CARE: CONSERVATIVE POLICY OPTIMIZATION FOR RLVR IN GENERATIVE CLASSIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

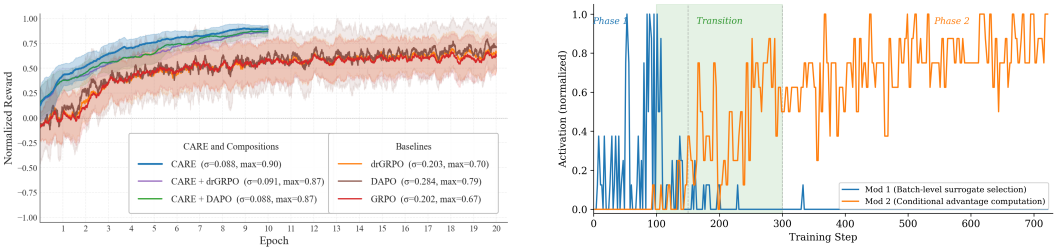
Reinforcement learning with verifiable rewards (RLVR) has become the predominant approach for post-training large language models on tasks where output correctness can be algorithmically verified, and GRPO-family optimizers (GRPO, DAPO, DR.GRPO), all of which inherit PPO's clipped surrogate, dominate this space. Their design assumptions of high-entropy action spaces and well-dispersed rewards are inherited from multi-step mathematical reasoning and open-ended generation. *Generative classification*, where the policy emits a class label directly as a short generated string, satisfies neither: the action distribution is low-entropy, and while every rollout is scored, the reward takes only a handful of distinct values. We identify two temporally distinct breakdowns of this clipped surrogate under these conditions. Early in training, elementwise surrogate minimization selects the *unclipped* branch for negative-advantage high-ratio (NAHR) samples, admitting exactly the aggressive updates that clipping exists to suppress. Later, as accuracy rises, prompt groups become uniform and mean-centered advantages vanish, starving the policy of gradient signal. We propose **C**onservative **A**dvantage **R**atio **E**nhancement (CARE), which addresses both with two modifications and no new hyperparameters. First, the elementwise minimum between clipped and unclipped surrogates is replaced by a single batch-aggregated magnitude comparison, committing the whole update to the less aggressive surrogate and closing the per-sample escape route. Second, advantage normalization is applied conditionally, falling back to the raw reward when the batch carries no reward variance. We further show that the batch-level conservative surrogate selection enables entropy recovery in collapsed-policy regimes without an explicit entropy bonus. Trained on RAGTruth QA for hallucination classification, CARE surpasses GRPO by absolute +11.5 points macro-F1 on in distribution test set over three seeds, and transfers to OOD hallucination classification on RAGTruth (Summarization, Data2Text), HaluEval (QA, Dialogue, Summarization), and Internal QA (Easy, Hard). CARE attains 64.40 macro-F1 on the multi-class ANLI benchmark against 59.20 for the strongest baseline.

1 INTRODUCTION

Reinforcement learning (RL) has become the preferred post-training paradigm for large language models (LLMs), driving recent advances in instruction following, mathematical reasoning, and code synthesis (Schulman et al., 2017; Ouyang et al., 2022; Shao et al., 2024; Agrawal et al., 2025). Within this landscape, *Reinforcement Learning with Verifiable Rewards* (RLVR) (Guo et al., 2025) has emerged as the preferred training regime. By deriving the signal directly from a programmatic correctness check (Cobbe et al., 2021), RLVR sidesteps the learned reward model (Stiennon et al., 2020), removing a confounding source of approximation error. The optimizers in widespread use, namely GRPO (Shao et al., 2024), DAPO (Yu et al., 2025), and DR.GRPO (Liu et al., 2025), share a single template: a clipped importance-weighted surrogate objective (PPO (Schulman et al., 2017)) combined with an unconditional group-normalized advantage.

This shared template was designed for *high-entropy* tasks: multi-step math reasoning, long-horizon code generation, and open-ended chat. Even when the terminal reward is binary, these tasks emit long, diverse output sequences, so any rollout batch contains varied trajectories and meaningful within-batch advantage dispersion. This assumption breaks in *generative classification* regime, where

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.



(a) Training reward convergence

(b) CARE's modification activation

Figure 1: Training dynamics on RAGTruth QA. **(a)** CARE shows narrow reward variance, reaching $\sim$0.9 normalized reward in 10 epochs. GRPO, DR.GRPO, and DAPO oscillate far more widely and stay below CARE even at 2$\times$ the budget (20 epochs), showing the signature of negative-advantage high-ratio (NAHR) instability early and gradient starvation late. **(b)** CARE's two modifications (Section 3) are separated in time: Mod 1 peaks early, where NAHR events are most frequent, then decays as the policy stabilises; Mod 2 takes over as uniform-reward batches dominate.

the policy emits a class label directly as a generated string rather than through a classification head (He et al., 2025). Short responses leave little room for the trajectory diversity that high-entropy tasks supply implicitly. Two properties characterize this regime and jointly govern optimizer behaviour: **(i) low action entropy**: the policy selects among a handful of short label strings rather than composing a free-form sequence, so the entropy of its output distribution is bounded by the size of the label set (two options for binary classification), where open-ended generation draws from a vocabulary of $\sim 10^5$ tokens at every position of a long sequence; and **(ii) low reward cardinality**: every rollout is scored, so the reward signal is *dense* rather than sparse in the sense familiar from RLVR, but it takes only a handful of distinct values ($R \in \{+1, -1, -2\}$; Section 3.1). As accuracy rises, a few-valued reward over a constrained action space leads to an increasing share of rollouts sharing identical rewards, so reward variation collapses (first within groups, then within whole batches) and it is that collapse the shared template is not built to survive.

The generative classification regime spans a broad range of production LLM workloads. Content moderation, intent classification, fact verification, and hallucination detection all require a discrete decision rather than an open-ended generation. We study it primarily through *hallucination detection* (Ji et al., 2023; Maynez et al., 2020): given a context, a question, and a candidate answer, the model must emit HALLUCINATED or NOT_HALLUCINATED. This setting also exposes the deployment constraint most acutely. Regulated industries such as finance, healthcare, and legal cannot route documents through API-served frontier models under latency, cost, and data-sovereignty constraints. That leaves self-hosted 7-13B small language models as the only deployable option for in-line factuality checking (Huang et al., 2025). The state-of-the-art compact detector still requires GPT-4-synthesized training data (Tang et al., 2024), a dependency RLVR removes.

