RETHINKING GROUP-RELATIVE ADVANTAGE ESTIMATION IN POLICY OPTIMIZATION FOR TEXT-TO-TEXT REGRESSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Fine-grained scalar rewards provide rich supervision for reinforcement learning with large language models and naturally support text-to-text regression, where numerical prediction error can directly define the reward. However, as model predictions improve, rewards within rollout groups become increasingly concentrated, a phenomenon we refer to as *reward collapse*. Existing group-relative advantage estimators either allow the learning signal to shrink with these reward differences or normalize away the extent of reward concentration. We introduce Min-Max GRPO (**MM-GRPO**), which normalizes centered rewards by their within-group range, maintaining a well-scaled learning signal while allowing relative reward diversity to determine each group's contribution to optimization. We additionally introduce Rank-based GRPO (**R-GRPO**), which discards reward magnitudes and constructs advantages from within-group ranks to verify the importance of maintaining a well-spread learning signal throughout training. Across two real-world text-to-text regression benchmarks and two model sizes, MM-GRPO achieves the strongest overall regression performance, while R-GRPO remains competitive despite using only reward ordering. Additional analyses show that MM-GRPO and R-GRPO's improvements extend to long-tail target values and multilingual out-of-distribution inputs.

1 INTRODUCTION

Large language models (LLMs) have emerged as effective text-to-text regressors, generating numerical predictions from natural-language inputs while reasoning over heterogeneous structured and unstructured information (Vacareanu et al., 2024; Jacobs et al., 2024; Tang et al., 2024; Zhang et al., 2026). Reinforcement learning (RL) is particularly well suited to this setting because prediction error directly provides a scalar reward reflecting the distance between a prediction and its target. Recent work has demonstrated the effectiveness of RL for text-to-text regression, both by directly optimizing regression-based rewards and by developing regression-aware RL objectives that explicitly leverage numerical structure (Chen et al., 2025; Zhang et al., 2026).

By utilizing prediction errors directly, continuous rewards provide fine-grained feedback that distinguishes differences among sampled predictions. However, the informativeness of these continuous rewards changes over the course of training. As the policy improves, sampled predictions tend to move closer to their targets, causing rewards within a rollout group to become increasingly concentrated, a phenomenon we refer to as *reward collapse* (Figure 1 (a) and (b)). This is a natural consequence of learning, but it changes how reward differences are translated into advantages. With mean-centered advantages, as in Dr.GRPO (Liu et al., 2025), the learning signal shrinks as rewards collapse. Standard GRPO (Shao et al., 2024), in contrast, normalizes by the within-group standard deviation, maintaining the scale of the advantages but assigning equal advantage standard deviations to all rollout groups. Thus, as training progresses, existing estimators either allow the learning signal to diminish with the collapsing reward scale or normalize away the extent of reward concentration.

Prior work in binary-reward settings shows that within-group reward variation can provide a useful learning signal. DAPO (Yu et al., 2026) uses dynamic sampling to discard groups with uniform rewards and retain groups containing both successful and unsuccessful responses, while related work

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



Figure 1: **Left panels: Within-group rewards concentrate as training progresses in text-to-text regression.** Across rollout groups, panels (a) and (b) show the distribution of within-group reward averages and standard deviations at the beginning of training and later in training. **Right panel: MM-GRPO adapts advantage magnitude according to reward dispersion relative to range.** The histograms show two illustrative groups of $G = 64$ rewards with different ranges and standard deviations, where $c_g = \sigma_g/(r_{\max,g} - r_{\min,g})$. Standard GRPO assigns the two groups equal advantage magnitude (Equation 8), whereas MM-GRPO assigns a larger advantage scale to the group with greater relative reward dispersion.

uses within-group reward variance to prioritize training samples (Jiang et al., 2025). These methods establish that a group's reward distribution can help determine its value for learning. With binary rewards, however, this distribution is largely characterized by the proportion of successful responses. Continuous regression rewards provide richer structure: groups can differ in how broadly their rewards are distributed across the observed range. This motivates using relative reward dispersion to continuously determine each group's advantage scale.

Motivated by these observations, we introduce Min-Max GRPO (MM-GRPO), an advantage estimator designed to mitigate the negative effect of reward collapse while preserving differences in reward diversity across groups. MM-GRPO normalizes centered rewards by their within-group range. Because the reward range contracts together with the raw reward differences as training progresses, this normalization prevents the advantage scale from collapsing with the rewards. At the same time, the resulting advantage scale depends on how broadly rewards are distributed within their observed range, causing groups with greater relative reward diversity to contribute more strongly to the policy update. MM-GRPO therefore provides a continuous, scale-invariant mechanism for prioritizing more diverse rollout groups without discarding groups or relying on the absolute reward scale.

Additionally, we examine the importance of maintaining a well-spread learning signal under reward collapse by considering a setting in which reward magnitudes are discarded entirely. We introduce Rank-based GRPO (R-GRPO), which constructs advantages from within-group reward ranks, preserving reward ordering while discarding the numerical differences between rewards. By producing advantages whose spread is independent of the absolute reward scale, R-GRPO serves as a diagnostic for the role of maintaining learning-signal diversity as rewards collapse. We illustrate the resulting MM-GRPO and R-GRPO training pipelines in Figure 2, highlighting how each method transforms within-group rewards into advantages for policy optimization.

We evaluate MM-GRPO and R-GRPO on two real-world text-to-text regression tasks spanning two model scales. MM-GRPO achieves the strongest overall performance, improving both average

regression accuracy and long-tail performance over existing group-relative advantage estimators. R-GRPO also performs competitively despite discarding reward magnitudes entirely, and in several settings outperforms estimators that preserve them. Both methods further show robust out-of-distribution performance across multilingual inputs. Overall, these results highlight the importance of maintaining an informative, well-spread learning signal as rewards become increasingly concentrated during training.

Our contributions are threefold. First, we identify and characterize reward collapse in text-to-text regression, where within-group rewards become increasingly concentrated as the policy improves, and examine how existing group-relative advantage estimators behave under this phenomenon. Second, we introduce MM-GRPO, a range-normalized advantage estimator that mitigates the effect of reward collapse while allowing groups with greater relative reward dispersion to contribute more strongly to learning. Third, we introduce R-GRPO, a rank-based estimator that maintains a well-spread learning signal independent of the absolute reward scale, providing a diagnostic for studying the importance of preserving learning-signal diversity as rewards collapse. Across two real-world regression tasks, our methods improve regression performance, long-tail behavior, and multilingual out-of-distribution robustness.

2 Related Works

Text-to-Text Regression with LLMs Recent work has shown that LLMs can serve as flexible regressors by generating numerical predictions directly as text, often outperforming traditional regression heads while handling heterogeneous and unstructured inputs without explicit feature engineering Song & Bahri; Akhauri et al. (2025b); Song et al.; Akhauri et al. (2025a); Tang et al.; Vedula et al. (2025). However, most existing methods rely on supervised finetuning (SFT) with cross-entropy (CE) loss, which optimizes token matching rather than numerical distance and is therefore insensitive to the magnitude of prediction errors. Regression-aware approaches address this limitation by incorporating numerical distance into the learning signal Zausinger et al. (2025); Lukasik et al. (2024; 2025); Chiang et al. (2025), however they remain within supervised or inference-time frameworks and do not study how RL objectives should be designed for LLM regression.

RL for LLM Regression RL provides a natural alternative to SFT for regression because it directly optimizes sequence-level rewards from prediction errors, allowing models to optimize continuous regression metrics. GenRe^2 shows that combining existing RL algorithms with regression error as the reward can outperform supervised objectives Chen et al. (2025). REAL proposes a regression-aware RL framework that accounts for the ordinal structure of numerical targets Zhang et al. (2026); however, it is designed for LLM-as-a-judge tasks with bounded, single-digit rating scales and is not directly applicable to continuous floating-point targets spanning wider, task-dependent numerical ranges. Recent work has also explored rank-based policy optimization for non-verifiable RLHF settings (Choi et al., 2026) and rank-structured recommendation generation (Zhu et al., 2026). In contrast, this work focuses on regression RL with verifiable continuous rewards.

Dynamic Sampling in Group-Based RL Dynamic-sampling methods for verifiable reasoning filter prompts that produce homogeneous reward groups. DAPO discards groups with uniform binary rewards and samples additional prompts to maintain nonzero-advantage updates (Yu et al., 2026), while GRESO attempts to predict and skip such prompts before rollout generation (Zheng et al., 2026). These methods make a discrete decision to retain or discard each prompt. AVSPO similarly intervenes when groups become uniform in binary reward settings (He et al., 2026). Our work instead considers continuous regression rewards that progressively shrink without necessarily becoming identical and studies how the reward distribution within each sampled group can continuously determine its advantage scale, without filtering prompts according to a selection threshold.

