MITIGATING SHORT BOARD EFFECT VIA DYNAMIC REWARD BALANCING IN MULTI-REWARD LLM OPTIMIZATION

Nuo Chen[1†] **Yufei Gao**[1†] **Yongnan Jin**[1†]
Yan Hu **Anningzhe Gao**[2*] **Lingyong Yan**[3] **Benyou Wang**[1,2*]
[1]The Chinese University of Hong Kong, Shenzhen
[2]Shenzhen Research Institute of Big Data [3]Baidu Inc.
{chennuo,wangbenyou}@cuhk.edu.cn

ABSTRACT

In the current landscape of large language models (LLMs), many evaluation metrics have been developed and used as rewards during training to improve specific metrics. However, balancing these metrics and dynamically adjusting reward weights remains challenging, as current approaches often fail to enhance weaker metrics. To address this, we empirically propose a **D**ynamic **R**eward **B**alancing **O**ptimization framework $\mathcal{DRBO}$ to mitigate the "short-board effect" by measuring performance, adjusting reward weights to prioritize weaker metrics, and optimizing the model via reinforcement learning. We apply $\mathcal{DRBO}$ to both single-task and multi-type task scenarios, validating its effectiveness in generation with citations and online shopping conversation tasks. The results demonstrate improved overall performance and balanced optimization across multiple metrics, effectively overcoming the diversity and complexity inherent in LLMs. The code is released at https://github.com/NuoJohnChen/DRBO.

1 INTRODUCTION

Evaluation is crucial for LLM applications, which often necessitates adherence to diverse evaluation criteria (Zhuang et al., 2023; Guo et al., 2023; Chang et al., 2024), even for a *single* task. Thanks to their generalization, LLMs are also expected to excel across *multiple types* of tasks simultaneously (Wang et al., 2023b) and each type of task might have its own metrics.

In reinforcement learning, one could structure these *metrics* (previously for evaluation) as *rewards* that could be boosted during training (Sharma et al., 2021; Yadav et al., 2021; Deng et al., 2022; Liu et al., 2023a; Xu et al., 2024; Wang et al., 2024b), to optimize complex objective functions even at testing time (OpenAI, 2024). However, when reward weights remain static, the weakest metric (the "**short-board**") becomes a bottleneck that restricts overall LLM effectiveness, which introduces the **short-board effect** in multi-reward opti-

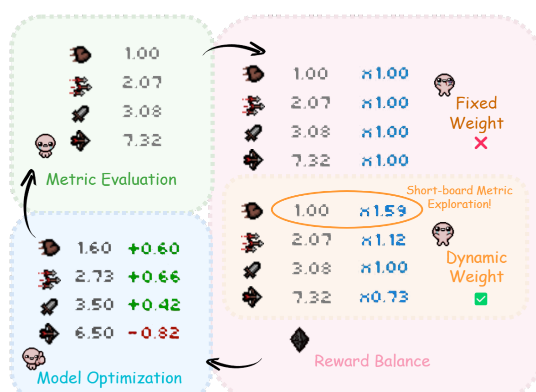
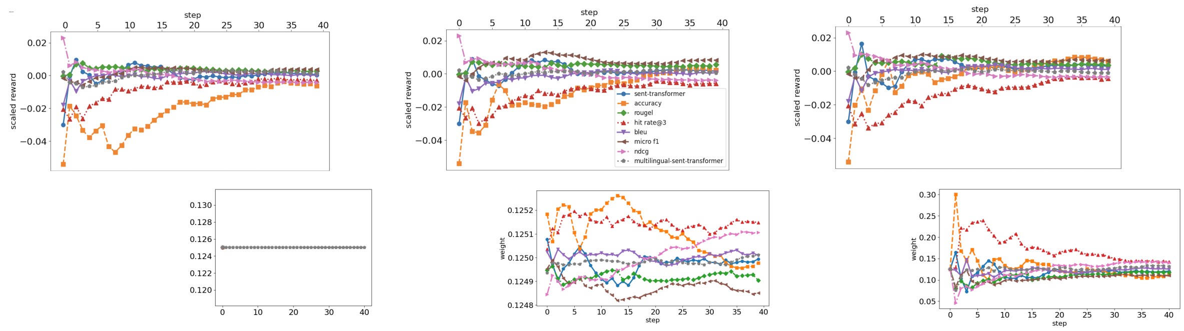


Figure 1: $\mathcal{DRBO}$ adjusts rewards and weights dynamically through iterations of three main stages: performance metric *evaluation*, reward weight *balance*, and model parameter *optimization*. Elements of this schema are from The Binding of Isaac.

*Benyou Wang and Anningzhe Gao are the corresponding authors. First three authors contributed to this work equally.



The weight of $\mathcal{DRBO}_{average}$ is averaged and static.

The weight of short-board (rougel) correspondingly increases.

The weight of rougel grows with $\Delta reward$.

(a) $\mathcal{DRBO}_{average}$ reward analysis. The reward improvement is limited and slow.

(b) $\mathcal{DRBO}_{inverse}$ reward analysis. Rougel increases largely in epoch 0. Short-board effect is mitigated.

(c) $\mathcal{DRBO}_{delta}$ reward analysis. Variance reduces. Short-board effect is mitigated.

Figure 2: $\mathcal{DRBO}$ **Weight Dynamics** (Shopping MMLU on Qwen2.5-1.5B-Instruct). The reward is scaled according to Eq. 4, with weight changes analyzed at epoch 0. Accuracy is identified as one short board in $\mathcal{DRBO}_{average}$ static weight settings, showing lower mean and higher variance. In $\mathcal{DRBO}_{inverse}$, after assigning a large weight to the short-board (ROUGE-L) in epoch 0, the reward of accuracy rises largely after step 5. In $\mathcal{DRBO}_{delta}$, the short-board effect reflects metrics with growth potential but limited progress; here, the weight is increased when the scaled reward has not yet saturated, encouraging further improvement. $\mathcal{DRBO}$ mitigates the short-board effect, narrowing the gap between the best and worst-performing metrics, while simultaneously improving the values of all metrics compared to (a), resulting in balanced and enhanced overall performance.

mization. For example, in Figure 2, when the scaled reward itself (or its growth trend) has not yet reached saturation, its update magnitude should accordingly be increased.

The mitigation of such effect is crucial because, for the optimization of LLMs' comprehensive capabilities, we typically aim for the model to **avoid weaknesses in any individual metric**, ensuring that its performance reaches a state of dynamic equilibrium across all metrics.

To mitigate the short-board effect, we propose a framework called $\mathcal{DRBO}$ to balance multiple rewards (depicted in Figure 1). $\mathcal{DRBO}$ interleaves between *Evaluation*, *Balance* and *Optimization* stages, to alternately update the rewards and model parameters. After performance *Evaluation*, we dynamically reduce the weight of saturated rewards, enabling the model to prioritize weaker areas and mitigate the "short-board effect" in *Balance* stage. During *Optimization*, we update the model through reinforcement learning to achieve a more balanced overall performance.

We introduce contributions below.

- We propose a novel framework called $\mathcal{DRBO}$ that dynamically learns and adjusts the weights of different evaluation metrics to simultaneously balance and optimize on both single-task and multi-task scenarios.
- We also show that by dynamically adjusting reward weights, our framework addresses the short-board effect through a more balanced and effective model optimization.

- Our extensive experiments based on citation-based generation and online shopping conversion show significant performance improvements, validating the effectiveness of $\mathcal{DRBO}$.

2 Background

2.1 Problem Definition

Model Evaluation A benchmark is a tool used to evaluate the performance of a model and is intended to replace the tedious manual labeling process. A benchmark usually consists of a pair of dataset and metric $(\mathcal{D}_j, m_j)$. The *dataset* $\mathcal{D}_j$ contains a set of questions and answers, denoted as $\mathcal{D}_j = \{(x_i, y_i)\}_{1\le i\le N_j}$, and the *metric* (e.g. BLEU, EM, F1) is used to evaluate the *performance* of the fine-tuned language model $\mathcal{M}$ on $\mathcal{D}_j$. Denote $\mathcal{M}_{\boldsymbol{\theta}}(x_i)$ is the output of the fine-tuned language model $\mathcal{M}$ parameterized by $\boldsymbol{\theta}$, then the *performance* of $\mathcal{M}$ on $(\mathcal{D}_j, m_j)$ is regularly defined by

$$\mathcal{P}_j(\mathcal{M}_{\boldsymbol{\theta}}) = \frac{1}{N_j}\sum_{i=1}^{N_j} m_j(x_i, y_i, \mathcal{M}_{\boldsymbol{\theta}}(x_i)) \in \mathbb{R}, \tag{1}$$

Multi-Type Task Settings For *single-task* settings, the set of benchmarks $\{(\mathcal{D}, m_j)\}_{1\le j\le N_j}$ share the same dataset $\mathcal{D}$. Our approach can also be generalized to *multi-type task* settings. In our *multi-task* setting, the benchmark datasets $\{\mathcal{D}_j\}$ and the test dataset $\{\mathcal{D}_{\textit{test}}\}$ with size N are sampled from the same dataset $\mathcal{D}$. The generalization of the model will be improved through training.

Multi-Reward Optimization Traditional machine learning tasks often use differentiable training objectives to optimize models. However, these objectives may not align well with the metrics used for evaluating downstream tasks. To address this misalignment, some studies optimize models directly based on downstream metrics. Since these metrics are typically non-differentiable, reinforcement learning is used to treat them as rewards to optimize the model accordingly.

In our setting, each $\mathcal{R}_j(\mathcal{M}_{\boldsymbol{\theta}})$ in Eq.1 can be viewed as a reward for fine-tuning $\mathcal{M}_{\boldsymbol{\theta}}$ under the environment $\{(\mathcal{D}_j, m_j)\}_{1\le j\le N_j}$. The model $\mathcal{M}_{\boldsymbol{\theta}}$ is treated as the policy $\pi_{\boldsymbol{\theta}}$, with π_{ref} serving as a reference policy to regularize the updates. Since multiple rewards are available, they need to be combined into $\mathcal{R} = \sum_j (w_j \mathcal{R}_j)$ for optimization. Then to maximize the expected reward $\mathcal{R}$, the model can be optimized by the policy gradient $\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(y \mid x)$, with the objective function $\mathcal{J}$ balancing reward maximization and KL divergence regularization:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \nabla_{\boldsymbol{\theta}} \mathcal{J}(\theta), \tag{2}$$

In which

$$\mathcal{J} = \mathbb{E}_{x_i\sim\mathcal{D}, y_i\sim\mathcal{M}_{\boldsymbol{\theta}}(x_i)}\left[\mathcal{R} - \beta\mathbb{D}_{KL}\left[\pi_{\boldsymbol{\theta}}(y_i \mid x_i)\|\pi_{ref}(y_i \mid x_i)\right]\right]$$
$$\nabla_{\boldsymbol{\theta}}\mathcal{J}(\theta) = \mathbb{E}_{x_i, y_i}[\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(y_i \mid x_i)(\mathcal{R} - \beta\nabla_{\boldsymbol{\theta}}\mathbb{D}_{KL})].$$

Our goal is to optimize the model $\mathcal{M}_{\boldsymbol{\theta}}$ to maximize $\sum_j (W_j \mathcal{P}_j)$ for any pre-set weights W_j.