This shared template exhibits two temporally distinct failure modes in this regime. **(i) Early-training instability:** The elementwise surrogate selection permits negative-advantage, high-ratio (NAHR) samples to bypass the clip in the destabilizing direction, injecting excessive gradient magnitude that produces reward spikes and policy oscillation. **(ii) Late-training gradient starvation:** As accuracy rises with training, prompt groups become uniform (same reward), and mean-centered advantage yields zero gradient, starving the policy of any learning signal. In high-entropy tasks, reward diversity keeps uniform groups rare, and NAHR events are correspondingly infrequent (Zhai et al., 2026). In generative classification, the constrained action space removes this implicit regularization, and both surface as structural properties of the template rather than as tuning artifacts, as formalized in Section 3. Both leave a visible signature on the baseline training curves (Figure 1a).

Variants of this template mitigate the symptoms without addressing the root cause: they adjust the clip bounds, the advantage estimator, or which prompt groups enter the batch, but each retains the elementwise surrogate selection and the mean-centered advantage that produce the two failure modes (Section 2). We instead propose CARE (*Conservative Advantage Ratio Enhancement*), which repairs both inside the existing objective. *Batch-level conservative surrogate selection* (Mod 1) replaces the elementwise min with a single batch-level magnitude comparison. *Conditional advantage computation* (Mod 2) bypasses normalization when the batch carries no reward variance. Neither

adds new hyperparameters, and the two fixes naturally align with the temporal structure of the failure modes, so no phase of training is left unprotected (Figure 1b).

Our contributions are:

- **A mechanistic account of two failure modes.** We characterize low-entropy generative classification by low action entropy and low-reward cardinality. We locate early-training instability in the branch that elementwise minimum selects for NAHR samples (Proposition 1), and late-training collapse in the probability that a group is uniform (Proposition 2).
- **CARE: two modifications, no new hyperparameters.** Batch-level conservative surrogate selection (Mod 1) closes the NAHR escape route; conditional advantage computation (Mod 2) preserves gradient flow on zero reward variance batches. Because CARE repairs the shared template rather than replacing it, it *composes* with existing optimizers; we demonstrate this with DAPO and DR.GRPO (Table 1).
- **Consistent gains, and entropy recovery without a bonus.** +11.5 macro-F1 over GRPO on RAGTruth QA (Niu et al., 2024) across three seeds, holding on two backbones and seven out-of-distribution evaluations, and 64.40 on multi-class ANLI (Nie et al., 2020) against 59.20. CARE reaches $\sim$0.9 normalized reward within 10 epochs, a level the primary baselines do not reach even at twice that budget. The same batch-level conservatism recovers entropy without an explicit bonus (Section 5).

2 Related Work

2.1 Policy Optimization for RLVR

The shared template. Widely used optimizers follow a single design: a clipped importance-weighted surrogate paired with an unconditional group-normalized advantage. GRPO (Shao et al., 2024) normalizes the advantage by the mean and standard deviation of rewards over the rollout group sharing a prompt. DR.GRPO (Liu et al., 2025) removes the standard-deviation denominator and the per-token length normalization, retaining mean centering. DAPO (Yu et al., 2025) decouples the clip into separate lower and upper bounds and filters at group granularity, discarding every prompt group whose rollouts share the same accuracy and over-sampling to refill the batch. All three retain PPO's surrogate selection mechanism (Schulman et al., 2017), whose interaction with negative advantages is a known source of early instability (Zheng et al., 2023), and all three centre the advantage on a mean. These are the two mechanisms Section 3 shows to break once the label set is bounded.

Prior repairs to the surrogate. TRULY-PPO (Wang et al., 2020) replaces the ratio-based clip trigger with a KL trust-region condition and, when the policy leaves that region, applies a rollback penalty on the KL divergence of each sample. The repair is principled but costly to adopt: it introduces two new hyperparameters (the rollback strength α and the trust-region radius δ) and evaluates a per-state KL divergence, while still resolving each sample independently. CARE's Mod 1 computes the same per-sample ratios and surrogates as GRPO, but rather than deciding sample by sample, it compares the batch-mean surrogates by magnitude and keeps the signed value of whichever is smaller as the loss. This closes the per-sample escape route while preserving the correct gradient direction, and adds no new hyperparameters to those GRPO already uses. Zhai et al. (2026) report gradient measurements that locate the same region independently: on mathematical reasoning, NAHR tokens carry the largest mean gradient magnitude of any bucket they report, despite covering a fraction of a percent of negative tokens. Their setting is high-entropy, so the region is a rare outlier there; under a bounded label set, it becomes a characteristic feature of early training.

Prior repairs to advantage computation. REINFORCE++ (Hu et al., 2025a) shows prompt-level normalization to be a biased estimator and normalizes over the global batch instead, the bias reducing as the pool grows. Gao et al. (2026) motivate batch-level normalization from the classification setting, treating the entire batch as a single group with a batch-mean baseline. Their setting avoids the degenerate uniform-batch case because both the self-reward and teacher-reward functions are continuous. CARE's Mod 2 adopts the same batch-level pooling. Batch-level pooling, however, does not by itself resolve the degenerate case: once the batch reward is uniform, the batch mean centers every advantage at zero. This failure mode remains rare in multi-step reasoning, but under a bounded label set it is where late training spends most of its steps (Section 3.3). DAPO discards the affected

groups outright, at a sampling cost that grows with accuracy; CARE instead retains these groups and uses them as informative training signal.

In each case the modification is made *around* the clipped surrogate rather than to it. Decoupled bounds widen the region where the clip is inert, dynamic sampling changes which groups reach the loss, and a wider pool changes the population the mean is taken over. The per-sample decision and the mean-centered advantage survive all of them. CARE moves only the selection decision, choosing the batch-mean surrogate with smaller magnitude and leaves the remaining choices (clip bounds, batch filtering, normalization pooling) free, which is why it composes with existing optimizers (Table 1). Appendix B gives the full loss for each method.

