust gains, proving the scalability of WPO. For synthetic data, although GPT-4o performs slightly better, the fully open-source Qwen3-Next-80B still delivers stable gains and significantly surpasses the KTO baseline, confirming WPO's strong robustness to data sources and its independence from proprietary commercial LLMs.

5.5 Detailed Analysis

Impact of hard negative sample scale. We investigate the impact of negative sample size during RL on CMeIE-V2. As shown in Figure 2, the model achieves optimal performance with 3,000 synthetic hard negative samples. This indicates that insufficient hard negative samples fail to fully exploit model potential, while an excessive amount hinders effective exploration, as noted in prior work (Xu et al., 2025a).

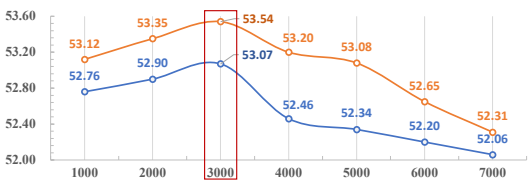


Figure 2: Results of hard sample scale on CMeIE-V2.

Table 6: Error type distribution of Qwen3-8B and Qwen3-4B on CMeIE-V2.

Error Type	Qwen3-8B		Qwen3-8B (WPO)		Qwen3-4B		Qwen3-4B (WPO)	
	Quantity	Ratio (%)	Quantity	Ratio (%)	Quantity	Ratio (%)	Quantity	Ratio (%)
Entity Span Error	9,403	90.85	6,742	86.87	9,383	90.36	6,289	84.87
# subject	5,009	53.27	3,415	50.65	4,714	50.24	3,313	52.68
# object	4,394	46.73	3,327	49.35	4,669	49.76	2,976	47.32
Entity Type Error	274	2.65	203	2.62	296	2.85	200	2.70
# subject	15	5.47	16	7.88	20	6.76	13	6.50
# object	259	94.53	187	92.12	276	93.24	187	93.50
Relation Error	482	4.66	384	4.95	510	4.91	392	5.29
Triplet Not Found	191	1.85	432	5.57	195	1.88	529	7.14
Sum	10,350	100.00	7,761	100.00	10,384	100.00	7,410	100.00

Table 7: Performance on the PMD. Med. = medical accuracy, $\text{Cli}_{Log.}$ = clinical logic, Commu. = communication, Emp. = empathy.

Method	Qwen3-8B					Qwen3-4B				
	Med.	$\text{Cli}_{Log.}$	Com.	Emp.	Avg.	Med.	$\text{Cli}_{Log.}$	Com.	Emp.	Avg.
Baseline LLMs										
No tuning	85.15	77.04	86.25	58.56	79.36	82.14	75.64	83.29	56.73	77.11
w/ SFT 50% D_{tr}	85.22	79.47	87.76	59.16	80.57	84.01	77.15	84.04	57.32	78.60
w/ SFT 50% D_{tr} + SFT 50% D_{tr}	86.23	80.00	88.28	59.89	81.31	84.12	76.99	88.61	57.46	79.05
w/ SFT 100% D_{tr}	86.02	80.47	89.50	61.11	81.66	84.64	78.58	88.75	58.79	80.04
Effectiveness of RL										
w/ SFT + RL with KTO	86.07	79.16	87.92	58.58	80.74	83.85	77.04	85.91	57.68	78.72
WPO (Ours)	**90.90**	**88.52**	**95.51**	**70.74**	**88.39**	**86.04**	**84.43**	**89.78**	**68.26**	**83.99**

Error analysis. For deeper insight into WPO, we analyse its prediction errors on CMeIE-V2, categorising them into entity span errors, entity type errors, relation errors, and triplet not found errors, as detailed in Table 6. Notably, the entity is subdivided into a subject or an object. Our analysis reveals: 1) WPO substantially reduces the number of error samples across all error types for both Qwen3-4B and Qwen3-8B models. 2) The rise in triplet not found cases suggests that the trained model prioritizes precision over recall.