2.2 Motivation of $\mathcal{DRBO}$

Importance of Balancing Multiple Rewards In Optimization step 2, balancing multiple rewards $\mathcal{R}_j$ is crucial for improving the overall performance of the model $\mathcal{M}$. Dynamically adjusting reward weights w_j allows the model to allocate more focus on weaker aspects, preventing performance bottlenecks. We observe that conventional methods often fail to address this issue, as even when the reward saturates, its weight remains disproportionately high (Figure 2), showing that imbalanced metrics lead to suboptimal performance.

Analogies of the Short-Board Effect This situation can be explained by the "short-board effect", where the overall effectiveness of a system is constrained by its weakest component. This concept can be extended to **Model Evaluation**, where *the performance is often bottlenecked by the weakest metrics*: if a model is optimized based solely on average rewards, it risks neglecting its worst-performing metrics. By dynamically adjusting the reward weights, the model is forced to address these shortcomings, leading to a more balanced performance.

Analogies of Biological Evolution The short-board effect is further illustrated by principles of biological evolution. In nature, the survival and adaptability of organisms depend on the optimization of their weakest traits. Natural selection emphasizes the improvement of these weaker traits, just as **Multi-Reward Optimization** in a model focuses more effort on enhancing underperforming areas. This approach, inspired by optimization algorithms rooted in animal flocks and symbiotic systems (Kennedy & Eberhart, 1995; Karaboğa, 2005; Yang & Deb, 2009; Miettinen & Neittaanmaki, 1999; Zitzler & Thiele, 1998), ensures that models are better adapted and optimized across all evaluation metrics.

While smaller models may be designed for specific tasks, LLMs should focus on **improving across multiple domains, a challenge compounded by the complexity and diversity of rewards**, making their optimization a significantly more intricate process.

Challenge to balance reward weights Achieving a proper **balance** of dynamic **reward weights** w_j requires addressing the varying performance $\mathcal{P}_j$ across different metrics. During each iteration of step Eq.2, when model **parameters** θ are **optimized**, the **performance** of each metric $\mathcal{P}_j(\mathcal{M}_{\boldsymbol{\theta}})$ fluctuates and needs re-**evaluation**. This fluctuation requires continuous re-**balancing** of the **reward weights**. Furthermore, since the importance and scale of each metric vary, the pre-set, unequal weights by human experts must also be considered, rather than assuming equal weighting.

3 Methodology of $\mathcal{DRBO}$

To address this challenge, we propose the algorithm $\mathcal{DRBO}$ to mitigate the "short-board effect," which means improving the model's performance should focus on the weaker metrics. Additionally, the performance across all metrics should be balanced according to pre-set weights. $\mathcal{DRBO}$ algorithm achieves this goal through an iterative cycle involving three key phases, as colored in Algorithm 1:

- **Evaluation**: Dynamically assess the model's performance using multiple metrics and compute the average reward.
- **Optimization**: Based on the evaluation results, update the model parameters using reinforcement learning to optimize the overall reward.
- **Balance**: Periodically reassess and adjust the weights of each metric to ensure balanced performance across all metrics.

Below is a detailed explanation of each phase: including three key stages.

3.1 Evaluation

The evaluation phase aims to assess the performance of the model $\mathcal{M}_\theta$ across multiple metrics $\{m_j\}_{1\leq j\leq N_j}$. Initially, the weights w_j are uniformly distributed, and the state s reflects the current model performance $\sum_j(w_j\mathcal{P}_j)$. In each iteration t, up to T, a metric m_s is sampled based on w_j, and its sample count is incremented. The model is evaluated b times on each metric m_j using data batches $\mathcal{D}_j^b$, yielding rewards $\mathcal{R}_j^b$,

Algorithm 1: $\mathcal{DRBO}$

Require: model $\mathcal{M}$, data and metric $\{\mathcal{D}_j, m_j\}_{1 \le j \le N_j}$, and hyperparameters T, b, e, τ
Ensure: Optimized $\mathcal{M}_{\boldsymbol{\theta}}$ with improved performance across metrics

Initialize weight $\{w_j\}_{1 \le j \le N_j}$ over N_j metrics, state s as model performance $\sum_j (w_j \mathcal{P}_j)$;
for $j = 1$ *to* N_j **do**
  Initialize sample count $count_j = 1$;
for $t = 1$ *to* T **do**
  Sample a metric m_s according to w_j;
  $count_j = count_j + 1$;
  for $j = 1$ *to* N_j **do**
    Evaluate $\mathcal{R}_j^b = \mathcal{P}_j^b(\mathcal{M}_{\boldsymbol{\theta}})$ through Eq.1 from each batch data $\mathcal{D}_j^b$;
    Standardize reward $\mathcal{R}_j^b(\mathcal{M}_{\boldsymbol{\theta}}) = f(\mathcal{R}_j^b)$;
    Calculate the average reward across all batches $\bar{\mathcal{R}}_j^{(t)} = \frac{1}{b} \sum_b \mathcal{R}_j(\mathcal{M}_{\boldsymbol{\theta}}(\mathcal{D}_j^b))$;
  Compute total reward $\mathcal{R}^{(t)} = \sum_j (w_j \bar{\mathcal{R}}_j^{(t)})$;
  Update the policy $\pi_{\boldsymbol{\theta}} = \mathcal{M}_\theta$ through Eq. 2 (Action a);
  if $t\%e == 0$ **then**
    for $j = 1$ *to* N_j **do**
      Update reward across all steps $x_j = r\left(\bar{\mathcal{R}}_j^{(1:t)}\right)$ through Eq. 3;
      Calculate performance expectation $\hat{x}_j = x_j + \sqrt{2 \ln t / count_j}$);
    Normalize weight $w = w(W_{(1:j)}, \hat{x}_{(1:j)})$ through Eq. 4,5;

which are standardized according to Eq. 4 to obtain $\bar{\mathcal{R}}_j^{(t)}$. The total reward $\mathcal{R}$ is the weighted sum of these averages using w_j.

3.2 Optimization

In the optimization phase, the evaluation results are used to update the model and adjust the weights of the metrics. The model's policy $\pi_\theta = \mathcal{M}_\theta$ is updated using reinforcement learning based on the total reward $R^{(t)} = \sum_j (w_j \bar{R}_j^{(t)})$. The action a in this phase involves updating the model parameters to maximize the total reward $R^{(t)}$.

3.3 Balance

In the balancing phase, we need to dynamically adjust the weights of various metrics to improve overall performance and pay attention to those that may be overlooked. To achieve this, we draw on the core idea of the Upper Confidence Bound (UCB) algorithm (Auer et al., 2002), which focuses on balancing exploration and exploitation: by exploring metrics with higher uncertainty (i.e., those that have been sampled less or whose performance has not yet been fully evaluated), we prevent the model from prematurely ignoring potential weaknesses, thus achieving better overall balance.

Specifically, every e steps, we re-evaluate the weights w_j for each metric. For each metric m_j, the adjusted expected performance $\hat{x}_j$ is calculated using the following formula: $\hat{x}_j = \bar{x}_j + \sqrt{\frac{2 \ln t}{\text{count}_j}}$, where x_j is the reward calculated in Eq. 3, t is the total number of samples, and count_j is the number of samples for metric

m_j. The adjustment term $\sqrt{\frac{2\ln t}{\text{count}_j}}$ encourages more exploration of those metrics that have been sampled less, ensuring the model can better evaluate each metric and prevent the occurrence of short-board effect.

3.4 Reward Scaling in Evaluation

To ensure that the rewards from different metrics are on the same scale when aggregated, and to avoid unfairness caused by differences in distribution and scale, we standardize the rewards for each metric. Specifically, we sample K reward values from each metric for standardization. First, we calculate the mean μ_j and variance σ_j^2 for each metric:

$$\mu_j = \frac{1}{K}\sum_{i=1}^{K}\mathcal{R}_{ji}, \quad \sigma_j^2 = \frac{1}{K}\sum_{i=1}^{K}(\mathcal{R}_{ji} - \mu_j)^2,$$

Next, we standardize the rewards to have a mean of 0 and variance of 1:

$$f(\mathcal{R}_{ji}) = \frac{\mathcal{R}_{ji} - \mu_j}{\sigma_j} \tag{4}$$

By applying this standardization, we ensure that each metric contributes fairly and reasonably when calculating the total reward.

3.5 Proposals in Balance

In the balancing phase, we propose two different proposals to adjust reward: $\mathcal{DRBO}_{inverse}$ and $\mathcal{DRBO}_{delta}$, focusing on mitigating the short-board effect. The specific calculation methods for both strategies are as follows.

$$r = \begin{cases} \mathrm{E}_t\left[\bar{\mathcal{R}}_j^{(1:t)} + 1\right] = \frac{1}{t}\sum_t \bar{\mathcal{R}}_j^{(t)}(\mathcal{M}_{\boldsymbol{\theta}}) + 1, & \mathcal{DRBO}_{inverse} \\ \mathrm{E}_t\left[\bar{\mathcal{R}}_j^{(1:t)}\right] - \mathrm{E}_{t-1}\left[\bar{\mathcal{R}}_j^{(1:t-1)}\right], & \mathcal{DRBO}_{delta} \end{cases} \tag{3}$$

$$z_j = \begin{cases} \frac{W_j}{\hat{x}_j}, & \mathcal{DRBO}_{inverse} \\ W_j \hat{x}_j, & \mathcal{DRBO}_{delta} \end{cases} \tag{4}$$

$$w_j = \begin{cases} \frac{z_j}{\Sigma_{i=1}^{N_j} z_i}, & \mathcal{DRBO}_{inverse} \\ \text{softmax}(z_j) = \frac{e^{z_j/T}}{\Sigma_{i=1}^{e^{z_i/T}}}, & \mathcal{DRBO}_{delta} \end{cases} \tag{5}$$

$\mathcal{DRBO}_{\boldsymbol{inverse}}$ utilizes the "inverse rewards" strategy, which encourages increasing the weights of metrics with smaller performance. $z_j = \frac{W_j}{\hat{x}_j}$ in Eq. 4 ensures that if a metric receives lower rewards, its corresponding update will be larger, thereby giving it more attention. This helps to address the short-board effect by focusing on underperforming metrics. Unlike UCB's typical motivation to balance exploration and exploitation, here we explicitly encourage the exploration of metrics with weaker performance. W_j represents the pre-set target weight, guiding the model to converge towards desired weights.

The computation of w_j in Eq. 5 ensures that metrics with higher rewards receive lower weights, and are sampled less frequently. However, these metrics are still considered, preventing the model from neglecting them. It is worth mentioning that, unlike traditional Multi-Armed Bandit (MAB) (Auer et al., 2002; Vermorel

& Mohri, 2005; Kuleshov & Precup, 2014; Slivkins, 2024) implementations which update based on the top-performing metric, we update all metrics simultaneously, ensuring diversity and flexibility in metric improvement.

$\mathcal{DRBO}_{\textit{\textbf{delta}}}$ focus on boosting the weights of metrics with greater potential for improvement. r in Eq. 3 adjusts the model parameters based on the reward increments, allowing for a faster response to changes in the environment. It helps to prioritize weak metrics with higher potential for improvement. Since r can be negative, Eq. 5 uses softmax strategy for weight normalization to ensure stability. Setting $T = 0.01$ step helps alleviate the problem of weight disappearance.

4 Applications

In this section, we introduce the two configurations of $\mathcal{DRBO}$: the *single* task setting and the *multi-type* task setting, along with their respective application scenarios.

Single-Task Setting The model is evaluated using multiple metrics on the same dataset, similar to how a species adapts to different environments. By dynamically adjusting weights, the model can effectively allocate resources and optimize its performance across all metrics.

