From General Reward to Targeted Reward: Improving Open-ended Long-context Generation Models

Zhihan Guo[1], **Jiele Wu**[2], **Wenqian Cui**[1], **Yifei Zhang**[3,*]
Minda Hu[1], **Yufei Wang**[4], **Irwin King**[1,*]
[1]The Chinese University of Hong Kong, Hong Kong SAR, China
[2]National University of Singapore, Singapore, Singapore
[3]Nanyang Technological University, Singapore, Singapore
[4]Macquarie University, Sydney, Australia
{zhguo22,king}@cse.cuhk.edu.hk, yifeiacc@gmail.com

Abstract

Current research on long-form context in Large Language Models (LLMs) primarily focuses on the understanding of long-contexts, the Open-ended Long Text Generation (Open-LTG) remains insufficiently explored. Training a long-context generation model requires curation of gold-standard reference data, which is typically nonexistent for informative Open-LTG tasks. However, previous methods only utilize general assessments as reward signals, which limits accuracy. To bridge this gap, we introduce **ProxyReward**, an innovative reinforcement learning (RL) based framework, which includes a dataset and a reward signal computation method. Firstly, **ProxyReward Dataset** generation is accomplished through simple prompts that enables the model to create automatically, obviating extensive labeled data or significant manual effort. Secondly, **ProxyReward Signal** offers a targeted evaluation of information comprehensiveness and accuracy for specific questions. The experimental results indicate that our method ProxyReward surpasses even GPT-4-Turbo. It can significantly enhance performance by 20% on the Open-LTG task when training widely used open-source models, while also surpassing the LLM-as-a-Judge approach. Our work presents effective methods to enhance the ability of LLMs to address complex open-ended questions posed by humans. The code is available at: https://github.com/zhihan-guo/ProxyReward/.

1 Introduction

Open-ended Long Text Generation (Open-LTG) represents a significant challenge in the field of large language model (LLM) research(Kumar et al., 2024; Lee et al., 2022), owing to its inherent openness and complexity (Li et al., 2023; Sudhakaran et al., 2023; Brown et al., 2020; Touvron et al., 2023). While current research has made substantial progress in enhancing LLMs' ability to understand long contexts (Wu et al., 2024b), the complementary challenge of generating coherent, informative, and contextually grounded outputs spanning thousands of tokens remains critically underexplored (Bai et al., 2024c; Li et al., 2024a,b; Liu et al., 2025). As illustrated in Figure 1, designing targeted training reward signals represents an essential component in addressing this challenge.

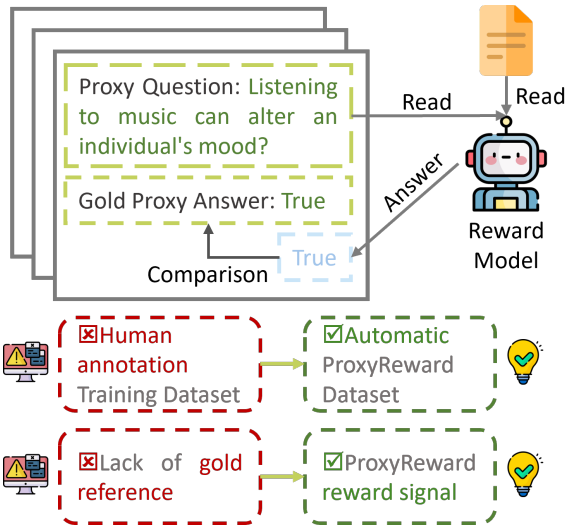

Figure 1: Illustration of the ProxyReward model and its two advantages: automatic over human data label annotation and the existence of reward signals for Open-LTG problems.

The Open-LTG task faces several fundamental obstacles that have impeded progress in this domain (Que et al., 2024; Liu et al., 2024b). Unlike traditional LLM generation tasks which have clear reference answers, Open-LTG lacks standard answers due to the inherent complexity of information in long texts (Köksal et al., 2023). Consequently, Open-LTG responses do not follow a single pattern, resulting in insufficient reward signals during model training (Zhang et al., 2024). Moreover, annotation of Open-LTG data requires

*Corresponding author

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 5152–5167
November 4-9, 2025

the creation of substantial high-quality long-form content (Pham et al., 2024), traditionally demanding domain experts with extensive knowledge, leading to inefficiency and prohibitive costs (Micheletti et al., 2024). Thirdly, experts' subjective preferences regarding long-text quality often result in inconsistent evaluations (Tan et al., 2024; Bai et al., 2024b; Guo et al., 2024a), while the diversity of long-form content further complicates efforts to translate subjective assessments to objective evaluation metrics (Xu et al., 2023; Krishna et al., 2023).

To bridge this critical gap, this paper introduces ProxyReward, an innovative reinforcement learning (RL)-based framework designed specifically for open-ended long-context generation. The framework consists of two key components. First, the ProxyReward dataset includes 9,271 long-form meta-questions across multiple domains and associated multiple proxy question-answer pairs generated via simple prompts. This dataset is constructed automatically without the need of extensive labeled data or significant manual effort. Second, the ProxyReward signal transforms the subjective and challenging task of evaluating long-form content quality into a long-context understanding task at which has been fully explored on the long-context LLMs (Li et al., 2024b; Guo et al., 2024b). This approach effectively provides targeted reward signals for more effective model training.

The central innovation of ProxyReward lies in its unique (meta-question, proxy question-answer pairs) structure. For each meta-question, the proxy question-answer pairs comprise multiple questions and corresponding boolean answers, which similar to reading comprehension tests. Our approach leverages these proxy question-answer pairs to derive informative reward signals for reinforcement learning, eliminating the need for gold standard responses and encouraging the generation of more informative and comprehensive outputs. By transforming subjective evaluation of long-form content as an objective assessment task grounded in the reading comprehension abilities of language models, ProxyReward effectively addresses the core challenges of Open-LTG tasks.

Our main contributions are as follows:

- The ProxyReward Dataset is constructed through a simple, scalable process without relying on predefined response patterns, eliminating the need for costly and labor-intensive supervised annotation.
- The ProxyReward serves as a targeted reward mechanism. It leverages the long-context understanding capabilities of LLMs to generate informative reward signals that guide model optimization without relying on traditional gold references.
- The ProxyReward dataset is also a multi-objective RL benchmark. It features dozens of fine-grained proxy checks for each meta-question, designed to assess coverage and accuracy in Open-LTG.
- Experiments demonstrate that ProxyReward significantly enhances performance on the ProxyQA benchmark (increase 20%) applied to Qwen and Llama models, and even surpassing GPT-4-Turbo. This validates its effectiveness for the Open-LTG task.

2 Related Work

2.1 Long-context Language Model

Long-context language models are designed to overcome the context length limitations of language models, enabling them to support a wide range of long context tasks effectively (Xiong et al., 2023; Ma et al., 2024; Guo et al., 2025b). A prominent line of research focuses on improving the transformer architecture through efficient attention mechanisms (Tay et al., 2020, 2022; Zaheer et al., 2020; Jiang et al., 2024), structured state space models (Gu et al., 2021; Poli et al., 2023) or memory recurrent (Bulatov et al., 2022), in order to alleviate the context length limitations. However, these approaches often involve significant approximations that deviate from full attention, making them less compatible with fine-tuning pre-trained large language models (Ma et al., 2024). Another active direction involves extending the context window via continual pre-training and supervised fine-tuning on longer sequences (Xiong et al., 2024; Peng et al., 2024; Chen et al., 2023; Fu et al., 2024; Bai et al., 2024a). While these methods typically requires higher computational costs, they generally achieve superior performance on a variety of long-context tasks. More recently, (Jin et al., 2025) combine policy network with multi-turn search engine calling have shown significant success. In this work, we focus on long-context generation task by designing a targeted reward signal.

