ALARM: Align Language Models via Hierarchical Rewards Modeling

Anonymous ACL submission

Abstract

We introduce ALARM, a novel framework modeling hierarchical rewards in reinforcement learning from human feedback (RLHF), which is designed to enhance the alignment of large language models (LLMs) with human preferences. The framework addresses the limitations of current alignment approaches, which often struggle with the inconsistency and sparsity of human supervision signals, by integrating holistic rewards with aspect-specific rewards. This integration enables more precise and consistent guidance of language models towards desired outcomes, particularly in complex and open text generation tasks. By employing a methodology that filters and combines multiple rewards based on their consistency, the framework provides a robust mechanism for improving model alignment. We validate our approach through applications in long-form question answering and machine translation tasks, employing `gpt-3.5-turbo-1106` for pair-wise comparisons, and demonstrate significant improvements over existing baselines. Our work underscores the effectiveness of hierarchical rewards modeling in refining LLM training processes for better human preference alignment. We release our codes at https://ALaRM-anonymized.github.io.

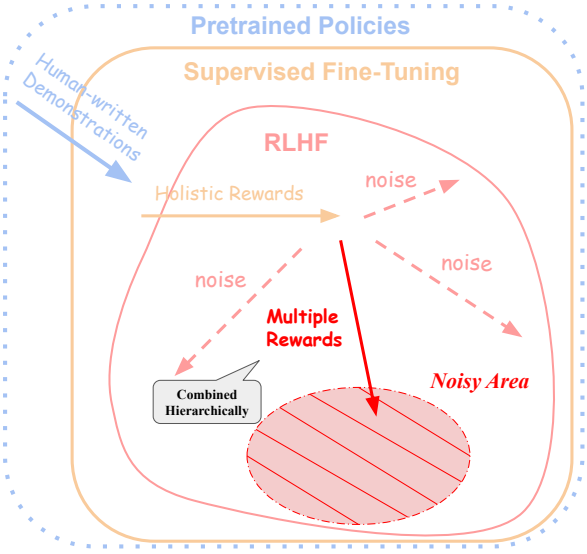

Figure 1: Illustration of our key ideas. The pretrained policies are first supervised fine-tuned on human-written demonstrations and then trained through RLHF given a holistic reward learned from human-ranked comparisons. The shadowed "noisy" area better aligns with human preference, which is hard to reach for solely a noisy holistic reward. We propose to utilize multiple rewards hierarchically for more accurate and consistent supervision signals and thus guide the policies into the noisy area.

1 Introduction

Current LLM-assisted AI systems have shown remarkable performance in a wide range of tasks (Brown et al., 2020; Chen et al., 2021; Touvron et al., 2023; Wang et al., 2023a), and benefit from different forms of human supervision signals (Wei et al., 2022; Stiennon et al., 2020). While supervised learning relies on human-written demonstrations to unlock the emergent abilities gained from pretraining on huge text and code corpora, RLHF utilizes generation comparisons labeled by humans to further fine-tune the LLMs for better alignment with human expectations, which has been demonstrated to be able to reduce undesired model generations like harmful contents (Bai et al., 2022) or hallucinations (Ouyang et al., 2022).

However, collecting human supervision signals can be nontrivial work. As recent LLMs are capable of doing more complex work and even surpass human performance in some areas, it becomes more difficult to write good enough demonstrations even for human experts (Lai et al., 2023). Comparisons between several model-generated texts can be intuitively easier to get from a crowd-sourcing platform though, previous research (Krishna et al., 2023; Wu et al., 2023) has revealed that human annotations can be inconsistent and unreliable in

evaluations between two or more model outputs for complex tasks like long text generations. While stable rewards are the key to successful reinforcement learning, the sparsity nature of current holistic rewards further stresses this challenge (Cao et al., 2024). Moreover, various scenarios hidden behind the tasks can differentiate preference standards (Jang et al., 2023), thus degrading the annotation consistency and value alignment on downstream applications, e.g., concise vs. comprehensive summary in text summarization, and literary vs. technical style in machine translation. Then we ask the question, what if humans are not able to even compare the preferences for two model generations in a reliable way?

To mitigate these challenges, we introduce a new framework ALARM modeling both holistic and aspect-specific rewards in hierarchical reinforcement learning, which is motivated by 1) fine-grained RLHF (Wu et al., 2023) that categorizes different error types for more accurate and cheaper annotation, 2) research in reinforcement learning (Pateria et al., 2021) that designs a hierarchical architecture for better reward shaping. At the core of our framework is to seek stronger supervision signals: As shown in Figure 1, solely using the holistic reward can make it difficult to reach the shadowed "noisy" area which represents better alignment with human preference, and thus we employ multiple rewards combined in a hierarchical way for accurate and consistent guidance into the noisy area. Firstly, we list several aspect-specific rewards corresponding to the task and perform the selection by their inconsistency with the holistic reward in pair-wise comparison. Later in the RLHF training process, the chosen rewards are combined with the holistic reward as a whole once the sampled generation receives a high holistic reward, which is above a certain threshold value. The aspect-specific rewards can either come from reward models trained on comparison datasets annotated along a specific dimension (e.g., honesty) or simply be toolkit-calculated metrics (e.g., token counts), with an arbitrary density at either the token level or the sequence level. In addition, we proactively transform the aspect-specific rewards to guarantee that their cumulative values are positive, thereby motivating the policy to surpass the threshold for higher returns, which enriches the hierarchical structure's effectiveness.