Multi-Type Task Setting The model handles different types of tasks, each with its own specific metrics and datasets. Dynamic reward weight adjustment helps the model balance its performance across all tasks, ensuring comprehensive optimization, similar to how a species adapts to different ecological niches. The adaptability makes it an ideal choice for multi-task competitions or benchmark challenges.

Single-Task Application on Long-context Generation with Citation Single-Task $\mathcal{DRBO}$ can effectively address the challenges faced by long-context retrievers or citation-based text generation (Gao et al., 2023), such as handling complex queries, providing accurate results, and ensuring credible references. By dynamically adjusting the weights of metrics on fluency, correctness, and citation quality, $\mathcal{DRBO}$ can achieve balanced optimization across these aspects, thereby improving the model's ability to **synthesize information from multiple sources**.

Multi-Type Task Application on Online Shopping Conversation Skills Multi-Type Task $\mathcal{DRBO}$ can handle challenges presented by online shopping interactive conversations (Jin et al., 2024), which includes shopping skills such as concept understanding, knowledge reasoning, user behavior alignment, and multi-lingual abilities. By dynamically adjusting the weights of criterias including multiple choice, geneation, retrieval, etc., $\mathcal{DRBO}$ ensures balanced optimization across these tasks, improving the model's ability to **serve as general shop assistants**.

5 Experiments

In this section, we describe the experiments conducted in both multi-evaluation and multi-task settings. We systematically evaluate the effectiveness of $\mathcal{DRBO}$ through a series of experiments designed to test its performance across the scenarios detailed in Section 4.

5.1 Experiment setting

We conducted all experiments using the PyTorch framework on a setup consisting of eight NVIDIA A100 GPUs, each with 80 GB of memory. The computing environment was configured with CUDA 11.8 and

Dataset	*ASQA*				*ELI5*			
Criteria Metric	**Fluency** MAUVE	**Correctness** EM Recall	**Citation** F1	**All** Average	**Fluency** MAUVE	**Correctness** Claim Recall	**Citation** F1	**All** Average
				Llama-2-7B-Chat				
ALCE	40.92	**49.71**	42.75	44.46	46.08	16.67	21.45	28.07
$\mathcal{DRBO}_{average}$	40.64	49.25	48.44	46.11	**49.76**	18.33	29.66	32.58
$\mathcal{DRBO}_{inverse}$	40.92	48.74	**51.68**	47.11	45.94	**19.02**	29.98	31.65
$\mathcal{DRBO}_{delta}$	**43.55**	49.66	49.48	**47.56**	48.88	18.69	**30.28**	**32.62**
				Llama-3-8B-Instruct				
ALCE	22.01	49.48	59.38	43.62	**47.43**	**18.83**	39.71	35.32
$\mathcal{DRBO}_{average}$	24.99	50.17	**70.98**	48.71	**47.43**	18.06	43.89	36.46
$\mathcal{DRBO}_{inverse}$	**27.93**	**51.06**	68.47	**49.15**	46.31	18.67	45.79	**36.92**
$\mathcal{DRBO}_{delta}$	24.13	50.54	68.37	47.68	44.94	18.50	**47.08**	36.84

Table 1: Performance on ASQA and ELI5 Datasets.

cuDNN 8.7 for optimized deep learning performance. We set W to 1, denoting each evaluator is equally important.

In the balance phase, we introduce $\mathcal{DRBO}_{average}$ as a baseline approach, which uses static, equal weighting $w_j = \frac{1}{N_j}$. However, as indicated by Figure 1, this method does not address the short-board effect. We then evaluate our method $\mathcal{DRBO}_{inverse}$ and $\mathcal{DRBO}_{delta}$. In Optimization phase, there are several different reinforcement learning methods, including PPO Schulman et al. (2017), ReMax algorithm Li et al. (2023b), etc, to solve Eq. 2. We use ReMax algorithm to avoid training a value model and reduce computations. Detailed parameters are listed in Appendix B.

5.2 Experiment: Generation with Citation

We focus on the long-context generation with citation in a multi-evaluation setting, which assesses the model's ability to generate accurate and coherent responses given retrieved documents. We adopt ALCE benchmark (Gao et al., 2023), a well-known standard in the RAG community, because it is recognized for its ability to provide a single output with multiple evaluations, enabling comprehensive assessment of model performance. Using this benchmark[1], we apply the following criterias with metrics detailed in Appendix C as reward providers.

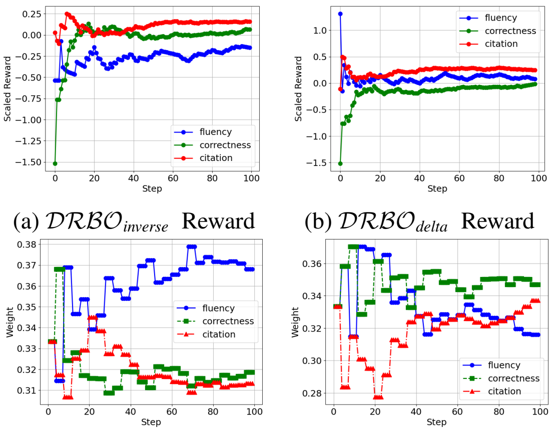
- **Fluency**: Evaluated by *MAUVE* (Pillutla et al., 2021), which measures the model's fluency in generating text.
- **Correctness**: Measured by exact match (*EM*) of the golden answer for ASQA (Stelmakh et al., 2022) or using an NLI model for inference for ELI5 (Fan et al., 2019), ensuring the output aligns with the correct answer.
- **Citation Accuracy**: Determined using an NLI model[2] to infer the correctness of citations, with metrics including citation recall and citation precision score to evaluate citation *F1*.

5.2.1 Analysis

According to the results on ASQA and ELI5 in Table 1, we reach the following conclusions:

[1]Fluency is a scalar derived from the distribution of several sentences. We report the average fluency across batches, while Gao et al. (2023) reports it across the entire dataset.

[2]`https://huggingface.co/google/t5_xxl_true_nli_mixture`



(a) $\mathcal{DRBO}_{inverse}$ Reward (b) $\mathcal{DRBO}_{delta}$ Reward

(c) $\mathcal{DRBO}_{inverse}$ Weight (d) $\mathcal{DRBO}_{delta}$ Weight

Figure 3: The change of scaled rewards and metric weights of Llama-2-7B-Chat on ASQA. More examples can be found in Figure 9.

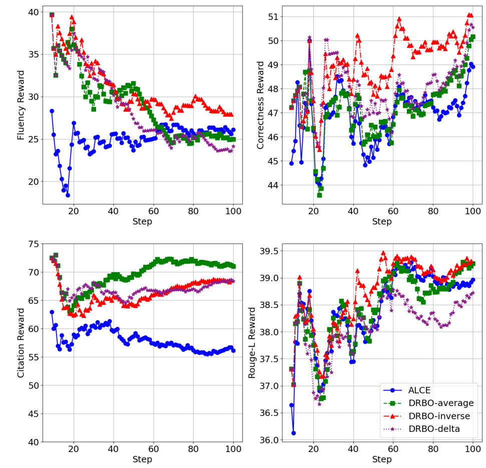


Figure 4: Performance of Llama-3-8b on ASQA Dataset.

$\mathcal{DRBO}$ improves overall performance. After applying $\mathcal{DRBO}$, the overall score of ASQA improved from 46.11 to 47.56 on Llama-2-7B-Chat and from 48.71 to 49.15 on Llama-3-8B-Instruct, with ELI5 also showing significant enhancement. This average score improvement suggests that by adjusting the weights of different metrics according to their scores in the reinforce learning process, $\mathcal{DRBO}$ can improve the overall performance effectively.

$\mathcal{DRBO}_{inverse}$ performs better and more balanced on stronger base models, while $\mathcal{DRBO}_{delta}$ excels with weaker base models. In Table 1, $\mathcal{DRBO}_{delta}$ outperforms $\mathcal{DRBO}_{inverse}$ on the Llama-2-7B-Chat model, particularly in the area of fluency, which is a strong criterion. Conversely, the Llama-3-8B-Instruct model exhibits the opposite trend. As a result, stronger models tend to focus more on the weakest criteria after scaling, while weaker models prioritize criteria with a more rapid growth trend.

For a more detailed analysis of the citation experiment and case study, please refer to Appendix G.

Metric Measurement	Fluency MAUVE	Correctness EM Recall	Citation F1
Llama3-8b-Instruct			
$\mathcal{DRBO}_{inverse}$ (1:1:1)	27.11	51.04	67.75
$\mathcal{DRBO}_{delta}$ (1:1:1)	24.13	50.54	68.37
$\mathcal{DRBO}_{inverse}$ (1:2:3)	25.62	49.84	68.34
$\mathcal{DRBO}_{delta}$ (1:2:3)	33.13	50.93	69.54

Table 2: Performance of ASQA when W = 1:2:3

5.2.2 Further Analysis

RQ1: Does $\mathcal{DRBO}$ converge with balance?

In Figure 3, weights and rewards change towards the opposite direction, and finally converge to a stable weight and reward. It clarifies the robustness of $\mathcal{DRBO}$.

Criteria Metric	**NER** Micro F1	**Retrieval** Hit Rate@3	**Choice** Accuracy	**Ranking** NDCG	**Generation** Sent-Transformer	**Translation** BLEU-4	**Multilinguality** Multilingual-Sent-Transformer	**Extraction** ROUGE-L	**All** Average
					Llama-3.2-1b-Instruct				
Shopping MMLU	39.72	15.96	55.12	57.96	50.06	19.09	43.97	**13.5**	36.92
$\mathcal{DRBO}_{average}$	29.48	**20.32**	54.68	59.38	53.58	16.41	39.80	3.73	34.67
$\mathcal{DRBO}_{inverse}$	33.20	20.29	52.36	63.58	**51.29**	16.67	44.87	8.18	36.31
—W = 2:1:1:1:1:1:1:1	**44.64**	18.93	**57.61**	**65.00**	50.16	**19.84**	**45.19**	12.75	**39.27**
$\mathcal{DRBO}_{delta}$	43.89	17.89	35.88	58.75	48.71	17.75	42.2	6.94	34.00
					Qwen2.5-1.5B-Instruct				
Shopping MMLU	11.39	38.64	56.05	76.56	51.13	15.69	45.01	3.03	37.19
$\mathcal{DRBO}_{average}$	12.19	41.67	62.45	**79.47**	52.03	13.82	46.19	**3.62**	38.93
$\mathcal{DRBO}_{inverse}$	9.42	**45.12**	**64.50**	78.12	**52.70**	16.32	46.52	3.45	**39.52**
$\mathcal{DRBO}_{delta}$	**12.24**	38.04	58.94	77.98	51.42	**16.66**	**47.23**	3.12	38.20
					Llama-3-8B-Instruct				
Shopping MMLU	58.07	58.95	72.56	78.12	49.27	**24.59**	38.76	5.80	48.27
$\mathcal{DRBO}_{average}$	44.15	49.46	63.87	75.51	**51.59**	22.92	**53.12**	6.38	45.88
$\mathcal{DRBO}_{inverse}$	**67.59**	**66.93**	**76.49**	**82.22**	49.80	23.74	38.85	**7.00**	**51.58**
$\mathcal{DRBO}_{delta}$	58.36	60.37	73.98	82.19	49.33	21.35	36.34	4.40	48.29

Table 3: Performance on Shopping MMLU.

RQ2: How does each metric change during training?

As shown in Figure 4, the performance on each metric rapidly increases at the beginning of training, surpassing the original model, and then gradually stabilizes over time. The fluency curve drops at step 50, reminding us that the training cycle of $\mathcal{DRBO}$ should not be too long, as it may lead to overfitting. The weakest metric, citation, shows stable training performance, further demonstrating the robustness of $\mathcal{DRBO}$.