2.2 GENERATIVE CLASSIFICATION

A separate line of work studies LLMs that emit a class label directly as a generated token rather than through a classification head (He et al., 2025). He et al. (2025) and Li et al. (2025) (vision-language setting) independently find that explicit chain-of-thought is unnecessary, and often harmful, for rule-based reinforcement fine-tuning on classification. Jungo & Fischer (2025) find that rule-based RL generalizes better out-of-distribution than supervised fine-tuning (SFT) on document image classification. Closest to our application, RL4HS (Su et al., 2025) applies GRPO with a span-level F1 reward to hallucination span detection on RAGTruth, retaining the surrogate we modify; Appendix A reviews the wider hallucination detection literature.

He et al. (2025) studies generative classification most broadly among prior work, running a comprehensive sweep over training and inference-time strategies for classifying with LLMs, including a comparison of RL optimizers (GRPO, REINFORCE++, and DPO (Rafailov et al., 2024)) from which they select REINFORCE++ for downstream use on the strength of its higher accuracy and simpler sampling; we carry that choice into our own comparison (Table 1). In that sweep, the optimizer is one configurable choice among several; we instead take the GRPO template itself as the object of study, identify its failures once the output space is bounded (Propositions 1 and 2), and repair them directly.

3 METHODOLOGY

3.1 RLVR FOR GENERATIVE CLASSIFICATION

We consider a policy π_θ that, given an input x built from a prompt template, emits a short string a from a bounded classification label set $\mathcal{Y}$ (binary for hallucination detection; multi-class for ANLI). The reward is calculated using a programmatic check of the generated string against the gold label y:

$$R(a, y) = \begin{cases} +1 & a = y \quad \text{(correct prediction)} \\ -1 & a \neq y \quad \text{(incorrect prediction)} \\ -2 & a \text{ is empty or violates the output format} \end{cases} \tag{1}$$

Each episode contains one input, one generated label, and one terminal reward. Thus, no temporal credit assignment or value function is required. Each training step samples N inputs and generates G completions for each input, forming a group $\mathcal{G}_q$. As discussed in Section 1, the bounded label set and few-valued reward place this setting in the low-entropy, low-reward-cardinality regime. GRPO computes advantage at group level:

$$A_i = \frac{R_i - \text{mean}(R_q)}{\text{std}(R_q)} \tag{2}$$

where mean and standard deviation are computed over the G completions sharing input q. Importance-sampling (IS) ratios $r_i(\theta) = \pi_\theta(a_i|x_i)/\pi_{\theta_{\text{old}}}(a_i|x_i)$, with $\pi_{\theta_{\text{old}}}$ the policy that generated the rollouts, enable data reuse across K inner optimization epochs per rollout batch (Schulman et al., 2017; Queeney et al., 2021). The GRPO clipped surrogate objective is given by:

$$L_{\text{GRPO}}(\theta) = -\frac{1}{N}\sum_{q=1}^{N}\frac{1}{G}\sum_{i\in\mathcal{G}_q}\min\big(r_i(\theta)\,A_i,\ \text{clip}(r_i(\theta), 1{-}\varepsilon, 1{+}\varepsilon)\,A_i\big) \tag{3}$$

with $\varepsilon = 0.2$. Because the gold label y is available at training time, this setting also admits a purely supervised alternative maximising $\log \pi_\theta(y \mid x)$ against the gold labels and so we show results on

cross-entropy SFT (Section 4). Which branch the elementwise min in Eq. 3 selects depends on the sign of A_i; this asymmetry is the source of the first failure mode . The unconditional computation of the mean-centered advantage in Eq. 2 is the source of the second .

3.2 Failure Mode 1: Asymmetric Clipping

For a sample with advantage A_i and ratio r_i, let $\text{surr}_{1,i} = r_i \cdot A_i$ and $\text{surr}_{2,i} = \text{clip}(r_i, 1-\varepsilon, 1+\varepsilon) \cdot A_i$. When $A_i > 0$ and $r_i > 1+\varepsilon$: min selects $\text{surr}_{2,i}$ (clipped, conservative as intended). When $A_i < 0$ and $r_i > 1+\varepsilon$, $r_i A_i < (1+\varepsilon) A_i$ (both negative; $r_i A_i$ has larger magnitude), so min selects $r_i A_i$ i.e. **the unclipped, more aggressive update** (the gradient-active band in Figure 2a).

Proposition 1 (Asymmetric Update)**.** *For* NAHR *samples ($A_i < 0$, $r_i > 1+\varepsilon$), the elementwise* min *selects the unclipped surrogate with excess magnitude $\Delta_i = (r_i - (1+\varepsilon)) \cdot |A_i| > 0$. The expected batch excess is $\mathbb{E}[\text{excess}] = f \cdot \mathbb{E}\big[(r - (1+\varepsilon)) \cdot |A| \;\big|\; r>1+\varepsilon,\ A<0\big]$, where f is the* NAHR *fraction. The selected branch is linear in r_i and no algorithmic ceiling applies to it, since the term that would impose one is precisely the branch the* min *discards.*

NAHR is primarily an early-training phenomenon because it requires both an incorrect prediction and a large importance ratio. Its frequency falls as accuracy improves and the policy stabilises. However, the remaining events can grow in magnitude as incorrect predictions become sparse outliers. Appendix D derives these dynamics. They are consistent with the early reward oscillations in Figure 1a.