We apply our framework to two text generation tasks: long-form question answering (QA) and machine translation (MT). Long-form QA represents difficult annotations for complex tasks, and MT represents more flexible and various preference standards behind the tasks. For each of them, we employ `gpt-3.5-turbo-1106` as the evaluator for pair-wise comparison. We empirically demonstrate that our framework significantly outperforms the compared methods. The ablation study and corresponding analysis further show that our framework provides stronger supervision signals toward human preference in both scenarios.

Collectively, we highlight our contributions as follows: 1) To our knowledge, we are the first to propose a framework that hierarchically models both holistic and aspect-specific rewards in RLHF, 2) we investigate how to perform reward selection to mitigate rewards conflicting, 3) we prove the effectiveness of our framework as pursuing more accurate and consistent supervision signals on two text generation tasks, with corresponding ablation study and analysis.

2 Framework

Our framework originates from RL-based text generation (Ryang and Abekawa, 2012; Buck et al., 2018), meanwhile modeling multiple rewards including both holistic and aspect-specific rewards in a hierarchical way. We first introduce the widely used RLHF as background knowledge. Then, we describe why we should do reward selection proactively, and how we do that. After that, we present our method to hierarchically model those rewards in detail.

2.1 Reinforcement Learning from Human Feedback

In the context of RL-based text generation, we define a language model parameterized by θ as a policy π_θ. Token generation is considered decision-making, and the policy completes an episode when the language model generates an EOS token or meets the length limit.

In this way, RLHF aims to optimize the policy π_θ to maximize the reward from a preference model R with Proximal Policy Optimization (PPO) (Schulman et al., 2017), a broadly used reinforcement learning algorithm for human preference alignment. The preference model R is learned on human-annotated comparison datasets to predict a single scalar that correctly ranks two or more model generations in the same comparison batch. The opti-

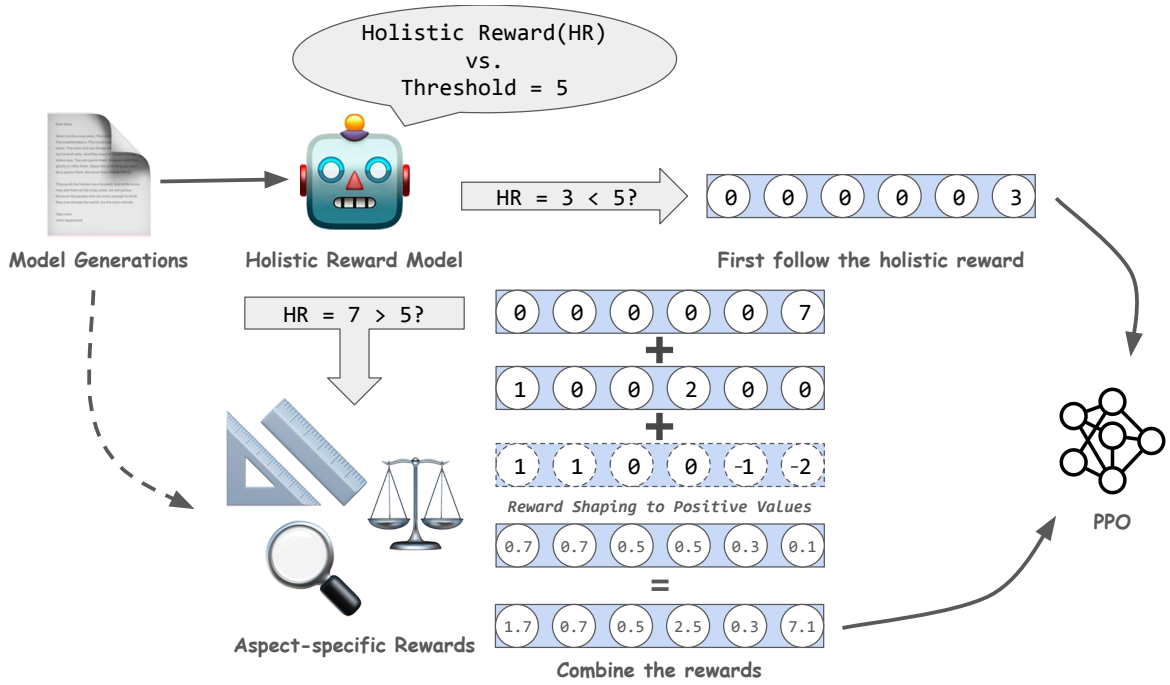

Figure 2: Illustration of hierarchical rewards modeling.

mization objective is also formulated as follows:

$$\arg\max_{\theta} \mathbb{E}_{\tau \sim \pi_\theta}[R(\tau)] \quad (1)$$

where τ denotes a trajectory produced by the policy π_θ, and $R(\tau)$ is the reward associated with the trajectory τ, as evaluated by the preference model R. Here we see how the reward model R deeply affects RLHF and thus it's crucial to ensure the accuracy and consistency of its prediction.