Algorithm 1: Synthetic preference alignment pipeline

Input: Reference model $\mathcal{G}_{ref}$, preference dataset $\mathcal{D}$, iterations W

for $w = 1, 2, \ldots, W$ **do**
  $\mathcal{D}_w \leftarrow w$-th partition of $\mathcal{D}$;
  for $m \in \mathcal{D}_w$ **do**
    Generate response r using $\mathcal{G}_{ref}(m)$;
  $\{y, y_w, y_l\} \leftarrow$ Rank responses by preference;
  $\mathcal{D}_w \leftarrow \{(m, y_w, y_l) \mid m \in \mathcal{D}_w\}$;
  $\mathcal{G}_{\theta_t} \leftarrow \arg\min_\theta \mathcal{L}(\theta)$ as defined in Equation 2;
  where r^*_θ is given by Equation 3;
  $\mathcal{G}_{ref} \leftarrow \mathcal{G}_{\theta_t}$;

2.2 Improving LLM with AI Feedback

Reinforcement learning from human feedback (RLHF) is crucial for aligning LLMs with human values, enables them to pursue diverse goals by learning from human feedback (Ouyang et al., 2022; Yuan et al., 2023; Rafailov et al., 2023; Song et al., 2024; Zhang et al., 2025). However, collecting high-quality pairwise human preference data is both expensive and time-consuming (Bai et al., 2024a; Wu et al., 2024a). To address this, synthetic preference data generated by LLMs presents a promising alternative, offering scalability at a significantly lower cost. Following this direction, (Bai et al., 2024a) first introduced LLM-generated critiques for evaluating whether model outputs are harmful, using human-annotated harmful prompts as a reference. (Dubois et al., 2023) further leveraged API-based LLMs to select preferred model responses, reducing human involvement. Reinforcement Learning with AI Feedback (RLAIF) (Lee et al., 2024) extends this idea by using another LLM as a verifier to approximate human judgment. Building on this, (Yang et al., 2024) later found that using better prompts (self-improve) that direct harmful or harmless responses can surpass RLAIF. More recently, (Yuan et al., 2024) demonstrated that combining iterative fine-tuning with high-quality prompts generated via in-context learning can yield surprisingly strong performance.

2.3 LLM-as-a-Judge

Before the era of LLMs, striking a balance between comprehensive and scalable evaluation remained a long-standing challenge (Gu et al., 2024; Wang et al., 2024). Subjective methods such as expert-driven assessments (Gao et al., 2023; Shi et al., 2024) have long been considered the gold standard due to their ability to provide holistic reasoning and fine-grained contextual understanding. These approaches are costly, difficult to scale, and often suffer from inconsistency. In contrast, objective evaluation methods, such as automatic metrics offer strong scalability and consistency (Papineni et al., 2002; Lin, 2004). These metrics rely heavily on surface-level lexical overlaps, making them difficult to evaluate outputs that require deeper semantic understanding (Schluter, 2017). The "LLM-as-a-Judge" paradigm has emerged as a promising alternative that combines the advantages of both paradigms: the contextual understanding of human evaluation and the scalability of automated metrics (Dubois et al., 2023; Fernandes et al., 2023; Bai et al., 2023). Studies have also used "LLM-as-a-Judge" to train reward models and curate preference data (Lee et al., 2024; Chen et al., 2024; Li et al., 2024c). However, previous methods mostly utilizing provide general assessments as reward signal, which often lack accuracy and specificity.

3 Preliminaries

3.1 Preference Alignment

Let $\mathcal{D} = \{(x, y^+, y^-)\}$ denote a dataset of preferences, where x is an input prompt, y^+, y^- are the responses labeled as preferred and dis-preferred, respectively. The purpose of preference alignment is to designing a policy π that maps prompts to responses, maximizing a reward that reflects human preferences using the Bradley–Terry (BT) model:

$$p(y_1 \succ y_2|x) = \sigma\left(r^*(x, y_1) - r^*(x, y_2)\right), \quad (1)$$

where $r^*(x, y_1)$ represents the oracle reward of a response given a prompt, and $\sigma(z) = \{1 + exp(-z)\}^{-1}$ is the sigmoid function, mapping differences in rewards to probabilities. A parameterized reward model r_θ is estimated by solving a maximum likelihood estimation (MLE) objective:

$$\mathcal{L}(\theta) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}[\log \sigma(r^*_\theta(x, y_w) - r^*_\theta(x, y_l)], \quad (2)$$

where y_w, y_l are preferred and dis-preferred sample respectively. The direct preference optimization (DPO) (Rafailov et al., 2023) we used in this work chose the reward as:

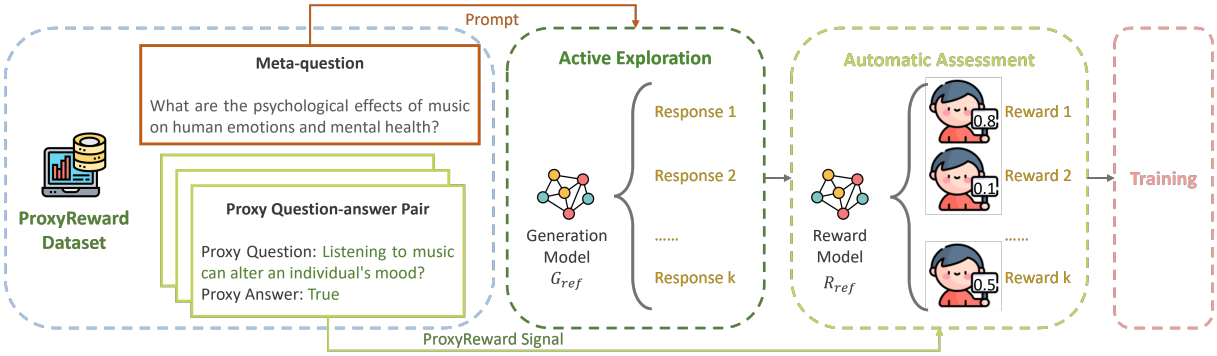

Figure 2: ProxyReward overview. Our framework operates in three stages: First, ProxyReward automatically generates a training dataset comprising meta-questions and their corresponding proxy question-answer pairs. Second, a generation model actively explores k diverse responses based on these meta-questions. Third, a reward model automatically assesses these responses using the corresponding proxy question-answer pairs to generate reward signals, which are subsequently utilized during model training.

$$r_\theta(x, y) = \beta \log \frac{\pi_\theta(y \mid x)}{\pi_{\text{ref}}(y \mid x)}, \tag{3}$$

to directly optimize the policy π_θ using the loss $L(\theta)$ in Equation 2, with the reward function r specified by Equation 3. As the reward is implicitly defined by the policy itself, the objective becomes fully dependent on θ eliminating the need for a separately trained reward model. Consequently, this reformulation significantly improves the computational efficiency of the alignment process.

3.2 Synthetic Preference Alignment Pipeline

Given the generation policy $\mathcal{G}$ parameterized by θ and an LLM-based reward model $\mathcal{R}$. As shown in Table 1, the synthetic preference alignment pipeline typically consists of the following stages:

Response Generation. Given a dataset of prompts $\mathcal{X} = \{x_1, \cdots, x_n\}$, the policy $\mathcal{G}_\theta$ generate a set of responses $\{y_i^1, y_i^2, \cdots\}$ which are intended to cover diverse output patterns for each prompt x_i.