Modification 1 (Mod 1): Batch-Level Conservative Surrogate Selection. We replace elementwise min with a batch-level magnitude comparison. Let $\overline{\text{surr}_1}$ and $\overline{\text{surr}_2}$ denote the batch-mean unclipped and clipped surrogates respectively:

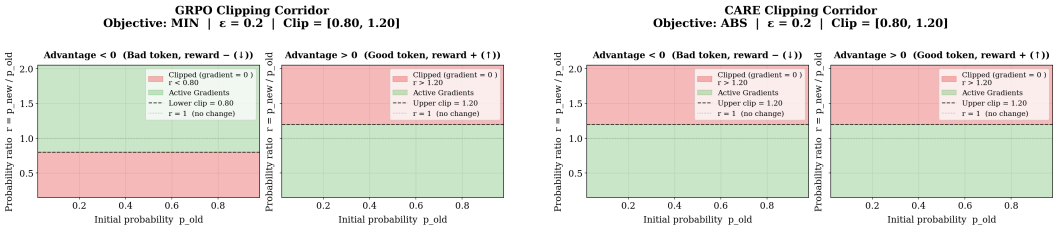
$$\overline{\text{surr}_1} = \frac{1}{NG} \sum_i r_i(\theta) \cdot A_i \tag{4}$$

$$\overline{\text{surr}_2} = \frac{1}{NG} \sum_i \text{clip}\big(r_i(\theta), 1-\varepsilon, 1+\varepsilon\big) \cdot A_i \tag{5}$$

The CARE policy loss is

$$L_{\text{CARE}}(\theta) = \begin{cases} -\overline{\text{surr}_2} & \text{if } |\overline{\text{surr}_2}| < |\overline{\text{surr}_1}| \\ -\overline{\text{surr}_1} & \text{otherwise} \end{cases} \tag{6}$$

where the absolute values act only as the selection criterion. This criterion applies the same notion of conservatism regardless of the advantage sign. It therefore closes the NAHR escape route identified in Proposition 1. When most ratios remain inside the clipping interval, the two batch means coincide and Mod 1 is inert. Aggregating the decision also reduces its variance; Appendix C formalizes this property. Algorithm 1 gives the implementation.



(a) GRPO clipping regions (b) CARE clipping regions

Figure 2: Gradient-active regions in the ratio–advantage plane for GRPO (left) and CARE (right). GRPO produces gradient in the NAHR region ($r > 1+\varepsilon$, $A < 0$) that Proposition 1 identifies as destabilizing, and is clipped in the opposing region ($r < 1-\varepsilon$, $A < 0$); CARE's batch-level criterion reverses both. Section 5 partitions the plane into four quadrants and shows that this reversal is what enables entropy recovery.

3.3 Failure Mode 2: Vanishing Advantage

The second failure mode originates in the mean centering of Eq. 2. We call a group *uniform* when all G completions sampled for a given input receive identical rewards. Mean centering assigns zero advantage to every completion in such a group, which therefore contributes no policy gradient.

Proposition 2 (Gradient Starvation)**.** *Let α_q denote the per-input accuracy for input q. Assume that the G completions per input are drawn independently, and that format violations are negligible such that rewards are effectively binary ($R_i \in \{+1, -1\}$). A group of G completions is uninformative under any mean-centered advantage ($A_i = 0\ \forall\, i$) with probability $P(\text{uniform}) = \alpha_q^G + (1-\alpha_q)^G$. The expected fraction of zero-gradient groups across a training step is $\mathbb{E}_q[\alpha_q^G + (1-\alpha_q)^G]$, which grows monotonically with model accuracy for $\alpha_q > 1/2$. For $G = 2$ and $\alpha_q = 0.9$, each group is uninformative with probability 0.82; at $\alpha_q = 0.95$, this rises to 0.905.*

The failure therefore worsens monotonically with training success: the more accurate the policy becomes, the fewer groups carry learning signal (Appendix C). No choice of pooling granularity removes it, since any $A_i = f(R_i, \bar{R})$ vanishes identically once every reward in the pool coincides. Widening the pool makes that coincidence rarer without eliminating it, and under a bounded label set it remains frequent enough to dominate late training (Appendix D).

Modification 2 (Mod 2): Conditional Advantage Computation. We propose bypassing advantage normalization whenever the batch carries no reward variance, using the raw reward in place of an advantage that mean centering would have zeroed:

$$A_i = \begin{cases} \dfrac{R_i - \mathrm{mean}(R)}{\mathrm{std}(R)} & \mathrm{std}(R) > 0 \\ R_i & \mathrm{std}(R) = 0 \end{cases} \tag{7}$$

where $\mathrm{mean}(R)$ and $\mathrm{std}(R)$ are computed over the full batch of $N \cdot G$ samples. Because the rewards of Eq. 1 are integers, $\mathrm{std}(R) = 0$ is an exact test and introduces no threshold hyperparameter. On an all-correct batch the raw reward retains the positive reinforcement that mean centering would erase; on an all-incorrect batch it delivers a corrective penalty. Either way the batch continues to contribute gradient. Lines 5–10 of Algorithm 1 give the implementation.

4 Experiments

4.1 Setup

We fine-tune Mistral 7B Instruct v0.3 (Jiang et al., 2023; AI, 2024) with LoRA (Hu et al., 2022) and report Macro-F1 (%) throughout. All CARE variants share one setup: AdamW (Loshchilov & Hutter, 2019) at learning rate 4×10^{-5} with cosine annealing and 5% warmup, K=4 inner PPO epochs, G=2, and 10 outer epochs on 8×H100s. Full configurations are in Table F.1, Appendix F.

Baselines. We compare against two groups. Our primary baselines are the three GRPO-family optimizers, GRPO (Shao et al., 2024), DAPO (Yu et al., 2025), and DR.GRPO (Liu et al., 2025), which share the template analyzed in Section 3. We also compose CARE with DAPO and DR.GRPO to show that it layers cleanly on top of each algorithm's own sample-level modifications. We include additional baselines outside the GRPO family to provide external checks on our diagnosis. SFT trains directly on the labels that define the reward. REINFORCE++ (Hu et al., 2025a) keeps the clipped surrogate but pools the advantage over the global batch, the same pooling adopted by Mod 2. TRULY-PPO (Wang et al., 2020) is the closest prior repair to the surrogate selection of Section 3.2: it replaces ratio clipping with a KL trust region and a per-sample rollback, targeting Failure Mode 1 while leaving Failure Mode 2 untouched. We run it with group-relative advantages at CARE's rollout configuration, so the surrogate is the only component separating it from GRPO (δ=0.03, α=5; Appendix B.2). We first ran the primary baselines under CARE's configuration where all three collapsed to near-trivial decision boundaries, as Section 3.2 predicts for K=4 inner epochs (Table G.1, Appendix G). To avoid exploiting this weakness, Table 1 reports each baseline under the strongest configuration we could produce, using the authors' recommended hyperparameters (Table F.1, Appendix). Every comparison below is therefore against the best version of each baseline we could produce.