Cross-Task generalization. To validate the generalization of WPO, we conduct cross-task analysis on EE and ABSA tasks in general. As shown in Table 2, WPO achieves SoTA performance on both Qwen3-4B and Qwen3-8B. Specifically, when using Qwen3-4B as the backbone model, WPO surpasses the previous SoTA method KTO by 1.43% and 0.66% in F1 score on the Ace05 and 14Lap, respectively. These results demonstrate the strong generalization ability of our WPO across IE tasks.

Cross-Scenario robustness. We further extend our evaluation to medical dialogue tasks. Unlike the other three tasks, which utilize verifiable rewards, we employ DeepSeek-R1 to assign reward scores across four distinct dimensions for this task. To mitigate the risk of reward hacking (Ding et al., 2025), we adopt GPT-5.1 as a separate judge model for evaluation. As shown in Table 7, WPO consistently achieves higher average scores than baseline methods across both Qwen3-4B and Qwen3-8B models, demonstrating its robustness in complex, non-verifiable scenarios.

6 Conclusion

In this work, we propose the WPO, which utilizes hard negative samples to enhance LLM performance on GIE tasks. Our method automatically expands pre-defined error types and utilizes error type distributions for data construction, along with a weighting mechanism for training. Extensive experimental analysis validates the cross-task generalization and cross-scenario robustness of WPO.

Limitations. Our method currently relies on a single-round error assessment, where the error-type distribution is estimated once to guide subsequent training. Future work will explore iterative re-evaluation and dynamic updating of the error distribution across multiple training rounds.

Ethics Statement

To ensure patient privacy, the PMD dataset contains no personally identifiable information (e.g., names or hospital details), eliminating any risk of privacy breaches. All data usage complies with applicable medical research ethics and regulatory standards.

Reproducibility Statement

To ensure reproducibility, we publicly release our code at an anonymous URL (`https://anonymous.4open.science/r/WPO-ECD8`). The datasets, hyperparameters, and evaluation metrics are described in detail in the Experiments section and the appendix.

AI Use Statement

In this work, we used generative AI tools for language polishing of the manuscript. We have reviewed all AI-assisted work and take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Shengnan An, Zexiong Ma, Zeqi Lin, Nanning Zheng, Jian-Guang Lou, and Weizhu Chen. Learning from mistakes makes llm better reasoner. *arXiv preprint arXiv:2310.20689*, 2023.

Xiaoyi Bao, Zhongqing Wang, Jinghang Gu, and Chu-Ren Huang. Revisiting classical chinese event extraction with ancient literature information. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8440–8451, 2025.

Zheng Cai, Maosong Cao, Haojiong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui Chen, Zhi Chen, Pei Chu, et al. Internlm2 technical report. *arXiv preprint arXiv:2403.17297*, 2024.

Jiale Cheng, Yida Lu, Xiaotao Gu, Pei Ke, Xiao Liu, Yuxiao Dong, Hongning Wang, Jie Tang, and Minlie Huang. Autodetect: Towards a unified framework for automated weakness detection in large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 6786–6803, 2024.

Darshan Deshpande, Varun Gangal, Hersh Mehta, Jitin Krishnan, Anand Kannappan, and Rebecca Qian. Trail: Trace reasoning and agentic issue localization. *arXiv preprint arXiv:2505.08638*, 2025.

Meidan Ding, Jipeng Zhang, Wenxuan Wang, Cheng-Yi Li, Wei-Chieh Fang, Hsin-Yu Wu, Haiqin Zhong, Wenting Chen, and Linlin Shen. Med-rewardbench: Benchmarking reward models and judges for medical multimodal large language models. *arXiv preprint arXiv:2508.21430*, 2025.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Model alignment as prospect theoretic optimization. In *Forty-first International Conference on Machine Learning*, 2024. URL `https://openreview.net/forum?id=iUwHnoENnl`.

Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Dan Zhang, Diego Rojas, Guanyu Feng, Hanlin Zhao, et al. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *arXiv preprint arXiv:2406.12793*, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

Jiwoo Hong, Noah Lee, and James Thorne. ORPO: Monolithic preference optimization without reference model. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 11170–11189, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.626. URL https://aclanthology.org/2024.emnlp-main.626/.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Peixin Huang, Xiang Zhao, Minghao Hu, Zhen Tan, and Weidong Xiao. Logic induced high-order reasoning network for event-event relation extraction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 24141–24149, 2025.

Minhao Jiang, Ken Ziyu Liu, Ming Zhong, Rylan Schaeffer, Siru Ouyang, Jiawei Han, and Sanmi Koyejo. Investigating data contamination for pre-training language models. *arXiv preprint arXiv:2401.06059*, 2024.

Ryo Kamoi, Sarkar Snigdha Sarathi Das, Renze Lou, Jihyun Janice Ahn, Yilun Zhao, Xiaoxin Lu, Nan Zhang, Yusen Zhang, Haoran Ranran Zhang, Sujeeth Reddy Vummanthala, Salika Dave, Shaobo Qin, Arman Cohan, Wenpeng Yin, and Rui Zhang. Evaluating LLMs at detecting errors in LLM responses. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=dnwRScljXr.

Xiao Liang, Zhong-Zhi Li, Yeyun Gong, Yang Wang, Hengyuan Zhang, Yelong Shen, Ying Nian Wu, and Weizhu Chen. Sws: Self-aware weakness-driven problem synthesis in reinforcement learning for llm reasoning. *arXiv preprint arXiv:2506.08989*, 2025.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=HPuSIXJaa9.

Yuhong Sun, Zhangyue Yin, Xuanjing Huang, Xipeng Qiu, and Hui Zhao. Error classification of large language models on math word problems: A dynamically adaptive framework. *arXiv preprint arXiv:2501.15581*, 2025.

Nguyen Xuan Thanh, Anh Duc Le, Quyen Tran, Thanh-Thien Le, Linh Ngo Van, and Thien Huu Nguyen. Few-shot, no problem: Descriptive continual relation extraction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 25282–25290, 2025.

Yongqi Tong, Dawei Li, Sizhe Wang, Yujia Wang, Fei Teng, and Jingbo Shang. Can llms learn from previous mistakes? investigating llms' errors to boost for reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3065–3080, 2024.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. Ace 2005 multilingual training corpus. *Linguistic Data Consortium, Philadelphia*, 57:45, 2006.

Kaishuai Xu, Tiezheng Yu, Wenjun Hou, Yi Cheng, Chak Tou Leong, Liangyou Li, Xin Jiang, Lifeng Shang, Qun Liu, and Wenjie Li. Subtle errors in reasoning: Preference learning via error-injected self-editing. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 31184–31203, 2025a.

Lu Xu, Hao Li, Wei Lu, and Lidong Bing. Position-aware tagging for aspect sentiment triplet extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2339–2349, 2020.

Zishan Xu, Shuyi Xie, Qingsong Lv, Shupei Xiao, Linlin Song, Sui Wenjuan, and Fan Lin. Diagnosing failures in large language models' answers: Integrating error attribution into evaluation framework. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 21148–21165, 2025b.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Zhiyuan Zeng, Yizhong Wang, Hannaneh Hajishirzi, and Pang Wei Koh. Evaltree: Profiling language model weaknesses via hierarchical capability trees. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=aV2hQN9vkp.

Zhongshen Zeng, Yinhong Liu, Yingjia Wan, Jingyao Li, Pengguang Chen, Jianbo Dai, Yuxuan Yao, Rongwu Xu, Zehan Qi, Wanru Zhao, et al. Mr-ben: A meta-reasoning benchmark for evaluating system-2 thinking in llms. *Advances in Neural Information Processing Systems*, 37: 119466–119546, 2024.

Ningyu Zhang, Mosha Chen, Zhen Bi, Xiaozhuan Liang, Lei Li, Xin Shang, Kangping Yin, Chuanqi Tan, Jian Xu, Fei Huang, et al. Cblue: A chinese biomedical language understanding evaluation benchmark. In *ACL*, pp. 7888–7915, 2022.

Zikang Zhang, Wangjie You, Tianci Wu, Xinrui Wang, Juntao Li, and Min Zhang. A survey of generative information extraction. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 4840–4870, 2025.

Jiayuan Zhu, Jiazhen Pan, Yuyuan Liu, Fenglin Liu, and Junde Wu. Ask patients with patience: Enabling llms for human-centric medical dialogue with grounded reasoning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 2846–2857, 2025.

A Novelty and Key Differences: WPO vs. SwS, RISE

WPO is a novel framework purpose-built for GIE, advancing beyond SwS and RISE through three core innovations.

1) Error Mining: while baselines rely on static or coarse-grained predefined categories, WPO utilizes automated dynamic induction to directly capture fine-grained, GIE-specific errors from failure samples.

2) Data Synthesis: unlike the approaches of SwS (positive samples only) and RISE (negative samples only), WPO generates high-quality positive and negative samples based on the mined errors to establish strict extraction boundaries.

3) Model Training: rather than applying standard GRPO/DPO with generic rewards, WPO introduces a sample weighting mechanism to address schema complexity imbalances and designs a multi-dimensional reward function explicitly tailored to evaluate structural integrity and extraction precision.

B Considering the Computational Cost of GPT-5.1

Our WPO module is not strictly bound to GPT-5.1. Lightweight models can achieve comparable results at a negligible cost. To demonstrate this, we evaluated the error mining task on 100 samples. While GPT-5.1 achieves 98% accuracy at $1.94, replacing it with 4o mini yields a highly competitive 95% accuracy while drastically reducing the cost to $0.19. Even a smaller model like Qwen3-8B maintains 87% accuracy at just $0.08. This 90% cost reduction with only a 3% drop in accuracy confirms that the performance gain can be effectively sustained at a negligible cost for real-world deployments.

C Sensitivity analysis of weights

To further validate the effectiveness of the weights, we conduct sensitivity analysis on CMeIE-V2. In detail, we compare four strategies: remove any one weight (Any one=0), remove any two weights (Any two=0), change only one with others fixed (Fix two) and remove all weights (ALL=0/1). As shown in Table A1, we observed that the selected weights achieves the best performance on CMeIE-V2 for both Qwen3-4B and Qwen3-8B.

Table A1: Sensitivity analysis of weights on CMeIE-V2.

Strategy	**Weights**			**Qwen3-4B**			**Qwen3-8B**		
	W_A	W_I	W_S	**P (%)**	**R (%)**	**F1(%)**	**P (%)**	**R (%)**	**F1(%)**
Any one = 0	[0.1, 0.9]	0	0.1	57.34	48.56	52.59	55.50	50.93	53.11
	[0.1, 0.9]	0.5	0	55.97	49.70	52.65	58.31	49.02	53.27
	0	0.5	0.1	56.97	48.82	52.58	57.47	49.18	53.00
Any two = 0	[0.1, 0.9]	0	0	56.39	48.81	52.33	56.87	49.72	53.06
	0	0	0.1	56.96	48.03	52.11	59.80	47.04	52.66
	0	0.5	0	57.14	48.46	52.45	57.57	48.83	52.84
Fix two	[0.1, 0.9]	0.7	0.1	56.08	49.14	52.38	57.16	49.77	53.21
	[0.1, 0.9]	0.3	0.1	57.01	48.39	52.35	57.40	49.68	53.26
	[0.1, 0.9]	0.5	0.5	56.97	49.11	52.75	58.11	49.19	53.28
	[0.1, 0.9]	0.5	0.9	56.09	48.92	52.26	57.17	49.31	52.93
	[0.2, 0.8]	0.5	0.1	56.21	49.56	52.67	57.25	49.60	53.15
	[0.4, 0.6]	0.5	0.1	56.60	49.37	52.74	56.81	50.24	53.33
All=0	0	0	0	54.06	49.49	51.68	54.11	50.58	52.28
All=1	1	1	1	56.40	48.50	52.15	57.40	48.98	52.86
WPO	[0.1, 0.9]	0.5	0.1	**57.98**	48.93	**53.07**	58.23	49.55	**53.54**