AI-based Reward Assignment. For each response y_i^j, reward score $r(x_i, y_i^j)$ is calculated by reward model $\mathcal{R}$, which acts as an automatic evaluator.

Policy Optimization. The policy $\mathcal{G}_\theta$ is then fine-tuned using the synthetic reward signal. DPO are commonly used to align the policy with the feedback provided by reward model $\mathcal{R}$.

4 Methodology

In this section, we introduce ProxyReward, a novel reinforcement learning (RL)-based framework for effective long-context generation. As shown in Figure 2, our approach comprises three key components. First, an automatically constructed **ProxyReward Dataset** (Section 4.1) that utilizes LLMs to generate diverse meta-questions with corresponding proxy question-answer pairs, avoiding fixed patterns; Second, an **Active Exploration** mechanism (Section 4.2) that generates various long-form contents; Third, an **Automatic Assessment** system (Section 4.3) that provides targeted reward signals to guide the optimization process. This iterative process enables the model to improve by exploring, evaluating, and incorporating superior response patterns across multiple iterations. By combining automated data construction with RL-based optimization, ProxyReward efficiently addresses the challenges of Open-LTG task.

4.1 ProxyReward Dataset Collection

One challenge of Open-LTG is that responses do not follow a fixed pattern, making it difficult to establish standard answers when constructing the training dataset. To address this issue, we designed a ProxyReward dataset to facilitate training data generation. This dataset consists of meta-questions set $M = \{m_1, m_2, ..., m_n\}$, where each meta-question requires lengthy context to answer thoroughly. For each meta-question m_i, we develop a corresponding list of proxy question-answer pairs represented as $P_i = \{(q_{i1}, a_{i1}), (q_{i2}, a_{i2}), ..., (q_{il}, a_{il})\}$. The main idea is to transform subjective expert evaluations of long-form text quality into objective reading comprehension questions that can be automatically assessed by LLMs.

These proxy question-answer pairs (q, a) are directly related to information covered by the meta-questions M. The proxy questions resemble read-

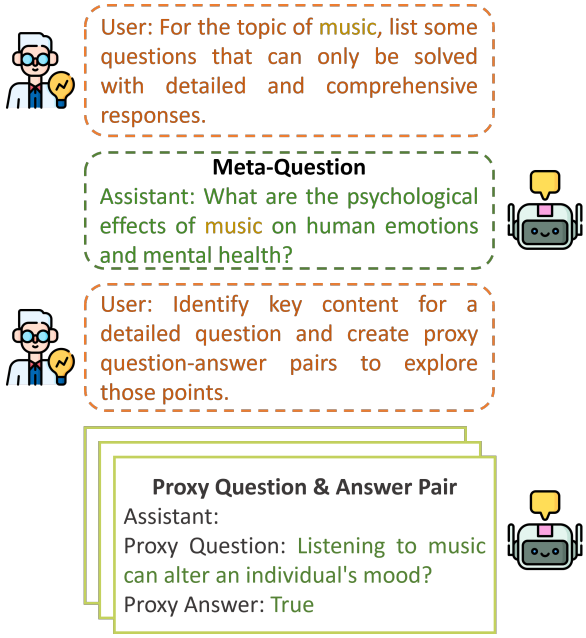

Figure 3: The pipeline of meta-questions and Proxy Question-Answer pairs generation in ProxyReward.

ing comprehension questions Q, with each one designed to assess a key information point that should be included in the long-form response text t. All Proxy answers are formulated as boolean values. We utilize these proxy question-answer pairs (q, a) as an objective checklist to quantify the information content of long-form responses. We ensure that most proxy questions have "True" as their expected answer, as "False" answers would indicate that the information in the proxy-question q is not sufficiently relevant to the meta-question m.

The construction method for the ProxyReward Dataset is straightforward. Instead of traditional manual annotation, we use a simple prompt to automatically generate data through LLM API calls. We begin by manually selecting 40 domains that typically require long-form answers, including Computer Science, Technology, History, Game, Policy, and others. As shown in Figure 3, the automatic data generation process using the LLM API consists of two key steps: First, for each domain, we prompted the LLM API to generate 9,271 meta-questions that require long-form contextual answers. Second, for each meta-question, we instructed the LLM API to generate approximately 15 boolean proxy question-answer pairs. The prompts details are shown in Appendix B. Detailed specifications of the scoring dataset are presented in Table 1.

Table 1: Statistics of the ProxyReward Dataset.

Dataset	ProxyReward
Domain	40
Meta-Question	9,271
Proxy Q&A	156,506
Metric	Proxy Accuracy

4.2 Active Exploration

A significant challenge in Open-LTG is the scarcity of large-scale human-annotated preference data exhibiting diverse patterns. To address this limitation, we implement Active Exploration, leveraging two distinct language models: a generation model $\mathcal{G}$ and a reward model $\mathbb{R}$. This approach proceeds in two phases. First, we extract a meta-question m from the ProxyReward Dataset and input it to our generation LLM, which produces k different responses $T_i = r_1, r_2, ..., r_k$. Second, we feed both the QA list from the Scoring Dataset and the k responses T_i into a higher-performing reward LLM. This reward model comprehends all responses t_{ik} and answers all questions q_{ij}, outputting an answer list $A_{ik} = a_{i1}, a_{i2}, ..., a_{ik}$. Through this process, we efficiently create a large-scale training dataset for Open-LTG that contains diverse patterns and high-quality annotations.

4.3 Automatic Assessment

To incorporate expert preferences while minimizing annotation costs, we introduce automatic assessment as the reward signal. Specifically, we hypothesize that the quality of a long-form response r_i is optimal when it enables a language model to accurately answer all associated proxy questions $P_i = \{(\hat{q}_{ij}, \hat{a}_{ij})\}_{j=1}^{l}$. A response r_i is deemed thorough and informative if it facilitates correct answers to all proxy questions, the prompt detail as shown in Appendix B. Conversely, if r_i fails to correctly answer a significant portion of the proxy questions, it lacks the necessary information and coherence required for a high-quality long-form response. Building on this intuition, we design the reward signal as a scoring function $\mathcal{S}(r_i)$ that quantifies the informativeness of r_i. The reward function is defined as:

$$\mathcal{S}(r_i) = \frac{\sum_{j=1}^{l} \mathcal{F}(a'_{ij}, \hat{a}_{ij})}{l}, \tag{4}$$

where $\mathcal{F}(a'_{ij}, \hat{a}_{ij})$ is a binary function that compares the predicted answer a'_{ij} for proxy question

Table 2: Comparison of Accuracy (%) with Standard Deviation (%) of ProxyQA across different settings. Each model runs 3 times inference and evaluation.