Table 1: Macro-F1 (%) for models trained on RAGTruth QA. The in-distribution column reports mean ± std over three seeds (per-seed breakdown in Table G.4); all out-of-distribution entries are single-seed (42).

	In-distribution	*Out-of-distribution*						
	RAGTruth	*RAGTruth*		*Internal QA*		*HaluEval*		
Method	**QA**	**Summ.**	**Data2Text**	**Easy**	**Hard**	**Dialogue**	**QA**	**Summ.**
Primary baselines: the GRPO *family*								
GRPO	70.44 ± 0.66	64.03	54.50	53.03	59.18	45.89	79.53	65.00
DR.GRPO	69.66 ± 0.50	64.56	43.97	51.89	58.35	44.72	79.98	63.11
DAPO	71.85 ± 0.47	67.05	50.91	54.52	59.73	43.61	79.96	63.37
*Non-*GRPO *objectives*								
SFT	79.77 ± 1.37	52.14	46.22	36.93	37.46	39.47	74.13	42.25
REINFORCE++	70.68 ± 1.03	62.68	50.08	50.48	52.59	45.62	77.75	61.04
TRULY-PPO†	79.12 ± 1.27	69.76	59.72	56.42	54.29	57.72	76.88	63.22
Ours-CARE	**81.90 ± 1.25**	**73.58**	52.63	**64.88**	**61.26**	49.44	82.00	**66.44**
Ours-CARE+DR.GRPO	80.08 ± 0.26	72.77	**68.20**	58.98	54.54	55.49	**84.22**	64.66
Ours-CARE+DAPO	79.44 ± 2.15	72.34	59.72	59.70	60.14	**57.89**	77.96	65.92

†Our port of TRULY-PPO; we are not aware of a public implementation for LLM training (Appendix B.2).

Datasets. We evaluate on three hallucination detection benchmarks. Two are public. **RAGTruth** (Niu et al., 2024) spans QA, Summarization, and Data-to-Text over retrieval-augmented (Lewis et al., 2020) outputs from four generator models, and **HaluEval** (Li et al., 2023; llm-semantic router, 2025) spans QA, Summary, and Dialogue. The third, **Internal QA**, is a proprietary financial-services benchmark with *Easy* and *Hard* test splits, the latter containing adversarially subtle hallucinations.[1] It contributes a domain shift the public sets do not cover. We train separately on RAGTruth and Internal QA, and evaluate both in-distribution and cross-domain. We additionally evaluate on round 2 (A2) of the multi-class Adversarial NLI benchmark (Nie et al., 2020), which tests generality beyond binary labels on public data. Full dataset descriptions and statistics are in Table F.2, Appendix F.

4.2 MAIN RESULTS

CARE and its compositions outperform every baseline. Table 1 reports results for models trained on RAGTruth QA. CARE reaches 81.90 in-distribution, exceeding GRPO by 11.5 points and the strongest primary baseline by 10.1 points. It leads five of eight evaluation columns. Its compositions lead the remaining three, so every column is won by CARE or one of its compositions. The same pattern holds when training on Internal QA (Appendix G). Among non-GRPO objectives, TRULY-PPO recovers much of the gap and reaches 79.12, supporting our diagnosis that surrogate selection is central. CARE remains 2.8 points ahead without introducing TRULY-PPO's two additional hyperparameters. REINFORCE++ reaches 70.68, showing that batch-level pooling alone does not address both failures.

CARE generalizes where supervised fine-tuning does not. Under the same model, data, and compute budget, SFT is a strong in-distribution competitor. Its 79.77 on RAGTruth QA is ahead of every primary baseline and behind only CARE. Out of distribution the ordering inverts sharply. SFT falls to 52.14 on RAGTruth Summarization, and to 42.25 and 39.47 on HaluEval Summarization and Dialogue, where CARE reaches 73.58, 66.44, and 49.44. Every RLVR variant transfers; SFT does not. The same contrast has been reported for supervised against reinforcement post-training more broadly (Chu et al., 2025; Jin et al., 2025). SFT optimizes toward one fixed target and tends to fit it, whereas a verifiable reward keeps refining the decision boundary and carries to domains unseen in training.

Generality beyond binary classification. On the multi-class Adversarial NLI benchmark, CARE attains 64.40 Macro-F1 (64.02 accuracy) against 59.20 for the strongest primary baseline (DAPO; Table G.5, Appendix), confirming that its benefits extend to multi-class classification.

[1]Internal QA cannot be released. It serves only as an additional out-of-distribution evaluation and as a second training corpus; every headline claim in this paper is measured on public data.

Table 2: Ablation on RAGTruth QA. All rows except the recommended GRPO baseline use CARE's training configuration. All entries are single seed (seed 42).

Condition	Macro F1	Acc
GRPO (recommended)	69.51	77.83
+ Mod 1, sample-level	73.35	80.11
+ Mod 1, batch-level	74.97	84.57
+ Mod 2	79.49	86.06
CARE (Mod 1 + Mod 2)	**80.41**	**86.86**

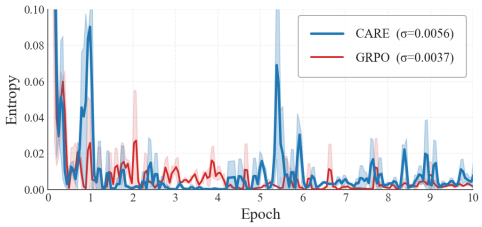


Figure 3: Training entropy for CARE vs GRPO (rolling mean, window $= 5$; shaded $= \pm 1\sigma$). CARE recovers entropy where GRPO stays collapsed (Section 5).

4.3 Ablations and Controls

Component contributions and decision granularity. We report the ablation on RAGTruth QA in Table 2. Both modifications are individually effective, giving $+5.46$ and $+9.98$ Macro F1 over GRPO at its recommended configuration, the stronger of its two settings in the table, and neither alone reaches the full result. Enabling both gives $+10.90$, consistent with the overlap in their temporal coverage (Figure 1b). The table also closes the gap between our per-sample analysis (Proposition 1) and the batch-level implementation of Mod 1. Applied at sample level, Mod 1 reaches 73.35 against 74.97 at batch level, consistent with the decision-variance reduction established in Proposition 3.