2.2 Reward Selection

Evaluation along a specific dimension of model generations instead of the general quality is demonstrated to be less noisy and more accurate for reward modeling (Wu et al., 2023). Therefore, in order to get more accurate and consistent supervision signals, we first list several aspect-specific rewards corresponding to a certain task intuitively. However, human preferences are intricate. Different decomposed aspects are associated and can even conflict with each other. A common way to balance them is the weighted sum method, which assigns a carefully chosen vector of weights for each aspect-specific reward, according to observations of either the performance during training or accuracy in pair-wise comparisons (Go et al., 2023). Nevertheless, this method still suffers from the over-optimization problem (Moskovitz et al., 2023), where the model loses individual information from every single aspect-specific reward and can't attribute changes in the composed reward to any of them.

Being different from simply combining all those aspect-specific rewards, our key idea is to stabilize the supervision signals. We need a "copilot" for the holistic reward. Thus we aim to resolve this challenge by discarding the conflicting rewards, and we select the rewards that are mostly consistent with the holistic reward. Therefore, we proactively conduct pair-wise comparisons on two sets of model generations from the same supervised fine-tuned model using greedy decoding and pure sampling respectively. Then we calculate the prediction inconsistency of whether the holistic reward and the aspect-specific reward predict differently about which answer is better.

2.3 Hierarchical Rewards Modeling

Hierarchical reinforcement learning has grown proudly in a wide range of decision-making tasks (Yang et al., 2018; Saleh et al., 2020; Wang et al., 2018), decomposing complex and challenging optimization objectives into simpler sub-tasks. Nevertheless, existing RLHF works usually employ a plain rewarding strategy that linearly assigns a single holistic reward (Sun et al., 2023) or a fixed combination of aspect-specific rewards (Go et al., 2023), which also ignores the close relations between the holistic reward and the aspect-specific rewards.

With these motivations, we propose an approach to modeling multiple rewards in hierarchical reinforcement learning, utilizing both holistic and aspect-specific rewards. In this way, we consider the optimization objective that aligns the language models with human preference deep into the noisy area depicted in Figure 1, as a challenging decision-making task. Thus we propose a decomposition of this task into two less complex sub-tasks which ought to be addressed sequentially: 1) Directly follow the holistic reward until the model generation receives a high holistic reward, which indicates the generation is generally good and meets human preference at a relatively high level, 2) optimize the combination of the holistic and aspect-specific rewards, which as a whole provides more accurate and consistent supervision signals towards the noisy area. Differential from the plain weighted-sum method that directly uses the combined rewards for the whole training, we generally follow the supervision of the holistic reward and gently turn the steering wheel only when it's not enough to solely rely on the holistic reward to reach the noisy area.

As illustrated in Figure 2, we first follow the regular RLHF process to sample some generations from the policy. Then we employ a preference model that predicts a single scalar as the holistic reward for each of them. Here we utilize UltraRM-13B (Cui et al., 2023) as a zero-shot reward model to predict holistic rewards for all experiments reported in this paper. For each sampled generation, if it receives a holistic reward lower than a certain threshold value, we directly assign it as the final reward. Otherwise, we calculate the aspect-specific rewards and combine all the rewards together. To design a more effective hierarchical architecture for the above reward modeling, we make sure that the generations receiving a holistic reward above the threshold always get higher cumulative rewards than those receiving a lesser one below this threshold. We do this by reward shaping, transforming the aspect-specific rewards into positive values via the sigmoid function.

3 Long-Form Question Answering

Long-form QA is a complicated task that aims to produce elaborated responses covering background information, explanations, or discussions corresponding to the questions. This process is difficult and even humans are not able to write good demonstrations or do accurate and consistent comparisons in a reliable manner. Following we show how we address this issue using ALARM.

Aspect-specific Rewards	% Inconsistency
Factuality Rate	21.56
Relevance Rate	40.27
Completeness	45.15
Response Length	54.16

Table 1: Results of inconsistency with the holistic reward for each listed aspect-specific reward in long-form QA.

Rewards	% Win Rate
Holistic Reward	73.67
Factuality Rate (↑)	67.07
Relevance Rate (↑)	47.53
Completeness	57.66
Response Length	47.04

Table 2: Win rates of the greedy decoding for all rewards in long-form QA. The higher the factuality rate and the relevance rate, the better.

3.1 Task Settings

We utilize the most settings and release[1] from Wu et al. (2023): QA-Feedback dataset, supervised fine-tuned T5-large as the initial policy, and three fine-grained reward models.

Dataset. QA-Feedback dataset is extracted from ASQA (Stelmakh et al., 2023), and then reformed into a reading comprehensive setting, which gives an ambiguous factoid question and a set of related knowledge sources from the Wikipedia corpus, and requires the language model to generate a long-form response. QA-Feedback has a training set of 3,853 examples, a development set of 500, and a test set of 948 in total. And we keep the division of this dataset in the experiment.

Initial Policy. The policy model is T5-large (Raffel et al., 2020) initialized with supervised fine-tuning on 1K training examples.

Reward Models. We reuse the three fine-grained reward models designed in Wu et al. (2023), which are called the relevance reward model R_{ϕ_1}, the

[1] https://github.com/allenai/FineGrainedRLHF

Methods	Holistic Reward	Factuality Reward	Weighted Sum	ALARM	Avg.
% Win Rates by the Holistic Reward					
Holistic Reward	-	55.86 ± 4.22	50.36 ± 0.95	49.06 ± 0.48	51.76 ± 1.66
Factuality Reward	44.13 ± 4.22	-	45.04 ± 3.73	42.73 ± 3.91	43.97 ± 3.92
Weighted Sum	49.64 ± 0.95	54.95 ± 3.73	-	48.55 ± 1.14	51.05 ± 0.67
ALARM	50.94 ± 0.48	57.27 ± 3.91	51.45 ± 1.14	-	53.22 ± 1.76
% Win Rates by the Factuality Rate (↑)					
Holistic Reward	-	48.31 ± 3.49	50.89 ± 1.50	44.42 ± 2.58	47.87 ± 1.48
Factuality Reward	51.68 ± 3.49	-	52.76 ± 4.01	48.70 ± 3.17	51.05 ± 3.49
Weighted Sum	49.11 ± 1.50	47.24 ± 4.01	-	44.65 ± 4.40	47.00 ± 3.00
ALARM	55.58 ± 2.58	51.30 ± 3.17	55.35 ± 4.40	-	54.08 ± 2.15
% Win Rates Evaluated by `gpt-3.5-turbo-1106`					
Holistic Reward	-	44.11 ± 3.12	44.16 ± 6.28	40.94 ± 1.87	43.07 ± 1.81
Factuality Reward	55.88 ± 3.12	-	53.55 ± 6.09	50.77 ± 4.01	53.40 ± 4.21
Weighted Sum	55.84 ± 6.28	46.45 ± 6.09	-	46.03 ± 5.94	49.44 ± 5.93
ALARM	59.06 ± 1.87	49.23 ± 4.01	53.96 ± 5.94	-	54.08 ± 2.18

Table 3: Win rates on the test set calculated by the holistic reward, the factuality rate, and the evaluator `gpt-3.5-turbo-1106` respectively in long-form QA.

Methods	HR(↑)	FR(↑)	Length
Holistic Reward	0.608	0.736	87.8
Factuality Reward	0.538	0.752	116.8
Weighted Sum	0.598	0.738	98.5
ALARM	0.617	0.752	97.6

Table 4: Evaluation results on the test set for each training in long-form QA. Here HR represents the mean of the holistic reward. FR represents the mean of the factuality rate. Length represents the mean token counts of each generation.

factuality reward model R_{ϕ_2}, the completeness reward model R_{ϕ_3}, representing three categories of different error types. They all use Longformer-base (Beltagy et al., 2020) as the backbone model and do predictions at different levels of density. R_{ϕ_1} is trained to predict whether the generation contains irrelevance, repetition, or incoherence errors at the sub-sentence level, with a binary classification accuracy of 69.6 and an F1-score of 68.5 on the development set. R_{ϕ_2} learns to detect incorrect or unverified facts for each sentence according to given knowledge sources, having an accuracy of 77.8 and an F1-score of 67.5. R_{ϕ_3} is developed to measure the holistic information completeness for the full sequence, with an accuracy of 70.9 in pairwise comparison.

3.2 Reward Selection

Listing Corresponding Rewards. Following the task settings, we have three aspect-specific reward models. We also consider the response token counts as one corresponding reward in this task.

Calculating Inconsistency To proactively filter out appropriate rewards that mostly aid the holistic reward, together as more accurate and consistent training signals, we conduct pair-wise comparisons to check the inconsistency with the holistic reward of those four candidate rewards.

We first employ the initial policy to get two sets of model generations on the training set, using greedy decoding and pure sampling respectively, and two generations from different sets to the same question form a comparison pair. For each aspect-specific reward, we calculate the inconsistency as the percentage of the pairs that the holistic reward prefers one, while the aspect-specific reward prefers another. We disregard all the ties that have the same holistic reward. For the relevance reward and the factuality reward, they can't make an apple-to-apple comparison due to their sentence

or sub-sentence level reward density. Thus we compute the overall correct rate in all the slices of each generation for direct comparison, named as factuality rate and relevance rate.

As shown in Table 1, the factuality rate shows significantly lower inconsistency than the other three rewards, indicating it's more suitable to be the "copilot" of the holistic reward. Also, we compared the win rates of the greedy decoding generations to the pure sampling ones. The win rate is calculated as described in subsection 3.3. As shown in Table 2, the factuality rate has more similar win rates to the holistic reward, which further proves its consistency. Therefore, we select the factuality reward for hierarchical rewards modeling.

3.3 Experimental Setup

Rewards Modeling. We z-normalize the holistic reward in a set of generations D_p which is produced through pure sampling on the training set. The factuality reward is shaped to positive values to ensure the hierarchical structure by the sigmoid function. The threshold value is set to 0.6, which is around the top 30% in D_p. We simply add the rewards up to combine them and only adjust the weight of the holistic reward.

Reinforcement Learning Training. We use the pure sampling strategy in the reinforcement learning process and use greedy decoding for the development set and the test set evaluation. Beginning with the supervised fine-tuned initial policy, we train our model for 2 epochs on the training set. And as we set the exploration frequency to 4, the training runs for about 30K episodes. We utilize LoRA (Hu et al., 2022) for the training.

Evaluation. Besides the mean value of the holistic reward and the factuality rate, we calculate the win rates between different models in pair-wise comparison. Following Jang et al. (2023), the win rate is formulated as:

$$\textit{Win Rate} = \frac{\textit{Win}}{\textit{Win} + \textit{Lose}} \tag{2}$$

where all the ties are disregarded. We also calculate the win rate in general quality by employing `gpt-3.5-turbo-1106` as the evaluator for pair-wise comparison. To mitigate the positional bias (Wang et al., 2023b) in LLM-as-a-judge, we do prompting twice with the two generations swapped. We only consider the comparison valid when both promptings prefer the same generation, indicating the evaluator is faithful enough to overcome the positional bias. We use the prompt from AlpacaEval (Dubois et al., 2024).

Compared Methods. We compare our framework to three methods. **ALARM** represents our proposed hierarchical rewards modeling approach. **Holistic Reward** represents the baseline using the holistic reward as the only supervision signal. **Factuality Reward** represents solely using the factuality reward without reward shaping to train the policy. **Weighted Sum** is the plain weighted sum method that directly adds the holistic reward and the factuality reward together for training.

3.4 Main Results

We run all experiments in this paper across three different seeds and all results are averaged over the three independent runs. Table 4 shows the evaluation results on the test set of the mean values of each reward and the token counts. We can see that ALARM leads to significantly higher holistic reward than other methods, meanwhile reaching the highest factuality rate. As expected, except for ALARM, Holistic Reward gets the highest holistic reward value and Factuality Reward gets the highest factuality rate. Weighted Sum balances these two rewards instead. Table 3 represents the win rates between the four methods. And we can see ALARM holds the best under all three different metrics, which further proves that ALARM provides a stronger supervision signal than other methods.

4 Machine Translation

Machine translation can be considered a text generation task that involves converting a piece of text from the source to the target language while preserving the original meaning, context, and cultural nuances. This task requires not only a deep understanding of the grammatical structure and vocabulary of both the source and target languages but also an appreciation of their idiomatic expressions and cultural references. However, this feature can differentiate people's preferences and form noise (Marchisio et al., 2019; Savoldi et al., 2021), due to the wide range of application scenarios and user groups, thus posing barriers to accurate and consistent alignment annotation. Following we show how we use ALARM on this task.

4.1 Task Settings

Dataset. We utilize Europarl (Tiedemann, 2012), a Spanish-English dataset contains transcripts of European Parliamentary proceedings. We select 100K samples from it and arrange them into a training set of 63K, a development set of 2K, and a test set of 30K in total.

Initial Policy. We initialize the policy model with mT5-base (Xue et al., 2021), which is then supervised fine-tuned on the training set for 5 epochs. The initial policy has a BLEU score of 31.57 on the test set.

4.2 Reward Selection

Aspect-specific Rewards	% Inconsistency
Grammar Error Rate	25.33
Language Confidence	31.54
Readability	42.96
Response Length	43.97

Table 5: Results of inconsistency with the holistic reward for each listed aspect-specific reward in MT.

Rewards	% Win Rate
Holistic Reward	88.10
Grammar Error Rate (↓)	41.30
Language Confidence (↑)	47.06
Readability	49.30
Response Length	48.96

Table 6: Win rates of the greedy decoding for all rewards in MT. The lower the grammar error rate and the lower the language confidence, the better.

Listing Corresponding Rewards. We first intuitively list three corresponding rewards, named grammar reward, language confidence, and readability. All of them are calculated through python-wrapped toolkits. The grammar reward is based on LanguageTool[2], which can detect grammar errors at the word level. We define the grammar reward as assigning negative values to the grammar-error-related tokens. The language confidence is based on Lingua[3], which computes the language likelihood in a set of languages by n-gram models as a single scalar. The readability is to measure the general difficulty of reading the text, which can be calculated by Textstat[4] at the full sequence level.

Calculating Consistency. The same as in subsection 3.2, we conduct pair-wise comparisons on two sets of generations produced by greedy decoding and pure sampling respectively. We define the grammar error rate of a generation as the token counts divided by the number of grammar errors.

As shown in Table 5 and Table 6, the grammar reward stands out with lower inconsistency and better win rates than the other three rewards. Thus we choose the grammar reward for the following experiments.

4.3 Experimental Setup

We follow most setups in subsection 3.3. Evaluation and compared methods are the same.

Rewards Modeling. We do z-normalization to the holistic reward and set the threshold value to 0.5, around the top %30. Due to the token level density of the grammar reward, we do reward shaping for every token including the 0-rewarded tokens by the sigmoid function to ensure the hierarchical structure.

Reinforcement Learning Training. With the pure sampling strategy in training, we train our models for around 13K episodes.

4.4 Main Results

Table 8 represents the mean values of separate rewards on the test set, where ALARM has the highest holistic reward and a comparable grammar error rate to the other methods. As shown in Table 6, ALARM continues to outperform the others in the win rates, evaluated by the holistic reward, the grammar error rate, and `gpt-3.5-turbo-1106` respectively. These results together strongly prove the effectiveness of our framework on this task.

5 Ablation Study

Without Combination. To examine whether ALARM helps with more accurate and consistent supervision signals by utilizing both holistic and aspect-specific rewards, we compare the methods using separate rewards individually in the experiments. As shown in Table 3, 4, 7, 8, ALARM leads to better results along both dimensions.

[2] https://github.com/languagetool-org/languagetool

[3] https://github.com/pemistahl/lingua-py

[4] https://github.com/textstat/textstat

Methods	Holistic Reward	Grammar Reward	Weighted Sum	ALARM	Avg.
% Win Rates by the Holistic Reward					
Holistic Reward	-	51.52 ± 0.77	50.18 ± 0.89	48.30 ± 0.66	50.00 ± 0.76
Grammar Reward	48.47 ± 0.77	-	48.69 ± 0.15	47.07 ± 0.16	48.08 ± 0.32
Weighted Sum	49.82 ± 0.89	51.31 ± 0.15	-	48.26 ± 0.15	49.80 ± 0.25
ALARM	51.70 ± 0.66	52.93 ± 0.16	51.74 ± 0.15	-	52.12 ± 0.26
% Win Rates by the Grammar Error Rate (↓)					
Holistic Reward	-	48.10 ± 2.83	46.53 ± 3.58	57.80 ± 4.29	50.81 ± 0.40
Grammar Reward	51.90 ± 2.83	-	49.37 ± 5.11	57.53 ± 5.29	52.93 ± 3.82
Weighted Sum	53.46 ± 3.58	50.63 ± 5.11	-	59.37 ± 4.58	54.49 ± 3.88
ALARM	42.20 ± 4.29	42.47 ± 5.29	40.63 ± 4.58	-	41.76 ± 4.46
% Win Rates Evaluated by* `gpt-3.5-turbo-1106`					
Holistic Reward	-	51.03 ± 0.42	51.27 ± 1.79	48.97 ± 1.70	50.43 ± 1.00
Grammar Reward	48.97 ± 0.42	-	48.98 ± 2.77	47.85 ± 2.09	48.60 ± 0.90
Weighted Sum	48.73 ± 1.79	51.02 ± 2.77	-	47.98 ± 3.05	49.24 ± 2.30
ALARM	51.03 ± 1.70	52.15 ± 2.09	52.02 ± 3.05	-	51.73 ± 2.11

Table 7: Win rates on the test set calculated by the holistic reward, the grammar error rate, and the evaluator `gpt-3.5-turbo-1106` respectively in MT. *: We choose a smaller set of 3K examples randomly selected from the test set to reduce the annotation cost.

Methods	HR(↑)	% GER(↓)	Length
Holistic Reward	0.919	1.203	36.7
Grammar Reward	0.908	1.190	36.5
Weighted Sum	0.918	1.197	36.5
ALARM	0.928	1.203	37.0

Table 8: Evaluation results on the test set for each training in MT. Here HR represents the mean of the holistic reward. GER represents the mean of the grammar error rate. Length represents the mean token counts of each generation.

Without Hierarchical Structure. We compare our framework with the plain weighted sum method to prove the importance of the hierarchical structure. Most results by Weighted Sum can be considered a balance of the holistic and aspect-specific rewards which is hard to win both, indicating ALARM takes advantage of the hierarchical rewards modeling for stronger supervision signals.

6 Related Work

Hierarchical Reinforcement Learning. Designing a hierarchical rewarding structure that decomposes a long-horizon reinforcement learning task into simpler sub-tasks has shown promising performance in traditional reinforcement learning problems (Florensa et al., 2017; Levy et al., 2019; Kulkarni et al., 2016; Gupta et al., 2020). Motivated by this, ALARM utilizes hierarchical reinforcement learning to align language models.

Human Preference Alignment. AI alignment with human preference has been one of the key research topics in the NLP community as LLMs show notable performance yet are prone to generating unexpected content (Song et al., 2023; Wang et al., 2023c; Zhou et al., 2023; Yuan et al., 2023). RLHF is a popular algorithm for alignment and many related methods are proposed (Dai et al., 2023; Yu et al., 2023; Zhang et al., 2024).

7 Conclusion

We propose ALARM, the first framework hierarchically modeling both holistic and aspect-specific rewards in RLHF. We explore how to proactively do reward selection for better compatibility with the holistic reward. We prove the effectiveness of ALARM with comprehensive experiments, corresponding ablation study, and analysis on two text generation tasks.

Limitations

Our framework needs specifically designed rewards for every task, posing challenges to scale up the application scenarios. Better automatic reward selection is needed. Our evaluation employs OpenAI's API, which has an extra cost and may encounter rate limitations and unstable response times for normal users.

Ethics Statement

Our study complies with the ACL ethics policy, involving no direct human or animal subjects and presenting no discernible ethics concerns. The datasets QA-Feedback, Europarl, and the toolkits textstat, Lingua, and LanguageTool are publicly available and we ensure transparency and reproducibility. We confirm that the research and methodologies do not involve harmful practices or potential misuse. We are committed to upholding the highest integrity and ethical responsibility standards in our work.

References

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Christian Buck, Jannis Bulian, Massimiliano Ciaramita, Wojciech Gajewski, Andrea Gesmundo, Neil Houlsby, and Wei Wang. 2018. Ask the right questions: Active question reformulation with reinforcement learning. In *International Conference on Learning Representations*.

Meng Cao, Lei Shu, Lei Yu, Yun Zhu, Nevan Wichers, Yinxiao Liu, and Lei Meng. 2024. Drlc: Reinforcement learning with dense rewards from llm critic.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. 2021. Evaluating large language models trained on code.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Wei Zhu, Yuan Ni, Guotong Xie, Zhiyuan Liu, and Maosong Sun. 2023. Ultrafeedback: Boosting language models with high-quality feedback.

Josef Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong Yang. 2023. Safe rlhf: Safe reinforcement learning from human feedback.

Yann Dubois, Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2024. Alpacafarm: A simulation framework for methods that learn from human feedback.

Carlos Florensa, Yan Duan, and Pieter Abbeel. 2017. Stochastic neural networks for hierarchical reinforcement learning. In *International Conference on Learning Representations*.

Dongyoung Go, Tomasz Korbak, Germán Kruszewski, Jos Rozen, and Marc Dymetman. 2023. Compositional preference models for aligning lms.

Abhishek Gupta, Vikash Kumar, Corey Lynch, Sergey Levine, and Karol Hausman. 2020. Relay policy learning: Solving long-horizon tasks via imitation and reinforcement learning. In *Proceedings of the Conference on Robot Learning*, volume 100 of *Proceedings of Machine Learning Research*, pages 1025–1037. PMLR.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large

language models. In *International Conference on Learning Representations*.

Joel Jang, Seungone Kim, Bill Yuchen Lin, Yizhong Wang, Jack Hessel, Luke Zettlemoyer, Hannaneh Hajishirzi, Yejin Choi, and Prithviraj Ammanabrolu. 2023. Personalized soups: Personalized large language model alignment via post-hoc parameter merging.

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, Dubrovnik, Croatia. Association for Computational Linguistics.

Tejas D Kulkarni, Karthik Narasimhan, Ardavan Saeedi, and Josh Tenenbaum. 2016. Hierarchical deep reinforcement learning: Integrating temporal abstraction and intrinsic motivation. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.

Yuhang Lai, Chengxi Li, Yiming Wang, Tianyi Zhang, Ruiqi Zhong, Luke Zettlemoyer, Wen-Tau Yih, Daniel Fried, Sida Wang, and Tao Yu. 2023. DS-1000: A natural and reliable benchmark for data science code generation. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 18319–18345. PMLR.

Andrew Levy, Robert Platt, and Kate Saenko. 2019. Hierarchical reinforcement learning with hindsight. In *International Conference on Learning Representations*.

Kelly Marchisio, Jialiang Guo, Cheng-I Lai, and Philipp Koehn. 2019. Controlling the reading level of machine translation output. In *Proceedings of Machine Translation Summit XVII: Research Track*, pages 193–203, Dublin, Ireland. European Association for Machine Translation.

Ted Moskovitz, Aaditya K. Singh, DJ Strouse, Tuomas Sandholm, Ruslan Salakhutdinov, Anca D. Dragan, and Stephen McAleer. 2023. Confronting reward model overoptimization with constrained rlhf.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.

Shubham Pateria, Budhitama Subagdja, Ah-hwee Tan, and Chai Quek. 2021. Hierarchical reinforcement learning: A comprehensive survey. *ACM Comput. Surv.*, 54(5).

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1).

Seonggi Ryang and Takeshi Abekawa. 2012. Framework of automatic text summarization using reinforcement learning. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 256–265, Jeju Island, Korea. Association for Computational Linguistics.

Abdelrhman Saleh, Natasha Jaques, Asma Ghandeharioun, Judy Shen, and Rosalind Picard. 2020. Hierarchical reinforcement learning for open-domain dialog. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8741–8748.

Beatrice Savoldi, Marco Gaido, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2021. Gender bias in machine translation. *Transactions of the Association for Computational Linguistics*, 9:845–874.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Feifan Song, Bowen Yu, Minghao Li, Haiyang Yu, Fei Huang, Yongbin Li, and Houfeng Wang. 2023. Preference ranking optimization for human alignment.

Ivan Stelmakh, Yi Luan, Bhuwan Dhingra, and Ming-Wei Chang. 2023. Asqa: Factoid questions meet long-form answers.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. In *Advances in Neural Information Processing Systems*, volume 33, pages 3008–3021. Curran Associates, Inc.

Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liang-Yan Gui, Yu-Xiong Wang, Yiming Yang, Kurt Keutzer, and Trevor Darrell. 2023. Aligning large multimodal models with factually augmented rlhf.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association (ELRA).

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller,

Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, Wayne Xin Zhao, Zhewei Wei, and Ji-Rong Wen. 2023a. A survey on large language model based autonomous agents.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Qi Liu, Tianyu Liu, and Zhifang Sui. 2023b. Large language models are not fair evaluators.

Xin Wang, Wenhu Chen, Jiawei Wu, Yuan-Fang Wang, and William Yang Wang. 2018. Video captioning via hierarchical reinforcement learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yufei Wang, Wanjun Zhong, Liangyou Li, Fei Mi, Xingshan Zeng, Wenyong Huang, Lifeng Shang, Xin Jiang, and Qun Liu. 2023c. Aligning large language models with human: A survey.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. 2022. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Zeqiu Wu, Yushi Hu, Weijia Shi, Nouha Dziri, Alane Suhr, Prithviraj Ammanabrolu, Noah A Smith, Mari Ostendorf, and Hannaneh Hajishirzi. 2023. Fine-grained human feedback gives better rewards for language model training. *arXiv preprint arXiv:2306.01693*.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Zhaoyang Yang, Kathryn Merrick, Lianwen Jin, and Hussein A. Abbass. 2018. Hierarchical deep reinforcement learning for continuous action control. *IEEE Transactions on Neural Networks and Learning Systems*, 29(11):5174–5184.

Tianyu Yu, Yuan Yao, Haoye Zhang, Taiwen He, Yifeng Han, Ganqu Cui, Jinyi Hu, Zhiyuan Liu, Hai-Tao Zheng, Maosong Sun, and Tat-Seng Chua. 2023. Rlhf-v: Towards trustworthy mllms via behavior alignment from fine-grained correctional human feedback.

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.

Shun Zhang, Zhenfang Chen, Sunli Chen, Yikang Shen, Zhiqing Sun, and Chuang Gan. 2024. Improving reinforcement learning from human feedback with efficient reward model ensemble.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. Lima: Less is more for alignment.

A Experiments

Training Details. We use the TRL[5] implementation for the PPO algorithm. We perform all training using LoRA and BF16. We use 8×48G NVIDIA A6000 GPU for the training, which takes around 2 hours for the long-form QA task and around 1 hour for the MT task.

Evaluation. Figure 3 shows our prompt to call `gpt-3.5-turbo-1106` in pair-wise comparisons for both tasks.

[5] https://github.com/huggingface/trl

```
<|im_start|>system
You are a helpful instruction-following assistant that prints the best model by selecting the best outputs for a given instruction.
<|im_end|>
<|im_start|>user
Select the output (a) or (b) that best matches the given instruction. Choose your preferred output, which can be subjective. Your answer should ONLY contain: Output (a) or Output (b). Here's an example:

# Example:
## Instruction:
Give a description of the following job: "ophthalmologist"

## Output (a):
An ophthalmologist is a medical doctor who pokes and prods at your eyes while asking you to read letters from a chart.

## Output (b):
An ophthalmologist is a medical doctor who specializes in the diagnosis and treatment of eye diseases and conditions.

## Which is best, Output (a) or Output (b)?
Output (b)

Here the answer is Output (b) because it provides a comprehensive and accurate description of the job of an ophthalmologist. In contrast, output (a) is more of a joke.

# Task:
Now is the real task, do not explain your answer, just say Output (a) or Output (b).

## Instruction:
{instruction}

## Output (a):
{output_1}

## Output (b):
{output_2}

## Which is best, Output (a) or Output (b)?
<|im_end|>
```

Figure 3: Evaluation prompt for `gpt-3.5-turbo-1106`.