Model	Base	LLM-as-a-Judge	ProxyReward (Iter 1)	ProxyReward (Iter 2)
Qwen2.5-1.5B-Instruct	18.30$_{\pm 0.27}$	12.88$_{\pm 0.35}$	**22.10**$_{\pm 0.25}$	21.54$_{\pm 0.90}$
Qwen2.5-3B-Instruct	27.00$_{\pm 0.29}$	16.69$_{\pm 0.42}$	27.58$_{\pm 0.33}$	**28.54**$_{\pm 0.32}$
Qwen2.5-7B-Instruct	32.37$_{\pm 0.12}$	25.73$_{\pm 0.24}$	33.23$_{\pm 0.45}$	**35.07**$_{\pm 0.10}$
Llama-3.2-1B-Instruct	18.49$_{\pm 0.18}$	11.84$_{\pm 0.58}$	19.97$_{\pm 0.32}$	**20.08**$_{\pm 0.41}$
Llama-3.2-3B-Instruct	25.16$_{\pm 0.32}$	21.16$_{\pm 0.26}$	26.96$_{\pm 0.30}$	**28.59**$_{\pm 0.32}$
Llama-3.1-8B-Instruct	25.02$_{\pm 0.11}$	23.43$_{\pm 0.33}$	29.38$_{\pm 0.24}$	**30.11**$_{\pm 0.57}$

$\hat{q}_{ij}$ with the reference answer $\hat{a}_{ij}$:

$$\mathcal{F}(a'_{ij}, \hat{a}_{ij}) = \begin{cases} 1, & \text{if } a'_{ij} = \hat{a}_{ij}, \\ 0, & \text{if } a'_{ij} \neq \hat{a}_{ij}. \end{cases} \quad (5)$$

Here, $a'_{ij} = R_{ref}(t_i, \hat{q}_{ij})$ represents the predicted answer generated by the reward model R_{ref} when evaluating the response r_i against the proxy question $\hat{q}_{ij}$. The reference answer $\hat{a}_{ij}$ serves as the ground truth for $\hat{q}_{ij}$.

The function $\mathcal{S}(r_i)$ measures the proportion of correctly answered proxy questions, replacing rigid evaluation metrics for long-form text quality with a more adaptable framework. This flexible approach enables the reward signal to effectively steer the reinforcement learning process, fostering the generation of responses that are not only informative but also deeply aligned with the context of the tasks.

5 Experiment

5.1 Setup

Baseline Models For trainable baselines, we use Llama (including Llama-7B (Touvron et al., 2023), Llama-2-7B (Touvron et al., 2023), Llama-2-13b (Touvron et al., 2023), Llama-3.2-1B-Instruct (Grattafiori et al., 2024), Llama-3.2-3B-Instruct (Grattafiori et al., 2024) and Llama-3.1-8B-Instruct (Grattafiori et al., 2024)), and Qwen Instruct (including Qwen2.5-1.5B-Instruct (Team, 2024), Qwen2.5-3B-Instruct (Team, 2024), and Qwen2.5-7B-Instruct (Team, 2024)) as backbone models. Training-free baselines include GPT (including GPT-3.5-Turbo, GPT-4, GPT-4-Turbo, GPT-4o-mini (OpenAI, 2024a) and GPT-4o (OpenAI, 2024b)), DeepSeek (including DeepSeek-V3 (Liu et al., 2024a) and DeepSeek-R1 (Guo et al., 2025a)), Bing (Bing, 2023). We also compare out method with LLM-as-a-Judge (Gu et al., 2024). The LLM-as-a-Judge prompt is shown in Appendix B.

Evaluation and Metrics We assess performance using the ProxyQA benchmark (Tan et al., 2024), an innovative dataset designed for evaluating long-text generation. ProxyQA consists of in-depth, human-curated meta-questions across various domains, each accompanied by specific proxy questions and pre-annotated answers.

Model Settings We utilize the GPT-4o-mini API to construct the ProxyReward dataset. All methods are trained using consistent hyperparameters across the board. Training is conducted on $4 \times L40$ GPUs. The learning rate for DPO is set at 5e-7, with a batch size of 2 and a gradient accumulation step of 8. The maximum completion length is 2048 tokens, and we train for 5 epochs. To enhance the diversity of the outputs generated by the LLMs, we implement a temperature setting of 0.8 for the trainable models. For the reward model, we employ GPT-4o-mini (OpenAI, 2024a) to compute training rewards. For evaluation purposes, we utilize GPT-4o (OpenAI, 2024b) to determine the ProxyQA score.

5.2 How does ProxyReward Performance Compare to Trainable Baselines?

Our proposed ProxyReward demonstrated substantial improvements on open-source models. The experimental results clearly demonstrate the effectiveness of our ProxyReward method across different model architectures and parameter scales. As shown in Table 2, ProxyReward consistently outperforms both the base models across all tested configurations. Notably, when applied to the Qwen2.5-7B-Instruct model, our method achieves a remarkable score of 35.07, surpassing even GPT-4-Turbo (33.94) which is a significantly larger proprietary

Table 3: Comparison of ProxyQA Accuracy (%) between ProxyReward and closed-source LLMs. We use *Iter 2* ProxyReward results for both Qwen and Llama.

Model	ACC
Qwen2.5-7B-Instruct	**35.07**
Llama-3.1-8B-Instruct	30.11
GPT-3.5-Turbo	23.94
GPT-4-0613	27.19
GPT-4-Turbo	33.94
GPT-4o-mini	37.57
GPT-4o	44.98
DeepSeek-V3	42.93
DeepSeek-R1	**48.65**
ReAct (GPT-4)	17.15
ReAct (GPT-4-Turbo)	21.19
Bard (Gemini Pro)	25.00
New Bing (Creative Mode)	39.37

model. This performance superiority highlights the efficiency of our approach in enhancing model capabilities without requiring the massive computational resources typically associated with larger models.

The improvement pattern is consistent across different model scales. For smaller models like Qwen2.5-1.5B-Instruct, ProxyReward boosts performance from 18.30 to 22.10, representing a 20.8% relative improvement. Similarly, for medium-sized models such as Qwen2.5-3B-Instruct, our method increases the score from 27.00 to 28.54. The enhancement is equally evident in the Llama series, where Llama-3.1-8B-Instruct shows a substantial improvement from 25.02 to 30.11, demonstrating a 20.3% relative gain.

These results validate that ProxyReward provides an efficient framework for enhancing model performance across architectures and scales, offering a viable approach to achieve state-of-the-art performance with smaller models.

Our analysis reveals that long-form texts in scientific domains (computer science, medicine) consistently receive higher scores than those in humanities (literature, history). This discrepancy arises because scientific questions typically have more standardized answers with clear evaluation criteria, while humanities questions involve more subjective interpretation and diverse perspectives.

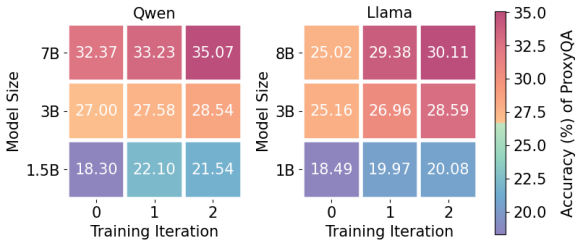

Figure 4: Effects of model size and iteration on ProxyQA score for 3×3.

5.3 How does ProxyReward Performance Compare to LLM-as-a-Judge Performance?

Compared to the LLM-as-a-Judge approach, the ProxyReward signal offers a targeted evaluation of information comprehensiveness and accuracy for specific questions, resulting in higher performance. To evaluate the effectiveness of our proposed ProxyReward signal, we conducted comparative experiments against the conventional LLM-as-a-Judge approach using Qwen2.5 models at 1.5B, 3B and 7B parameter scales. The results demonstrate a substantial performance advantage for ProxyReward across all model sizes.

As shown in Table 2, across both Qwen and Llama families, ProxyReward outperforms LLM-as-a-Judge at all parameter scales. For Qwen2.5 models (1.5B, 3B, 7B), ProxyReward improves absolute scores by 9.22, 11.85, and 9.34 points, corresponding to relative gains of 71.6%, 71.1%, and 36.3%. Similarly, for Llama models (1B, 3B, 8B), ProxyReward achieves gains of 1.59, 3.43, and 5.09 points, with relative improvements of 8.6%, 13.6%, and 20.4%.