3 RL Finetuning for Text-to-Text Regression

We finetune language models for text-to-text regression using group-based reinforcement learning. Each example consists of a text input x and a ground-truth numerical value y. Given x, the behavior policy $\pi_{\theta_{\text{old}}}$ samples a group of G textual responses $\{o_i\}_{i=1}^{G}$, where each response $o_i = (o_{i,1}, \ldots, o_{i,|o_i|})$ is a sequence of tokens.

Following Dr.GRPO (Liu et al., 2025), we optimize the policy using the clipped objective

$$J(\theta) = \mathbb{E}_{x,\{o_i\}_{i=1}^{G}} \left[\frac{1}{G} \sum_{i=1}^{G} \sum_{t=1}^{|o_i|} \min \left(\rho_{i,t}(\theta) A_i, \text{clip} \left(\rho_{i,t}(\theta), 1 - \epsilon_{\text{clip}}, 1 + \epsilon_{\text{clip}} \right) A_i \right) \right], \quad (1)$$

where

$$\rho_{i,t}(\theta) = \frac{\pi_\theta(o_{i,t} \mid x, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t} \mid x, o_{i,<t})}. \quad (2)$$

is the policy ratio, ϵ_{clip} is the policy-ratio clipping threshold, and A_i is the advantage for response o_i, shared across all of its tokens.

The advantage A_i is computed from the rewards assigned to the sampled responses. For text-to-text regression, each response o_i contains a numerical prediction $\hat{y}_i$, which we extract from the generated text and compare with the ground-truth value y. In this work, we define the reward as the negative absolute error:

$$r_i = -|\hat{y}_i - y|. \quad (3)$$

The methods proposed in this work share this optimization objective and differ only in how the response-level advantages A_i are constructed. In the following subsection, we review existing group-relative advantage estimators. We then introduce our proposed estimators in Section 4.

3.1 Existing Advantage Estimators

Existing group-based methods center rewards within each rollout group but differ in how they determine advantage scale. Dr.GRPO uses the centered rewards directly, without scale normalization:

$$A_i^{\text{Dr.GRPO}} = r_i - \mu. \quad (4)$$

Standard GRPO (Shao et al., 2024) additionally divides by the within-group reward standard deviation:

$$A_i^{\text{GRPO}} = \frac{r_i - \mu}{\sigma + \epsilon}. \quad (5)$$

Batch GRPO (Liu et al., 2026) instead uses a standard deviation $\sigma_{\mathcal{B}}$ computed across all rewards in the optimization batch $\mathcal{B}$:

$$A_i^{\text{Batch}} = \frac{r_i - \mu}{\sigma_{\mathcal{B}} + \epsilon}. \quad (6)$$

Consequently, Dr.GRPO leaves advantage scale proportional to the raw within-group reward dispersion, GRPO makes it approximately constant for every group with non-identical rewards, and Batch GRPO scales all groups using a statistic shared across the batch. These formulations highlight a central design choice in group-based policy optimization: how a group's observed rewards determine the advantages used to update the policy.

4 Methodology

For text-to-text regression, we seek an advantage formulation that remains informative as within-group rewards concentrate while still making use of the fine-grained structure of continuous regression rewards. We introduce two alternative advantage estimators. MM-GRPO normalizes centered rewards by their within-group range, preserving relative reward differences while allowing the reward distribution of each group to determine its advantage scale. R-GRPO instead constructs advantages from within-group reward ranks, discarding reward magnitudes entirely and allowing us to test whether reward ordering alone provides a sufficient learning signal. Figure 2 illustrates how both methods transform the rewards within each sampled group into response-level advantages.

4.1 MM-GRPO: Adaptive Group Weighting

We first introduce Min-Max GRPO (MM-GRPO), an advantage estimator designed to maintain a stable within-group advantage scale while retaining information about the distribution of rewards within each rollout group. MM-GRPO constructs advantages by centering the rewards within a group and scaling them by the group's observed reward range:

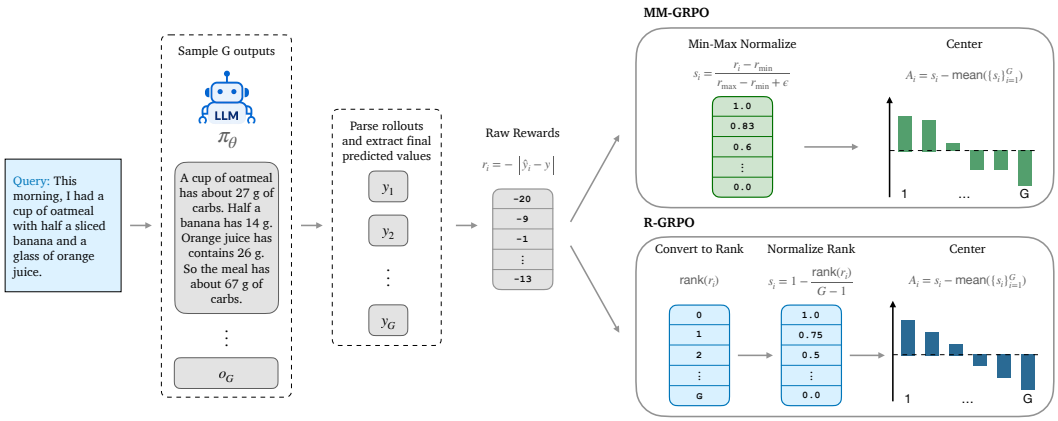

Figure 2: **Overview of MM-GRPO and R-GRPO.** For each input query, the policy samples a group of rollouts and computes regression-error-based rewards from the generated predictions. MM-GRPO applies within-group min-max normalization before mean-centering, while R-GRPO transforms rewards into within-group ranks before mean-centering to construct advantages. Both methods preserve a well-spread advantage signal even when raw regression rewards become concentrated during training.

$$A_i^{\mathrm{MM}} = \alpha \frac{r_i - \mu}{r_{\max} - r_{\min} + \epsilon}, \tag{7}$$

where $r_{\max}$ and $r_{\min}$ are the maximum and minimum reward values in the group, $\alpha > 0$ is a constant scaling factor, and ϵ is a numerical stabilizer. Min-max normalization maps the minimum and maximum rewards to approximately 0 and 1, while mean-centering shifts all values without changing their range. Consequently, every group has an advantage range of approximately α. Groups can nevertheless differ in how their advantages are distributed within this shared range. MM-GRPO therefore maintains a stable advantage range as rewards concentrate, rather than allowing advantage scale to shrink as in Dr.GRPO. At the same time, it allows advantage variance to reflect the distribution within each group's range instead of fixing it at $\sigma = 1$ as in GRPO. Consequently, the resulting advantage scale explicitly depends on how broadly the rewards are distributed within their range.

Adaptive group scaling. To characterize how each group's reward distribution determines the scale of its advantages, we define the advantage magnitude for a group g with G rollouts using the root mean square value of its advantages:

$$M_g = \sqrt{\frac{1}{G}\sum_{i=1}^{G} A_{g,i}^2} \tag{8}$$

From Equation 7, it follows that M_g under MM-GRPO is

$$M_g^{\mathrm{MM}} = \alpha\sqrt{\frac{G-1}{G}}\frac{\sigma_g}{r_{\max,g} - r_{\min,g} + \epsilon} \approx \alpha\sqrt{\frac{G-1}{G}}c_g, \tag{9}$$

where $c_g = \sigma_g/(r_{max,g} - r_{min,g})$ and the factor $\sqrt{(G-1)/G}$ arises because σ_g is computed using the sample standard deviation. Because G is fixed across rollout groups, the factor $\sqrt{(G-1)/G}$ is constant, and variation in advantage magnitude across groups is determined entirely by the ratio c_g. Panel c in Figure 1 illustrates this scaling effect using two examples of rollout groups with 64 rollouts each. In this sense, MM-GRPO separates two roles of normalization: the reward range controls the overall within-group advantage range, while the ratio of reward standard deviation to range determines the group's effective advantage scale.