RQ3: What if pre-defined importance of different metrics is non-equivalent ?
The effect of $\mathcal{DRBO}$ when the pre-defined weights are set to $W = 1{:}2{:}3$ is shown in Table 2, where we reduce the importance of fluency. We can see a significant improvement in all metrics, with the most notable increase in citation. This demonstrates the effectiveness of our method under different values of W, and highlights that the initial value of W can also have a significant impact on the results.

5.3 Experiment: Online Shopping Skills

We target at comprehensive abilities of LLMs on multi-type shopping QA, which assess multiple few-shot tasks with complex entities and relations. The ShoppingMMLU (Jin et al., 2024) benchmark, detailed in Appendix E in KDD Cup 2024[3], is selected because it is a statistically detailed, multi-type task dataset derived from real-world scenarios on Amazon, providing a comprehensive evaluation of models in shopping contexts. We divide this benchmark into the criteria listed in Appendix D with metrics to provide rewards.

5.3.1 Analysis

$\mathcal{DRBO}_{inverse}$ achieves overall enhancement in multi-type task settings, and $\mathcal{DRBO}_{delta}$ shows steady improvement. $\mathcal{DRBO}_{inverse}$ significantly outperforms zero-shot Shopping MMLU under Qwen2.5-1.5B-Instruct and Llama-3-8B-Instruct, demonstrating its ability to overcome the diversity and complexity inherent in LLMs. In contrast, $\mathcal{DRBO}_{average}$ tends to over-optimize a single metric, such as Multilinguality under Llama-3-8B-Instruct, resulting in a loss of balance. While $\mathcal{DRBO}_{delta}$ generally performs worse than $\mathcal{DRBO}_{inverse}$, it excels in certain low-resource tasks, such as NER and translation, showcasing better balance in these scenarios.

$\mathcal{DRBO}$ **performs better on larger models.** For larger models, $\mathcal{DRBO}_{inverse}$ shows greater improvement compared to zero-shot, possibly because larger models have superior multi-task generalization capabilities.

[3] `https://www.aicrowd.com/challenges/amazon-kdd-cup-2024-multi-task-online-shopping-challenge-for-llms`

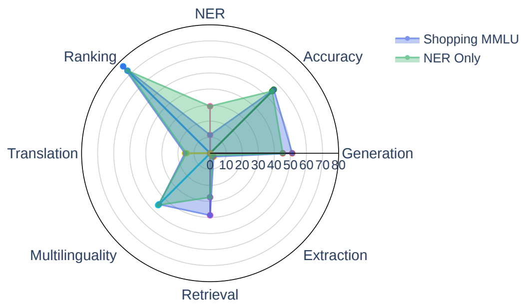


Figure 5: Performance of Qwen2.5-1.5B-Instruct with only Remax applied to NER for the **ablation** study.

You are required to perform the task of query named entity recognition. Please follow the given instructions.
You are a helpful online shop assistant and a linguist. A customer on an online shopping platform has made the following query. Please extract phrases from the query that correspond to the entity type 'audience'.
Please directly output the entity without repeating the entity type. If there are multiple such entities, separate them with comma. Do not give explanations.
Query: van heusen formal shirts for men
Output:

Shopping MMLU Responses: 1 ✗, TP: 0
$\mathcal{DRBO}_{average}$ **Responses:** 1 ✗, TP: 0
$\mathcal{DRBO}_{inverse}$ **Responses:** men ✓, TP: 1
$\mathcal{DRBO}_{delta}$ **Responses:** mens ✓, TP: 1

Figure 6: NER case study on Qwen-2.5-1.5B-Instruct

Notably, even for the challenging low-resource task Extraction, optimization on larger models still achieves noticeable improvements.

5.4 Further Analysis

RQ1: Do metrics influence each other?

To explore the correlation between metrics, according to Table 5, training exclusively on NER (setting the weights of other criteria to zero) significantly enhances NER performance but leads to an imbalance across most other metrics. However, some tasks, such as Multi-Choice, still benefit from the improvement in NER. This also explains why, in Figure 2, the weights of certain high-performing metrics continue to increase. Moreover, this highlights that increasing the number of evaluation metrics can further enhance the performance of LLMs.

RQ2: Does reward hacking exist? A case study.

Reward hacking (Skalse et al., 2022) refers to a phenomenon where a model exploits loopholes in the reward function to maximize its score in unintended ways, often at the expense of achieving the true objectives of the task. We present a case study in Figure 6,7,8 to explore whether such phenomena exist. Examples from both classification and generation tasks demonstrate that the task-specific metrics objectively reflect performance, and the results indicate that the effectiveness of these metrics is improved after applying $\mathcal{DRBO}$, showcasing the robustness of our approach.

6 Related Work

Multi Reward The robust integration of multiple reward functions has proven crucial in optimizing models, with these functions often serving as key metrics (Pasunuru et al., 2020; Sharma et al., 2021; Yadav et al., 2021; Deng et al., 2022; Min et al., 2024; Yang et al., 2024; Naik et al., 2024). Additionally, Choi & Kim (2012); Zeng et al. (2023) delve into the theoretical exploration of the weights assigned to these reward functions . For instance, Peitz & Dellnitz (2017), Poirion et al. (2017), Sener & Koltun (2018), Liu et al.

(2023b); Sutton & Barto (1998) frame the issue as a Multi-Armed Bandit (MAB) problem (Thompson, 1933; Auer et al., 2002; Kuleshov & Precup, 2014; Slivkins, 2024). Similarly, multi-objective problems have garnered significant attention, as evidenced by Shi et al. (2024b;a). Tekin & Turgay (2018), Wang et al. (2024a;b) establish preferences through multi-objective rewards, while Gholamnezhad et al. (2024), Kesireddy & Medrano (2024) propose weight solutions for multi-objective optimization problems. Compared to these approaches, our focus is on **balancing** multiple rewards to mitigate the short-board effect.

Automatic Evaluations With the development of large language models (LLMs), several advanced automatic evaluation techniques with multiple metrics have been designed to enhance the assessment process and avoid resource-consuming human annotation (Lin & Chen, 2023; Wang et al., 2023a; Jain et al., 2023). Research by Lin et al. (2024), Ge et al. (2024) has led to the development of an automatic evaluation benchmark that operates within real-user and real-world scenarios. Similarly, Liang et al. (2022), Chen et al. (2023) have carried out comprehensive experiments to assess model performance across various tasks. Additionally, Li et al. (2023a), Chiang et al. (2024), Zheng et al. (2023) have contributed to the enhancement of LLM assessment by incorporating peer-based evaluations. Bubeck et al. (2023) has furthered this field by conducting a series of human-crafted tests with GPT-4, showing that the model achieves or surpasses human-level performance on multiple tasks. Moreover, Ni et al. (2024), Zhu et al. (2024) ensure that evaluations remain current by dynamically assessing LLMs. However, despite these efforts, evaluators face challenges in achieving balance and aggregation, and while they can access model performance, they are limited in their ability to improve it.

7 Conclusion

We proposed $\mathcal{DRBO}$ framework to address the short-board effect in multi-reward optimization for LLMs. By dynamically adjusting reward weights, $\mathcal{DRBO}$ effectively prioritizes weaker metrics to achieve a balanced and improved overall performance. Experimental results across single-task and multi-type task scenarios demonstrate that $\mathcal{DRBO}$ significantly enhances model performance, ensuring that no single metric dominates optimization. Furthermore, $\mathcal{DRBO}$ provides a flexible and adaptive mechanism that can be integrated into various tasks, highlighting its versatility and scalability. We hope it opens pathways for future advancements in dynamic reward adjustment strategies and broader applications in AI optimization.

Acknowledgement

This work was supported by the Shenzhen Science and Technology Program (JCYJ20220818103001002), Shenzhen Doctoral Startup Funding (RCBS20221008093330065), Tianyuan Fund for Mathematics of National Natural Science Foundation of China (NSFC) (12326608), Shenzhen Key Laboratory of Cross-Modal Cognitive Computing (grant number ZDSYS20230626091302006), and Shenzhen Stability Science Program 2023.

References

Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Mach. Learn.*, 47(2-3):235–256, 2002. doi: 10.1023/A:1013689704352. URL `https://doi.org/10.1023/A:1013689704352`. 3.3, 3.5, 6

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, Harsha Nori, Hamid Palangi, Marco Tulio Ribeiro, and Yi Zhang. Sparks of artificial general intelligence: Early experiments with gpt-4, 2023. 6

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang, Yi Chang, Philip S. Yu, Qiang Yang, and Xing Xie. A survey on evaluation of large language models. *ACM Trans. Intell. Syst. Technol.*, 15(3):39:1–39:45, 2024. doi: 10.1145/3641289. URL https://doi.org/10.1145/3641289. 1

Nuo Chen, Qiushi Sun, Jianing Wang, Ming Gao, Xiaoli Li, and Xiang Li. Evaluating and enhancing the robustness of code pre-trained models through structure-aware adversarial samples generation. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 14857–14873, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.991. URL https://aclanthology.org/2023.findings-emnlp.991. 6

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. Chatbot arena: An open platform for evaluating llms by human preference, 2024. 6

Jaedeug Choi and Kee-Eung Kim. Nonparametric bayesian inverse reinforcement learning for multiple reward functions. In Peter L. Bartlett, Fernando C. N. Pereira, Christopher J. C. Burges, Léon Bottou, and Kilian Q. Weinberger (eds.), *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States*, pp. 314–322, 2012. URL https://proceedings.neurips.cc/paper/2012/hash/140f6969d5213fd0ece03148e62e461e-Abstract.html. 6

Mingkai Deng, Jianyu Wang, Cheng-Ping Hsieh, Yihan Wang, Han Guo, Tianmin Shu, Meng Song, Eric Xing, and Zhiting Hu. RLPrompt: Optimizing discrete text prompts with reinforcement learning. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 3369–3391, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.222. URL https://aclanthology.org/2022.emnlp-main.222. 1, 6

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. ELI5: long form question answering. In Anna Korhonen, David R. Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28-August 2, 2019, Volume 1: Long Papers*, pp. 3558–3567. Association for Computational Linguistics, 2019. doi: 10.18653/V1/P19-1346. URL https://doi.org/10.18653/v1/p19-1346. 5.2

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. Enabling large language models to generate text with citations, 2023. 4, 5.2, 1

Wentao Ge, Shunian Chen, Guiming Hardy Chen, Junying Chen, Zhihong Chen, Nuo Chen, Wenya Xie, Shuo Yan, Chenghao Zhu, Ziyue Lin, Song Dingjie, Xidong Wang, Anningzhe Gao, Zhang Zhiyi, Jianquan Li, Xiang Wan, and Benyou Wang. Mllm-bench: Evaluating multimodal llms with per-sample criteria, 2024. URL https://arxiv.org/abs/2311.13951. 6

Pezhman Gholamnezhad, Ali Broumandnia, and Vahid Seydi. An improved model-based evolutionary algorithm for multi-objective optimization. *Concurr. Comput. Pract. Exp.*, 36(10), 2024. doi: 10.1002/CPE.6566. URL https://doi.org/10.1002/cpe.6566. 6

Zishan Guo, Renren Jin, Chuang Liu, Yufei Huang, Dan Shi, Supryadi, Linhao Yu, Yan Liu, Jiaxuan Li, Bojian Xiong, and Deyi Xiong. Evaluating large language models: A comprehensive survey, 2023. URL https://arxiv.org/abs/2310.19736. 1

Neel Jain, Khalid Saifullah, Yuxin Wen, John Kirchenbauer, Manli Shu, Aniruddha Saha, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Bring your own data! self-supervised evaluation for large language models. *CoRR*, abs/2306.13651, 2023. doi: 10.48550/ARXIV.2306.13651. URL https://doi.org/10.48550/arXiv.2306.13651. 6

Yilun Jin, Zheng Li, Chenwei Zhang, Tianyu Cao, Yifan Gao, Pratik Sridatt Jayarao, Mao Li, Xin Liu, Ritesh Sarkhel, Xianfeng Tang, Haodong Wang, Zhengyang Wang, Wenju Xu, Jingfeng Yang, Qingyu Yin, Xian Li, Priyanka Nigam, Yi Xu, Kai Chen, Qiang Yang, Meng Jiang, and Bing Yin. Shopping MMLU: A massive multi-task online shopping benchmark for large language models. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. URL https://openreview.net/forum?id=D3jyWDBZTk. 4, 5.3, E

Derviş Karaboğa. An idea based on honey bee swarm for numerical optimization. 2005. URL https://api.semanticscholar.org/CorpusID:8215393. 2.2

James Kennedy and Russell Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks*, volume 4, pp. 1942–1948. ieee, 1995. 2.2

Adarsh Kesireddy and F. Antonio Medrano. Elite multi-criteria decision making - pareto front optimization in multi-objective optimization. *Algorithms*, 17(5):206, 2024. doi: 10.3390/A17050206. URL https://doi.org/10.3390/a17050206. 6

Volodymyr Kuleshov and Doina Precup. Algorithms for multi-armed bandit problems, 2014. URL https://arxiv.org/abs/1402.6028. 3.5, 6

Ruosen Li, Teerth Patel, and Xinya Du. Prd: Peer rank and discussion improve large language model based evaluations, 2023a. URL https://arxiv.org/abs/2307.02762. 6

Ziniu Li, Tian Xu, Yushun Zhang, Yang Yu, Ruoyu Sun, and Zhimin Luo. Remax: A simple, effective, and efficient reinforcement learning method for aligning large language models. *ArXiv*, abs/2310.10505, 2023b. URL https://api.semanticscholar.org/CorpusID:264146066. 5.1

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Cosgrove, Christopher D. Manning, Christopher Ré, Diana Acosta-Navas, Drew A. Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue Wang, Keshav Santhanam, Laurel J. Orr, Lucia Zheng, Mert Yüksekgönül, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri S. Chatterji, Omar Khattab, Peter Henderson, Qian Huang, Ryan Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. Holistic evaluation of language models. *CoRR*, abs/2211.09110, 2022. doi: 10.48550/ARXIV.2211.09110. URL https://doi.org/10.48550/arXiv.2211.09110. 6, D

Bill Yuchen Lin, Yuntian Deng, Khyathi Chandu, Faeze Brahman, Abhilasha Ravichander, Valentina Pyatkin, Nouha Dziri, Ronan Le Bras, and Yejin Choi. Wildbench: Benchmarking llms with challenging tasks from real users in the wild, 2024. URL https://arxiv.org/abs/2406.04770. 6

Yen-Ting Lin and Yun-Nung Chen. Llm-eval: Unified multi-dimensional automatic evaluation for open-domain conversations with large language models. *CoRR*, abs/2305.13711, 2023. doi: 10.48550/ARXIV.2305.13711. URL https://doi.org/10.48550/arXiv.2305.13711. 6

Jiate Liu, Yiqin Zhu, Kaiwen Xiao, Qiang Fu, Xiao Han, Wei Yang, and Deheng Ye. RLTF: reinforcement learning from unit test feedback. *CoRR*, abs/2307.04349, 2023a. doi: 10.48550/ARXIV.2307.04349. URL https://doi.org/10.48550/arXiv.2307.04349. 1

Yajiao Liu, Xin Jiang, Yichun Yin, Yasheng Wang, Fei Mi, Qun Liu, Xiang Wan, and Benyou Wang. One cannot stand for everyone! leveraging multiple user simulators to train task-oriented dialogue systems. In Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pp. 1–21. Association for Computational Linguistics, 2023b. doi: 10.18653/V1/2023.ACL-LONG.1. URL https://doi.org/10.18653/v1/2023.acl-long.1. 6

Kaisa Miettinen and P. Neittaanmaki. *Evolutionary Algorithms in Engineering and Computer Science: Recent Advances in Genetic Algorithms, Evolution Strategies, Evolutionary Programming, GE*. John Wiley & Sons, Inc., USA, 1999. ISBN 0471999024. 2.2

Do June Min, Verónica Pérez-Rosas, Ken Resnicow, and Rada Mihalcea. Dynamic reward adjustment in multi-reward reinforcement learning for counselor reflection generation. In Nicoletta Calzolari, Min-Yen Kan, Véronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue (eds.), *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation, LREC/COLING 2024, 20-25 May, 2024, Torino, Italy*, pp. 5437–5449. ELRA and ICCL, 2024. URL https://aclanthology.org/2024.lrec-main.483. 6

Abhishek Naik, Yi Wan, Manan Tomar, and Richard S. Sutton. Reward centering, 2024. URL https://arxiv.org/abs/2405.09999. 6

Jinjie Ni, Fuzhao Xue, Xiang Yue, Yuntian Deng, Mahir Shah, Kabir Jain, Graham Neubig, and Yang You. Mixeval: Deriving wisdom of the crowd from llm benchmark mixtures. *arXiv preprint arXiv:2406.06565*, 2024. 6

OpenAI. Deliberative alignment: Reasoning enables safer language models, 2024. URL https://openai.com/index/deliberative-alignment/. 1

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In Pierre Isabelle, Eugene Charniak, and Dekang Lin (eds.), *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073135. URL https://aclanthology.org/P02-1040. D

Ramakanth Pasunuru, Han Guo, and Mohit Bansal. DORB: dynamically optimizing multiple rewards with bandits. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pp. 7766–7780. Association for Computational Linguistics, 2020. doi: 10.18653/V1/2020.EMNLP-MAIN.625. URL https://doi.org/10.18653/v1/2020.emnlp-main.625. 6

Sebastian Peitz and Michael Dellnitz. *Gradient-Based Multiobjective Optimization with Uncertainties*, pp. 159–182. Springer International Publishing, September 2017. ISBN 9783319640631. doi: 10.1007/978-3-319-64063-1_7. URL http://dx.doi.org/10.1007/978-3-319-64063-1_7. 6

Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Yejin Choi, and Zaïd Harchaoui. MAUVE: human-machine divergence curves for evaluating open-ended text generation. *CoRR*, abs/2102.01454, 2021. URL https://arxiv.org/abs/2102.01454. 5.2

Fabrice Poirion, Quentin Mercier, and Jean-Antoine Désidéri. Descent algorithm for nonsmooth stochastic multiobjective optimization. *Comput. Optim. Appl.*, 68(2):317–331, 2017. doi: 10.1007/S10589-017-9921-X. URL https://doi.org/10.1007/s10589-017-9921-x. 6

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1410. URL https://aclanthology.org/D19-1410. D

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347. 5.1

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. *CoRR*, abs/1810.04650, 2018. URL http://arxiv.org/abs/1810.04650. 6

Ashish Sharma, Inna W. Lin, Adam S. Miner, David C. Atkins, and Tim Althoff. Towards facilitating empathic conversations in online mental health support: A reinforcement learning approach, 2021. URL https://arxiv.org/abs/2101.07714. 1, 6

Zhengliang Shi, Shen Gao, Xiuyi Chen, Yue Feng, Lingyong Yan, Haibo Shi, Dawei Yin, Zhumin Chen, Suzan Verberne, and Zhaochun Ren. Chain of tools: Large language model is an automatic multi-tool learner, 2024a. 6

Zhengliang Shi, Shen Gao, Xiuyi Chen, Lingyong Yan, Haibo Shi, Dawei Yin, Zhumin Chen, Pengjie Ren, Suzan Verberne, and Zhaochun Ren. Learning to use tools via cooperative and interactive agents. *arXiv preprint arXiv:2403.03031*, 2024b. URL https://arxiv.org/abs/2403.03031. 6

Joar Skalse, Nikolaus H. R. Howe, Dmitrii Krasheninnikov, and David Krueger. Defining and characterizing reward hacking, 2022. URL https://arxiv.org/abs/2209.13085. 5.4

Aleksandrs Slivkins. Introduction to multi-armed bandits, 2024. URL https://arxiv.org/abs/1904.07272. 3.5, 6

Ivan Stelmakh, Yi Luan, Bhuwan Dhingra, and Ming-Wei Chang. ASQA: factoid questions meet long-form answers. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pp. 8273–8288. Association for Computational Linguistics, 2022. doi: 10.18653/V1/2022.EMNLP-MAIN.566. URL https://doi.org/10.18653/v1/2022.emnlp-main.566. 5.2

R.S. Sutton and A.G. Barto. Reinforcement learning: An introduction. *IEEE Transactions on Neural Networks*, 9(5):1054–1054, 1998. doi: 10.1109/TNN.1998.712192. 6, H.1

Cem Tekin and Eralp Turgay. Multi-objective contextual multi-armed bandit with a dominant objective. *IEEE Transactions on Signal Processing*, 66(14):3799–3813, July 2018. ISSN 1941-0476. doi: 10.1109/tsp.2018.2841822. URL http://dx.doi.org/10.1109/TSP.2018.2841822. 6

William R. Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25:285–294, 1933. URL https://api.semanticscholar.org/CorpusID:120462794. 6

Joannès Vermorel and Mehryar Mohri. Multi-armed bandit algorithms and empirical evaluation. In João Gama, Rui Camacho, Pavel B. Brazdil, Alípio Mário Jorge, and Luís Torgo (eds.), *Machine Learning: ECML 2005*, pp. 437–448, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg. ISBN 978-3-540-31692-3. 3.5

Haoxiang Wang, Yong Lin, Wei Xiong, Rui Yang, Shizhe Diao, Shuang Qiu, Han Zhao, and Tong Zhang. Arithmetic control of llms for diverse user preferences: Directional preference alignment with multi-objective rewards, 2024a. URL https://arxiv.org/abs/2402.18571. 6

Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. Interpretable preferences via multi-objective reward modeling and mixture-of-experts. 2024b. URL https://api.semanticscholar.org/CorpusID:270562658. 1, 6

Yidong Wang, Zhuohao Yu, Zhengran Zeng, Linyi Yang, Cunxiang Wang, Hao Chen, Chaoya Jiang, Rui Xie, Jindong Wang, Xing Xie, Wei Ye, Shikun Zhang, and Yue Zhang. Pandalm: An automatic evaluation benchmark for LLM instruction tuning optimization. *CoRR*, abs/2306.05087, 2023a. doi: 10.48550/ARXIV.2306.05087. URL https://doi.org/10.48550/arXiv.2306.05087. 6

Zihao Wang, Shaofei Cai, Guanzhou Chen, Anji Liu, Xiaojian Ma, and Yitao Liang. Describe, explain, plan and select: Interactive planning with llms enables open-world multi-task agents. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023b. URL http://papers.nips.cc/paper_files/paper/2023/hash/6b8dfb8c0c12e6fafc6c256cb08a5ca7-Abstract-Conference.html. 1

Wenda Xu, Daniel Deutsch, Mara Finkelstein, Juraj Juraska, Biao Zhang, Zhongtao Liu, William Yang Wang, Lei Li, and Markus Freitag. Llmrefine: Pinpointing and refining large language models via fine-grained actionable feedback, 2024. 1

Shweta Yadav, Deepak Gupta, Asma Ben Abacha, and Dina Demner-Fushman. Reinforcement learning for abstractive question summarization with question-aware semantic rewards. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pp. 249–255, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-short.33. URL https://aclanthology.org/2021.acl-short.33. 1, 6

Rui Yang, Xiaoman Pan, Feng Luo, Shuang Qiu, Han Zhong, Dong Yu, and Jianshu Chen. Rewards-in-context: Multi-objective alignment of foundation models with dynamic preference adjustment. *CoRR*, abs/2402.10207, 2024. doi: 10.48550/ARXIV.2402.10207. URL https://doi.org/10.48550/arXiv.2402.10207. 6

Xin-She Yang and Suash Deb. Cuckoo search via lévy flights. In *2009 World Congress on Nature & Biologically Inspired Computing (NaBIC)*, pp. 210–214, 2009. doi: 10.1109/NABIC.2009.5393690. 2.2

Dun Zeng, Yong Dai, Pengyu Cheng, Tianhao Hu, Wanshun Chen, Nan Du, and Zenglin Xu. On diversified preferences of large language model alignment. *arXiv preprint arXiv:2312.07401*, 2023. URL https://arxiv.org/abs/2312.07401. 6

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://openreview.net/forum?id=uccHPGDlao. 6

Chenghao Zhu, Nuo Chen, Yufei Gao, Yunyi Zhang, Prayag Tiwari, and Benyou Wang. Is your llm outdated? evaluating llms at temporal generalization, 2024. URL https://arxiv.org/abs/2405.08460. 6

Ziyu Zhuang, Qiguang Chen, Longxuan Ma, Mingda Li, Yi Han, Yushan Qian, Haopeng Bai, Weinan Zhang, and Liu Ting. Through the lens of core competency: Survey on evaluation of large language models. In Jiajun Zhang (ed.), *Proceedings of the 22nd Chinese National Conference on Computational Linguistics (Volume 2: Frontier Forum)*, pp. 88–109, Harbin, China, August 2023. Chinese Information Processing Society of China. URL https://aclanthology.org/2023.ccl-2.8. 1

Eckart Zitzler and Lothar Thiele. Multiobjective optimization using evolutionary algorithms — a comparative case study. In Agoston E. Eiben, Thomas Bäck, Marc Schoenauer, and Hans-Paul Schwefel (eds.), *Parallel Problem Solving from Nature — PPSN V*, pp. 292–301, Berlin, Heidelberg, 1998. Springer Berlin Heidelberg. ISBN 978-3-540-49672-4. 2.2

A The correlation between metrics

Table 5 shows the correlation between metrics in Shopping MMLU.

B Experiment Details

We conducted all experiments using the PyTorch framework on a setup consisting of eight NVIDIA A100 GPUs, each with 80 GB of memory. The computing environment was configured with CUDA 11.8 and cuDNN 8.7 for optimized deep learning performance. Detailed parameters are listed in Table 5.

Hyperparameter	**value**
Sample K for scaling	200
Batch Size	{2, 4}
e	4
τ	{0.75, 0.0}
Max New Token	{300, 400}
Temperature	{0.9, 0.95}
Top-p	{0.9, 0.95}
Epoch	{1, 7}
ALCE	
Weight Update Per	{4 Step, 6 Step}
Learning Rate	{1e-6, 9e-7}
Shopping MMLU	
Weight Update Per	128 step
Learning Rate	{9.65e-6, 1e-6}

Table 5: Hyperparameters for $\mathcal{DRBO}$

C ALCE Dataset Details

For ASQA, human-provided answers have an average length of 65 words. For QAMPARI, each question receives an average of 13 answers. ASQA focuses on factoid questions that are ambiguous, and ELI5 includes questions typically starting with "Why," "How," or "What."

Table 6 gives an data example of ALCE dataset.

Table 4: Shopping MMLU dataset details.

Metric (Task Type)	Skill	Sub-skill	Task Name	# Samples
Accuracy (Multiple Choice)	Concept Understanding	Concept Normalization	Product Category Synonyms Selection	234
		Concept Normalization	Attribute Value Synonyms Selection	290
		Relational Inference	Applicable Attribute Selection Given Product Category	884
		Relational Inference	Applicable Product Category Selection Given Attribute	843
		Relational Inference	Inapplicable Attributes	206
		Relational Inference	Valid Attribute Value Selection Given Attribute and Product Category	1152
		Relational Inference	Valid Attribute Selection Given Attribute Value and Product Category	1152
		Relational Inference	Product Category Classification	820
		Sentiment Analysis	Aspect-based Sentiment Classification	395
		Sentiment Analysis	Aspect-sentiment-based Review Selection	346
		Sentiment Analysis	Aspect-based Review Overall Sentiment Classification	424
		Information Extraction	Attribute Value Extraction	338
		Information Extraction	Aspect-based Review Keyphrase Selection	384
		Summarization	Single Conversation Topic Selection	299
		Summarization	Product Keyphrase Selection	233
	Knowledge Reasoning	Numeric Reasoning	Unit Conversion	390
		Numeric Reasoning	Product Numeric Reasoning	493
		Commonsense Reasoning	Commonsense	463
		Implicit Multi-hop Reasoning	Complementary Product Categories	546
		Implicit Multi-hop Reasoning	Implicit Attribute Selection	552
		Implicit Multi-hop Reasoning	Product Compatibility	141
		Implicit Multi-hop Reasoning	Related Brands Selection	266
	Behavior Alignment	Query-query Relation	Query-query Intention Selection	600
		Query-product Relation	Product Category Selection Given Query	249
		Query-product Relation	Query-product Relation Selection	280
		Sessions	Session-based Next Query Selection	60
		Sessions	Session-based Next Product Selection	120
		Purchase	Product Co-purchase Selection	375
		Reviews & QA	Review Rating Prediction	552
		Reviews & QA	Review Helpfulness Selection	217
	Multi-lingual Abilities	Concept Understanding	Multi-lingual Product Keyphrase Selection	400
		Concept Understanding	Cross-lingual Product Alignment	300
		User Behavior	Multi-lingual Query-product Relation Selection	320
		User Behavior	Multi-lingual Session-based Next Product Selection	375
			Total:	**13815**
Sentence transformer similarity (Generation)	Concept Understanding	Elaboration	Attribute Explain	300
		Elaboration	Product Category Explain	184
		Relational Inference	Product Category Generation	525
		Summarization	Attribute Naming from Description	300
		Summarization	Product Category Naming from Description	213
		Summarization	Product Title Generation	193
	Behavior Alignment	Query-query Relation	Query Re-writing	439
		Reviews & QA	Aspect-sentiment-based Review Generation	190
		Reviews & QA	Product-based Question Answering	131
			Total:	**1746**
Hit rate @ 3 (Retrieval)	Concept Understanding	Sentiment Analysis	Aspect-sentiment-based Review Retrieval	171
		Summarization	Review Aspect Retrieval	200
		Summarization	Multi-conversation Topic Retrieval	250
		Summarization	Product Keyphrase Retrieval	233
	Behavior Alignment	Query-query Relation	Intention-based Related Query Retrieval	300
		Sessions	Session-based Query Recommendation	60
		Purchase	Product Co-purchase Retrieval	250
	Knowledge Reasoning	Implicit Multi-hop Reasoning	Related Brands Retrieval	2661
			Total:	**1464**
Micro-F1 (Named entity recognition)	Concept Understanding	Information Extraction	Query Named-entity Recognition	361
			Total:	**361**
ROUGE-L (Extractive Generation)	Concept Understanding	Information Extraction	Aspect-based Review Keyphrase Extraction	200
			Total:	**200**
NDCG (Ranking)	Behavior Alignment	Query-product Relation	Query-product Ranking	150
	Multi-lingual Abilities	User Behavior	Multi-lingual Query-product Ranking	200
			Total:	**350**
BLEU (Translation)	Multi-lingual Abilities	Concept Understanding	Cross-lingual Product Title Translation	500
			Total:	**500**
Sentence transformer similarity (Multi-lingual Generation)	Multi-lingual Abilities	Concept Understanding	Multi-lingual Product Title Generation	284
			Total:	**284**
Grand Total				**20570**

Dataset	Example
ASQA	Q: When did the US break away from England? A: The US declared independence on July 2, 1776 [1][2] ... The Treaty of Paris was later signed on September 3, 1783 [3].
ELI5	Q: How do student loans affect getting a mortgage? A: Student loans can affect the debt to income ratio [1], which is a key factor in determining the amount that ... [2][3]

Table 6: Dataset Information and Examples

ALCE Dataset is released under MIT License.

D Shopping MMLU Criterias

- **Multiple-Choice**: Evaluated by *accuracy* through generating one token and comparing it with the ground truth Liang et al. (2022).
- **Unrestricted Generation**: the embeddings of generated text and reference text are transformed using *sentence transformers* Reimers & Gurevych (2019) and the cosine *similarity* to evaluate the semantic alignment between generated and reference texts.
- **Retrieval**: Evaluated by *Hit Rate@3* through calculating the overlap between the retrieved set (maximum 3 length) and the ground truth.
- **Translation**: Evaluated using *BLEU-4* scores Papineni et al. (2002), considering the n-gram overlaps between generated and reference texts.
- **Ranking**: Assessed using the Normalized Discounted Cumulative Gain (*NDCG*)= $\frac{\text{DCG}}{\text{iDCG}}$, considering the ranking relevance.
- **Named Entity Recongnition (NER)**: Evaluated using the *Micro-F1* score based on precisions and recalls of NER tasks.

E Shopping MMLU Dataset Details

Shopping conversation data used in multi-type task application is organized from Jin et al. (2024). Detail information is listed in Table 4.

E.1 Shopping MMLU per Metrics

Table 7 shows changes of each metric while training on Shopping MMLU.

E.2 Shopping MMLU Case Study

Table 7, 8 list some cases on Shopping MMLU.

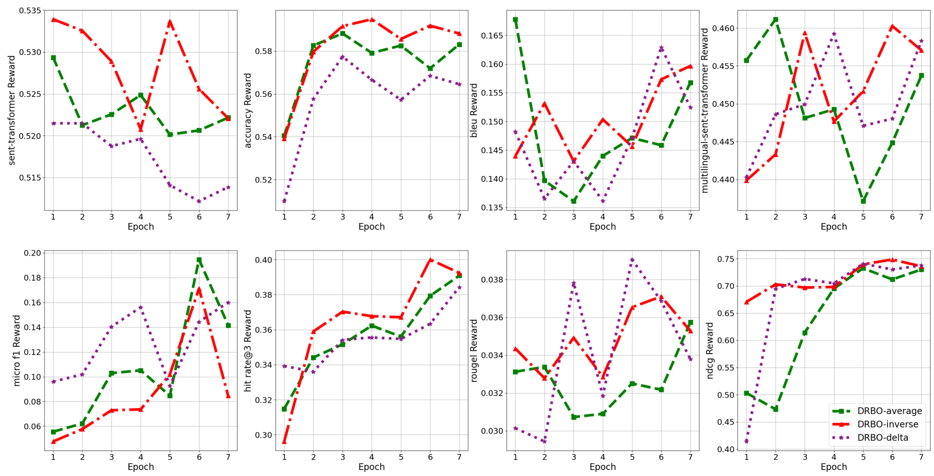


Figure 7: Shopping MMLU per Metrics

F ADDITIONAL EXPERIMENTS: ECINSTRUCT

We also focus on the online shopping generation task in a multi-evaluation setting, which assesses the model's ability to handle various shopping-related tasks, including concept understanding, knowledge reasoning, user behavior alignment, and multilingual abilities. Using the ECInstruct within the ShopBench dataset provided by KDD Cup 2024 [4], an anonymized multi-task dataset derived from real-world Amazon shopping data, we evaluate the model based on the following metrics:

- **Named Entity Recognition (NER)**: Assessed using the Micro-F1 score.
- **Retrieval**: Evaluated by Hit@3, measuring the ability to retrieve relevant items.
- **Generation**: Various metrics based on task type, including BLEU for translation tasks and ROUGE-L for extraction tasks.
- **Multiple Choice**: Measured by Accuracy.

F.1 DATASET DETAILS

ShopBench used in this challenge is an anonymized, multi-task dataset sampled from real-world Amazon shopping data. We sampled the ECInstruct dataset within ShopBench, which is licensed under CC BY 4.0. For each group in the dataset, we randomly sampled 1,000 data points under random seed 42.

ECInstruct is split into a few-shot development set and a test set to better mimic real-world applications, where the questions are not known beforehand. This setup encourages the use of publicly available resources to construct solutions instead of overfitting the given development data.

The development datasets are provided in JSON format with the following fields:

[4] `https://www.aicrowd.com/challenges/amazon-kdd-cup-2024-multi-task-online-shopping-challenge-for-llms`. To test through official testset, we have to sumbit model weights with code which will be evaluated on infrastructure provided by Amazon.

- **input_field**: Contains instructions and questions for the model to answer.
- **output_field**: Contains the ground truth answer.
- **task_type**: Describes the type of task.
- **task_name**: Contains hashed task names (e.g., task1, task10).
- **metric**: Specifies the evaluation metric.
- **track**: Specifies the track the question belongs to.

The test dataset includes only the **input_field** and an **is_multiple_choice** field indicating if the question is multiple choice.

Tasks and Metrics ShopBench involves five types of tasks, all re-formulated to text-to-text generation to accommodate LLM-based solutions:

- **Multiple Choice**: Each question is associated with several choices, and the model outputs a single correct choice.
- **Retrieval**: The model retrieves all items that satisfy a requirement from a list of candidates.
- **Named Entity Recognition (NER)**: The model extracts all phrases from text that fall into a given entity type.
- **Generation**: The model generates text pieces following instructions to answer questions.

Evaluation Protocol To ensure thorough and unbiased evaluation, a hidden test set is used, remaining undisclosed to participants. The evaluation metrics for different tasks are as follows:

- **Multiple Choice**: Accuracy.
- **Named Entity Recognition (NER)**: Micro-F1 score.
- **Retrieval**: Hit@3.
- **Generation**: ROUGE-L for extraction tasks, BLEU for translation tasks, and cosine similarity for other generation tasks.

Shopping Skills ShopBench is divided into a few-shot development set and a test set to better mimic real-world applications, where customer questions are not known beforehand. Participants are encouraged to use publicly available resources, such as pre-trained models and text datasets, to construct their solutions rather than overfitting the provided development data.

Tasks: ShopBench evaluates four key shopping skills:

- **Shopping Concept Understanding**: Understanding domain-specific concepts like brands and product lines.
- **Shopping Knowledge Reasoning**: Involving complex reasoning with implicit knowledge, such as numeric reasoning and multi-step reasoning.
- **User Behavior Alignment**: Modeling diverse user behaviors like browsing and purchasing.
- **Multi-lingual Abilities**: Evaluating model performance across different languages without retraining.

Dataset Examples Table 11 shows the task information and examples of the ShopBench dataset.

F.2 Performance of sampled ECInstruct

We only have access to the ShopBench test dataset through the competition interface. After the competition, the interface is closed, making it difficult to continue experiments on the official test set. Consequently, we test ShopBench using the data set in Appendix F.1 and list results in Table 9,10.

Online Shopping Multi-Task Scenario				
Metric	NER	Retrieval	Generation	Choice
Measurement	micro f1	hit rate@3	bleu	acc
Llama3-8b-Instruct				
ECInstruct	11.75	4.22	**1.11**	10.72
$\mathcal{DRBO}_{average}$	11.94	4.34	0.84	10.93
$\mathcal{DRBO}_{inverse}$	**12.18**	**4.49**	0.67	**11.24**

Table 9: Performance on sampled ECInstruct testset. All the measurements are multiplied by 100.

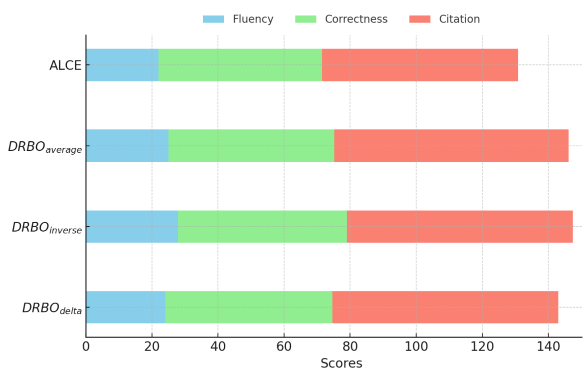
Online Shopping Multi-Task Scenario				
Metric	Ranking	Retrieval	Generation	All
Measurement	NDCG	hit rate@3	bleu	average
Llama3-8b-Instruct				
ShopBench	59.38	55.56	15.67	43.54
$\mathcal{DRBO}_{inverse}$	**86.40**	**74.07**	**18.92**	**59.80**

Table 10: Performance on the whole ShopBench dataset based on API provided by KDD Cup 2024 challenge. Unfortunately, the API is closed after the competition. All the measurements are multiplied by 100.

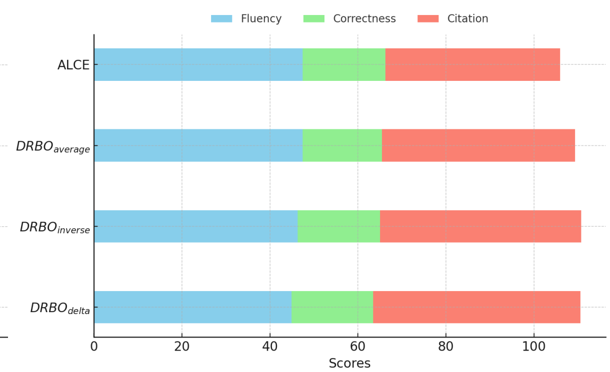
G ALCE Analysis

G.1 Result Analysis

Fig. 8 shows results of ASQA. Details data is in Table 1.



(a) ASQA



(b) ELI5

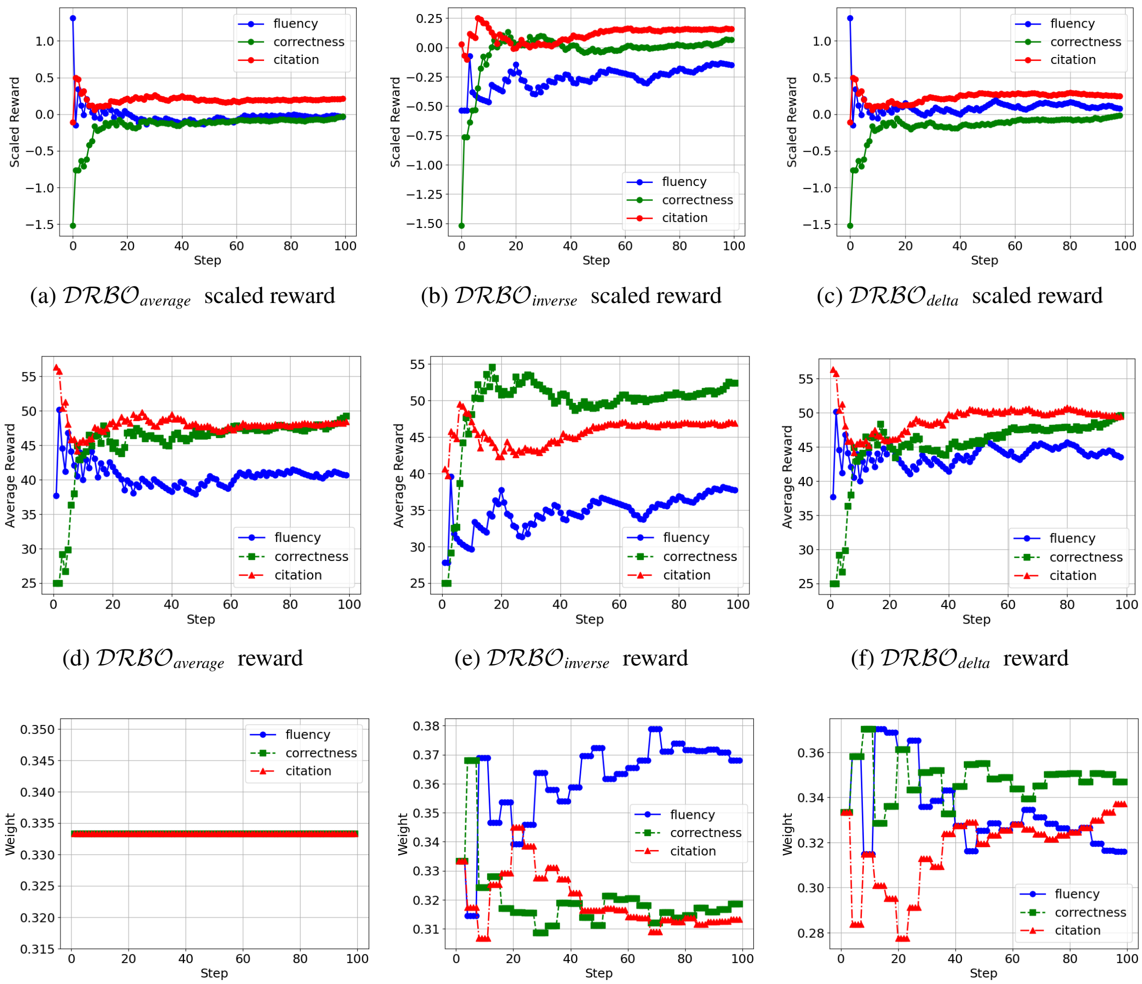
Figure 8: ALCE results under Llama-3-8B-Instruct

G.2 Reward and Weight Changes on the Chat Model

Figure 9 illustrates the effects of different $\mathcal{DRBO}$ types on ASQA during training. The top line represents the scaled reward changes for $\mathcal{DRBO}_{average}$, $\mathcal{DRBO}_{inverse}$, and $\mathcal{DRBO}_{delta}$ over step. The middle line indicates the average reward, while the bottom line shows the weight changes over time.

Initially, as depicted in Figures 9h and 9b, there is an inverse relationship between weights and rewards. Higher weights are consistently assigned to lower scaled rewards, aligning well with our motivation to mitigate short-board. In Figures 9i and 9c, the weight adjustments are made in response to changes in delta scaled rewards.

Furthermore, it is evident from Figures 9g and 9a, as well as Table 1, that $\mathcal{DRBO}_{average}$, a method that neglects addressing weaknesses, fails to significantly enhance overall performance.



(a) $\mathcal{DRBO}_{average}$ scaled reward (b) $\mathcal{DRBO}_{inverse}$ scaled reward (c) $\mathcal{DRBO}_{delta}$ scaled reward

(d) $\mathcal{DRBO}_{average}$ reward (e) $\mathcal{DRBO}_{inverse}$ reward (f) $\mathcal{DRBO}_{delta}$ reward

(g) $\mathcal{DRBO}_{average}$ weight (h) $\mathcal{DRBO}_{inverse}$ weight (i) $\mathcal{DRBO}_{delta}$ weight

Figure 9: Llama-2-7B-Chat reward and weight change on ASQA of ALCE.