These results confirm our hypothesis that unlike LLM-as-a-Judge approaches, which provide general assessments of text quality, the ProxyReward signal delivers more targeted evaluation of information comprehensiveness and accuracy for specific questions. This approach appears particularly advantageous when scaling to larger model sizes, indicating better alignment with the evaluation criteria required for our task.

5.4 How does ProxyReward Performance Compare to Closed-source Baselines?

Our method can improve small size models to exceed GPT-4-Turbo. As shown in Table 3, Qwen2.5-7B-Instruct achieves an accuracy of 35.07% on the ProxyQA benchmark, surpassing GPT-4-Turbo (33.94%) despite having a substantially smaller pa-

Table 4: Ablation study of ProxyQA Accuracy (%) in the Qwen 2.5 series.

Model	1.5B	3B	7B
ProxyReward - Precision	12.57	17.49	25.67
ProxyReward - Accuracy (Iter 1)	**22.10**	27.58	33.23
ProxyReward - Accuracy (Iter 2)	21.54	**28.54**	**35.07**

rameter count. This is a remarkable achievement considering the vast resource difference between these models.

The performance gap is even more pronounced when comparing to GPT-3.5-Turbo (23.94%) and GPT-4-0613 (27.19%), with our Qwen2.5-7B-Instruct outperforming them by 11.13 and 7.88 percentage points respectively. While Llama-3.1-8B-Instruct achieves a lower accuracy of 30.11%, it still exceeds GPT-3.5-Turbo and GPT-4-0613, demonstrating the effectiveness of our approach across different model architectures.

6 Ablation Study

In this section, we conduct an ablation study to systematically investigate the impact of various factors on the performance of our proposed model. We address two critical questions: first, we explore whether multiple training iterations with ProxyReward can enhance model performance Section (6.1); second, we evaluate the effectiveness of different ProxyReward metrics to identify the most suitable one for our tasks Section (6.2).

6.1 Can ProxyReward Multiple Training Iterations Improve Performance?

The results reveal a consistent pattern of improvement when applying ProxyReward, especially with multiple training iterations. Across all tested models, multiple iterations of ProxyReward consistently outperform single iterations. For example, Qwen2.5-3B-Instruct improves from 27.58 to 28.54, while Llama-3.1-8B-Instruct increases from 29.38 to 30.11. Even the smallest models benefited, with Qwen2.5-1.5B-Instruct and Llama-3.2-1B-Instruct showing improvements of 21.54 and 20.08, respectively.

Interestingly, the conventional LLM-as-a-Judge approach consistently underperformed compared to both baseline models and our method, suggesting limitations in directly applying judgment signals without our proposed proxy mechanism. This performance gap was especially pronounced in smaller models, highlighting the scalability advantages of our approach across model sizes.

These results collectively validate that ProxyReward not only enhances model performance but can achieve state-of-the-art results that exceed those of much larger models when applied iteratively, offering an efficient pathway to improved capabilities without the computational costs associated with scaling model size.

6.2 Which ProxyReward Metric Should We Choose?

The results demonstrate that accuracy-based metrics consistently outperform precision-based metrics across all model scales. In particular, accuracy measurements after multiple iterations show the most substantial performance gains for larger models. As shown in Table 4, based on our ablation study examining different evaluation metrics across varying model sizes (1.5B, 3B, and 7B parameters), we established a systematic methodology for metric selection. When comparing the metrics directly, we observe that accuracy after the second iteration yields the best overall performance for the 7B parameter model (35.0), representing a significant improvement over precision-based evaluation (25.67). For mid-sized models (3B parameters), second-iteration accuracy also demonstrates superior performance (28.54), though with a less pronounced advantage compared to first-iteration accuracy (27.58).

Interestingly, for the smallest model (1.5B parameters), first-iteration accuracy (22.10) slightly outperforms second-iteration accuracy (21.54), suggesting that additional iterations may introduce noise rather than refinement at smaller parameter scales. This pattern indicates that optimal metric selection should be calibrated to model size, with larger models benefiting from extended iterative evaluation approaches.

6.3 Whether higher ProxyReward scores correlate with human preference?

The results demonstrate that accuracy-based metrics consistently outperform precision-based metrics across all model scales.

Setup We evaluated the correlation among our ProxyReward metric, LLM-as-a-Judge and human judgments through pairwise comparisons. We sam-

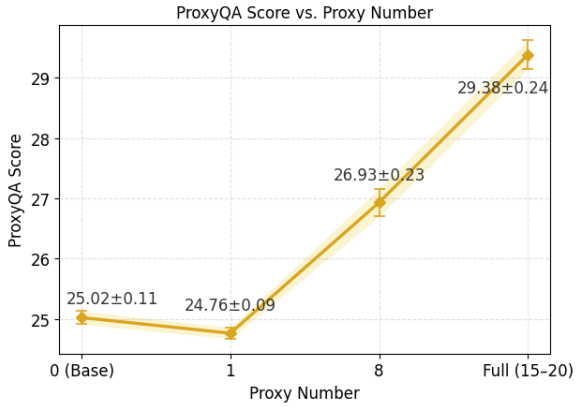

Figure 5: Effect of proxy-question count on ProxyQA performance. Increasing the number of proxy questions per meta-question improves accuracy, with the full setting (15–20 proxies) achieving the highest score (29.38 ± 0.24) versus the base (25.02 ± 0.11). Error bars denote ±1 standard error across runs.

Table 5: Comparison of LLM-as-a-Judge and ProxyReward

LLM-as-a-Judge	ProxyReward
50.00	80.00

pled 10 meta-questions from the ProxyReward dataset and used Qwen2.5-1.5B-Instruct to generate two long text responses for each. ProxyReward scores were generated using GPT-4o-mini. Three AI experts evaluated the long text responses, with the majority vote determining the preferred response.

Result ProxyReward demonstrated strong alignment with human judgment, achieving an 80.00% agreement rate with majority human decisions. For comparison, a standard LLM-as-a-Judge approach using GPT-4o-mini achieved only a 50.00% agreement rate. These findings provide robust evidence that the proposed ProxyReward can effectively assess LLMs' capabilities in generating long-text content.

6.4 How Does the Number of Proxy Questions per Meta-Question Affect Performance?

Experimental results show that increasing the number of proxy questions leads to higher performance. We conduct the ablation study on Llama-3.1-8B-Instruct. As shown in Figure 5, using a single proxy question decreases performance, as it cannot comprehensively verify long text quality. Eight proxy questions show improvement over the baseline but remain less effective than the full set. The results demonstrate the effectiveness of ProxyReward.

6.5 How Does Filtering Mechanisms Affect Performance?

Setup The quality of responses automatically generated by the generation model varies considerably, which significantly impacts training outcomes. To address this challenge, we implement data selection based on scores. First, we recognize that meta-questions vary in difficulty—for simpler questions, consistently produces high-scoring responses, while for more challenging questions, the scores of generated responses show greater variance. Consequently, we prioritize meta-questions that exhibit higher variance in scores for the same response. This approach prevents our dataset from being dominated by simple questions, which would reduce training efficiency. Second, to avoid model optimization direction contrast, we filter cases where a single meta-question has multiple preference indicators. From the remaining data, we select the highest and lowest scoring responses to create preference pairs, which serve as reference responses for training model. The selection details are in Table 7.

Result We compare the performance of proposed data selection method and random selection on Llama-3.1-8B-Instruct. Our filtering method significantly outperforms random selection, as shown in the Table 6.

7 Conclusion

We tackle Open-LTG by addressing two obstacles—limited gold references for training and the weakness of generic assessment signals. To address these challenges, we introduced **ProxyReward**, a novel reinforcement learning (RL)-based framework that comprises two main components. First, our dataset generation methodology eliminates the need for extensive labeled data or manual effort by employing automatic creation of training examples. Second, our ProxyReward signal design provides precise evaluation of information comprehensiveness and accuracy, moving beyond the limitations of general assessment methods. Experiments show improvements of up to 20% on Open-LTG with open-source models, surpassing LLM-as-a-Judge and even GPT-4-Turbo. ProxyReward provides a practical path to more coherent, informative, and contextually grounded long-form responses to complex queries.

Limitations

While ProxyReward demonstrates significant improvements in open-ended long text generation, several limitations remain. The reliance on LLM-generated proxy question-answer pairs introduces potential biases and errors inherent to the underlying models, which may affect the objectivity and coverage of the reward signals. Additionally, the framework's effectiveness depends on the quality and diversity of automatically generated meta-questions, which may not fully capture the complexity of real-world user queries across all domains. Finally, although ProxyReward reduces the need for manual annotation, it still requires access to high-performing LLMs for reward computation, which may pose computational and cost challenges for some practitioners.

Acknowledgment

The work described in this paper was partially supported by the Research Grants Council of the Hong Kong Special Administrative Region, China (CUHK 2410072, RGC R1015-23).

References

Yushi Bai, Xin Lv, Jiajie Zhang, Yuze He, Ji Qi, Lei Hou, Jie Tang, Yuxiao Dong, and Juanzi Li. 2024a. Longalign: A recipe for long context alignment of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1376–1395.

Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, and 1 others. 2024b. Longbench v2: Towards deeper understanding and reasoning on realistic long-context multitasks. *arXiv preprint arXiv:2412.15204*.

Yushi Bai, Jiahao Ying, Yixin Cao, Xin Lv, Yuze He, Xiaozhi Wang, Jifan Yu, Kaisheng Zeng, Yijia Xiao, Haozhe Lyu, and 1 others. 2023. Benchmarking foundation models with language-model-as-an-examiner. *Advances in Neural Information Processing Systems*, 36:78142–78167.

Yushi Bai, Jiajie Zhang, Xin Lv, Linzhi Zheng, Siqi Zhu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2024c. Longwriter: Unleashing 10,000+ word generation from long context llms. *arXiv preprint arXiv:2408.07055*.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1533–1544, Seattle, Washington, USA. Association for Computational Linguistics.

Bing. 2023. Ai-powered bing with chatgpt's gpt-4. Language model.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Aydar Bulatov, Yury Kuratov, and Mikhail Burtsev. 2022. Recurrent memory transformer. *Advances in Neural Information Processing Systems*, 35:11079–11091.

Lichang Chen, Shiyang Li, Jun Yan, Hai Wang, Kalpa Gunaratna, Vikas Yadav, Zheng Tang, Vijay Srinivasan, Tianyi Zhou, Heng Huang, and 1 others. 2024. Alpagasus: Training a better alpaca with fewer data. In *The Twelfth International Conference on Learning Representations*.

Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. 2023. Extending context window of large language models via positional interpolation. *arXiv preprint arXiv:2306.15595*.

Yann Dubois, Chen Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy S Liang, and Tatsunori B Hashimoto. 2023. Alpacafarm: A simulation framework for methods that learn from human feedback. *Advances in Neural Information Processing Systems*, 36:30039–30069.

Patrick Fernandes, Daniel Deutsch, Mara Finkelstein, Parker Riley, André FT Martins, Graham Neubig, Ankush Garg, Jonathan H Clark, Markus Freitag, and Orhan Firat. 2023. The devil is in the errors: Leveraging large language models for fine-grained machine translation evaluation. In *Proceedings of the Eighth Conference on Machine Translation*, pages 1066–1083.

Yao Fu, Rameswar Panda, Xinyao Niu, Xiang Yue, Hannaneh Hajishirzi, Yoon Kim, and Hao Peng. 2024. Data engineering for scaling language models to 128k context. In *International Conference on Machine Learning*.

Mingqi Gao, Jie Ruan, Renliang Sun, Xunjian Yin, Shiping Yang, and Xiaojun Wan. 2023. Human-like summarization evaluation with chatgpt. *arXiv preprint arXiv:2304.02554*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Albert Gu, Karan Goel, and Christopher Ré. 2021. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396*.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, and 1 others. 2024. A survey on llm-as-a-judge. *arXiv preprint arXiv:2411.15594*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025a. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Zhihan Guo, Jiele Wu, Wenqian Cui, Yifei Zhang, Minda Hu, Yufei Wang, and Irwin King. 2025b. From general to targeted rewards: Surpassing gpt-4 in open-ended long-context generation. *Preprint*, arXiv:2506.16024.

Zhihan Guo, Yifei Zhang, Zhuo Zhang, Zenglin Xu, and Irwin King. 2024a. Fedhlt: Efficient federated low-rank adaption with hierarchical language tree for multilingual modeling. In *Companion Proceedings of the ACM Web Conference 2024*, WWW '24, page 1558–1567, New York, NY, USA. Association for Computing Machinery.

Zhihan Guo, Yifei Zhang, Zhuo Zhang, Zenglin Xu, and Irwin King. 2024b. FedLFC: Towards efficient federated multilingual modeling with LoRA-based language family clustering. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1519–1528, Mexico City, Mexico. Association for Computational Linguistics.

Huiqiang Jiang, Yucheng Li, Chengruidong Zhang, Qianhui Wu, Xufang Luo, Surin Ahn, Zhenhua Han, Amir H Abdi, Dongsheng Li, Chin-Yew Lin, and 1 others. 2024. Minference 1.0: Accelerating prefilling for long-context llms via dynamic sparse attention. *arXiv preprint arXiv:2407.02490*.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. 2025. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*.

Abdullatif Köksal, Timo Schick, Anna Korhonen, and Hinrich Schütze. 2023. Longform: Effective instruction tuning with reverse instructions. *arXiv preprint arXiv:2304.08460*.

Kalpesh Krishna, Erin Bransom, Bailey Kuehl, Mohit Iyyer, Pradeep Dasigi, Arman Cohan, and Kyle Lo. 2023. Longeval: Guidelines for human evaluation of faithfulness in long-form summarization. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1650–1669.

Ishita Kumar, Snigdha Viswanathan, Sushrita Yerra, Alireza Salemi, Ryan A Rossi, Franck Dernoncourt, Hanieh Deilamsalehy, Xiang Chen, Ruiyi Zhang, Shubham Agarwal, and 1 others. 2024. Longlamp: A benchmark for personalized long-form text generation. *arXiv preprint arXiv:2407.11016*.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Kellie Ren Lu, Thomas Mesnard, Johan Ferret, Colton Bishop, Ethan Hall, Victor Carbune, and Abhinav Rastogi. 2024. RLAIF: Scaling reinforcement learning from human feedback with AI feedback.

Nayeon Lee, Wei Ping, Peng Xu, Mostofa Patwary, Pascale N Fung, Mohammad Shoeybi, and Bryan Catanzaro. 2022. Factuality enhanced language models for open-ended text generation. *Advances in Neural Information Processing Systems*, 35:34586–34599.

Jiaqi Li, Mengmeng Wang, Zilong Zheng, and Muhan Zhang. 2024a. Loogle: Can long-context language models understand long contexts? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16304–16333.