Relation to GRPO. The adaptive scaling induced by MM-GRPO can be understood directly in relation to the standard GRPO advantage estimator. Recall that GRPO normalizes centered rewards by the within-group standard deviation (Equation 5). Then, from Equations 7 and 5 (when ϵ is small), it follows that

$$A_{g,i}^{\mathrm{MM}} \approx \alpha \frac{\sigma_g}{r_{\max,g} - r_{\min,g}} A_{g,i}^{\mathrm{GRPO}} = \alpha c_g A_{g,i}^{\mathrm{GRPO}} \tag{10}$$

This relationship shows that MM-GRPO preserves the relative ordering and within-group structure of the GRPO advantages while allowing their overall scale to vary across groups. Standard GRPO approximately standardizes every group to the same advantage variance, whereas MM-GRPO rescales each group's GRPO advantages by the distribution-dependent factor αc_g. Groups with larger relative reward dispersion therefore receive a larger multiplicative scaling relative to GRPO, while groups with smaller relative dispersion receive a smaller scaling. Figure 1 visualizes this effect for groups with low and high relative reward dispersion.

4.2 R-GRPO: From Reward Magnitudes to Reward Ranks

The reward collapse illustrated in Figure 1 (a) and (b) raises the question of whether the relative ordering between sampled responses can still provide an informative learning signal to guide optimization when reward magnitudes become increasingly concentrated. Motivated by this idea, we propose R-GRPO, a ranking-based advantage estimator. As illustrated in Figure 2, R-GRPO uses within-group ranks to convert continuous rewards into scale-invariant relative comparisons, mitigating the effect of reward collapse.

Specifically, we first compute the negative absolute error reward r_i for each response and then rank the responses within a group to obtain $\mathrm{rank}(r_i) \in \{0, \ldots, G-1\}$. We then define the normalized rank score s_i^{rank} and compute the corresponding advantages using the ranked scores:

$$s_i^{\mathrm{rank}} = 1 - \frac{\mathrm{rank}(r_i)}{G-1}; \qquad A_i^{\mathrm{R-GRPO}} = \alpha \left(s_i^{\mathrm{rank}} - \frac{1}{G} \sum_{j=1}^{G} s_j^{\mathrm{rank}} \right), \tag{11}$$

where $\alpha > 0$ is a scaling factor.

Ranking produces a uniformly distributed reward signal that is independent of reward scale. The goal is to provide a reliable reward signal that remains consistent, even when absolute reward differences are small. Additionally, by removing magnitude information, R-GRPO serves as a probe to test whether training relies on fine-grained reward magnitudes for text-to-text regression or if it can sufficiently utilize relative comparisons between rollouts when a well-spread reward signal is maintained.

5 Experimental Setup

5.1 Tasks & Datasets

We evaluate our method across two real-world text-to-text regression tasks.

Nutrition Estimation Accurate nutrition monitoring is an important component of healthcare, particularly for conditions such as diabetes that require careful management of carbohydrate intake (Buck et al., 2022). We study carbohydrate estimation from natural-language meal descriptions. For training, we use FoodData Central (FDC) (USDA, 2019) to construct individual foods and meal combinations, and use GPT-4o-mini to generate corresponding natural-language descriptions (see Appendix C.1 for details). The resulting dataset contains 50,446 training examples and 1,000 validation examples. We test on NutriBench (Hua et al., 2024), a real-world benchmark comprising 4,826 meal descriptions.

Product Price Prediction We consider the *Appliances* category of the Amazon Reviews dataset for the price prediction task (Hou et al., 2024). Each product is represented as free-form text ob-

tained by combining its metadata fields (e.g., name, category, average user rating, dimensions). Detailed preprocessing steps and input formatting are provided in Appendix C.2. The resulting dataset consists of 32,163 training examples, 946 validation examples, and 3,784 test examples.

We provide representative examples for both datasets in Appendix C.5.

5.2 Baselines

Advantage Estimation We compare our methods against common group-relative advantage estimation methods described in section 3. We evaluate Dr.GRPO (Liu et al., 2025), which only applies a mean-centering transformation to the rewards. Batch-GRPO (Liu et al., 2026) was motivated by the observation that batch-level standard deviation could provide a more stable estimate and reduce sensitivity to low within-group variance. Lastly, we also evaluate the standard GRPO advantage formulation which applies group z-score normalization (Shao et al., 2024)).

Dynamic Sampling. We evaluate the dynamic sampling strategy introduced by DAPO (Yu et al., 2026), which removes rollout groups with zero reward variance and samples additional prompts to maintain a fixed number of groups that contribute nonzero advantages. This provides a discrete prompt-selection baseline for comparison with MM-GRPO's continuous adaptive group weighting.

Supervised Finetuning. We additionally compare against supervised finetuning with distilled reasoning. For each training example, GPT-4o-mini is provided with the input and ground-truth numerical target and prompted to generate an example reasoning trace ending in the target value. The Qwen models are then finetuned on these teacher-generated responses using CE loss.

5.3 Training Details

We finetune Qwen3-4B and Qwen3-1.7B Yang et al. (2025) with a learning rate of 3×10^{-6}. For the 4B models, we generate 8 rollouts per prompt and use a training batch size of 2,048. For the 1.7B models, we generate 64 rollouts per prompt and use a training batch size of 512. Following Dr.GRPO, no KL penalty is applied for MM-GRPO and R-GRPO. All baseline training details remain faithful to the original recipes; a KL penalty is used for Standard GRPO and Batch GRPO. For DAPO, we follow the original recipe and apply dynamic sampling that discards prompt groups with zero reward variance and continues sampling until the training batch is filled (with a generation batch $1.5\times$ the training batch size). For Qwen3-1.7B experiments, R-GRPO and MM-GRPO advantages are scaled with a constant factor determined using a separate held-out validation set (details can be found in Appendix B). All experiments are run on a single node with 4 GPUs. Training is performed using VERL Sheng et al. (2024) with the vLLM rollout backend and tensor parallelism size 1. We use a PPO mini-batch size of 256 and a micro-batch size of 16 and 8 per GPU for the nutrition estimation and product price prediction tasks, respectively.

6 Results

6.1 MM-GRPO Leads to Better Reward Utilization

Table 1 reports the main results on nutrition estimation and product price prediction. MM-GRPO achieves the overall best performance. In general, all RL-based methods substantially improve performance relative to the base model, confirming that RL is effective for finetuning LLMs for regression tasks. On nutrition estimation, Dr.GRPO improves the Qwen3-4B model from an R^2 score of -0.264 to 0.694, while similar gains are observed on product price prediction. However, existing normalization strategies provide limited additional improvements. DAPO, GRPO, and Batch GRPO achieve performance comparable to Dr.GRPO, but neither method consistently improves final regression performance across tasks. In contrast, MM-GRPO achieves the strongest overall performance, obtaining the best MAE, R^2, and RMSE across both tasks and model sizes. The improvements are particularly pronounced for Qwen3-1.7B. On nutrition estimation, MM-GRPO reduces MAE from 15.875 for the strongest baseline to 12.690, while increasing R^2 from 0.449 to 0.608. On price prediction, it increases R^2 by more than 17% from 0.504 to 0.591.

Table 1: Held-out performance on nutrition estimation and price prediction for Qwen-3 models. Avg. R^2 is the unweighted mean of the two task-specific R^2 values and is not reported when either value is missing. Best and second-best values within each model size are shown in bold and underlined, respectively.

	Nutrition Estimation			Price Prediction			
Method	$\mathbf{R^2}\uparrow$	$\mathbf{MAE}\downarrow$	$\mathbf{RMSE}\downarrow$	$\mathbf{R^2}\uparrow$	$\mathbf{MAE}\downarrow$	$\mathbf{RMSE}\downarrow$	$\mathbf{Avg.\ R^2}\uparrow$
Qwen3-4B							
Base Model	-0.264	18.779	39.017	-0.089	21.459	29.744	-0.177
SFT + Distillation	0.675	10.285	19.772	0.190	15.205	25.660	0.433
Dr.GRPO	0.694	9.951	19.185	0.586	11.342	18.348	0.640
GRPO	0.690	9.858	19.313	0.304	<u>11.020</u>	23.779	0.497
Batch GRPO	0.693	11.176	19.244	0.607	11.214	17.865	0.650
DAPO	0.628	9.817	21.157	0.131	11.893	26.574	0.380
R-GRPO	<u>0.732</u>	<u>9.750</u>	<u>17.953</u>	<u>0.610</u>	11.212	<u>17.810</u>	<u>0.671</u>
MM-GRPO	**0.756**	**8.978**	**17.140**	**0.621**	**10.902**	**17.542**	**0.689**
Qwen3-1.7B							
Base Model	-64.630	46.919	280.970	-0.261	20.626	32.026	-32.446
SFT + Distillation	0.467	**12.173**	25.340	-1.758	20.078	47.342	-0.646
Dr.GRPO	0.375	17.634	27.434	0.498	12.928	20.207	0.437
GRPO	0.443	15.910	25.902	0.504	12.641	20.082	0.474
Batch GRPO	0.449	15.875	25.748	0.352	14.786	22.952	0.401
DAPO	0.449	16.087	25.764	<u>0.556</u>	<u>12.461</u>	<u>19.004</u>	0.503
R-GRPO	<u>0.490</u>	14.367	<u>24.794</u>	0.536	12.515	19.411	<u>0.513</u>
MM-GRPO	**0.608**	<u>12.690</u>	**21.719**	**0.591**	**11.716**	**18.227**	**0.600**

Table 2: Performance beyond average in-distribution performance for Qwen3-4B. Long-tail performance is evaluated on the top 10% highest-value examples in each dataset, while multilingual performance is evaluated on nutrition estimation across nine unseen languages. Best and second-best values in each column are shown in bold and underlined, respectively.