Group size and inner epochs. Ablating the group size $G \in \{2, 4, 8\}$ with all other hyperparameters fixed leaves CARE essentially unchanged, the spread across the three settings being smaller than its seed-to-seed variation (Table G.6, Appendix). Batch-level advantage estimation draws on the full batch rather than the group, so a smaller group does not inflate estimator variance, and we default to G=2 to minimize the number of generations while retaining adequate signal for stable policy optimization. Together with the matched-configuration runs of Section 4.1, this attributes CARE's +11.5 over GRPO to the two modifications rather than to group size or inner epochs.

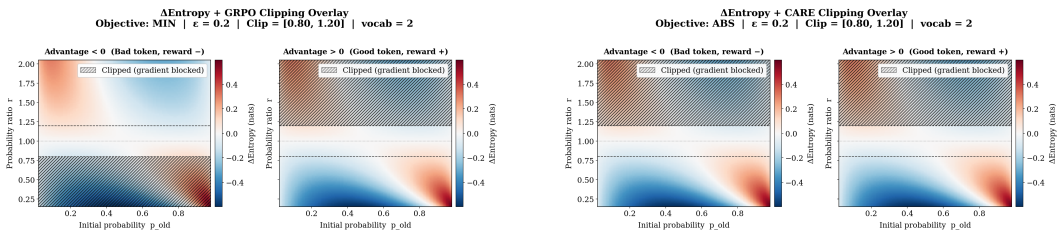
Seeds and backbones. Three-seed runs on Mistral 7B give CARE 81.90 ± 1.25 against GRPO's 70.44 ± 0.66 on RAGTruth QA (Table G.4, Appendix), and a full replication on LLaMA 3.1-8B (Grattafiori et al., 2024) reproduces CARE's advantage across all evaluation settings (Table G.3, Appendix). CARE's improvements are therefore neither seed-dependent nor architecture-specific.

5 Analysis: Entropy Recovery via Conservative Selection

Mod 1 prevents gradient overshoot due to NAHR samples (Section 3.2). We now show that the same mechanism produces a complementary benefit: entropy recovery in collapsed-policy regimes by activating gradients in a region where GRPO is clipped.

Gradient-active regions. We partition the ratio-advantage plane (Figure 2) by the sign of A and the position of r relative to the clip interval $[1-\varepsilon, 1+\varepsilon]$. Inside that interval the clipped and unclipped surrogates coincide, so both objectives are gradient-active there by construction; collapsing it leaves four quadrants: Q1 ($r > 1+\varepsilon$, $A > 0$), Q2 ($r > 1+\varepsilon$, $A < 0$, the NAHR zone), Q3 ($r < 1-\varepsilon$, $A < 0$), and Q4 ($r < 1-\varepsilon$, $A > 0$). By Proposition 1, GRPO is gradient-active in Q2 and Q4, and clipped in Q1 (its intended conservatism) and Q3 (Figure 2a). CARE's batch-level conservative surrogate selection produces the entropy-recovering pattern. In Q3, the unclipped term $r \cdot A$ has *smaller* magnitude than the clipped term $(1-\varepsilon) \cdot A$ (since $r < 1-\varepsilon$ and both products are negative). The conservative criterion $|\overline{\text{surr}_1}| < |\overline{\text{surr}_2}|$ is satisfied, and CARE selects the unclipped surrogate keeping the gradient active. In Q2, the unclipped surrogate has larger magnitude, so CARE selects the clipped surrogate, zeroing the gradient. CARE is thus active in Q3 and Q4, clipped in Q1 and Q2 (Figure 2b).[2]

[2]The quadrant argument is stated per sample. CARE decides once per batch, so a sample follows this pattern only when its batch resolves to the corresponding surrogate.



(a) GRPO ΔEntropy overlay (b) CARE ΔEntropy overlay

Figure 4: ΔEntropy heatmap with each method's clipping regions overlaid. GRPO (left) blocks gradients exactly where suppression would raise entropy; CARE (right) permits them, enabling entropy recovery without an explicit bonus.

How Q3 gradient activation recovers entropy. Under low-entropy generative classification the policy tends to collapse towards a near-deterministic distribution, placing high probability p_{old} on a single label. The only corrective signal comes from negative-advantage tokens, that is, from misclassified inputs. Such tokens enter each rollout batch at $r = 1$ and are driven downward along the ratio axis over the K inner epochs if the optimizer successfully suppresses them, crossing into Q3 once $r < 1-\varepsilon$. Q3 is therefore not a corner case in this regime but the destination of every successful suppression: the more effectively the optimizer corrects a collapsed policy, the more of its negative-advantage mass ends up there. GRPO clips at that boundary, freezing the gradient and leaving probability mass on the incorrect label, so the collapse the clip is meant to guard against is precisely what it preserves. CARE keeps suppressing across the remaining inner epochs, redistributes that mass, and recovers entropy. Figure 3 shows this playing out: both methods start from identical entropy (0.277), GRPO collapses to near zero by epoch 4–5 and remains trapped, while CARE recovers after collapsing to near zero, intermittently rather than monotonically, consistent with a batch-level trigger that fires only when the batch carries enough negative-advantage mass to make the unclipped surrogate the less aggressive of the two. Figure 4 localizes the mechanism. Each panel overlays one method's clipping regions on a heatmap of ΔEntropy, the entropy change a gradient step at that point would produce. Positive ΔEntropy sits at high p_{old} and negative advantage, the Q3 corner a collapsed policy drives its misclassified samples into. GRPO blocks gradients there and the collapse persists; CARE permits them.

6 Conclusion and Limitations

We identified two failures of the GRPO objective in low-entropy generative classification: NAHR-driven instability and vanishing mean-centered advantages. CARE addresses them through batch-level conservative surrogate selection (Mod 1) and conditional advantage computation (Mod 2), without adding hyperparameters. Across RAGTruth QA, CARE improves over GRPO by 11.5 macro-F1 points and retains its advantage across seven out-of-distribution evaluations. It also composes with DR.GRPO and DAPO. Beyond stabilizing optimization, its batch-level selection enables entropy recovery without an explicit entropy bonus. These results position CARE as a direct replacement for the GRPO objective in generative classification.