Tianle Li, Ge Zhang, Quy Duc Do, Xiang Yue, and Wenhu Chen. 2024b. Long-context llms struggle with long in-context learning. *arXiv preprint arXiv:2404.02060*.

Xian Li, Ping Yu, Chunting Zhou, Timo Schick, Omer Levy, Luke Zettlemoyer, Jason E Weston, and Mike Lewis. 2024c. Self-alignment with instruction backtranslation. In *The Twelfth International Conference on Learning Representations*.

Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. 2023. Contrastive decoding: Open-ended text generation as optimization. In *The 61st Annual Meeting Of The Association For Computational Linguistics*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and 1 others. 2024a. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Jiaheng Liu, Dawei Zhu, Zhiqi Bai, Yancheng He, Huanxuan Liao, Haoran Que, Zekun Wang, Chenchen Zhang, Ge Zhang, Jiebin Zhang, and 1 others. 2025. A comprehensive survey on long context language modeling. *arXiv preprint arXiv:2503.17407*.

Xiang Liu, Peijie Dong, Xuming Hu, and Xiaowen Chu. 2024b. Longgenbench: Long-context generation benchmark. *arXiv preprint arXiv:2410.04199*.

Xuezhe Ma, Xiaomeng Yang, Wenhan Xiong, Beidi Chen, Lili Yu, Hao Zhang, Jonathan May, Luke Zettlemoyer, Omer Levy, and Chunting Zhou. 2024. Megalodon: Efficient llm pretraining and inference with unlimited context length. *Advances in Neural Information Processing Systems*, 37:71831–71854.

Nicolo Micheletti, Samuel Belkadi, Lifeng Han, and Goran Nenadic. 2024. Exploration of masked and causal language modelling for text generation. *arXiv preprint arXiv:2405.12630*.

OpenAI. 2024a. Gpt-4o mini. https://www.openai.com/. Language model.

OpenAI. 2024b. Hello gpt-4o. https://openai.com/index/hello-gpt-4o/. Language model.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. 2024. Yarn: Efficient context window extension of large language models. In *The Twelfth International Conference on Learning Representations*.

Chau Minh Pham, Simeng Sun, and Mohit Iyyer. 2024. Suri: Multi-constraint instruction following for long-form text generation. *arXiv preprint arXiv:2406.19371*.

Michael Poli, Stefano Massaroli, Eric Nguyen, Daniel Y Fu, Tri Dao, Stephen Baccus, Yoshua Bengio, Stefano Ermon, and Christopher Ré. 2023. Hyena hierarchy: Towards larger convolutional language models. In *International Conference on Machine Learning*, pages 28043–28078. PMLR.

Haoran Que, Feiyu Duan, Liqun He, Yutao Mou, Wangchunshu Zhou, Jiaheng Liu, Wenge Rong, Zekun Moore Wang, Jian Yang, Ge Zhang, and 1 others. 2024. Hellobench: Evaluating long text generation capabilities of large language models. *arXiv preprint arXiv:2409.16191*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741.

Natalie Schluter. 2017. The limits of automatic summarisation according to rouge. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 41–45. Association for Computational Linguistics.

Lin Shi, Chiyu Ma, Wenhua Liang, Weicheng Ma, and Soroush Vosoughi. 2024. Judging the judges: A systematic investigation of position bias in pairwise comparative assessments by llms. *arXiv preprint arXiv:2406.07791*.

Feifan Song, Bowen Yu, Minghao Li, Haiyang Yu, Fei Huang, Yongbin Li, and Houfeng Wang. 2024. Preference ranking optimization for human alignment. *Preprint*, arXiv:2306.17492.

Shyam Sudhakaran, Miguel González-Duque, Matthias Freiberger, Claire Glanois, Elias Najarro, and Sebastian Risi. 2023. Mariogpt: Open-ended text2level generation through large language models. *Advances in Neural Information Processing Systems*, 36:54213–54227.

Haochen Tan, Zhijiang Guo, Zhan Shi, Lu Xu, Zhili Liu, Yunlong Feng, Xiaoguang Li, Yasheng Wang, Lifeng Shang, Qun Liu, and 1 others. 2024. Proxyqa: An alternative framework for evaluating long-form text generation with large language models. *arXiv preprint arXiv:2401.15042*.

Yi Tay, Mostafa Dehghani, Samira Abnar, Yikang Shen, Dara Bahri, Philip Pham, Jinfeng Rao, Liu Yang, Sebastian Ruder, and Donald Metzler. 2020. Long range arena: A benchmark for efficient transformers. *arXiv preprint arXiv:2011.04006*.

Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. 2022. Efficient transformers: A survey. *ACM Computing Surveys*, 55(6):1–28.

Qwen Team. 2024. Qwen2. 5: A party of foundation models, september 2024. *URL https://qwenlm. github. io/blog/qwen2*, 5.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Tianlu Wang, Ilia Kulikov, Olga Golovneva, Ping Yu, Weizhe Yuan, Jane Dwivedi-Yu, Richard Yuanzhe Pang, Maryam Fazel-Zarandi, Jason Weston, and Xian Li. 2024. Self-taught evaluators. *arXiv preprint arXiv:2408.02666*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. *Preprint*, arXiv:2203.11171.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.

Tianhao Wu, Weizhe Yuan, Olga Golovneva, Jing Xu, Yuandong Tian, Jiantao Jiao, Jason Weston, and Sainbayar Sukhbaatar. 2024a. Meta-rewarding language models: Self-improving alignment with llm-as-a-meta-judge. *arXiv preprint arXiv:2407.19594*.

Tsung-Han Wu, Giscard Biamby, Jerome Quenum, Ritwik Gupta, Joseph E Gonzalez, Trevor Darrell, and David M Chan. 2024b. Visual haystacks: A vision-centric needle-in-a-haystack benchmark. *arXiv preprint arXiv:2407.13766*.

Wenhan Xiong, Jingyu Liu, Igor Molybog, Hejia Zhang, Prajjwal Bhargava, Rui Hou, Louis Martin, Rashi Rungta, Karthik Abinav Sankararaman, Barlas Oguz, and 1 others. 2023. Effective long-context scaling of foundation models. *arXiv preprint arXiv:2309.16039*.

Wenhan Xiong, Jingyu Liu, Igor Molybog, Hejia Zhang, Prajjwal Bhargava, Rui Hou, Louis Martin, Rashi Rungta, Karthik Abinav Sankararaman, Barlas Oguz, and 1 others. 2024. Effective long-context scaling of foundation models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4643–4663.

Fangyuan Xu, Yixiao Song, Mohit Iyyer, and Eunsol Choi. 2023. A critical evaluation of evaluations for long-form question answering. In *The 61st Annual Meeting Of The Association For Computational Linguistics*.

Kevin Yang, Dan Klein, Asli Celikyilmaz, Nanyun Peng, and Yuandong Tian. 2024. Rlcd: Reinforcement learning from contrastive distillation for lm alignment. In *The Twelfth International Conference on Learning Representations*.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. Tree of thoughts: Deliberate problem solving with large language models. *Preprint*, arXiv:2305.10601.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. 2024. Self-rewarding language models. *Proceedings of the 41th International Conference on Machine Learning*.

Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. 2023. Rrhf: Rank responses to align language models with human feedback without tears. *arXiv preprint arXiv:2304.05302*.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and 1 others. 2020. Big bird: Transformers for longer sequences. *Advances in neural information processing systems*, 33:17283–17297.

Jiajie Zhang, Zhongni Hou, Xin Lv, Shulin Cao, Zhenyu Hou, Yilin Niu, Lei Hou, Yuxiao Dong, Ling Feng, and Juanzi Li. 2024. Longreward: Improving long-context large language models with ai feedback. *arXiv preprint arXiv:2410.21252*.

Qiyuan Zhang, Fuyuan Lyu, Zexu Sun, Lei Wang, Weixu Zhang, Wenyue Hua, Haolun Wu, Zhihan Guo, Yufei Wang, Niklas Muennighoff, Irwin King, Xue Liu, and Chen Ma. 2025. A survey on test-time scaling in large language models: What, how, where, and how well? *Preprint*, arXiv:2503.24235.

Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. 2022. Automatic chain of thought prompting in large language models. *Preprint*, arXiv:2210.03493.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, and Ed Chi. 2023. Least-to-most prompting enables complex reasoning in large language models. *Preprint*, arXiv:2205.10625.

A Appendix

A.1 How Transferable Is the Proposed Method to Other Tasks?

Experimental results show that ProxyReward has good transferability on other tasks. We have added the evaluation results to our paper.

Setup We evaluated ProxyReward on the Web-based Question Answering (WQA) benchmark (Berant et al., 2013), which is widely recognized for short-term answer evaluation. We employed Qwen2.5-7B-Instruct as our response generator. Our comparison includes seven established baselines: Zero-shot, Few-shot Prompting, Chain-of-Thought (CoT) (Wei et al., 2023), Self-Consistency (SC) (Wang et al., 2023), Tree of Thoughts (ToT) (Yao et al., 2023), Auto-Chain-of-Thought (Auto-CoT) (Zhang et al., 2022), and Least-to-Most prompting (Zhou et al., 2023).

Result ProxyReward demonstrates exceptional transferability to short-term answer tasks, achieving a 47.34% F1 score that surpasses all baseline methods. The performance improvement is particularly notable compared to the next best method (Few-shot at 44.70%), representing a 5.9% relative improvement. Our method shows consistent gains across different question types within the WQA benchmark, with particularly strong performance on factual queries (+7.2% over Few-shot) and definitional questions (+6.3% over Least-to-Most). These results provide compelling evidence that ProxyReward effectively generalizes to the short-term answer task.

Table 6: Filtering mechanisms comparison between Random and ProxyReward.

Random	ProxyReward
24.71	29.38

B Prompts

In this part, we first introduce a response generation prompt.

Prompt for Response Generation

Write a well-structured and extensive report to answer the question below.

Question: {META-QUESTION}

Second, we present the prompts used in the two main components of ProxyReward dataset construction: meta-question and proxy question-answer pair generation.

Prompt for Meta-question Generation

You are a data scientist. For the specified topic, please provide a list of questions that require detailed and comprehensive responses. Your tone should be formal.
Topic: {TOPIC}

Here is an example:

Input:
Topic: Computer Science

Output:
Data parallelism, model parallelism, and pipeline parallelism play a vital role in the training of large-scale language models. What are the representative works and frameworks among these technologies? Please introduce these technologies and frameworks in detail.

Prompt for Proxy Question-answer Pair Generation

You are a data scientist. Your task is to generate proxy question-answer pairs based on given meta-question.

Meta-questions are defined as questions that require detailed and comprehensive responses.

For a given meta-question, please identify the key content necessary for formulating a detailed question and create more than 15 proxy question-answer pairs to explore these points.

Each proxy question should incorporate a key aspect of the meta-question.

The corresponding proxy answers should be one of the following: {True, False, Not Mentioned}, indicating the correctness and relevance of each proxy question to the meta-question.

Meta-question: {META-QUESTION}

Here is an example:

Input:
Meta-question: Contrastive learning has greatly promoted the progress of the learning of sentence embeddings. Please introduce some effective contrastive learning methods in sentence embedding.

Output:
1. **Question**: The hierarchical sampling strategy first selects a subset of negative samples based on their relevance to positive samples, then randomly samples from this subset to form hard negatives.
Answer: True

2. **Question**: Given a sentence, EDA (Easy Data Augmentation) randomly chooses and applies one of four simple operations: Synonym replacement (SR), Random insertion (RI), Random swap (RS), and Random deletion (RD).
Answer: True

3. **Question**: SBERT (Sentence-BERT) relies on siamese and triplet network architectures to learn sentence embeddings such that the sentence similarity can be estimated by cosine similarity between pairs of embeddings.
Answer: True

4. **Question**: BERT-flow was proposed to transform the embedding into a smooth and isotropic Gaussian distribution via normalizing flows.
Answer: True

5. **Question**: IS-BERT (Info-Sentence BERT) adopts a self-supervised learning objective based on mutual information maximization to learn good sentence embeddings in an unsupervised manner.
Answer: True

Thirdly, we present the prompt of ProxyReward automatic assessment.

Prompt for ProxyReward Signal

Read the provided document and determine whether the question or statement below is "True", "False" or "Not mentioned".

Use only the information in the text to make your decision. Do not rely on prior knowledge or information outside of the given text.

If the text does not provide enough information to make a decision, respond with "Not mentioned".

Your are required to explain how you answer the question, and then select the final answer from "True", "False" and "Not Mentioned".

Document: {DOCUMENT}

Question: {QUESTION}

Finally, we show the prompts used in LLM-as-a-Judge baseline.

Prompt for LLM-as-a-Judge Reward Signal

Evaluate the quality of the given response to the question.

Rate the response on four dimensions: accuracy, coherence, factuality, and comprehensiveness. Use a scale from 1 (worst) to 10 (best).

- Accuracy: Assess how well the response addresses the question and provides correct information.
- Coherence: Evaluate the clarity and logical flow of the response.
- Factuality: Check for the presence of verifiable facts and data.
- Comprehensiveness: Determine if the response covers all relevant aspects of the question.

Be strict in your evaluation, and aim to use the full scale. Consider the following criteria for scoring:

- A score of 1-3 indicates major flaws in multiple dimensions.
- A score of 4-6 indicates moderate issues or inconsistencies.
- A score of 7-8 reflects generally good quality with minor flaws.
- A score of 9-10 is reserved for exemplary responses that excel in all dimensions.

Question: {META-QUESTION}

Response: {DOCUMENT}

Table 7: Statistical characteristics (%) of response quality distributions across different models and reward approaches.

Method	Base Model	Low Reward	High Reward	Mean Reward
LLM-as-a-Judge	Qwen2.5-1.5B-Instruct	50.53	80.98	$73.00_{\pm 0.08}$
	Qwen2.5-3B-Instruct	78.88	99.92	$99.02_{\pm 0.04}$
	Qwen2.5-7B-Instruct	39.22	77.58	$62.42_{\pm 0.11}$
ProxyReward-Precision	Qwen2.5-1.5B-Instruct	3.35	99.24	$82.80_{\pm 0.24}$
	Qwen2.5-3B-Instruct	6.87	99.24	$84.90_{\pm 0.20}$
	Qwen2.5-7B-Instruct	1.25	99.92	$84.18_{\pm 0.24}$
ProxyReward-Accuracy	Qwen2.5-1.5B-Instruct	18.70	63.53	$37.94_{\pm 0.12}$
	Qwen2.5-3B-Instruct	27.77	70.05	$49.12_{\pm 0.12}$
	Qwen2.5-7B-Instruct	28.19	72.57	$50.65_{\pm 0.12}$
	Llama-3.2-1B-Instruct	34.45	62.01	$48.02_{\pm 0.08}$
	Llama-3.2-3B-Instruct	7.58	53.51	$23.34_{\pm 0.12}$
	Llama-3.1-8B-Instruct	9.52	62.54	$30.84_{\pm 0.15}$