	Long-Tail Performance		Multilingual Performance
Method	**Nutrition** $\mathbf{MAE}\downarrow$	**Price** $\mathbf{MAE}\downarrow$	**Nutrition** $\mathbf{MAE}\downarrow$
Base Model	53.049	54.045	66.784
SFT + Distillation	29.578	46.783	35.953
Dr.GRPO	30.295	34.518	31.452
GRPO	29.562	35.261	32.444
Batch GRPO	36.702	35.561	34.940
DAPO	31.616	33.820	36.193
R-GRPO	<u>28.252</u>	<u>32.997</u>	<u>31.201</u>
MM-GRPO	**24.613**	**32.974**	**30.080**

6.2 Reward Ordering Alone Provides a Strong Learning Signal

As shown in Table 1, across both tasks, rank-based advantage estimation consistently matches or outperforms standard magnitude-based GRPO variants, despite discarding fine-grained reward magnitudes entirely. This suggests that maintaining a stable and well-spread advantage signal is more important for LLM regression RL optimization than preserving precise scalar reward differences. Specifically, R-GRPO achieves consistently stronger performance across both benchmarks and model sizes. On nutrition estimation, R-GRPO improves the Qwen3-4B model from an R^2 of 0.694 with Dr.GRPO to 0.732, while simultaneously reducing MAE and RMSE. Similar improvements are observed on price prediction, where R-GRPO achieves better R^2 and regression error compared to existing RL baselines. Notably, these gains are achieved using only within-group reward ordering, without access to exact reward magnitudes. This indicates that relative comparisons between sampled rollouts provide a sufficiently informative learning signal for regression RL.

6.3 Performance Beyond Average In-Distribution Accuracy

While average in-distribution metrics provide an overall measure of regression performance, they can obscure behavior in challenging regions of the target space and do not capture generalization under distribution shift. We therefore conduct two additional analyses using Qwen3-4B: performance on long-tail target values and out-of-distribution generalization to multilingual inputs.

6.3.1 Long-Tail Target Values

Table 2 (left) evaluates MAE on the top 10% highest-value examples in each dataset. MM-GRPO achieves the strongest long-tail performance on both tasks, reducing MAE to 24.613 on nutrition estimation and 32.974 on price prediction. R-GRPO also performs competitively, achieving the second-best performance on both tasks. The calibration plots in Appendix Figure 3 further show that Dr.GRPO increasingly under-predicts as target values rise, whereas R-GRPO and MM-GRPO remain closer to the identity line and achieve lower calibration error across both tasks and model sizes. Together, these results show that the proposed methods reduce systematic underprediction in sparsely represented regions of the target space, where accurate prediction is particularly challenging.

6.3.2 Multilingual Generalization

In real-world settings, users may describe meals in different languages and use language- or culture-specific food terms (Winata et al., 2025). We therefore evaluate the proposed and baseline methods in an out-of-distribution setting under linguistic shift using a multilingual nutrition estimation benchmark spanning nine unseen languages (Italian, Spanish, French, Amharic, Tagalog, Chinese, Lao, Portuguese, and Romanian), resulting in 897 test samples. We provide dataset construction details in Appendix C.3 and examples in different languages in Appendix C.5. Results are shown in Table 2 (right); MM-GRPO achieves the strongest multilingual performance with Qwen3-4B, obtaining an MAE of 30.080. R-GRPO achieves the second-best performance with an MAE of 31.201, while both proposed methods outperform the existing GRPO variants. These results show that the improvements from MM-GRPO extend beyond the English evaluation setting to previously unseen languages.

7 Conclusion

In this work, we study group-relative advantage estimation for text-to-text regression through the lens of reward collapse, the increasing concentration of within-group rewards as the policy improves. We examine how existing advantage estimators interact with this phenomenon: unnormalized advantages shrink with the reward scale, while standard GRPO normalizes by the within-group standard deviation, removing the effect of reward concentration on the overall advantage scale.

We introduce MM-GRPO, which normalizes centered rewards by their within-group range. This formulation mitigates the effect of reward collapse on the learning signal while allowing differences in relative reward diversity to determine how strongly each group contributes to optimization. We additionally introduce R-GRPO as a diagnostic for the importance of maintaining a well-spread learning signal, using within-group ranks to produce advantages whose spread is independent of the absolute reward scale. Across two real-world regression tasks and two model scales, MM-GRPO achieves the strongest overall performance, while R-GRPO remains competitive despite discarding reward magnitudes entirely. Both approaches also exhibit robust behavior on long-tail targets and multilingual out-of-distribution inputs.

Overall, our results suggest that how reward differences are transformed into learning signals becomes increasingly important as rewards concentrate during training. More broadly, they motivate advantage estimators that account not only for the absolute scale of rewards, but also for the structure of reward variation within each rollout group.

AI USE STATEMENT

In this work, we used generative AI tools to generate natural-language meal descriptions and synthetic reasoning traces, as described in Appendix C, and to aid and polish the writing. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

MM-GRPO and R-GRPO are drop-in advantage reformulations requiring no learned reward model or model-architecture changes, making them easily portable to group-relative RL fine-tuning pipelines. Implementation and experimental details are provided in Sections 4–5 and Appendices B–C.

REFERENCES

Yash Akhauri, Bryan Lewandowski, Cheng-Hsi Lin, Adrian N Reyes, Grant C Forbes, Arissa Wongpanich, Bangding Yang, Mohamed S Abdelfattah, Sagi Perel, and Xingyou Song. Performance prediction for large systems via text-to-text regression. *arXiv preprint arXiv:2506.21718*, 2025a.

Yash Akhauri, Xingyou Song, Arissa Wongpanich, Bryan Lewandowski, and Mohamed S Abdelfattah. Regression language models for code. *arXiv preprint arXiv:2509.26476*, 2025b.

Sina Buck, Collin Krauss, Delia Waldenmaier, Christina Liebing, Nina Jendrike, Josef Högel, Boris M Pfeiffer, Cornelia Haug, and Guido Freckmann. Evaluation of meal carbohydrate counting errors in patients with type 1 diabetes. *Experimental and Clinical Endocrinology & Diabetes*, 130(07):475–483, 2022.

Ming Chen, Sheng Tang, Rong-Xi Tan, Ziniu Li, Jiacheng Chen, Ke Xue, and Chao Qian. Beyond token-level supervision: Unlocking the potential of decoding-based regression via reinforcement learning. *arXiv preprint arXiv:2512.06533*, 2025.

Cheng-Han Chiang, Hung-yi Lee, and Michal Lukasik. Tract: Regression-aware fine-tuning meets chain-of-thought reasoning for llm-as-a-judge. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2934–2952, 2025.

Kyuseong Choi, Dwaipayan Saha, Woojeong Kim, Anish Agarwal, and Raaz Dwivedi. Gopo: Policy optimization using ranked rewards. *arXiv preprint arXiv:2602.03876*, 2026.

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

Xixiang He, Qiyao Sun, Ao Cheng, Xingming Li, Xuanyu Ji, Hailun Lu, Runke Huang, and Qingyong Hu. Advantage collapse in group relative policy optimization: Diagnosis and mitigation. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=MKNimf9bIx.

Namgyu Ho, Laura Schmid, and Se-Young Yun. Large language models are reasoning teachers. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: long papers)*, pp. 14852–14882, 2023.

Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. Bridging language and items for retrieval and recommendation. *arXiv preprint arXiv:2403.03952*, 2024.