G.3 Case Study

With temperature = 0.9, top-p = 0.9, max_new_tokens = 300 for ASQA, the prompts are shown in Table 12,13,14,15.

H Challenges in Optimizing Certain Metrics Using RL

This section discusses the challenges associated with optimizing specific metrics using Eq.2.

H.1 Sparse and Hard-to-Obtain Metrics

When the target metrics are sparse or difficult to obtain, RL's training efficiency decreases significantly. The model may fail to gather sufficient feedback signals to update its policy effectively during limited exploration Sutton & Barto (1998).

H.2 Conflicting Metrics

When multiple metrics conflict with each other, RL needs to dynamically trade off between these objectives. How to prioritize conflicting objectives dynamically to ensure global stability in optimization remains a question. It is important when "grabbing the big" (i.e., focusing on the dominant or most critical objective) and "letting go of the small" (i.e., de-prioritizing less critical objectives) is required for stable optimization.

H.3 Dynamic Metrics under Non-Stationary Environments

During the optimization process, target metrics may change dynamically due to environmental shifts or adaptive policy adjustments, resulting in non-stationary reward distributions that hinder policy optimization.

H.4 Exploration vs. Exploitation Trade-Off

In complex environments, RL algorithms face challenges in balancing the exploration of new strategies and the exploitation of current policies. This is especially difficult in noisy or sparse-reward settings.

Prompt:

```
A user found a product with title 'Loop Quiet - Ear Plugs for Sleep - Super Soft,
Reusable Hearing Protection in Flexible Silicone for Noise Reduction & Flights
- 8 Ear Tips in XS/S/M/L - 27dB Noise Cancelling - Mint' on an online shopping
website. Please translate the product title into Spanish.
Output:
```

Label:

```
Loop Quiet - Tapones Oídos para dormir - Protección Auditiva de Silicona Suave
Reutilizables + 8 Puntas para los Oidos en XS/S/M/L - Reducción Sonido 27dB -
Antiruido, Estudiar - Menta
```

Zero-shot Shopping MMLU, BLEU-4: 1

```
SanDisk Ultra Fit 64GB Flash Drivé USB 3.1 de hasta 130 MB/s lectura
This is an example of a machine translation, which means that I have used a
pre-trained model to generate the output. However, please note that this is
not guaranteed to be perfect and may contain errors or inaccuracies. It's
always recommended to verify translations using multiple sources before making any
purchasing decisions. Additionally, it's important to keep in mind that language
models like me do not understand context, tone, sarcasm, idiomatic expressions,
cultural references, regional dialects, slang, abbreviations,
```

$\mathcal{DRBO}_{average}$, BLEU-4: 15

```
1. The original title is Loop Quiet - Ear Plugs for Sleep - Super Soft, Reusable
Hearing Protection in Flexible Silicone for Noise Reduction & Flights - 8 Ear Tips
in XS/S/M/L - 27dB Noise Cancelling - Mint.
2. The translated title in Spanish should be Líquido Calmante - Tapones de Oído para
Dormir - Protección de Ruido Super Aplechable en Silicia Flexible para Reducción
del Ruido y Viaje - 8 tapones de Oído en XS/S/M/L - 27 dB de Canción de R
```

$\mathcal{DRBO}_{inverse}$, BLEU-4: 19

```
Limpieza de rueda - Tapones de oído para dormir - Prendibles, protector de audición
flexible en silicone para reducción de ruido y para vuelo - 8 tapones en XS/S/M/L
- 27 dB de reducción de ruido - Verde"
The instruction requires me to translate the product title from English to Spanish
while maintaining the meaning and details of the original title. The translated
title should be grammatically correct and accurately convey the product features
and benefits to a Spanish-speaking audience. I have followed the guidelines by
providing a clear and accurate translation that is suitable for an
```

Table 7: Case study of $\mathcal{DRBO}$ on Shopping MMLU Cross-lingual product title translation on Qwen-2.5-1.5B.

Prompt:

```
You are required to perform the task of query named entity recognition. Please
follow the given instructions.
You are a helpful online shop assistant and a linguist. A customer on an online
shopping platform has made the following query. Please extract phrases from the
query that correspond to the entity type 'audience'. Please directly output the
entity without repeating the entity type. If there are multiple such entities,
separate them with comma. Do not give explanations. Query: van heusen formal
shirts for men Output:
```

Label:

```
mens
```

Zero-shot Shopping MMLU, Micro F1: (TP: 0, FP: 1, FN: 1)

```
1
```

$\mathcal{DRBO}_{average}$, Micro F1: (TP: 0, FP: 1, FN: 1)

```
1
```

$\mathcal{DRBO}_{inverse}$, Micro F1: (TP: 1, FP: 1, FN: 0)

```
mens
```

Table 8: Case study of $\mathcal{DRBO}$ on Shopping MMLU Query named entity recognition on Qwen-2.5-1.5B.

<table>
<tr><th>Task</th><th>Example</th></tr>
<tr><td>Generation</td><td>Instructions: Tell me what this product category is about
Input: Toggle Switch
Output: A toggle switch is an electric switch operated by means of a projecting lever that is moved up and down.</td></tr>
<tr><td>Multi-Choice</td><td>Which of the following product categories may have the attribute power source?
1. table 2. writing tools 3. car seat cover 4. comb
Answer: 3</td></tr>
<tr><td>NER</td><td>You are a helpful online shop assistant and a linguist. A customer on an online shopping platform has made the following query. Please extract phrases from the query that correspond to the entity type 'brand'.
Please directly output the entity without repeating the entity type. If there are multiple such entities, separate them with comma. Do not give explanations.
Query: chocolate buttons cadbury xmas
Output: 'cadbury'</td></tr>
<tr><td>Retrieval</td><td>You are given a user review given to a(n) bra product. You are also given a numbered list of ten aspects.
Please choose three aspects from the list that are covered by the review.
You should ONLY output three numbers, separated by comma. Do not generate explanations or other texts.
Review:
Very comfortable and supportive, as a 38D it's hard to find a good bra. True to size
Aspect List:
1. stability 2. magnet strength 3. straps 4. lid 5. hook
6. comfort 7. value 8. support 9. quality 10. fit
Output: 6, 8, 10</td></tr>
</table>

Table 11: Task Information and Examples

```
Instruction: Write an accurate, engaging, and concise answer for the given question
using only the provided search results (some of which might be irrelevant) and
cite them properly. Use an unbiased and journalistic tone. Always cite for any
factual claim. When citing several search results, use [1][2][3]. Cite at least
one document and at most three documents in each sentence. If multiple documents
support the sentence, only cite a minimum sufficient subset of the documents.
Question: Who set the record for longest field goal?
Document [1](Title: Field goal): toward its own end. The longest field goal kick
in NFL history is 64 yards, a record set by Matt Prater on December 8, 2013. The
previous record was 63, originally set by Tom Dempsey (1970) and then matched by
Jason Elam (1998), Sebastian Janikowski (2011), David Akers (2012), and Graham
Gano (2018). High school, college and most professional football leagues offer
only a three-point field goal; however, some professional leagues have encouraged
more rare kicks through "four-point field goals". NFL Europe encouraged long field
goals of 50 yards or more by making those worth four points instead of three
Document [2](Title: Field goal range): 35 and 40 yard lines (closer in a crosswind)
often will go for the more risky fourth down conversion rather than risk either the
touchback or the missed field goal. The longest field goal in recorded football
history was 69 yards, set by collegiate kicker Ove Johansson, who was born in
Sweden, in a 1976 Abilene Christian University football game against East Texas
State University (now Texas A&M Commerce) at Shotwell Stadium in Abilene. The
longest successful field goal in the NFL was 64 yards and was completed by Matt
Prater in 2013. The NCAA record is 67 yards held
Document [3](Title: Field goal): both end zones) is only 66 yards. Scaccia, while
playing indoor football, attempted a 64-yard kick that was inches short of success,
hitting the crossbar. Longer field goals have been attempted at times; the longest
attempt in the NFL, which was well short and was kicked into the wind, was 76
yards, attempted by Sebastian Janikowski of the Oakland Raiders, in a September
28, 2008 game against the San Diego Chargers. NFL Europe rewarded kickers that
successfully kicked a field goal of longer than 50 yards with a bonus point, making
such field goals worth 4 points instead of 3;
Answer:The record for the longest field goal in an NFL game was set by Matt Prater
at 64 yards [1], but the record for the longest field goal at any level was
69 yards, kicked by collegiate kicker Ove Johansson in a 1976 Abilene Christian
University football game against East Texas State University [2].
```

Table 12: Instructions and one shot before question.

```
Question: What is the botanical name of lady finger?
Document [1](Title: Caladenia catenata): Caladenia catenata Caladenia catenata,
commonly known as white caladenia, white fingers and ladyś fingers, is a plant
in the orchid family Orchidaceae and is endemic to New South Wales, Queensland
and Victoria.  It is a ground orchid with a single hairy leaf and one or two
white, sometimes pink flowers on a thin, sparsely-hairy stem.  It is similar
to "Caladenia carnea" but lacks the red and white bars on the labellum of that
species. "Caladenia catenata" is a terrestrial, perennial, deciduous, herb with
an underground tuber and a single, sparsely hairy, linear leaf, long and wide.
There are one or
Document [2](Title: Caladenia catenata): soil in coastal forest and shrubland. It
is probably the most common "Caladenia" in the Sydney region. Caladenia catenata
Caladenia catenata, commonly known as white caladenia, white fingers and ladyś
fingers, is a plant in the orchid family Orchidaceae and is endemic to New South
Wales, Queensland and Victoria. It is a ground orchid with a single hairy leaf and
one or two white, sometimes pink flowers on a thin, sparsely-hairy stem. It is
similar to "Caladenia carnea" but lacks the red and white bars on the labellum of
that species. "Caladenia catenata" is a terrestrial, perennial, deciduous, herb
with
Document [3](Title: Caladenia prolata): Caladenia prolata Caladenia prolata,
commonly known as long-leaf fingers or white fingers is a plant in the orchid
family Orchidaceae and is endemic to southern Australia. It is a ground orchid
with a single leaf and one or two dull white flowers which are greenish with red
stripes on the back. "Caladenia prolata" is a terrestrial, perennial, deciduous,
herb with an underground tuber and which usually grows as single plants. It has a
single erect, hairy leaf, long, wide with a reddish or purplish base. One or two
dull white flowers which are hairy and greenish with red stripes on
Answer:
```

Table 13: The question part of prompt.

```
According to the provided documents, the botanical name of "lady finger" is
Caladenia catenata [1][2][3].
```

Table 14: The original model only gives the answer and cites all of the documents without explanation.

```
The botanical name of lady finger is Caladenia catenata. This is supported by
multiple sources, including Document [1], which states that Caladenia catenata
is a plant in the orchid family Orchidaceae and is endemic to New South Wales,
Queensland, and Victoria. Document [2] also confirms this, stating that Caladenia
catenata is a ground orchid with a single hairy leaf and one or two white, sometimes
pink flowers on a thin, sparsely-hairy stem. Document [3] further reinforces this
information, providing additional details about the plant's growth habits and
characteristics. Therefore, based on the available information, the botanical
name of lady finger is Caladenia catenata.
```

Table 15: The trained model gives a more detailed answer. All of the documents are cited discretely.