Limitations and future work. CARE is designed for low-entropy generative classification. Its behavior in high-entropy settings such as multi-step reasoning remains untested. Mod 2 should rarely activate there, since a batch with identical rewards is unlikely. Mod 1, by contrast, can still alter the update whenever ratios leave the clip interval. Its impact in that regime remains to be characterized. Another limitation is that the shift from group-level to batch-level normalization (Section 3.3) trades per-prompt difficulty calibration for statistical stability. A phase-dependent hybrid normalization scheme could be a future direction which uses group-level normalization while groups remain informative, and switches to batch-level normalization thereafter. Such a scheme would preserve per-prompt calibration early in training while avoiding learning-signal starvation later, potentially extending CARE to curriculum-structured pipelines. Another direction could be an entropy-aware schedule for K which increases the number of inner epochs once entropy has collapsed, giving Mod 1 more opportunities to redistribute probability mass without incurring additional generation cost.

ETHICS STATEMENT

Data. This work studies an optimization algorithm for generative classification. It does not involve human subjects, crowdsourced annotation of personal data, or the release of a new dataset. RAGTruth, HaluEval, and ANLI are public benchmarks used under their original licenses. Internal QA is proprietary financial-services data. It was assembled from internal documents under the data-governance policy of the institution that owns it, screened for personally identifiable information before use, and accessed only within that institution's approved compute environment. We do not release Internal QA, and no headline claim in this paper depends on it (Section 4.1).

Asymmetric error costs. A hallucination detector functions as a safety filter, and its two error modes have asymmetric consequences. A false positive flags a faithful answer and consumes reviewer time; a false negative allows an unsupported claim to reach the user. In finance, healthcare, legal, and enterprise retrieval-augmented settings, the latter error can cause material harm. Our main metric, macro-F1, weights both classes equally and therefore does not encode application-specific cost asymmetries. Deployers of a CARE-trained detector should choose operating procedures according to the error costs and risk tolerance of their own application, rather than relying solely on the aggregate numbers reported here.

Calibration, thresholds, and abstention. Our policy emits a discrete label string rather than a calibrated score. This has two practical consequences. First, the decision procedure produces no score of its own. A score can be recovered post hoc and the standard diagnostics then apply to it, but nothing in training encourages it to be calibrated, since the reward of Eq. 1 scores only which label is emitted. Second, because there is no explicit decision threshold, the usual mechanism for selecting a recall-oriented operating point is unavailable. Practitioners could address this in several ways, none of which we evaluate here: using the token-level likelihood of the emitted label as a score, sampling multiple completions and using agreement as a confidence proxy, or adding an explicit ABSTAIN option to $\mathcal{Y}$ so that uncertain inputs are routed to human review. The abstention approach would require a reward that prices abstention between a correct and an incorrect prediction, modifying Eq. 1; we leave this extension to future work.

Class imbalance. The hallucinated class is a minority in several evaluation sets: 18% in RAGTruth QA and 23% in RAGTruth Summarization. We balance the training split (Appendix F) and report macro-F1 rather than accuracy so that minority-class performance remains visible. Deployed traffic may be even more skewed, since hallucinations should be rare in a well-engineered pipeline; as the base rate falls, precision at any fixed recall will generally degrade. Our out-of-distribution results (Table 1) provide evidence of transfer, but they are not a substitute for evaluation on the target deployment distribution.

Automation bias and fairness. We identify no direct dual-use risk specific to the optimizer. CARE is a general-purpose modification to a policy-gradient objective and is agnostic to the downstream task. The application, however, introduces an important risk: even an accurate detector can induce automation bias if reviewers stop checking outputs marked as faithful. We therefore recommend using such detectors to prioritize human review rather than to replace it. We did not audit differential error rates across demographic, dialectal, or other protected subgroups because the benchmarks used in this work do not include the attributes required for such an analysis. Such an audit is necessary before deployment in settings subject to fair-lending or comparable fairness obligations.

Conflicts of interest. The authors declare no conflicts of interest.

REPRODUCIBILITY STATEMENT

All headline empirical claims are evaluated on public benchmarks. The reported +11.5 macro-F1 improvement over GRPO is measured on RAGTruth QA, the multi-class result is measured on ANLI, and the ablations, group-size sweep, matched-configuration runs, multi-seed runs, and backbone replication are all conducted on RAGTruth QA. RAGTruth, HaluEval, and ANLI are publicly available.

The information needed to reproduce the public-data results is provided in the paper and appendix. Appendix E gives the full CARE procedure in pseudocode (Algorithm 1). Appendix B specifies the complete loss formulations for CARE and all baselines, including our implementation of TRULY-PPO (Appendix B.2), for which we are not aware of a public LLM training implementation. Appendix C contains proofs for the formal claims. Appendix F reports the model choice, LoRA configuration, optimizer settings, and per-algorithm hyperparameters (Table F.1), together with dataset statistics (Table F.2), the span-to-binary label conversion, and the dataset-balancing procedure and random seed. Prompt templates for all tasks are provided in Appendix H.

We are unable to release source code for institutional reasons. To support reproducibility despite this restriction, we specify both CARE modifications directly in the paper: Appendix E gives line-level pseudocode, and Appendix B gives the full objective for CARE and for every baseline in the comparison. The modifications require only a small number of changes to a standard GRPO implementation and introduce no additional hyperparameters.

The Internal QA benchmark is proprietary financial-services data and cannot be released. It is used only as an additional out-of-distribution evaluation and as a second training corpus; no headline claim in this paper depends on it. We report its summary statistics in Table F.2 and describe its construction in Appendix F, so that the experimental setup can be understood and approximated on comparable data.

AI USE STATEMENT

We used generative AI tools only to assist with writing and editing this paper. The tools were applied to author-drafted text to improve clarity, readability, grammar, and flow. The scientific content, including the problem formulation, failure-mode diagnosis, CARE algorithm, theoretical claims, proofs, implementation, data preparation, experimental design, and interpretation of results, was developed by the authors.

All AI-assisted edits were reviewed by the authors. The authors checked edited passages against the corresponding technical claims, results, and citations, and made the final decisions about wording and inclusion. No numerical result, citation, theorem, proof, algorithmic contribution, or experimental conclusion is included without author verification. The authors take full responsibility for the final content of the paper.