Andong Hua, Mehak Preet Dhaliwal, Ryan Burke, Laya Pullela, and Yao Qin. Nutribench: A dataset for evaluating large language models on nutrition estimation from meal descriptions. *arXiv preprint arXiv:2407.12843*, 2024.

Ryan Jacobs, Maciej P Polak, Lane E Schultz, Hamed Mahdavi, Vasant Honavar, and Dane Morgan. Regression with large language models for materials and molecular property prediction. *arXiv preprint arXiv:2409.06080*, 2024.

Guochao Jiang, Wenfeng Feng, Guofeng Quan, Chuzhan Hao, Yuewei Zhang, Guohua Liu, and Hao Wang. Vcrl: Variance-based curriculum reinforcement learning for large language models. *arXiv preprint arXiv:2509.19803*, 2025.

C Leclercq, P Allemand, A Balcerzak, F Branca, RF Sousa, A Lartey, M Lipp, VP Quadros, and P Verger. Fao/who gift (global individual food consumption data tool): A global repository for harmonised individual quantitative food consumption studies. *Proceedings of the Nutrition Society*, 78(4):484–495, 2019.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=5PAF7PAY2Y.

Zihe Liu, Jiashun Liu, Yancheng He, Weixun Wang, Jiaheng Liu, Ling Pan, Xinyu Hu, Shaopan Xiong, Ju Huang, Jian Hu, Shengyi Huang, Siran Yang, Jiamang Wang, Wenbo Su, and Bo Zheng. Tricks or traps? a deep dive into RL for LLM reasoning. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=R0JM3BWP7W.

Michal Lukasik, Harikrishna Narasimhan, Aditya Krishna Menon, Felix Yu, and Sanjiv Kumar. Regression aware inference with llms. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 13667–13678, 2024.

Michal Lukasik, Zhao Meng, Harikrishna Narasimhan, Yin-Wen Chang, Aditya Krishna Menon, Felix Yu, and Sanjiv Kumar. Better autoregressive regression with llms via regression-aware fine-tuning. In *The Thirteenth International Conference on Learning Representations*, 2025.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*, 2024.

Xingyou Song and Dara Bahri. Decoding-based regression. *Transactions on Machine Learning Research*.

Xingyou Song, Oscar Li, Chansoo Lee, Bangding Yang, Daiyi Peng, Sagi Perel, and Yutian Chen. Omnipred: Language models as universal regressors. *Transactions on Machine Learning Research*.

Eric Tang, Bangding Yang, and Xingyou Song. Understanding llm embeddings for regression. *Transactions on Machine Learning Research*.

Eric Tang, Bangding Yang, and Xingyou Song. Understanding llm embeddings for regression. *arXiv preprint arXiv:2411.14708*, 2024.

USDA. Fooddata central. https://fdc.nal.usda.gov/, 2019. Accessed: 2024-09-30.

Robert Vacareanu, Vlad-Andrei Negru, Vasile Suciu, and Mihai Surdeanu. From words to numbers: Your large language model is secretly a capable regressor when given in-context examples. *arXiv preprint arXiv:2404.07544*, 2024.

Nikhita Vedula, Dushyanta Dhyani, Laleh Jalali, Boris N Oreshkin, Mohsen Bayati, and Shervin Malmasi. Quantile regression with large language models for price prediction. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 12396–12415, 2025.

Genta Indra Winata, Frederikus Hudi, Patrick Amadeus Irawan, David Anugraha, Rifki Afina Putri, Wang Yutong, Adam Nohejl, Ubaidillah Ariq Prathama, Nedjma Ousidhoum, Afifa Amriani, et al. Worldcuisines: A massive-scale benchmark for multilingual and multicultural visual question answering on global cuisines. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 3242–3264, 2025.

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

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *Advances in Neural Information Processing Systems*, 38:113222–113244, 2026.

Jonas Zausinger, Lars Pennig, Anamarija Kozina, Sean Sdahl, Julian Sikora, Adrian Dendorfer, Timofey Kuznetsov, Mohamad Hagog, Nina Wiedemann, Kacper Chlodny, et al. Regress, don't guess: A regression-like loss on number tokens for language models. In *International Conference on Machine Learning*, pp. 73995–74017. PMLR, 2025.

Yasi Zhang, Tianyu Chen, Mingyuan Zhou, Oscar Leong, Ying Nian Wu, and Michal Lukasik. Real: Regression-aware reinforcement learning for llm-as-a-judge. *arXiv preprint arXiv:2603.17145*, 2026.

Haizhong Zheng, Yang Zhou, Brian Bartoldson, Bhavya Kailkhura, Fan Lai, Jiawei Zhao, and Beidi Chen. Act only when it pays: Efficient reinforcement learning for llm reasoning via selective rollouts. *Advances in Neural Information Processing Systems*, 38:124321–124346, 2026.

Yaochen Zhu, Harald Steck, Dawen Liang, Yinhan He, Vito Claudio Ostuni, Jundong Li, and Nathan Kallus. Rank-GRPO: Training LLM-based conversational recommender systems with reinforcement learning. In *The Fourteenth International Conference on Learning Representations*, 2026.

A Calibration Plots

Figure 3 presents the predicted-versus-ground-truth calibration plots on both nutrition estimation and product price prediction. Consistent with the main results discussed in Section 6, Dr.GRPO exhibits stronger saturation and systematic under-prediction in high-value regions, while both R-GRPO and MM-GRPO more closely follow the identity function. The improvement is particularly pronounced on nutrition estimation, where stable advantage formulations substantially reduce long-tail calibration error relative to Dr.GRPO. To quantify systematic prediction bias across the target range, we compute the integrated absolute error (IAE) between the binned mean-prediction curve and the ideal identity line. We partition the ground-truth values into bins and compute the mean prediction in each nonempty bin as

$$\bar{\hat{y}}_i = \frac{1}{|\mathcal{S}_i|} \sum_{j \in \mathcal{S}_i} \hat{y}_j, \tag{12}$$

where $\mathcal{S}_i$ is the set of samples assigned to bin i and $\hat{y}_j$ is the prediction for sample j. The IAE is then defined and numerically approximated using the composite trapezoidal rule as

$$\mathrm{IAE} = \int_{x_1}^{x_B} \left| \bar{\hat{y}}(x) - x \right| \, dx \approx \sum_{i=1}^{B-1} \frac{\left| \bar{\hat{y}}_i - x_i \right| + \left| \bar{\hat{y}}_{i+1} - x_{i+1} \right|}{2} \left(x_{i+1} - x_i \right), \tag{13}$$

where B is the number of nonempty bins, x_i is the representative ground-truth coordinate of bin i, and $\bar{\hat{y}}_i$ is its mean prediction. Thus, $\left| \bar{\hat{y}}_i - x_i \right|$ measures the vertical deviation of the binned mean-prediction curve from the ideal line $\hat{y} = x$. Lower IAE indicates better agreement between predicted and ground-truth values.

B Advantage Scaling

Policy-gradient updates are directly weighted by the advantage assigned to each sampled response. Therefore, advantage magnitude controls the effective scale of the policy update. R-GRPO and MM-GRPO produce bounded rank- or min-max-based scores. This keeps advantage estimation robust to reward collapse, but it also makes the overall advantage magnitude an explicit optimization choice. In R-GRPO, rank-based advantages lie in the range of $[-0.5, 0.5]$, with adjacent ranks separated by

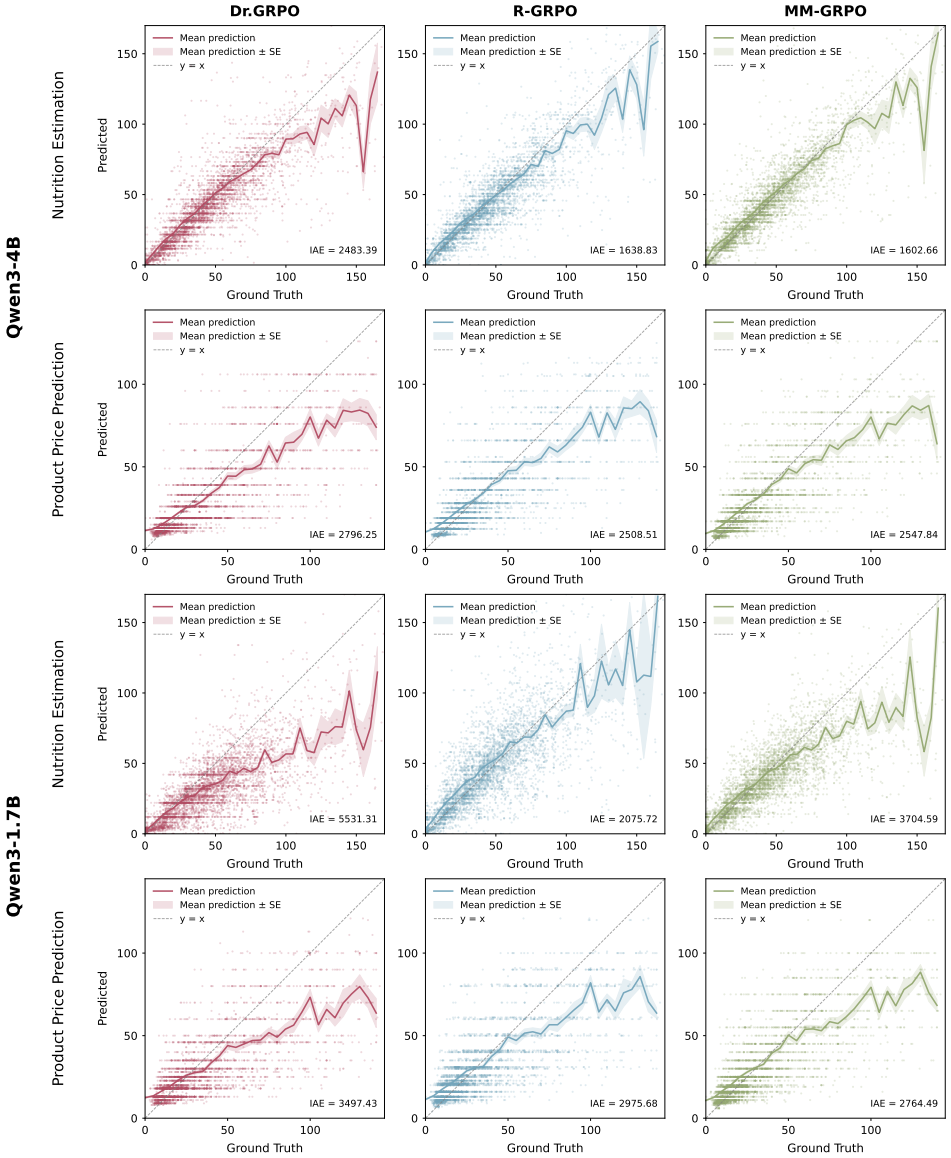

Figure 3: Predicted versus ground-truth values for nutrition estimation and product price prediction under Dr.GRPO, R-GRPO, and MM-GRPO using Qwen3-4B and Qwen3-1.7B as base models. Scatter points denote individual predictions. Solid lines show the mean prediction obtained by binning samples according to ground-truth values using fixed-width bins of size 5, while the shaded regions indicate the standard error of the mean within each bin. Dashed lines correspond to the ideal prediction function y=x. The integral of absolute error (IAE), displayed in the bottom-right of each panel, measures the accumulated absolute deviation between the mean prediction curve and the identity line across the target range.

$1/(G-1)$. For $G = 64$, this produces a rank spacing of approximately 0.016, so neighboring ranks induce only small differences in update weight unless the advantages are rescaled. In MM-GRPO, min-max normalization produces advantages in $[-1, 1]$. Consequently, multiplicative scaling is introduced to control the effective update magnitude after the advantage range has been stabilized. Table 3 evaluates this effect for Qwen3-1.7B on a held-out validation set, and the results are used to select the value of this hyperparameter. Performance varies with the scaling factor for both R-GRPO and MM-GRPO, indicating that advantage magnitude remains an important factor even after advantage spread is fixed. This analysis identifies advantage scaling as a separate calibration problem

from reward collapse: R-GRPO and MM-GRPO stabilize the relative advantage structure within each group, while the scaling factor determines the strength of the resulting policy update.

Table 3: Advantage-scaling validation performance for R-GRPO and MM-GRPO. Best values within each method are shown in bold.

Scale	Nutrition Estimation MAE ↓	Product Price Prediction MAE ↓
R-GRPO		
1	13.783	13.886
8	13.527	**12.819**
16	**12.545**	14.642
32	13.551	14.177
MM-GRPO		
1	13.201	13.174
4	**12.135**	**12.400**
8	14.201	12.544
16	13.956	13.500

C Additional Dataset Construction Details

C.1 Nutrition Estimation Data

We generate natural language meal descriptions for the training and validation sets using GPT-4o-mini, based on entries from FoodData Central (FDC) (USDA, 2019), the food composition database maintained by the U.S. Department of Agriculture (USDA). Specifically, we provide the LLM with the food names and portion sizes from FDC entries, and prompt it to synthesize realistic, free-form meal descriptions in the same style using the prompt shown in Box C.1. Representative examples of the training data are provided in Box C.5

Box C.1: Nutrition Estimation Data Generation Prompts

Single-Item Meal Description Generation Prompt

Role: You are a meal description bot. Your style should be conversational and natural. Your tone should be matter-of-fact. Do not use flowery or overly descriptive words or phrasing.

Objective: Given an input, describe the meal in plain language as a regular person would.

Instructions:

- Provide five different meal descriptions in a dictionary with keys "description1", "description2", etc.
- Extract the **exact food names and serving sizes** from each description and store them in the "items" field.
- Ensure consistency between "description" and "items"—the "items" field must not omit details that appear in the description.
- Return only the meal descriptions and items without additional information.

Context: Use the following examples of user input and model output as a guide:

Input1

```
{
  "parts": [
    "smoothie, made with spinach, banana,
     almond milk, protein powder"
  ],
  "units": [
    "1.0 glass"
```

```
  ]
}
```

Output1

```
{
  "description1": {
    "description": "I am having a glass of smoothie
made with spinach, banana, almond milk,
and protein powder for lunch.",
    "items": "a glass of smoothie made with
spinach, banana, almond milk, and
protein powder"
  },
  etc
}
```

Input2

```
{
  "parts": [
    "Beverages, NESTEA, tea, black,
     ready-to-drink, lemon"
  ],
  "units": [
    "1.0 serving (8 fl oz)"
  ]
}
```

Output2

```
{
  "description1": {
    "description": "8 fl oz of NESTEA lemon black tea",
    "items1": "8 fl oz of NESTEA lemon black tea"
  },
  etc.
}
```

Input3

```
{
  "parts": [
    "pepperoni pizza"
  ],
  "units": [
    "1.0 slice"
  ]
}
```

Output3

```
{
  "description1": {
    "description": "One slice of pepperoni pizza",
    "items1": "one slice of pepperoni pizza"
  },
  etc.
}
```

Input4

```
{
  "parts": [
    "Breakfast tart, lowfat"
  ],
  "units": [
    "1 Pop Tart"
```

```
  ]
}
```

Output4

```
{
  "description1": {
    "description": "I have a low-fat Breakfast Tart.",
    "items1": "a low-fat Breakfast Tart."
  },
  etc.
}
```

Task: For each input, create five different meal descriptions in plain, conversational language. Additionally, output the natural food names including natural serving sizes in the descriptions.

Multi-Item Meal Description Generation Prompt

Task: Combine two meal descriptions into one, creating a natural-sounding sentence as if spoken by a person. If two queries describe different meals, merge them into a single meal description.

Input1

```
[
  "43g garlic bread.",
  "100g grilled chicken with steamed vegetables.",
  "100g tomato soup."
]
```

Output1

```
43g garlic bread, 100g grilled chicken with
steamed vegetables, and 100g tomato soup.
```

Input2

```
[
  "For breakfast, I am having 136g banana.",
  "My lunch consists of 106g refrigerated
whole wheat naan bread."
]
```

Output2

```
For lunch, I am having 136g banana and
106g refrigerated whole wheat naan bread.
```

Instruction: Return only the combined meal descriptions, excluding any additional information.

C.2 Product Price Prediction Data

For our primary experiments, we focus on the Appliances category, which consists of 94.3K items. For each product, we construct a textual input by concatenating relevant product metadata fields (e.g., item name, category, user rating, and descriptive attributes). We filter the dataset to retain only items that contain at least one descriptive feature and have an associated price label, and we remove outliers whose prices fall above the 90th percentile. Representative examples of the training data are provided in Box C.6

C.3 Multilingual Nutrition Estimation

For out-of-distribution evaluation under linguistic shift, we build a multilingual nutrition estimation test set from dietary records in the WHO Global Individual Food Consumption Data Tool (GIFT) (Leclercq et al., 2019). The benchmark includes meals in nine languages (Italian, Spanish, French, Amharic, Tagalog, Chinese, Lao, Portuguese, and Romanian). For each record, we use

GPT-4o-mini to generate a natural-language meal description while preserving the record's original language. Rather than translating English records into the target languages, this construction preserves local terminology and reduces translation-induced artifacts. The generation prompt is provided in Box C.2 and representative examples are provided in Box C.7.