REFERENCES

Pulkit Agrawal, Rukma Talwadker, Aditya Pareek, and Tridib Mukherjee. FAST-Q: Fast-track exploration with adversarially balanced state representations for counterfactual action estimation in offline reinforcement learning. In *Companion Proceedings of the ACM Web Conference 2025 (WWW '25)*, pp. 85–94. Association for Computing Machinery, 2025. doi: 10.1145/3701716.3715224.

Mistral AI. Mistral-7b-instruct-v0.3 large language model. `https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3`, 2024.

Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V. Le, Sergey Levine, and Yi Ma. SFT memorizes, RL generalizes: A comparative study of foundation model post-training. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Yingying Gao, Shilei Zhang, Runyan Yang, Zihao Cui, and Junlan Feng. B-GRPO: Unsupervised speech emotion recognition based on batched-group relative policy optimization. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2026.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Honghui Ding, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jingchang Chen, Jingyang Yuan, Jinhao Tu, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaichao You, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645 (8081):633–638, September 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL http://dx.doi.org/10.1038/s41586-025-09422-z.

Mingqian He, Fei Zhao, Chonggang Lu, Ziyan Liu, Yue Wang, and Haofu Qian. GenCLS++: Pushing the boundaries of generative classification in LLMs through comprehensive SFT and RL studies across diverse datasets. *arXiv preprint arXiv:2504.19898*, 2025.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

Jian Hu, Jason Klein Liu, Haotian Xu, and Wei Shen. REINFORCE++: Stabilizing critic-free policy optimization with global advantage normalization. *arXiv preprint arXiv:2501.03262*, 2025a.

Jian Hu, Xibin Wu, Wei Shen, Jason Klein Liu, Weixun Wang, Songlin Jiang, Haoran Wang, Hao Chen, Bin Chen, Wenkai Fang, Xianyu, Yu Cao, Haotian Xu, and Yiming Liu. OpenRLHF: A ray-based easy-to-use, scalable and high-performance RLHF framework. In Ivan Habernal, Peter Schulam, and Jörg Tiedemann (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 656–666, Suzhou, China, November 2025b. Association for Computational Linguistics. ISBN 979-8-89176-334-0. doi: 10.18653/v1/2025.emnlp-demos.48. URL https://aclanthology.org/2025.emnlp-demos.48/.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55, January 2025. ISSN 1558-2868. doi: 10.1145/3703155. URL http://dx.doi.org/10.1145/3703155.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38, 2023.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier,

Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023. URL https://arxiv.org/abs/2310.06825.

Hangzhan Jin, Sitao Luan, Tianwei Ni, Sicheng Lyu, Guillaume Rabusseau, Reihaneh Rabbany, Doina Precup, and Mohammad Hamdaqa. RL fine-tuning heals OOD forgetting in SFT. *arXiv preprint arXiv:2509.12235*, 2025.

Michael Jungo and Andreas Fischer. Rule-based reinforcement learning for document image classification with vision language models. In *Document Analysis and Recognition – ICDAR 2025 Workshops*, pp. 292–309. Springer Nature Switzerland, 2025. doi: 10.1007/978-3-032-09368-4_18.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474, 2020.

Junyi Li, Xiaoxue Cheng, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. HaluEval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6449–6464. Association for Computational Linguistics, 2023.

Ming Li, Jike Zhong, Shitian Zhao, Yuxiang Lai, Haoquan Zhang, Wang Bill Zhu, and Kaipeng Zhang. To think or not to think: A study of thinking in rule-based visual reinforcement fine-tuning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

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 (COLM)*, 2025.

llm-semantic router. Halueval span-level dataset (RAGTruth-normalized). Hugging Face Hub, 2025. Prompts normalized to RAGTruth format for cross-dataset compatibility.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Potsawee Manakul, Adian Liusie, and Mark Gales. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 9004–9017. Association for Computational Linguistics, 2023.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 1906–1919. Association for Computational Linguistics, 2020.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, KaShun Shum, Randy Zhong, Juntong Song, and Tong Zhang. RAGTruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10862–10878, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.585.

OpenAI. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

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. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744, 2022.

James Queeney, Ioannis Ch. Paschalidis, and Christos G. Cassandras. Generalized proximal policy optimization with sample reuse. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, 2021.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, pp. 53728–53741, 2024.

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

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

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

Hsuan Su, Ting-Yao Hu, Hema Swetha Koppula, Kundan Krishna, Hadi Pouransari, Cheng-Yu Hsieh, Cem Koc, Joseph Yitan Cheng, Oncel Tuzel, and Raviteja Vemulapalli. Learning to reason for hallucination span detection. *arXiv preprint arXiv:2510.02173*, 2025.

Liyan Tang, Philippe Laban, and Greg Durrett. MiniCheck: Efficient fact-checking of LLMs on grounding documents. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 8818–8847. Association for Computational Linguistics, 2024.

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. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. TRL: Transformer reinforcement learning. `https://github.com/huggingface/trl`, 2020.

Yuhui Wang, Hao He, and Xiaoyang Tan. Truly proximal policy optimization. In *Proceedings of the 35th Conference on Uncertainty in Artificial Intelligence (UAI)*, volume 115 of *Proceedings of Machine Learning Research*, pp. 113–122. PMLR, 2020.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. DAPO: An open-source LLM reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. AlignScore: Evaluating factual consistency with a unified alignment function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11328–11348. Association for Computational Linguistics, 2023.

Zepeng Zhai, Meilin Chen, Jiaxuan Zhao, Junlang Qian, Lei Shen, and Yuan Lu. Rewards as labels: Revisiting RLVR from a classification perspective. *arXiv preprint arXiv:2602.05630*, 2026.

Rui Zheng, Shihan Dou, Songyang Gao, Yuan Hua, Wei Shen, Binghai Wang, Yan Liu, Senjie Jin, Qin Liu, Yuhao Zhou, Limao Xiong, Lu Chen, Zhiheng Xi, Nuo Xu, Wenbin Lai, Minghao Zhu, Fei Huang, Tao Gui, Qi Zhang, and Xipeng Qiu. Secrets of RLHF in large language models part I: PPO. *arXiv preprint arXiv:2307.04964*, 2023.