Box C.2: Multilingual Nutrition Estimation Data Generation Prompt

You are generating realistic meal log descriptions for nutrition estimation.
Given food items and weights, write 3 diverse meal descriptions in the target language, like how different real people would log the same meal.
Requirements:

- All 3 descriptions must faithfully preserve the food items and quantities. Do not add or infer any nutritional values (calories, macros, etc.) that are not part of the food item names.
- Vary meaningfully across the following dimensions — each description should feel distinct:
  1. Brevity: one ultra-short (telegraphic), one mid-length, one fuller sentence
  2. Wording: different vocabulary, avoid repeating the same phrasing across descriptions
  3. Sentence structure: mix styles (e.g., bare list, full sentence, diary-style prose)
  4. Quantity explicitness: at least one omits exact grams, at least one includes them
  5. Level of detail: at least one summarizes the meal, at least one names every item
- Do not add opinions, subjective commentary (e.g., "quite filling"), taste/texture descriptions, or cooking inferences unless explicitly stated in the food item name.
- Do not prefix descriptions with numbers, bullets, or labels.
- The output should resemble real-world food diary or meal log entries.
- Some food item names may contain bilingual forms such as "Crisp eggplant / Mak Keua" or "Rice ()". Always use only the name that matches the target language.
- For non-English outputs, vary sentence structure as much as in English — avoid repeating the same "[quantity][food]" pattern across all 3 descriptions.

C.4 Reasoning Traces for Supervised Fine-Tuning

For the supervised finetuning baseline, we construct synthetic reasoning traces using GPT-4o-mini, following prior work on reasoning distillation for language models Guo et al. (2025); Ho et al. (2023). For each training example, the teacher model is provided with the task input and ground-truth numerical target and prompted to generate intermediate natural-language reasoning that supports the target value. The generated response consists of this reasoning followed by the ground-truth numerical prediction in the task-specific output format. Boxes C.3 and C.4 provide the prompts used for generation of the reasoning traces for the nutrition estimation and price prediction tasks, respectively. We provide examples from the nutrition estimation and price prediction datasets, along with their generated synthetic reasoning traces, in the following subsection C.5.

Box C.3: Nutrition Estimation Reasoning Trace Generation Prompt

Task
You are a nutrition assistant that explains how a meal's total carbohydrate count was determined. Given a meal description and its actual total carbohydrate count in grams, write a short step-by-step reasoning that:

- Parses the meal description into its discrete food or beverage items along with their serving sizes.
- Estimates the carbohydrate content (in grams) for each item.
- Explains how the per-item estimates combine into the overall total.

Your reasoning should be consistent with the provided total carbohydrate count. Do not restate the total or output any JSON/structured value in your response — provide only the step-by-step reasoning.

Now provide only the reasoning for the following query:

Box C.4: Product Price Prediction Reasoning Trace Generation Prompt

Task
You are a shopping assistant that explains why a product is priced the way it is.
Given the product information and its actual price below, write a short reasoning paragraph that explains why this product costs that amount. Consider relevant factors such as:

- The type of product and its typical market positioning
- Key features or specifications
- Brand or manufacturer
- Customer ratings or popularity signals (if available)
- Any other information that would reasonably affect price

Your reasoning should be consistent with the provided price. Do not restate the price in your response — provide only the reasoning paragraph.

Now provide only the reasoning for the following query:

C.5 Data Examples

We provide representative training examples together with their generated synthetic reasoning traces for both tasks. Examples from the nutrition estimation dataset are shown in Box C.5, while examples from the product price prediction dataset are shown in Box C.6. Box C.7 provides multilingual meal examples in different languages.

Box C.5: Nutrition Estimation Training Data Examples

Example 1:

Query:
100 grams of gluten-free granola with almond butter.

Ground Truth Carbohydrates:
60g

Synthetic Reasoning Trace:
1. First, we break down the meal description into its components: we have ”100 grams of gluten-free granola” and ”almond butter.”
2. Next, we estimate the carbohydrate content for each item. For gluten-free granola, a typical serving of 100 grams usually contains around 50 grams of carbohydrates. For almond butter, a standard serving size (about 2 tablespoons or 32 grams) typically has around 6-8 grams of carbohydrates. Since we don’t have the exact serving size for the almond butter in this case, we can estimate it contributes approximately 5 grams of carbohydrates.
3. Now, we combine the carbohydrate estimates from both items: 50 grams from the gluten-free granola and 5 grams from the almond butter. Adding these together gives us a total of 55 grams of carbohydrates. However, considering the total carbohydrate count provided is 60 grams, this suggests that the granola may have a slightly higher carbohydrate content than the average estimate, or there may be additional ingredients in the granola contributing to the carbohydrate total. Thus, the overall total aligns closely with the provided count when accounting for these variations.
Output: {“total_carbohydrates”: 60.0}

Example 2:

Query:
For dinner, I am enjoying a cup of chili with beans that was heated in the microwave.

Ground Truth Carbohydrates:
26.55g

Synthetic Reasoning Trace:

1. The meal description consists of one main item: a cup of chili with beans. Since no other food or beverage items are mentioned, we will focus solely on this item.
2. To estimate the carbohydrate content of the chili with beans, we need to consider the typical ingredients found in chili, which usually includes beans, tomatoes, and various seasonings. A standard serving of chili with beans (approximately one cup) generally contains around 20-30 grams of carbohydrates, primarily from the beans and any added vegetables or tomato products.
3. Given that the total carbohydrate count for this meal is 26.55 grams, we can conclude that the estimate for the cup of chili with beans aligns well with the average carbohydrate content for this type of dish. The beans contribute a significant portion of the carbohydrates, while any other ingredients in the chili help round out the total to reach the provided carbohydrate count. Thus, the estimate for the chili with beans effectively combines to match the overall total of 26.55 grams.
Output: {"total_carbohydrates": 26.55}

Example 3:

Query:
100g of chocolate fudge pudding.

Ground Truth Carbohydrates:
22.83g

Synthetic Reasoning Trace:
1. The meal description consists of a single food item: chocolate fudge pudding, with a serving size of 100 grams.
2. To estimate the carbohydrate content, we refer to nutritional databases or food labels for chocolate fudge pudding. Typically, chocolate fudge pudding contains approximately 22.83 grams of carbohydrates per 100 grams.
3. Since there is only one item in this meal, the estimated carbohydrate content of 22.83 grams from the chocolate fudge pudding directly corresponds to the overall total carbohydrate count for the meal. Thus, the calculation confirms that the total carbohydrates in the meal are accurately represented by the single item.
Output: {"total_carbohydrates": 22.83}

Box C.6: Product Price Prediction Training Data Examples

Example 1:

Query:
main_category: Appliances
title: SB08088378 Broan Appliance Damper
average_rating: 4.5
store: Broan
details: {"Brand Name": "Broan", "Item Weight": "0.01 Ounces", "Installation Type": "Freestanding", "Part Number": "SB08088378", "Form Factor": "Compact", "Batteries Required?": "No", "Date First Available": "December 1, 2017", "Brand": "Broan", "Pattern": "Solid"}

Synthetic Reasoning Trace:
The SB08088378 Broan Appliance Damper is priced at a premium due to its high-quality construction and the reputable brand behind it. Broan is well-known in the appliance industry for producing reliable and durable products, which adds to the perceived value. This particular damper features a compact form factor, making it ideal for various installation scenarios, and it is freestanding, offering flexibility for users. The item is lightweight, which contributes to its ease of installation and usability. Additionally, the product has an impressive average rating of 4.5, indicating strong customer satisfaction and reliability, which further justifies its price point. Overall, the combination of brand reputation, quality, and positive customer feedback positions this product as a worthwhile investment for consumers looking for a dependable appliance component.
{"price": 22.5}

Example 2:

Query:
main_category: Amazon Home
title: R.A.M Reliable Humidifier Wick Filter for Kenmore Quiet Comfort 13-4 Pack
average_rating: 4.6
store: Artist Unknown
categories: Appliances; Parts & Accessories; Humidifier Parts & Accessories; Replacement Wicks
details: {"Is Discontinued By Manufacturer": "No", "Batteries required": "No"}

Synthetic Reasoning Trace:

The R.A.M Reliable Humidifier Wick Filter for Kenmore Quiet Comfort is priced at a premium due to its high-quality construction and effectiveness in maintaining optimal humidity levels in your home. With an impressive average rating of 4.6, this product has garnered positive customer feedback, indicating its reliability and performance. The filter is designed specifically for the Kenmore Quiet Comfort model, ensuring a perfect fit and maximum efficiency, which adds to its value. Additionally, as a replacement part, it plays a crucial role in extending the lifespan of the humidifier by maintaining its functionality. The brand's reputation in the market for producing durable and effective replacement parts further justifies the price point, making it a worthwhile investment for consumers looking to enhance their home environment.
{"price": 32.31}

Example 3:

Query:
main_category: Amazon Home
title: Samsung DA63-07180A Refrigerator Door Bin Genuine Original Equipment Manufacturer (OEM) Part
average_rating: 5.0
store: SAMSUNG
categories: Appliances; Parts & Accessories; Refrigerator Parts & Accessories; Bins
details: {"Package Dimensions": "12 x 4 x 4 inches", "Item Weight": "0.01 ounces"}

Synthetic Reasoning Trace:
The Samsung DA63-07180A Refrigerator Door Bin is priced at a premium due to several factors. As a genuine OEM part, it is specifically designed for compatibility with Samsung refrigerators, ensuring a perfect fit and reliable performance that third-party alternatives may not guarantee. The high average rating of 5.0 indicates strong customer satisfaction and reliability, suggesting that buyers value quality and performance over cost. Additionally, Samsung is a well-established brand known for its quality appliances, which often commands higher prices due to its reputation for durability and innovation. The precise dimensions and lightweight nature of the product further suggest careful engineering, which is typical of OEM parts, contributing to its overall value in the market.
{"price": 79.16}

Box C.7: Examples of Multilingual Meal Descriptions

Spanish

Example 1: Un poco de pan dulce de panadería, unos 70 gramos aproximadamente. (39.47g carbohydrates)

Example 2: Para la comida, tomé un jugo de naranja industrializado, pesando 156 gramos, acompañado de 242 gramos de sopa de pasta tipo Maruchan. (39.27g carbohydrates)

Italian

Example 1: Per il bimbo, ho dato 150 grammi di succo di pera Nipiol e 120 grammi di yogurt alla frutta Plasmon con mela e banana. (45.66g carbohydrates)

Example 2: Per merenda ho mangiato una barretta fitness Nestlé al cioccolato di 27 grammi. (21.82g carbohydrates)

Portuguese

Example 1: 50g de café, 36g de cuscuz e 100g de ovos mexidos. (12.02g carbohydrates)

Example 2: Pão integral 50g com margarina 10g e adoçante artificial 2g, acompanhado de duas vitaminas de 300g cada, uma de maçã e outra mista. (119.01g carbohydrates)

Tagalog

Example 1: Isang maliit na ulam na may 3g gabi, 2g boiled galunggong, 60g white rice, plus 10g bawat isa ng nilutong kalabasa at upo. (20.0g carbohydrates)

Example 2: Niluto ko ang 40g na gabi, 40g na kalabasa, at 10g na malunggay, sinamahan ng 225g na puting kanin at 10g na tsabita na pinirito. (81.81g carbohydrates)

C.6 TASK PROMPTS

We provide the prompts used for the nutrition (carbohydrate) estimation and price prediction tasks in Boxes C.8 and C.9, respectively.

Box C.8: Nutrition Estimation System Prompt

For the given query including a meal description, you are asked to respond in the query's language.
In your response, think step by step as follows:

1. Parse the meal description into discrete food or beverage items along with their serving size. If the serving size of any item in the meal is not specified, assume it is a single standard serving based on common nutritional guidelines (e.g., USDA). Ignore additional information that does not relate to the item name and serving size.
2. For each food or beverage item in the meal, calculate the amount of carbohydrates in grams for the specific serving size.
3. Respond with a dictionary object containing the total carbohydrates in grams as follows: {"total_carbohydrates": total grams of carbohydrates for the serving}
   For the total carbohydrates, respond with just the numeric amount of carbohydrates without extra text. If you do not know the answer, set the value of "total_carbohydrates" to −1.

Follow the format of the following examples when answering.

Query:
"This morning, I had a cup of oatmeal with half a sliced banana and a glass of orange juice."

Answer:
The meal consists of 1 cup of oatmeal, 1/2 a banana, and 1 glass of orange juice.
1 cup of oatmeal has 27g carbs.
1 banana has 27g carbs, so half a banana has $(27 \times \frac{1}{2}) = 13.5$g carbs.
1 glass of orange juice has 26g carbs.
So the total grams of carbs in the meal $= (27 + 13.5 + 26) = 66.5$.
Output: {"total_carbohydrates": 66.5}

Query:
"I ate scrambled eggs made with 2 eggs and a toast for breakfast."

Answer:
The meal consists of scrambled eggs made with 2 eggs and 1 toast.
Scrambled eggs made with 2 eggs has 2g carbs.
1 toast has 13g carbs.
So the total grams of carbs in the meal $= (2 + 13) = 15$.
Output: {"total_carbohydrates": 15}

Query:
"Half a peanut butter and jelly sandwich."

Answer:
The meal consists of 1/2 a peanut butter and jelly sandwich.
1 peanut butter and jelly sandwich has 50.6g carbs, so half a peanut butter and jelly sandwich has $(50.6 \times \frac{1}{2}) = 25.3$g carbs.
So the total grams of carbs in the meal $= 25.3$.
Output: {"total_carbohydrates": 25.3}

These examples are in English. Make sure to use the query's language in your answer.

Box C.9: Product Price Prediction System Prompt

Task
You are a shopping assistant that estimates the price of a product based on its description.
Given the product information below, estimate the product's price in USD using general market knowledge and reasonable assumptions.
First, explain your reasoning step by step. In your reasoning, consider relevant factors such as:

- The type of product and its typical market positioning
- Key features or specifications
- Brand or manufacturer
- Customer ratings or popularity signals (if available)
- Any other information that would reasonably affect price

Use only the provided information and general marketplace knowledge.

After completing the reasoning, output the final estimate on the last line only, in the following exact structured format:

{"price": estimated_price_in_usd}

Important:

- The final line must contain only the dictionary, with no additional text.
- The value of "price" must be a single numeric value (e.g., 12.99).
- Do not include currency symbols.
- Do not include any text after the final dictionary.

Example 1:

Query:
main_category: Appliances
title: ERP 482469 Refrigerator Evaporator Motor Kit
average_rating: 3.8
features: Replacement evaporator motor, compatible with many refrigerator models, 1-year warranty
brand: ERP

Answer:
This is a small refrigerator evaporator motor replacement part from an aftermarket appliance parts brand. Such motors are lightweight, mass-produced, and typically priced lower than OEM parts. With average customer ratings and broad compatibility, similar parts usually sell in the $15–$30 range. Based on this, a low-$20s price is reasonable.

{"price": 21.39}

Example 2:

Query:
main_category: Tools & Home Improvement
title: Multi-Functional Movable Adjustable Base Telescopic Furniture Dolly with 8 Locking Rubber Caster Wheels
average_rating: 2.7
features: Adjustable base, 8 locking caster wheels, 2204 lb capacity, stainless steel tubes
brand: Greenesthome

Answer:
This is a heavy-duty adjustable appliance dolly designed for moving large appliances. The 8-wheel design, high stated weight capacity, and stainless steel construction place it above basic furniture dollies. However, it is from a lesser-known brand and has a low rating, suggesting mid-range quality rather than premium. Similar products typically sell between $50 and $120. A price in the upper-middle of that range is reasonable.

{"price": 88.99}

Example 3

Query:
main_category: Tools & Home Improvement
title: Swift Green Filters SGF-LGFR06 Rx Replacement Refrigerator Water Filter (1 pack)
average_rating: 1.7
features: NSF certified, activated carbon filter, compatible with LG refrigerators
brand: Swift Green Filters

Answer:
This is a third-party replacement refrigerator water filter. Aftermarket filters are usually much cheaper than OEM versions, which often cost $40–$60. The low rating and low popularity suggest budget positioning. Single-pack third-party filters commonly sell in the $15–$30 range. A low-to-mid $20 price point is reasonable.

{"price": 22.71}

Now answer the following query: