BALTO: Balanced Token-Level Policy Optimization for Hallucination Mitigation

Ning Li[1]*, **Zixuan Guo**[2]*, **Yan Xu**[2], **Wenbo Fei**[1], **Yifan Niu**[3], **Chang Luo**[2], **Yasheng Wang**[2]†, **Weiwen Liu**[1]†, **Yong Yu**[1], **Weinan Zhang**[1]†
[1]Shanghai Jiao Tong University
[2]Tencent
[3]The Hong Kong University of Science and Technology (Guangzhou)

Abstract

Hallucinations remain a major obstacle to deploying large language models (LLMs) in knowledge-intensive settings, where generated responses must be faithfully grounded in provided evidence. Reinforcement learning (RL) is a promising direction for hallucination mitigation, but response-level faithfulness rewards suffer from a granularity mismatch: localized hallucinations can cause supported content to receive spurious penalties. Although recent work introduces fine-grained feedback such as claim-level verification and token-level rewards, unbalanced credit assignment can still induce length, verbosity, or optimization-noise biases. We propose **BALTO**, a **Bal**anced **T**oken-level Policy **O**ptimization framework for hallucination mitigation. BALTO extracts checkable factual claims, verifies them against the reference context, and projects claim-level judgments to token-level labels. A balanced token-level credit assignment mechanism is introduced into the framework. This design redistributes probability mass from unsupported content toward faithful content, rather than suppressing the entire response. We systematically analyze the limitations of response-level rewards from a theoretical standpoint, and prove BALTO's advantages in training stability and optimization efficiency for hallucination mitigation. Experiments on ConFiQA, RAGTruth, and FinLLM-Eval show that BALTO achieves the highest faithfulness across all six model–benchmark settings and consistently outperforms existing post-training baselines in Q-Score, demonstrating a stronger faithfulness–informativeness trade-off. The code is released at `https://github.com/NingLi670/BALTO`.

1 Introduction

Large Language Models (LLMs) have demonstrated remarkable capabilities across a wide range of tasks [34, 52, 51]. However, their real-world deployment remains significantly constrained by faithfulness hallucinations—generating content that lacks factual support or contradicts provided evidence [39]. Such hallucinations pose significant risks in high-stakes domains such as healthcare [20], law [10], finance [18], and scientific discovery [45]. Unlike global factual failures, these hallucinations often appear as localized, unsupported spans (e.g., incorrect entities, dates, numerical values, or logical relations) embedded within otherwise coherent and accurate content. This locality makes them particularly difficult to detect and mitigate.

Currently, most reinforcement learning (RL)-based methods for mitigating hallucinations employ response-level rewards, assigning a single scalar score to the entire output. However, such coarse-

*Equal contribution.

†Correspondence to: Yasheng Wang (`asheryswang@tencent.com`), Weiwen Liu (`wwliu@sjtu.edu.cn`), and Weinan Zhang (`wnzhang@sjtu.edu.cn`).

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

grained supervision cannot distinguish hallucinated spans from faithful ones: a uniform penalty applied to all tokens may erroneously penalize correct information while providing only weak corrective signals to the actual hallucinated content. This not only diminishes the effectiveness of hallucination mitigation but may also induce reward hacking: models may learn to generate shorter, more conservative responses to avoid penalties, thereby reducing the overall informativeness and utility of the response.

Recent studies [43, 23] have moved beyond response-level supervision to incorporate fine-grained feedback signals, such as step-wise faithfulness verification and token-level dense rewards. Nevertheless, fine-grained feedback alone does not determine how policy-gradient credit should be assigned. Unbalanced token-level rewards may introduce length or faithful-density bias, while token-level updates whose magnitudes remain tied to response-level advantages can still inherit noise from global response-level rewards. These observations suggest that hallucination mitigation requires not only fine-grained detection, but also balanced token-level credit assignment.

Based on this premise, we propose **BALTO** (**Bal**anced **T**oken-Level Policy **O**ptimization), a reinforcement learning framework for fine-grained hallucination mitigation. BALTO localizes hallucinations at the token level and introduces a balanced credit assignment mechanism: hallucinated tokens receive negative advantages, faithful tokens receive positive compensatory advantages, and neutral tokens are excluded. By balancing the total negative and positive faithfulness-driven signals within each response, BALTO encourages the model to redistribute probability mass from hallucinated content toward faithful content, rather than suppressing the response as a whole.

We provide both theoretical and empirical analysis to demonstrate the benefits of BALTO. Theoretically, we show that response-level rewards induce spurious credit assignment under localized hallucinations, whereas balanced token-level credit assignment reduces unnecessary parametric updates and improves optimization stability and efficiency. Empirically, across ConFiQA, RAGTruth, and FinLLM-Eval with Qwen3-8B and Qwen3-4B, BALTO consistently outperforms both response-level and token-level post-training baselines in both faithfulness and Q-Score across all six model–benchmark settings. Compared with the strongest post-training baseline in each setting, BALTO improves Q-Score by 2.7 points on average and up to 4.9 points, while improving faithfulness by 3.1 points on average and up to 6.7 points. These results suggest that BALTO provides a more effective faithfulness–informativeness trade-off than prior post-training approaches, reducing hallucinations without collapsing into overly conservative generation.

Our main contributions are summarized as follows:

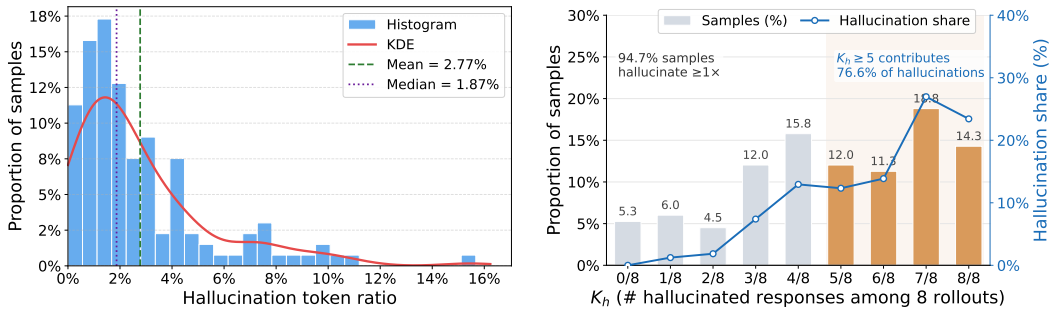
- **Balanced Token-level Hallucination Mitigation.** We propose BALTO, a token-level RL framework for fine-grained hallucination mitigation. BALTO converts claim-level factuality judgments into token-level optimization signals and uses balanced credit assignment to shift probability mass from hallucinated content toward faithful content.
- **Theoretical Credit Assignment Analysis.** We systematically analyze the credit assignment limitations of response-level faithfulness rewards and show that balanced token-level advantages reduce unnecessary parametric updates, improving training stability and optimization efficiency.
- **SOTA Performance.** Across ConFiQA, RAGTruth, and FinLLM-Eval with Qwen3-8B and Qwen3-4B, BALTO achieves the highest faithfulness in all six model–benchmark settings and consistently outperforms existing post-training methods in Q-Score, demonstrating a stronger faithfulness–informativeness trade-off.

2 Background and Motivation

2.1 Pilot Study: Hallucinations Are Token-Sparse but Response-Prevalent

Faithfulness hallucinations in RAG-style generation are often localized factual errors rather than complete response-level failures. Prior studies similarly analyze factual errors at the level of claims, atomic facts, or evidence-grounded statements [16, 1, 7, 32, 47]. To quantify this locality, we conduct a pilot study on FinLLM-Eval [24], a financial QA benchmark requiring answers to be grounded in numerical evidence.

As shown in Figure 1(a), hallucinated tokens account for only a small fraction of each response, with an average ratio of 2.77% and a median ratio of 1.87%. However, this token-level sparsity



(a) Hallucinated-token ratio per response.

(b) Hallucinated responses among eight rollouts.

Figure 1: Pilot study on FinLLM-Eval with Qwen3-8B. Hallucinations are sparse within individual responses but prevalent across rollouts. The shaded region in (b) marks majority-hallucinating groups.

does not imply response-level sparsity. Figure 1(b) shows that 94.7% of rollout groups contain at least one hallucinated response, and groups with majority-hallucinating rollouts ($K_h \geq 5$) contribute 76.6% of all hallucinations. Thus, hallucinations are sparse within responses but prevalent across sampled responses. This dual pattern makes response-level RL supervision inefficient: partially faithful responses with only a few erroneous tokens may receive the same binary penalty as severely hallucinated responses, motivating fine-grained and balanced token-level credit assignment.

2.2 Response-Level RL for Hallucination Mitigation

RL has become a common post-training paradigm for improving LLM faithfulness and alignment [34, 36, 6, 43, 23]. In a RAG setting, given an input $x = (q, d)$ with query q and reference documents d, the model generates a response $y = (y_1, \ldots, y_T)$ according to an autoregressive policy. In GRPO-style optimization, for each input x, a group of K responses $\{y_i\}_{i=1}^{K}$ is sampled from the old policy $\pi_{\theta_{\text{old}}}$. Given a response-level reward $r_i = R(x, y_i)$, GRPO computes the group-relative advantage: $A_i^{\text{resp}} = \frac{r_i - \text{mean}(\boldsymbol{r})}{\text{std}(\boldsymbol{r})}$, where $\boldsymbol{r} = (r_1, \ldots, r_K)$. The advantage formula applies when $\text{std}(\boldsymbol{r}) > 0$. This scalar advantage is broadcast to all tokens in the response. With lower and upper clipping widths $\epsilon_{\text{low}}, \epsilon_{\text{high}} \geq 0$, the clipped GRPO objective is

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{\{y_i\}_{i=1}^{K} \sim \pi_{\theta_{\text{old}}}(\cdot|x)} \left[\frac{1}{K} \sum_{i=1}^{K} \frac{1}{T_i} \sum_{t=1}^{T_i} \min \left(\frac{\pi_\theta(y_{i,t} \mid x, y_{i,<t})}{\pi_{\theta_{\text{old}}}(y_{i,t} \mid x, y_{i,<t})} A_i^{\text{resp}}, \text{clip} \left(\frac{\pi_\theta(y_{i,t} \mid x, y_{i,<t})}{\pi_{\theta_{\text{old}}}(y_{i,t} \mid x, y_{i,<t})}, 1 - \epsilon_{\text{low}}, 1 + \epsilon_{\text{high}} \right) A_i^{\text{resp}} \right) \right]. \tag{1}$$

This response-level assignment is problematic for localized hallucinations. Let $\mathcal{H}_i$ and $\mathcal{F}_i$ denote hallucinated and faithful factual tokens in response y_i. If a few hallucinated tokens cause a low faithfulness reward, then $A_i^{\text{resp}} < 0$, and the factual-token component of the update is proportional to

$$A_i^{\text{resp}} \left(\sum_{t \in \mathcal{H}_i} \nabla_\theta \log \pi_\theta(y_{i,t} \mid x, y_{i,<t}) + \sum_{t \in \mathcal{F}_i} \nabla_\theta \log \pi_\theta(y_{i,t} \mid x, y_{i,<t}) \right). \tag{2}$$

The first term suppresses hallucinated tokens, but the second term also suppresses faithful factual tokens. Since hallucinated tokens are typically sparse, this spurious penalty can dominate the faithfulness-driven update, reducing informativeness or encouraging overly conservative responses. This motivates moving beyond response-level rewards toward fine-grained hallucination feedback.

2.3 Fine-Grained Hallucination Feedback and the Remaining Assignment Gap

A natural remedy to response-level credit mismatch is to introduce finer-grained hallucination feedback. Prior work has explored claim-level correctness, atomic fact verification, statement-level faithfulness assessment, and dense process supervision for faithfulness alignment [25, 7, 12, 32, 47]. Recent RL-based hallucination mitigation methods further move toward token-level optimization. FSPO adjusts token-level advantages using step-wise faithfulness verification [23], while RLFH decomposes responses into atomic facts and converts statement-level truthfulness and informativeness

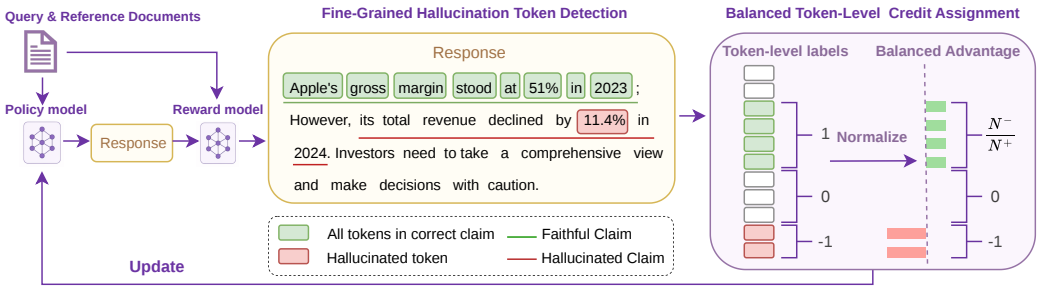


Figure 2: Overview of the BALTO framework is demonstrated. Initially, the reward model localize hallucinations at the token level. Subsequently, the balanced credit mechanism assigns token-level advantages: hallucinated tokens (red) receive -1 advantage , faithful factual tokens (green) receive positive compensatory advantages of N^-/N^+ , where N^- and N^+ are the numbers of hallucinated and faithful tokens respectively, and neutral tokens receive 0 advantage (white).

feedback into token-level dense rewards [43]. We provide a broader discussion of fine-grained hallucination mitigation and process supervision in Appendix A.

However, fine-grained feedback alone does not determine how policy-gradient credit should be assigned. Penalty-only token-level feedback may encourage shorter or less informative responses, while assigning fixed positive rewards to all faithful factual tokens may bias the model toward verbose factual padding. Similarly, token-level updates whose magnitudes remain tied to response-level advantages can still inherit noise from global response-level rewards.

These limitations motivate a balanced token-level assignment principle: hallucinated factual tokens should receive localized negative feedback, faithful factual tokens should receive compensatory positive feedback, and the total faithfulness-driven signal should be balanced within each response. In the next section, we instantiate this principle as a balanced token-level policy optimization framework for hallucination mitigation.

3 Methodology

We propose **BALTO**: **Bal**anced **T**oken-level Policy **O**ptimization, a fine-grained reinforcement learning framework for hallucination mitigation. As illustrated in Figure 2, BALTO consists of two stages. First, an LLM-based faithfulness evaluator extracts and verifies checkable factual claims from each sampled response. Second, BALTO projects claim-level judgments to token-level labels, constructs balanced token-level credits, and optimizes the policy with a GRPO-style clipped objective.

3.1 Fine-Grained Hallucination Token Detection

Given an input $x = (q, d)$ and a sampled response $y_i = (y_{i,1}, \ldots, y_{i,T_i})$, BALTO first identifies checkable factual claims in the response. A claim is defined as a minimal text span that expresses a verifiable factual assertion and can be independently checked against the reference document. We represent the extracted claims as $\mathcal{C}_i = \{c_{i,m}\}_{m=1}^{M_i}$, where each $c_{i,m} \subseteq \{1, \ldots, T_i\}$ denotes the token indices covered by the m-th claim. In practice, we focus on claims involving named entities, numerical values, dates, temporal expressions, attributes, and relations, which are common sources of faithfulness errors and align with atomic faithfulness evaluation [31, 29, 27]. The faithfulness evaluator verifies each claim against the reference document d and partitions the claims into faithful and hallucinated claim sets:

$$\mathcal{C}_i^+ = \{c_{i,m} \in \mathcal{C}_i : c_{i,m} \text{ is supported by } d\},$$
$$\mathcal{C}_i^- = \{c_{i,m} \in \mathcal{C}_i : c_{i,m} \text{ is unsupported or contradicted by } d\}.$$

We then project claim-level verification results back to token-level labels using the text-to-token alignment procedure in Appendix E.2. For faithful claims, the projection is straightforward: all tokens covered by the claim are labeled as faithful factual tokens. For hallucinated claims, however, assigning negative feedback to the entire claim span can be overly coarse, since a claim may contain only a small erroneous part. Therefore, for each hallucinated claim $c_{i,m} \in \mathcal{C}_i^-$, BALTO further localizes a minimal erroneous token subset $e_{i,m} \subseteq c_{i,m}$, where $e_{i,m}$ contains the tokens directly

responsible for the factual error, such as an incorrect entity, number, date, attribute, or relation. If the erroneous part cannot be reliably isolated, we conservatively set $e_{i,m} = c_{i,m}$. Formally, the token-level label is defined as

$$z_{i,t} = \begin{cases} -1, & \exists c_{i,m} \in \mathcal{C}_i^- \text{ such that } t \in e_{i,m}, \\ +1, & \exists c_{i,m} \in \mathcal{C}_i^+ \text{ such that } t \in c_{i,m} \text{ and } t \notin \bigcup_{c_{i,m'} \in \mathcal{C}_i^-} e_{i,m'}, \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

Here, $z_{i,t} = -1$ indicates that token $y_{i,t}$ is directly responsible for a hallucinated factual claim, $z_{i,t} = +1$ indicates that it belongs to a faithful factual claim, and $z_{i,t} = 0$ indicates that it is either outside checkable factual claims or is a faithful token inside a hallucinated claim. When a token is included in both faithful and hallucinated projections due to overlapping claims, the hallucination label takes precedence.

Finally, we define the hallucinated factual token set, faithful factual token set, and neutral token set as

$$\mathcal{H}_i = \{t \mid z_{i,t} = -1\}, \qquad \mathcal{F}_i = \{t \mid z_{i,t} = +1\}, \qquad \mathcal{N}_i = \{t \mid z_{i,t} = 0\}. \tag{4}$$

This projection differs from statement-level reward mapping, which typically assigns feedback to a whole matched statement or sentence [43]. BALTO instead localizes the factual error within a hallucinated claim and applies negative feedback only to the minimal erroneous tokens. For the hallucinated claim in Figure 2 that "total revenue declined by 11.4% in 2024", BALTO penalizes the token span corresponding to "11.4%" rather than the entire claim. This provides precise negative supervision while avoiding unnecessary suppression of surrounding faithful context. The resulting token sets $\mathcal{H}_i$, $\mathcal{F}_i$, and $\mathcal{N}_i$ are then used to construct balanced token-level advantages.

3.2 Balanced Token-Level Credit Assignment

BALTO directly assigns token-level advantages based on the factual status of tokens. For response y_i, let $N_i^- = |\mathcal{H}_i|$, $N_i^+ = |\mathcal{F}_i|$ denote the number of hallucinated factual tokens and faithful factual tokens, respectively. We define the balanced token-level advantage as

$$A_{i,t}^{\text{bal}} = \begin{cases} -1, & t \in \mathcal{H}_i, \\ \dfrac{N_i^-}{N_i^+}, & t \in \mathcal{F}_i,\ N_i^- > 0,\ N_i^+ > 0, \\ 0, & t \in \mathcal{N}_i \text{ or } N_i^- = 0. \end{cases} \tag{5}$$

Thus, hallucinated factual tokens receive a negative advantage, faithful factual tokens receive a positive compensation advantage, and tokens outside checkable factual claims do not contribute to faithfulness-driven optimization [3]. When both hallucinated and faithful factual tokens exist, BALTO satisfies an intra-response balance property:

$$\sum_{t=1}^{T_i} A_{i,t}^{\text{bal}} = \sum_{t \in \mathcal{H}_i} (-1) + \sum_{t \in \mathcal{F}_i} \frac{N_i^-}{N_i^+} = -N_i^- + N_i^+ \frac{N_i^-}{N_i^+} = 0. \tag{6}$$

Therefore, the total negative advantage assigned to hallucinated factual tokens is exactly matched by the total positive advantage assigned to faithful factual tokens. This encourages the model to redistribute probability mass from hallucinated factual content to supported factual content, rather than suppressing the entire response.

BALTO Objective Given an input x, we sample a group of K responses $\{y_i\}_{i=1}^K$ from the old policy $\pi_{\theta_{\text{old}}}$. For each response, we obtain token sets $\mathcal{H}_i$, $\mathcal{F}_i$, and $\mathcal{N}_i$ using the faithfulness evaluator, and assign balanced advantages according to Eq. 5. The BALTO objective is

$$\mathcal{J}_{\text{BALTO}}(\theta) = \mathbb{E}_{\{y_i\}_{i=1}^K \sim \pi_{\theta_{\text{old}}}(\cdot|x)} \left[\frac{1}{K} \sum_{i=1}^{K} \frac{1}{Z_i} \sum_{t \in \mathcal{H}_i \cup \mathcal{F}_i} \min \left(\frac{\pi_\theta(y_{i,t} \mid x, y_{i,<t})}{\pi_{\theta_{\text{old}}}(y_{i,t} \mid x, y_{i,<t})} A_{i,t}^{\text{bal}}, \text{clip} \left(\frac{\pi_\theta(y_{i,t} \mid x, y_{i,<t})}{\pi_{\theta_{\text{old}}}(y_{i,t} \mid x, y_{i,<t})}, 1 - \epsilon_{\text{low}}, 1 + \epsilon_{\text{high}} \right) A_{i,t}^{\text{bal}} \right) \right]. \tag{7}$$

[3] For faithful responses, where $N_i^- = 0$, all faithfulness-driven advantages are set to zero, avoiding an incentive to generate unnecessarily long responses containing redundant safe claims. If $N_i^- > 0$ but $N_i^+ = 0$, BALTO applies only the negative correction, since no faithful factual token is available for within-response compensation.

where

$$Z_i = \max(1, |\mathcal{H}_i| + |\mathcal{F}_i|) \tag{8}$$

normalizes the update over factual tokens. Since tokens in $\mathcal{N}_i$ have zero advantage, they do not contribute to the faithfulness-driven update.

Compared with response-level GRPO, BALTO provides localized and balanced supervision even when hallucinations are widespread at the response level. Compared with unbalanced token-level rewards, BALTO explicitly controls the total positive and negative advantage mass within each response. As a result, the policy receives precise corrective signals for hallucinated content while preserving response informativeness.

4 Theoretical Analysis

In this section, we analyze the theoretical advantages of BALTO over standard response-level GRPO. We first examine gradient variance under a correlated-token model, then discuss optimization efficiency across training stages.

To evaluate the optimization dynamics, we model the response-level reward r in GRPO as a Bernoulli variable: $r \sim \mathcal{B}(p)$, where $0 < p < 1$ is the probability for generating a faithful response. Here, $r = 1$ denotes a faithful response (y^+) and $r = 0$ denotes a hallucinated response (y^-). Therefore the mean and std of reward r are p and $\sqrt{p(1-p)}$. For this analysis, we use the population-normalized counterpart of the group-relative advantage in Section 2.2: $A = (r-p)/\sqrt{p(1-p)}$. In contrast, BALTO applies balanced local advantages A_t^{bal} as defined in Eq. 5, satisfying $\sum_{t=1}^{T} A_t^{\text{bal}} = 0$ when $N^- = 0$ or $N^+ > 0$ (Eq. 6).

Let a generated response $\mathbf{y}$ of length T be partitioned into three disjoint sets as defined in Eq. 4: the hallucinated factual token set $\mathcal{H}$ (size N^-), the faithful factual token set $\mathcal{F}$ (size N^+), and the neutral token set $\mathcal{N}$. We define the per-token gradient operator as $g_t = \nabla_\theta \log \pi_\theta(y_t \mid x, \mathbf{y}_{<t})$.

4.1 Enhancing Training Stability via Variance Reduction

To compare GRPO's response-wide credit with BALTO's localized updates, we analyze the raw gradient sums $\hat{\nabla} J_{\text{resp}} = A \sum_t g_t$ and $\hat{\nabla} J_{\text{BALTO}} = \sum_t A_t^{\text{bal}} g_t$ before clipping and length normalization. For fixed length T, condition on the reward and token-label configuration $\mathcal{L}$ and assume: **(A1)** $\text{Cov}(g_i, g_j \mid \mathcal{L}) = \rho_{ij}\Sigma$, where Σ is a common nonzero positive-semidefinite covariance matrix and $P = (\rho_{ij})$ is a correlation matrix with $\rho_{ii} = 1$; and **(A2)** the mean pairwise correlation $\bar{\rho} = \frac{1}{T(T-1)} \sum_{i \neq j} \rho_{ij} \geq 0$ for $T > 1$. These simplifying assumptions allow nonuniform correlations of either sign while requiring their average to be nonnegative.

Theorem 1 (Variance Reduction with Correlated Token Gradients)**.** *Under (A1)–(A2), define the scalar variance $V_m = \mathbb{E}_{\mathcal{L}}[\text{tr}\,\text{Cov}(\hat{\nabla} J_m \mid \mathcal{L})]$ for $m \in \{\text{resp}, \text{BALTO}\}$, and let $\sigma_g^2 = \text{tr}(\Sigma)$. Then $V_{\text{resp}} \geq T\sigma_g^2$ and $V_{\text{BALTO}} \leq 4\mathbb{E}[(N^-)^2]\sigma_g^2$. Consequently, $4\mathbb{E}[(N^-)^2] < T$ implies $V_{\text{BALTO}} < V_{\text{resp}}$, even when token gradients are correlated.*

Proof Sketch. For token weights w_t, (A1) gives conditional scalar variance $S\sigma_g^2$, where $S = \sum_t w_t^2 + \sum_{i \neq j} w_i w_j \rho_{ij}$. Since $|\rho_{ij}| \leq 1$, BALTO's weights -1, N^-/N^+, and 0 yield $S_{\text{BALTO}} \leq (\sum_t |w_t|)^2 = 4(N^-)^2$ when $N^-, N^+ > 0$. The same bound holds in the boundary cases. For GRPO, $S_{\text{resp}} = A^2[T + T(T-1)\bar{\rho}] \geq A^2 T$ by (A2). Taking expectations and using $\mathbb{E}[A^2] = 1$ proves the comparison. Thus the correlated-case bound scales as $\mathcal{O}(\mathbb{E}[(N^-)^2]\sigma_g^2)$, replacing the original independence-based bound. Appendix D.2 provides the complete derivation and relates these raw-sum bounds to the implemented objective. □

4.2 Maximizing Optimization Efficiency across Training Stages

As training progresses, the probability of generating a faithful response, denoted by p, gradually increases. An efficient optimization method should provide strong and reliable gradients throughout this process. However, standard global normalization suffers from gradient starvation during early training (low p) and disproportionate penalties during late-stage convergence (high p). BALTO

resolves both extremes by using bounded token-level advantages, guaranteeing efficient optimization across all stages.

Theorem 2 (High Optimization Efficiency in Asymptotic Regimes)**.** *As $p \to 0$, response-level GRPO suffers from vanishing expected gradients ($\|\mathbb{E}[\hat{\nabla} J_{\text{resp}}]\|_2 \to 0$), severely limiting early-stage learning efficiency. In contrast, BALTO maintains strong local learning signals to ensure efficient exploration. As $p \to 1$, response-level GRPO triggers unbounded divergent penalties ($A^- \to -\infty$), destroying late-stage optimization efficiency. BALTO, however, maintains strictly bounded updates ($A_t^{\text{bal}} \in [-1, 1]$), ensuring efficient and stable final convergence.*

Proof Sketch. For response-Level GRPO, the expected gradient magnitude is bounded by the normalization coefficient $\sqrt{p(1-p)}$. $\lim_{p\to 0}\|\mathbb{E}[\hat{\nabla} J_{\text{resp}}]\|_2 \propto \lim_{p\to 0}\sqrt{p} = 0$, which leads to gradient starvation when faithful responses are rare. As $p \to 1$, the penalty $A^- = -\sqrt{p/(1-p)} \to -\infty$, causing divergent updates. In contrast, BALTO's expected gradient magnitude is proportional to $(1-p)$. As $p \to 0$, $(1-p) \to 1$, so BALTO still provides dense and strong correction signals. As $p \to 1$, $(1-p) \to 0$, naturally decaying the gradient. Meanwhile, local advantages A_t^{bal} remain strictly bounded in $[-1, 1]$ preventing instability. (Detailed derivation in Appendix D.3). □

5 Experiments

5.1 Experimental Settings

Datasets. We evaluate our method on three benchmarks that cover various hallucination scenarios. **ConFiQA** [3] is a counterfactual QA dataset constructed from conflicting evidence, and we focus on the most challenging Multi-Conflicts (MC) subset. **RAGTruth** [30] is a RAG hallucination corpus with diverse data sources and task formats. **FinLLM-Eval** [24] is a financial domain QA dataset containing large amounts of financial facts and numerical information. We evaluate our method on FinLLM-Eval, a high-stakes benchmark that demands precise answers, as the financial domain is particularly sensitive to hallucinations: even minor factual or numerical errors can lead to misleading analyses and consequential decision-making risks.

Baselines. We use Qwen3-4B and Qwen3-8B [46] as base models and compare with three categories of methods discussed in Appendix A. (1) **Data-driven methods**: we construct preference pairs and train models using SFT and DPO. (2) **Response-level RL**: we implement GRPO [34] with two reward designs, including a binary hallucination reward (GRPO$_B$) and a dense reward based on the proportion of correct claims (GRPO$_D$). Most existing response-level hallucination optimization methods can be viewed as variants of GRPO$_D$. (3) **Token-level RL**: we compare with FSPO [23], which performs token-level advantage reshaping on the response level reward. We do not directly compare with RLFH [43], a PPO-based token-level method, due to its discrepancy on the self-assessment setting. Nevertheless, we perform RLFH-style experiments with its core components and conduct an ablation study in Section 5.3.

Evaluation. Following prior work [27, 47], we use an LLM-based evaluation pipeline to extract claims from model responses and verify their faithfulness against reference evidence. We report **faithfulness** (Faith.), which measures the proportion of responses without hallucinations, and **informativeness** (Info.), which evaluates whether the response preserves sufficient information compared to a reference response generated by the base model. We define **Q-Score** = Faith. $\times$ Info. to assert the comprehensive level of faithfulness and informativeness. To further evaluate the response helpfulness, for ConFiQA, we additionally report answer **accuracy** (Acc.). For RAGTruth and FinLLM-Eval, following [22, 7], we conduct pairwise comparisons between RL baselines and our BALTO by reporting the **win rate** (WR). Win Rate uses a separate judge, DeepSeek-v3.2, while ConFiQA Acc. uses rule-based reference-answer matching. Both provide complementary assessments independent of the training verifier, Qwen3.5-35B-A3B. More evaluation details are provided in Appendix E.3.

Implementation Details. We implement our method and the RL baselines using the verl [35] framework. Models are trained on the training split of each dataset and evaluated on the test set. When training, we use a batch size of 256 and a mini-batch size of 64, with 8 rollouts per prompt. The learning rate is set to 1×10^{-6}. Following DAPO [49], we use asymmetric clipping with $\epsilon_{\text{low}} = 0.2$ and $\epsilon_{\text{high}} = 0.28$. More details are in Appendix E.

5.2 Main Results

Table 1: Main results (%) on ConFiQA, RAGTruth, and FinLLM-Eval using Qwen3-8B and Qwen3-4B. **Bold** and underlined scores represent the best and second-best results. *[ref.]* denotes the reference method for relative metrics, with Info. computed against Base and WR computed against BALTO.

Model	Method	ConFiQA				RAGTruth				FinLLM-Eval			
		Faith.	Info.	Q-Score	Acc.	Faith.	Info.	Q-Score	WR	Faith.	Info.	Q-Score	WR
Qwen3-8B	Base	68.2	*[ref.]*	68.2	70.6	40.4	*[ref.]*	40.4	-	34.6	*[ref.]*	34.6	-
	SFT	71.2	68.0	48.4	71.6	56.2	66.7	37.5	-	48.1	70.7	34.0	-
	DPO	72.8	84.0	61.2	62.0	52.4	11.6	6.1	-	41.4	60.2	24.9	-
	$GRPO_B$	92.8	93.5	86.8	73.8	89.8	55.6	49.9	45.1	90.2	39.1	35.3	44.0
	$GRPO_D$	94.5	95.2	90.0	74.4	92.0	73.8	67.9	37.2	91.7	51.1	46.9	37.2
	FSPO	94.8	**97.3**	92.2	66.4	73.1	**84.9**	62.1	34.2	59.4	**82.0**	48.7	46.6
	BALTO	**97.5**	96.2	**93.8**	**80.2**	**98.4**	71.1	**70.0**	*[ref.]*	**92.5**	57.9	**53.6**	*[ref.]*
Qwen3-4B	Base	80.3	*[ref.]*	80.3	36.8	62.2	*[ref.]*	**62.2**	-	57.1	*[ref.]*	**57.1**	-
	$GRPO_B$	97.8	96.3	94.2	36.2	89.8	21.3	19.1	39.8	82.0	23.3	19.1	38.8
	$GRPO_D$	88.0	92.8	81.7	37.8	93.1	50.9	47.4	35.2	78.2	**43.6**	34.1	57.7
	FSPO	95.8	**98.5**	94.4	36.3	87.3	**57.3**	50.0	22.7	77.4	42.9	33.2	42.5
	BALTO	**99.0**	96.2	**95.2**	**38.6**	**94.0**	56.7	53.3	*[ref.]*	**88.7**	42.1	37.3	*[ref.]*

Table 1 reports the main results on ConFiQA, RAGTruth, and FinLLM-Eval. Overall, BALTO achieves the highest faithfulness across all datasets and both model scales, and obtains the best Q-Score among all post-training methods. Compared with the strongest post-training baseline in each setting, BALTO improves faithfulness by 3.1 points on average and up to 6.7 points, while improving Q-Score by 2.7 points on average and up to 4.9 points. These gains are particularly pronounced over response-level RL methods ($GRPO_B$ and $GRPO_D$), suggesting that balanced token-level credit assignment provides more precise optimization signals for localized hallucinations.

Although FSPO achieves higher informativeness on some datasets, it often does so at the cost of faithfulness, leading to a weaker overall faithfulness–informativeness trade-off. In contrast, BALTO consistently improves faithfulness while maintaining competitive informativeness, indicating that it suppresses unsupported content without simply shortening or over-conservatizing responses. Moreover, BALTO attains the highest accuracy on ConFiQA for both model scales, improving over the strongest post-training baseline by up to 5.8 points, and serves as the reference in most pairwise response comparisons on RAGTruth and FinLLM-Eval. These results show that BALTO preserves overall response quality while achieving stronger hallucination mitigation.

Additional experiments with Llama-3.1-8B [13] on RAGTruth and FinLLM-Eval also show the highest faithfulness and Q-Score among the evaluated methods, supporting generalization beyond the Qwen model family (Appendix F.1).

5.3 Ablation Study

Table 2: Ablation study on positive advantage assignment. "PA = 0" removes the positive advantage for faithful tokens, and "PA = 0.3" / "PA = 1" assign fixed positive advantage values.

Method	ConFiQA			RAGTruth			FinLLM-Eval		
	Faith.	Info.	Q-Score	Faith.	Info.	Q-Score	Faith.	Info.	Q-Score
BALTO	**97.5**	**96.2**	**93.8**	**98.4**	71.1	**70.0**	**92.5**	57.9	**53.6**
PA = 0	91.3	92.2	84.2	86.0	64.0	55.0	78.2	36.1	28.2
PA = 0.3	84.0	93.2	78.3	56.0	81.3	45.5	40.6	62.4	25.3
PA = 1	76.3	90.2	68.8	72.2	**89.8**	64.8	49.6	**77.4**	38.4

Balanced compensation preserves response quality. Table 2 examines positive advantage assignment. Removing positive compensation (PA = 0) lowers both faithfulness and informativeness across all three datasets. Fixed positive advantages (PA = 0.3 or 1) also yield lower faithfulness and Q-Score than BALTO, despite higher informativeness on RAGTruth and FinLLM-Eval. These fixed-PA settings isolate RLFH's fixed-reward design within our GRPO framework [43]. The balanced default achieves the highest faithfulness and Q-Score among these variants.

Table 3: Design and label-noise ablations (%) with Qwen3-8B on FinLLM-Eval.

Setting	Faith.	Info.	Q-Score
BALTO	**92.5**	**57.9**	**53.6**
Full claim penalty	82.4	46.8	38.6
Positive compensation $\times 0.5$	87.5	43.4	38.0
Positive compensation $\times 2.0$	90.2	54.1	48.8
Miss 20% hallucinated tokens	87.5	55.7	48.7
Mislabel 20% faithful tokens	76.5	43.4	33.2

Table 4: Human validation of the evaluator (%). Token localization is measured on correctly identified hallucinated claims.

Level	Precision	Recall
Claim extraction	98.4	88.8
Hallucination judgment	81.3	92.4
Error token localization	94.6	89.8

The balanced ratio gives the best trade-off. Table 3 shows that scaling the positive compensation N^-/N^+ by 0.5 or 2.0 also underperforms the balanced default (Q-Score: 38.0 and 48.8, respectively). Under-compensation reduces informativeness from 57.9 to 43.4, while over-compensation lowers faithfulness from 92.5 to 90.2.

Penalizing entire claims reduces response quality. Table 3 examines error localization and compensation strength with Qwen3-8B on FinLLM-Eval. Extending negative advantages from minimal erroneous spans to entire hallucinated claims reduces Q-Score from 53.6 to 38.6 and degrades both faithfulness and informativeness, supporting precise token localization.

Mislabeling faithful tokens causes larger losses. For the noise experiments in Table 3, we randomly drop 20% of hallucinated-token labels so they escape the negative penalty, or relabel 20% of faithful tokens as hallucinated, assigning them negative advantages and removing them from the positive compensation set. Q-Score falls to 48.7 and 33.2, respectively, corresponding to relative decreases of 9% and 38%. Under these perturbations, faithful-token mislabeling is more damaging than missed hallucinations, highlighting the importance of accurate negative labels.

5.4 Further Analysis

Evaluator Reliability. We conducted a 90-sample independent human annotation study (30 per dataset: ConFiQA, RAGTruth, and FinLLM-Eval) to evaluate three levels of the detection pipeline (Table 4). The evaluator achieves high claim extraction quality and strong hallucination detection, with hallucination judgment recall reaching 92.4%. Claim extraction and hallucination judgment use the same evaluator, Qwen3.5-35B-A3B, across GRPO$_B$, GRPO$_D$, FSPO, and BALTO, controlling the verifier model when comparing credit assignment mechanisms. For token localization, the evaluator uses its claim-verification reasoning to identify specific erroneous tokens, achieving 94.6% precision and 89.8% recall. These results support the reliability of BALTO's localized token-level feedback.

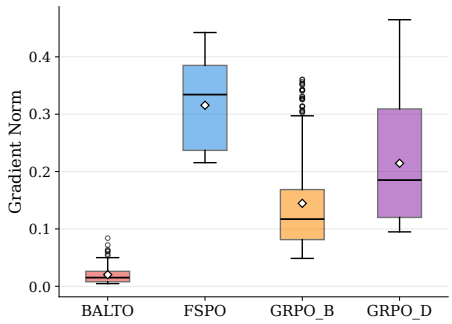


Figure 3: Gradient norm distribution on RAGTruth.

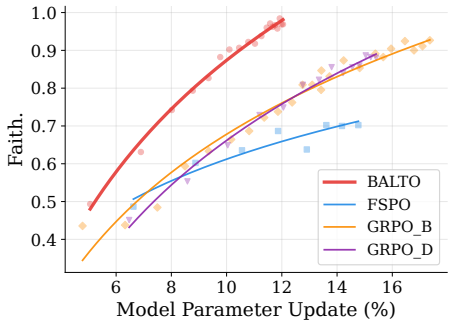


Figure 4: Faithfulness vs. model parameter update on RAGTruth. Curves are fitted using logarithmic functions $y = a \cdot \ln(x+1) + b$.

Training Dynamics. Figure 3 shows the distribution of policy gradient norms across training steps on the RAGTruth dataset. Compared with response-level RL methods, BALTO exhibits gradient norms that are an order of magnitude smaller (mean 0.020 vs. 0.145–0.214) and more tightly concentrated. This behavior reflects the sparse and balanced token-level advantage structure in BALTO: only hallucinated and corresponding faithful tokens receive non-zero advantages, while all other neutral

tokens are ignored. By restricting gradient updates to the most relevant tokens, BALTO reduces the influence of noisy signals from neutral tokens, resulting in lower variance and more stable optimization trajectories, as theoretically predicted in Theorem 1.

Figure 4 plots faithfulness against the fraction of model parameters whose stored values differ from initialization [28]. All parameters remain trainable during RL. BALTO achieves 0.98 faithfulness with only approximately 12% of parameters updated, whereas GRPO$_B$ requires around 17% of parameters updated to reach 0.93 faithfulness. This indicates that token-level signals guide the optimizer to focus on a smaller and more relevant subset of parameters, specifically those that are directly responsible for generating hallucinated content, while maintaining the model's broader capabilities acquired during pre-training. The practical implication is that BALTO can reduce hallucinations more effectively while minimizing the risk of degrading general language performance.

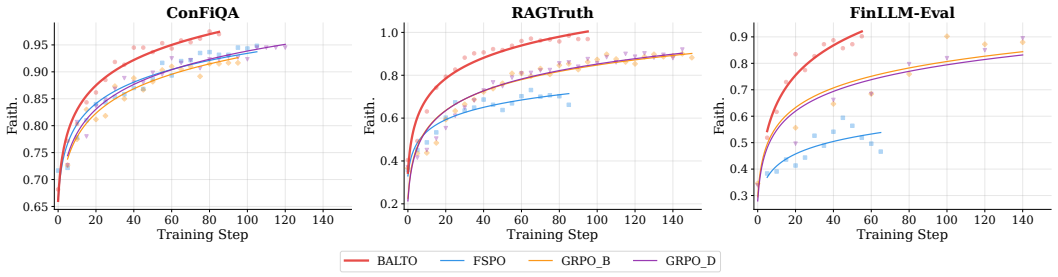


Figure 5: Faithfulness improvement during training across different RL methods on three datasets. Curves are fitted using logarithmic functions $y = a \cdot \ln(x + 1) + b$.

Efficiency. Figure 5 shows the improvement of faithfulness across training for different RL methods. BALTO consistently achieves faster early-stage gains and reaches higher final faithfulness with fewer training steps, reflecting the more informative token-level learning signals provided by balanced credit assignment. These observations align with Theorem 2, which demonstrates that BALTO maintains strong, bounded gradients across all stages of training, ensuring efficient exploration and stable convergence. In contrast, response-level methods such as GRPO$_B$ and GRPO$_D$ converge more slowly and are less effective on datasets with severe hallucinations, highlighting the advantage of fine-grained, balanced credit assignment for both performance and optimization efficiency.

6 Conclusion

In this work, we introduced BALTO, a balanced token-level policy optimization framework for fine-grained hallucination mitigation. Motivated by the localized nature of faithfulness hallucinations, BALTO addresses the credit assignment mismatch of response-level rewards by converting factuality judgments into token-level signals. By balancing positive and negative token-level credit assignment within each response, BALTO encourages probability mass to move from hallucinated content toward faithful content with balanced policy optimization.

Our theoretical analysis shows that balanced token-level credit assignment reduces unnecessary updates, lowers gradient variance, and provides bounded, stable optimization signals. Empirically, BALTO consistently improves the faithfulness–informativeness trade-off across ConFiQA, RAGTruth, and FinLLM-Eval with Qwen3-8B and Qwen3-4B, achieving stronger faithfulness and Q-Score than response-level and token-level RL baselines. These results highlight balanced fine-grained credit assignment as an effective direction for building more faithful and informative LLMs.

Acknowledgments and Disclosure of Funding

The SJTU team is supported by National Natural Science Foundation of China (62322603, 62502310) and Shanghai Municipal Science and Technology Major Project (2025SHZDZX025D08).

References

[1] Aisha Alansari and Hamzah Luqman. Large language models hallucination: A comprehensive survey. *Computer Science Review*, 61:100970, 2026.

[2] Patrice Béchard and Orlando Marquez Ayala. Reducing hallucination in structured outputs via retrieval-augmented generation. *arXiv preprint arXiv:2404.08189*, 2024.

[3] Baolong Bi, Shaohan Huang, Yiwei Wang, Tianchi Yang, Zihan Zhang, Haizhen Huang, Lingrui Mei, Junfeng Fang, Zehao Li, Furu Wei, et al. Context-dpo: Aligning language models for context-faithfulness. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 10280–10300, 2025.

[4] Alex J Chan, Hao Sun, Samuel Holt, and Mihaela Van Der Schaar. Dense reward for free in reinforcement learning from human feedback. *arXiv preprint arXiv:2402.00782*, 2024.

[5] Hongzhan Chen, Tao Yang, Shiping Gao, Ruijun Chen, Xiaojun Quan, Hongtao Tian, and Ting Yao. Discriminative policy optimization for token-level reward models. *arXiv preprint arXiv:2505.23363*, 2025.

[6] M Chen, L Sun, T Li, H Sun, Y Zhou, C Zhu, H Wang, JZ Pan, W Zhang, H Chen, et al. Research: Learning to reason with search for llms via reinforcement learning. arxiv 2025. *arXiv preprint arXiv:2503.19470*.

[7] Xilun Chen, Ilia Kulikov, Vincent-Pierre Berges, Barlas Oğuz, Rulin Shao, Gargi Ghosh, Jason Weston, and Wen-tau Yih. Learning to reason for factuality. *arXiv preprint arXiv:2508.05618*, 2025.

[8] Jie Cheng, Gang Xiong, Ruixi Qiao, Lijun Li, Chao Guo, Junle Wang, Yisheng Lv, and Fei-Yue Wang. Stop summation: Min-form credit assignment is all process reward model needs for reasoning. *arXiv preprint arXiv:2504.15275*, 2025.

[9] YS Chuang et al. Decoding by contrasting layers improves factuality in large language models (dola). *arXiv Preprint*, pages 1–8, 2024.

[10] Matthew Dahl, Varun Magesh, Mirac Suzgun, and Daniel E Ho. Large legal fictions: Profiling legal hallucinations in large language models. *Journal of Legal Analysis*, 16(1):64–93, 2024.

[11] Hanxing Ding, Liang Pang, Zihao Wei, Huawei Shen, and Xueqi Cheng. Retrieve only when it needs: Adaptive retrieval augmentation for hallucination mitigation in large language models. *arXiv e-prints*, pages arXiv–2402, 2024.

[12] Chengfeng Dou, Fan Yang, Fei Li, Jiyuan Jia, Qiang Ju, Shuai Wang, Tianpeng Li, Xiangrong Zeng, Yijie Zhou, Hongda Zhang, et al. Baichuan-m3: Modeling clinical inquiry for reliable medical decision-making. *arXiv preprint arXiv:2602.06570*, 2026.

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

[14] Tao He, Rongchuan Mu, Lizi Liao, Yixin Cao, Ming Liu, and Bing Qin. Good learners think their thinking: Generative prm makes large reasoning model more efficient math learner. *arXiv preprint arXiv:2507.23317*, 2025.

[15] Minda Hu, Bowei He, Yufei Wang, Liangyou Li, Chen Ma, and Irwin King. Mitigating large language model hallucination with faithful finetuning. *arXiv preprint arXiv:2406.11267*, 2024.

[16] 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.

[17] Kai Jiang, Qilai Zhang, Dongsheng Guo, Dengrong Huang, Sijia Zhang, Zizhong Wei, Fanggang Ning, and Rui Li. Ai-generated news articles based on large language models. In *Proceedings of the 2023 International Conference on Artificial Intelligence, Systems and Network Security*, pages 82–87, 2023.

[18] Haoqiang Kang and Xiao-Yang Liu. Deficiency of large language models in finance: An empirical examination of hallucination. *arXiv preprint arXiv:2311.15548*, 2023.

[19] Yannis Katsis, Sara Rosenthal, Kshitij Fadnis, Chulaka Gunasekara, Young-Suk Lee, Lucian Popa, Vraj Shah, Huaiyu Zhu, Danish Contractor, and Marina Danilevsky. MTRAG: A multi-turn conversational benchmark for evaluating retrieval-augmented generation systems. *Transactions of the Association for Computational Linguistics*, 13:784–808, 2025. doi: 10.1162/tacl.a.19.

[20] Yubin Kim, Hyewon Jeong, Shan Chen, Shuyue Stella Li, Chanwoo Park, Mingyu Lu, Kumail Alhamoud, Jimin Mun, Cristina Grau, Minseok Jung, et al. Medical hallucinations in foundation models and their impact on healthcare. *arXiv preprint arXiv:2503.05777*, 2025.

[21] Jiaqi Li, Yixuan Tang, and Yi Yang. Know the unknown: An uncertainty-sensitive method for llm instruction tuning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 2972–2989, 2025.

[22] Junliang Li, Yucheng Wang, Yan Chen, Yu Ran, Ruiqing Zhang, Jing Liu, Hua Wu, and Haifeng Wang. Knowledge-level consistency reinforcement learning: Dual-fact alignment for long-form factuality, 2026. URL `https://openreview.net/forum?id=Q04RwdeN9z`.

[23] Junyi Li and Hwee Tou Ng. Reasoning models hallucinate more: Factuality-aware reinforcement learning for large reasoning models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026. URL `https://openreview.net/forum?id=Igq7Dyc3OL`.

[24] Lichang Liang, Shaohui Wu, Yunlong Zhang, Yeyang Tang, and Fanyang Lu. finllm-eval: Evaluation methods for the logical, factual, and data accuracy of large models in the financial domain. `https://www.github.com/Tencent/finLLM-Eval`, 2025.

[25] Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The twelfth international conference on learning representations*, 2023.

[26] Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, et al. Deepseek-v3. 2: Pushing the frontier of open large language models. *arXiv preprint arXiv:2512.02556*, 2025.

[27] Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100, 2023.

[28] Sagnik Mukherjee, Lifan Yuan, Dilek Hakkani-Tur, and Hao Peng. Reinforcement learning finetunes small subnetworks in large language models. *arXiv preprint arXiv:2505.11711*, 2025.

[29] Feng Nan, Ramesh Nallapati, Zhiguo Wang, Cicero Nogueira dos Santos, Henghui Zhu, Dejiao Zhang, Kathleen McKeown, and Bing Xiang. Entity-level factual consistency of abstractive text summarization. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2727–2733, 2021.

[30] 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)*, pages 10862–10878, 2024.

[31] Artidoro Pagnoni, Vidhisha Balachandran, and Yulia Tsvetkov. Understanding factuality in abstractive summarization with FRANK: A benchmark for factuality metrics. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4812–4829, 2021.

[32] Baochang Ren, Shuofei Qiao, Da Zheng, Huajun Chen, and Ningyu Zhang. Knowrl: Exploring knowledgeable reinforcement learning for factuality. *arXiv preprint arXiv:2506.19807*, 2025.

[33] Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. A systematic survey of prompt engineering in large language models: Techniques and applications. *arXiv preprint arXiv:2402.07927*, 1, 2024.

[34] 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.

[35] 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.

[36] Huatong Song, Jinhao Jiang, Yingqian Min, Jie Chen, Zhipeng Chen, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. R1-searcher: Incentivizing the search capability in llms via reinforcement learning. *arXiv preprint arXiv:2503.05592*, 2025.

[37] Wei Sun, Wen Yang, Pu Jian, Qianlong Du, Fuwei Cui, Shuo Ren, and Jiajun Zhang. Ktae: A model-free algorithm to key-tokens advantage estimation in mathematical reasoning. *arXiv preprint arXiv:2505.16826*, 2025.

[38] Zhongxiang Sun, Qipeng Wang, Haoyu Wang, Xiao Zhang, and Jun Xu. Mechanistic detection and mitigation of hallucination in large reasoning models. In *The Fourteenth International Conference on Learning Representations*.

[39] Manveer Singh Tamber, Forrest Bao, Chenyu Xu, Ge Luo, Suleman Kazi, Minseok Bae, Miaoran Li, Ofer Mendelevitch, Renyi Qu, and Jimmy Lin. Benchmarking llm faithfulness in rag with evolving leaderboards. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 799–811, 2025.

[40] Hongze Tan, Zihan Wang, Jianfei Pan, Jinghao Lin, Hao Wang, Yifan Wu, Tao Chen, Zhihang Zheng, Zhihao Tang, and Haihua Yang. Gtpo and grpo-s: Token and sequence-level reward shaping with policy entropy. *arXiv preprint arXiv:2508.04349*, 2025.

[41] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

[42] Jerry Wei, Chengrun Yang, Xinying Song, Yifeng Lu, Nathan Hu, Jie Huang, Dustin Tran, Daiyi Peng, Ruibo Liu, Da Huang, Cosmo Du, and Quoc V. Le. Long-form factuality in large language models. In *Advances in Neural Information Processing Systems*, volume 37, pages 80756–80827, 2024.

[43] Xueru Wen, Jie Lou, Xinyu Lu, Yuqiu Ji, Xinyan Guan, Yaojie Lu, Hongyu Lin, Ben He, Xianpei Han, Debing Zhang, et al. On-policy self-alignment with fine-grained knowledge feedback for hallucination mitigation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 5215–5231, 2025.

[44] Guofu Xie, Yunsheng Shi, Hongtao Tian, Ting Yao, and Xiao Zhang. Capo: Towards enhancing llm reasoning through generative credit assignment. *arXiv preprint arXiv:2508.02298*, 2025.

[45] Guangzhi Xiong, Eric Xie, Corey Williams, Myles Kim, Amir Hassan Shariatmadari, Sikun Guo, Stefan Bekiranov, and Aidong Zhang. Toward reliable scientific hypothesis generation: Evaluating truthfulness and hallucination in large language models. *arXiv preprint arXiv:2505.14599*, 2025.

[46] 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.

[47] Taoye Yin, Haoyuan Hu, Yaxin Fan, Xinhao Chen, Xinya Wu, Kai Deng, Kezun Zhang, and Feng Wang. Mitigating hallucination in financial retrieval-augmented generation via fine-grained knowledge verification. *arXiv preprint arXiv:2602.05723*, 2026.

[48] Eunseop Yoon, Hee Suk Yoon, SooHwan Eom, Gunsoo Han, Daniel Nam, Daejin Jo, Kyoung-Woon On, Mark Hasegawa-Johnson, Sungwoong Kim, and Chang Yoo. Tlcr: Token-level continuous reward for fine-grained reinforcement learning from human feedback. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14969–14981, 2024.

[49] 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. *arXiv preprint arXiv:2503.14476*, 2025.

[50] Kangning Zhang, Haotian Fang, Xukun Luo, Hao Yin, Yang Gao, Peng Yan, Weiwen Liu, Weinan Zhang, and Yong Yu. Learning from Unreachable Rewards: Hint-Conditioned Reinforcement Learning for Generative Recommendation. In *Proceedings of the 35th ACM International Conference on Information and Knowledge Management*. Association for Computing Machinery, 2026. doi: 10.1145/3799682.3840842. URL `https://arxiv.org/abs/2608.11980`.

[51] Kangning Zhang, Weiwen Liu, Wenxiang Jiao, Kounianhua Du, Yuan Lu, Weinan Zhang, and Yong Yu. LoopTool: Closing the Data–Training Loop for Robust LLM Tool Calls. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 21144–21164, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.968. URL `https://aclanthology.org/2026.acl-long.968/`.

[52] Kechi Zhang, Jia Li, Ge Li, Xianjie Shi, and Zhi Jin. Codeagent: Enhancing code generation with tool-integrated agent systems for real-world repo-level coding challenges. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13643–13658, 2024.

[53] Yuxin Zhang, Meihao Fan, Ju Fan, Mingyang Yi, Yuyu Luo, Jian Tan, and Guoliang Li. Reward-sql: Boosting text-to-sql via stepwise reasoning and process-supervised rewards. *arXiv preprint arXiv:2505.04671*, 2025.

[54] Xufeng Zhao, Mengdi Li, Wenhao Lu, Cornelius Weber, Jae-Hee Lee, Kun Chu, and Stefan Wermter. Enhancing zero-shot chain-of-thought reasoning in large language models through logic. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 6144–6166, 2024.

[55] Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36:55006–55021, 2023.

A Related Work

A.1 Hallucination

In LLMs, hallucinations refer to syntactically fluent outputs that contain factual errors or contradict source materials [16, 1]. Prior mitigation strategies are primarily data-centric or model-centric. Data-centric approaches optimize curation quality [55], employ specialized prompting [17, 33], or utilize RAG [11, 2] and reasoning techniques like Chain-of-Thought or self-consistency[41, 54]. Conversely, model-centric methods refine internal mechanisms through faithful fine-tuning [15, 21] and contrastive decoding [9]. More recently, RL has emerged as a significant direction [43, 23], which we detail below.

A.2 RL for Faithfulness Alignment

While RL is critical for model alignment, prior approaches often employ coarse-grained rewards [6, 36], which may suffer from sparsity and imprecise guidance [25]. This has driven a shift towards fine-grained faithfulness alignment, where models are evaluated using claim-level correctness [7, 12] or atomic facts [32, 47]. However, these methods typically lack token-level precision or are constrained to specific domains. Furthermore, existing approaches that claim to utilize token-level rewards [38, 23, 43] either still depend on response-level signals or lack an effective credit balancing mechanism.

A.3 Fine-Grained Rewards and Process Supervision

To address the sparse reward issue inherent in traditional outcome-based models [25, 50], recent literature has shifted towards fine-grained process supervision, primarily integrating intermediate signals into PPO or GRPO frameworks. Within PPO, approaches assign credit at the step-level to reinforce error localization [8], or at the token-level utilizing mechanisms such as attention distributions, edit distances, or discriminative optimization [4, 48, 5]. Parallelly, GRPO-based methods explore diverse advantage estimation techniques, ranging from stepwise dynamic scaling and generative credit assignment [53, 44] to token-level advantage computation via decaying accumulation, entropy-based shaping, or statistical measures [14, 40, 37]. Despite these significant algorithmic advancements, genuinely integrating token-level rewards specifically for hallucination mitigation remains underexplored [43], a critical gap that our work directly addresses.

B Limitations

Despite its effectiveness, BALTO has several limitations. First, our method relies on an LLM-based faithfulness evaluator to extract and verify factual claims, and thus its performance is bounded by the accuracy and consistency of this evaluator. Errors in claim extraction, evidence matching, or token-level projection may introduce noisy advantage assignments, especially for implicit claims, long-range reasoning dependencies, or ambiguous evidence. Second, BALTO is primarily designed for retrieval-grounded faithfulness alignment, where reference documents are available for verification. Its applicability to open-domain generation without explicit evidence, or to tasks involving subjective, creative, or preference-based outputs, remains less explored.

C Broader Impact

A key positive impact of BALTO is its ability to provide more precise supervision for hallucination mitigation. This may improve the usefulness of retrieval-augmented systems, financial assistants, medical question-answering systems, and other applications where factual precision is critical. In addition, BALTO's variance-reduction properties and efficient optimization behavior may reduce training instability and computational waste during RL-based alignment.

At the same time, our method also introduces risks. BALTO relies on an external faithfulness evaluator to identify hallucinated and faithful tokens. If the evaluator itself contains biases, factual inaccuracies, or systematic verification errors, these imperfections may propagate into the optimization process and reinforce incorrect supervision signals.

D Complete Mathematical Proofs for Theoretical Analysis

In this appendix, we provide the complete mathematical derivations for the theorems presented in Section 4. We first establish the rigorous foundations of the policy gradient estimators used in our analysis.

D.1 Preliminaries and Definition of Policy Gradient Estimators

Let $y = (y_1, y_2, \ldots, y_T)$ be a response of length T generated by a language model policy $\pi_\theta(\cdot|x)$ given prompt x. The trajectory generation probability is given by $\pi_\theta(y \mid x) = \prod_{t=1}^{T} \pi_\theta(y_t \mid x, y_{<t})$.

We define the per-token score function (the gradient of the log-probability) as:

$$g_t = \nabla_\theta \log \pi_\theta(y_t \mid x, y_{<t}) \tag{9}$$

According to the log-derivative trick, the expectation of the score function over the action space is strictly zero: $\mathbb{E}_{y_t \sim \pi_\theta(\cdot|x, y_{<t})}[g_t \mid x, y_{<t}] = 0$.

The response-level score function, representing the gradient of the entire response, is the sum of the per-token score functions:

$$G(y) = \nabla_\theta \log \pi_\theta(y \mid x) = \sum_{t=1}^{T} \nabla_\theta \log \pi_\theta(y_t \mid x, y_{<t}) = \sum_{t=1}^{T} g_t \tag{10}$$

Bernoulli Reward Setting. To evaluate the optimization dynamics, we model the generation process as yielding two distinct types of trajectories:

- **Faithful response (y^+):** Generated with probability p, yielding a reward $r = 1$. It contains no hallucinations. We denote its response-level score function as $G(y^+) = \sum_{t=1}^{T^+} g_t^+$.
- **Hallucinated response (y^-):** Generated with probability $1 - p$, yielding a reward $r = 0$. It contains at least one hallucination. We denote its response-level score function as $G(y^-) = \sum_{t=1}^{T^-} g_t^-$.

Following our methodology, the token indices of any response are partitioned into three disjoint sets: the hallucinated factual token set $\mathcal{H}$ (size N^-), the faithful factual token set $\mathcal{F}$ (size N^+), and the neutral token set $\mathcal{N}$. For y^+, $N^- = 0$. For y^-, $N^- \geq 1$. Expectations $\mathbb{E}_{y^+}$ and $\mathbb{E}_{y^-}$ below are conditional on $r = 1$ and $r = 0$, respectively.

Derivation of the Response-Level GRPO Estimator. The standard RL objective maximizes the expected reward: $J_{\text{resp}}(\theta) = \mathbb{E}_{y \sim \pi_\theta}[R(y)]$. Using the score identity and the population-normalized advantage, the reward gradient satisfies:

$$\nabla_\theta J_{\text{resp}}(\theta) = \mathbb{E}_{y \sim \pi_\theta}[(r - p)G(y)] = \sigma\, \mathbb{E}_{y \sim \pi_\theta}[AG(y)] \tag{11}$$

where $A = \frac{r-\mu}{\sigma}$ is the normalized global advantage. Given $\mu = p$ and $\sigma = \sqrt{p(1-p)}$ for $0 < p < 1$, the respective advantages for y^+ and y^- are:

$$A^+ = \sqrt{\frac{1-p}{p}}, \quad A^- = -\sqrt{\frac{p}{1-p}} \tag{12}$$

We denote the raw normalized gradient signal for a single response by $\hat{\nabla} J_{\text{resp}}$. Its expectation is $\nabla_\theta J_{\text{resp}}/\sigma$, not the unnormalized reward gradient:

$$\hat{\nabla} J_{\text{resp}} = A \sum_{t=1}^{T} g_t \tag{13}$$

Derivation of the BALTO Estimator. BALTO assigns advantages at the token level. For the raw, unclipped surrogate with advantages held fixed, the expected gradient signal is:

$$\mathbb{E}[\hat{\nabla} J_{\text{BALTO}}] = \mathbb{E}_{y \sim \pi_\theta}\left[\sum_{t=1}^{T} A_t^{\text{bal}} \nabla_\theta \log \pi_\theta(y_t \mid x, y_{<t})\right] = \mathbb{E}_{y \sim \pi_\theta}\left[\sum_{t=1}^{T} A_t^{\text{bal}} g_t\right] \tag{14}$$

where the token-level advantages A_t^{bal} strictly follow the balanced assignment in Eq. 5. The corresponding raw gradient signal for a single response is:

$$\hat{\nabla} J_{\text{BALTO}} = \sum_{t=1}^{T} A_t^{\text{bal}} g_t \tag{15}$$

Here, $\hat{\nabla} J_m$ denotes a raw gradient signal rather than the gradient of the length-normalized, clipped objective in Section 2.2 or Eq. 7. The analysis is on-policy. Token weights and normalization statistics are held fixed during each surrogate update.

D.2 Proof of Theorem 1: Variance Reduction with Correlated Token Gradients

We analyze the raw gradient sums defined above, before clipping and length normalization, using the correlated-gradient assumptions (A1)–(A2) stated in Section 4.1. Let $\mathcal{L}$ fix the response reward and token-label configuration for a response of length T. Advantage weights are deterministic within each such configuration, while token gradients may remain correlated.

Conditional covariance decomposition. Write $\xi_t = g_t - \mathbb{E}[g_t \mid \mathcal{L}]$. Assumption (A1) states that

$$\mathbb{E}[\xi_i \xi_j^\top \mid \mathcal{L}] = \rho_{ij}(\mathcal{L})\Sigma, \qquad \rho_{ii} = 1, \tag{16}$$

where Σ is a common nonzero positive-semidefinite covariance matrix and $P = (\rho_{ij})$ is a correlation matrix. In particular, $|\rho_{ij}| \leq 1$. No independence, isotropy, or uniform pairwise correlation is required. Centering is conditional: the unconditional score identity $\mathbb{E}[g_t] = 0$ does not imply $\mathbb{E}[g_t \mid \mathcal{L}] = 0$.

For arbitrary fixed weights w_t, the conditional covariance decomposes into diagonal and cross-correlation terms:

$$\text{Cov}\left(\sum_{t=1}^{T} w_t g_t \mid \mathcal{L}\right) = S(w, \mathcal{L})\Sigma, \tag{17}$$

$$S(w, \mathcal{L}) = \underbrace{\sum_{t} w_t^2}_{D(w)} + \underbrace{\sum_{i \neq j} w_i w_j \rho_{ij}}_{C(w,\mathcal{L})}. \tag{18}$$

Taking the trace yields the scalar conditional variance

$$\begin{aligned} v_m(\mathcal{L}) &= \text{tr}\,\text{Cov}(\hat{\nabla} J_m \mid \mathcal{L}) \\ &= \mathbb{E}\left[\left\|\hat{\nabla} J_m - \mathbb{E}[\hat{\nabla} J_m \mid \mathcal{L}]\right\|_2^2 \mid \mathcal{L}\right] = S_m \sigma_g^2. \end{aligned} \tag{19}$$

where $\sigma_g^2 = \text{tr}(\Sigma) > 0$ and $S_m = S(w^{(m)}, \mathcal{L})$ for the weights of method m. We compare $V_m = \mathbb{E}_{\mathcal{L}}[v_m(\mathcal{L})]$ for $m \in \{\text{resp}, \text{BALTO}\}$. This conditioning keeps the reward-dependent weights inside their appropriate covariance calculation.

BALTO upper bound. BALTO assigns $w_t = -1$ to hallucinated tokens, $w_t = N^-/N^+$ to faithful factual tokens when $N^-, N^+ > 0$, and zero to neutral tokens. For notational convenience, write $h = N^-$ and $f = N^+$. The diagonal term is

$$D_{\text{BALTO}} = h + f\left(\frac{h}{f}\right)^2 = h + \frac{h^2}{f}. \tag{20}$$

Using $|\rho_{ij}| \leq 1$, we bound the cross term by grouping ordered pairs into hallucinated–hallucinated, faithful–faithful, and cross-set pairs:

$$\begin{aligned} |C_{\text{BALTO}}| &\leq \underbrace{h(h-1)}_{\mathcal{H}\text{–}\mathcal{H}} + \underbrace{f(f-1)\left(\frac{h}{f}\right)^2}_{\mathcal{F}\text{–}\mathcal{F}} + \underbrace{2hf\frac{h}{f}}_{\mathcal{H}\text{–}\mathcal{F}} \\ &= 4h^2 - h - \frac{h^2}{f}. \end{aligned} \tag{21}$$

Adding the diagonal term gives

$$0 \le S_{\mathrm{BALTO}} = D_{\mathrm{BALTO}} + C_{\mathrm{BALTO}} \le 4h^2. \tag{22}$$

Equivalently, this follows from $S_{\mathrm{BALTO}} \le (\sum_t |w_t|)^2 = (2h)^2$. The bound holds regardless of the signs or uniformity of individual correlations. When $h = 0$, all weights vanish. When $h > 0, f = 0$, Eq. 5 assigns only negative credit, yielding $S_{\mathrm{BALTO}} \le h^2 \le 4h^2$. Hence

$$V_{\mathrm{BALTO}} \le 4\mathbb{E}[h^2]\sigma_g^2 = 4(1-p)\mathbb{E}[h^2 \mid r = 0]\sigma_g^2. \tag{23}$$

GRPO lower bound. GRPO applies the uniform weight A to all T tokens. By (A2),

$$S_{\mathrm{resp}} = A^2 \left[T + \sum_{i \ne j} \rho_{ij} \right] = A^2[T + T(T-1)\bar{\rho}] \ge A^2 T. \tag{24}$$

For $T = 1$, the cross term is absent and the same lower bound holds. Under the Bernoulli reward model,

$$\mathbb{E}[A^2] = p\frac{1-p}{p} + (1-p)\frac{p}{1-p} = 1. \tag{25}$$

Taking expectations therefore gives

$$V_{\mathrm{resp}} \ge T\sigma_g^2. \tag{26}$$

Comparison and scope. Combining the two bounds,

$$\frac{V_{\mathrm{BALTO}}}{V_{\mathrm{resp}}} \le \frac{4\mathbb{E}[(N^-)^2]}{T}. \tag{27}$$

Thus, $4\mathbb{E}[(N^-)^2] < T$ implies $V_{\mathrm{BALTO}} < V_{\mathrm{resp}}$. For fixed hallucination counts, this becomes $N^- < \sqrt{T}/2$. Compared with the original independence-based analysis, the correlated case has the more conservative quadratic dependence on N^-. Sparsity expressed only as a small ratio N^-/T does not by itself guarantee this condition.

The comparison concerns fluctuations within reward/label configurations. By the law of total variance,

$$\operatorname{tr}\operatorname{Cov}(\hat{\nabla} J_m) = V_m + \operatorname{tr}\operatorname{Cov}_{\mathcal{L}}\left(\mathbb{E}[\hat{\nabla} J_m \mid \mathcal{L}]\right). \tag{28}$$

Without an additional bound on the second term, the theorem does not order the total scalar gradient variance. For the length-normalized, unclipped objectives, the corresponding scalar conditional variances are $S_{\mathrm{resp}}\sigma_g^2/T^2$ and $S_{\mathrm{BALTO}}\sigma_g^2/Z^2$, where $Z = \max(1, N^- + N^+)$. These normalization factors and clipping must be considered separately when relating the raw-sum comparison to training. The result provides formal motivation under (A1)–(A2), rather than a tight guarantee for the implemented optimizer. ■

D.3 Proof of Theorem 2: High Optimization Efficiency in Asymptotic Regimes

Optimization efficiency relies on the expected magnitude of the policy gradient providing a strong, informative signal. We analyze this expectation at the two extremes of alignment training: $p \to 0$ (cold-start) and $p \to 1$ (late-stage convergence).

Part I: Gradient Starvation in the Cold-Start Regime ($p \to 0$)

For Response-Level GRPO, we expand the expected gradient over the Bernoulli outcomes using the predefined response-level score functions $G(y^+)$ and $G(y^-)$:

$$\begin{aligned}
\mathbb{E}[\hat{\nabla} J_{\mathrm{resp}}] &= p \cdot \mathbb{E}_{y^+}\left[A^+ G(y^+)\right] + (1-p) \cdot \mathbb{E}_{y^-}\left[A^- G(y^-)\right] \\
&= p\sqrt{\frac{1-p}{p}}\mathbb{E}_{y^+}[G(y^+)] - (1-p)\sqrt{\frac{p}{1-p}}\mathbb{E}_{y^-}[G(y^-)] \\
&= \sqrt{p(1-p)}\Big(\mathbb{E}_{y^+}[G(y^+)] - \mathbb{E}_{y^-}[G(y^-)]\Big)
\end{aligned} \tag{29}$$

The overall magnitude of the learning signal is constrained by the global normalization coefficient $\sqrt{p(1-p)}$. Taking the limit as the faithful rate $p \to 0$:

$$\lim_{p\to 0} \|\mathbb{E}[\hat{\nabla} J_{\text{resp}}]\|_2 \propto \lim_{p\to 0} \sqrt{p(1-p)} = 0 \tag{30}$$

This mathematically demonstrates *gradient starvation*: when a task is exceedingly difficult and global rewards are uniformly 0, standard GRPO normalizes the advantages to zero, providing no effective optimization signal to correct errors.

Conversely, for BALTO, faithful responses y^+ yield zero advantage. The gradient expectation strictly depends on the hallucinated responses y^-:

$$\mathbb{E}[\hat{\nabla} J_{\text{BALTO}}] = p \cdot 0 + (1-p) \cdot \mathbb{E}_{y^-} \left[\sum_{t=1}^{T} A_t^{\text{bal}} g_t \right] \tag{31}$$

The magnitude of BALTO's expected gradient is governed by the pre-factor $(1-p)$. Taking the limit as $p \to 0$:

$$\lim_{p\to 0} \|\mathbb{E}[\hat{\nabla} J_{\text{BALTO}}]\|_2 \propto \lim_{p\to 0} (1-p) = 1 \tag{32}$$

BALTO achieves its theoretical maximum expected gradient magnitude precisely when the model's performance is at its lowest, guaranteeing highly efficient exploration and parameter updates during the cold-start phase.

Part II: Divergent Penalties in the Convergence Regime ($p \to 1$)

We now analyze the penalty assigned to a rare, hallucinated trajectory y^- as the model approaches near-perfect alignment ($p \to 1$). For standard GRPO, the penalty is determined by A^-:

$$\lim_{p\to 1} A^- = \lim_{p\to 1} -\sqrt{\frac{p}{1-p}} = -\infty \tag{33}$$

This divergence reveals a critical instability: a singular, rare mistake triggers an infinitely large negative update applied globally across the entire response. This disproportionate penalty destroys late-stage optimization efficiency by destabilizing previously well-aligned parameters.

In stark contrast, the advantages in BALTO are deterministically assigned based on local token sets according to Eq. 5:

$$A_t^{\text{bal}} \in \left\{ -1, \frac{N^-}{N^+}, 0 \right\} \tag{34}$$

Because hallucinations are a sparse minority of the response ($N^- \ll N^+$), the compensation advantage N^-/N^+ is a small positive fraction. Thus, all local advantages strictly remain bounded within $[-1, 1]$, entirely independent of the global faithful rate p.

Furthermore, analyzing the expected gradient magnitude for BALTO as $p \to 1$:

$$\lim_{p\to 1} \|\mathbb{E}[\hat{\nabla} J_{\text{BALTO}}]\|_2 \propto \lim_{p\to 1} (1-p) = 0 \tag{35}$$

This proves that BALTO ensures efficient and graceful final convergence: as the model ceases to hallucinate, the gradient expectation naturally decays to zero without ever triggering extreme penalties or gradient explosion. ■

E Implementation Details

E.1 Baselines.

Surprised Fine-Tuning (SFT). When constructing the SFT dataset, we first generated responses using the base model. We then employed Qwen3.5-35B-A3B as a reward model to detect hallucinations within these responses, with the prompt in Appendix H.1. For responses identified as containing hallucinations, we further used Qwen3.5-35B-A3B guided by the detection results to correct the original outputs, aiming to preserve the original content structure as much as possible. The prompt used for this correction process is provided in Appendix H.2. Both the original correct responses and the corrected responses were subsequently used as the training data for supervised fine-tuning.

Direct Preference Optimization (DPO). We constructed preference pairs using the SFT dataset: the negative samples were generated by the base model during the SFT data construction process, while the positive samples were obtained from the corrected responses. These preference pairs were then used to train the model with the DPO objective.

Group Relative Policy Optimization (GRPO). In the training process, we employed Qwen3.5-35B-A3B as a reward model to detect hallucinations within responses, with the prompt in Appendix H.1. For GRPO$_B$, it gives a 0/1 reward to indicate if the response has hallucination. For GRPO$_D$, we calculate the ratio of faithful claims as the reward.

FSPO. Similar to GRPO$_D$, we calculate the ratio of faithful claims as the reward to calculate the advantage. The advantage of the faithful claims in responses with negative advantage will be inverted. Similarily, The advantage of the hallucinated claims in responses with positive advantage will be inverted.

BALTO. As the same setting, we use the reward model to detect hallucination tokens within responses, with the prompt in Appendix H.1.

E.2 Token Alignment

The evaluator extracts claims, verifies them, and outputs erroneous text segments as `error_spans` in a single prompt call. We align these segments with the original rollout tokens in two phases.

Span location. We locate each substring in the response through cascaded matching: exact substring search, Markdown-insensitive matching, whitespace-insensitive matching, and combined Markdown, whitespace, and punctuation-normalized matching. The matched spans are mapped to character positions in the original response.

Character-to-token mapping. We re-encode the response with the training tokenizer and identify tokens overlapping the character spans. If the re-encoded token IDs match the original rollout IDs, tokenizer offsets directly provide the token indices. Otherwise, we decode the original token sequence prefix by prefix to build a character-to-token mapping and project the spans onto the original rollout tokens. If the minimal erroneous part cannot be reliably isolated, we use the entire hallucinated claim as described in Section 3.1.

E.3 Evaluation

LLM-based Evaluation. We use Qwen3.5-35B-A3B as the evaluator to extract claims from model responses and determine whether each claim is supported by the reference evidence. The detailed prompts used for claim extraction and hallucination detection are provided in Appendix H.1.

Faithfulness. Faithfulness (Faith.) is calculated as the proportion of responses without hallucinated claims over the test set.

Informativeness. Informativeness (Info.) measures whether a response preserves sufficient information compared to a reference response (generated by the base model). For each example, a response receives a score of 1 if it contains at least as many claims as the reference. Otherwise, it receives a score of 0. The final Info. score is averaged over the test set.

Q-Score. To jointly evaluate faithfulness and informativeness, we define: Q-Score = Faith. $\times$ Info., which reflects the overall response quality by jointly considering correctness and completeness. The multiplicative form explicitly penalizes responses that achieve high performance on only one dimension.

ConFiQA Accuracy. For ConFiQA, the paper [3] provides two metrics: P_c is the frequency of responses matching the context-faithful answer (excluding negations or the original answer), and P_o is the frequency of responses matching the original factual answer. We calculate the accuracy (Acc.) as: Acc. $= (P_c - P_o + 100)/2$, where P_c and P_o are expressed as percentages. A higher Acc. indicates better performance.

Win Rate. We follow prior work [22, 7] to compute the win rate (WR) using LLM-based pairwise comparison. We use DeepSeek-v3.2 [26] as the judge model with the prompt in Appendix H.3. For each prompt, responses from two models are compared twice with reversed ordering to mitigate position bias. Let $\mathrm{Win}(1)$ and $\mathrm{Win}(2)$ denote whether the candidate model is preferred in each ordering. The instance-level win indicator is defined as:

$$\mathrm{Win} = \frac{\mathrm{Win}(1) + \mathrm{Win}(2)}{2}.$$

The overall win rate over a test set of size N is:

$$\mathrm{WR} = \frac{1}{N} \sum_{i=1}^{N} \mathrm{Win}_i.$$

A win rate greater than 0.5 indicates that the candidate model is preferred over the compared model.

E.4 Experiments Compute Resources

We use 2x8 Nvidia H20 to conduct the experiments. The training time of Qwen3-8B and Qwen3-4B in one epoch is around 30 minites.

F Additional Results

F.1 Generalization to Llama-3.1-8B

We conduct additional experiments with Llama-3.1-8B [13] on RAGTruth and FinLLM-Eval to evaluate BALTO beyond the Qwen model family. Table 5 reports the results.

Table 5: Llama-3.1-8B results (%) on RAGTruth and FinLLM-Eval.

Method	RAGTruth			FinLLM-Eval		
	Faith.	Info.	Q-Score	Faith.	Info.	Q-Score
Base	39.6	*[ref.]*	39.6	41.9	*[ref.]*	41.9
GRPO$_B$	73.6	46.0	33.9	71.0	54.8	38.9
GRPO$_D$	91.3	53.6	48.9	69.9	67.7	47.3
FSPO	82.7	**77.1**	63.8	68.8	**68.8**	47.3
BALTO	**96.7**	69.1	**66.8**	**76.3**	63.4	**48.4**

BALTO achieves the highest faithfulness and Q-Score on both datasets. On RAGTruth, it reaches 96.7% faithfulness and 66.8% Q-Score. On FinLLM-Eval, it achieves 76.3% faithfulness and 48.4% Q-Score. These results support the effectiveness of balanced token-level credit assignment beyond the Qwen model family.

F.2 Results on MT-RAG

We evaluate BALTO on MT-RAG [19] using reference-answer-based lexical metrics: Precision, Recall, Token F1, and ROUGE-L. Table 6 shows that BALTO achieves the highest scores across all four metrics. Its ROUGE-L improves by 4.78 points over the base model, compared with 2.74 for GRPO$_D$ and 1.66 for FSPO. These results provide complementary evidence of improved response quality through better overlap with reference answers on an additional benchmark.

Table 6: Results on MT-RAG (%).

Method	Precision	Recall	Token F1	ROUGE-L
Base	35.03	44.69	36.26	23.58
GRPO$_D$	39.16	44.40	38.40	26.32
FSPO	37.20	45.53	38.00	25.24
BALTO	**40.75**	**47.39**	**40.42**	**28.36**

F.3 Frontier Model Reference Results

We evaluate DeepSeek-v3.2-thinking and Claude-Opus-4.8 on all three benchmarks using the same faithfulness evaluation protocol. Table 7 provides reference results alongside Qwen3-8B. BALTO-trained Qwen3-8B achieves higher faithfulness than both frontier models on ConFiQA and RAGTruth, while the frontier models perform better on FinLLM-Eval. The ConFiQA results are consistent with BALTO's explicit training to stay grounded in the provided counterfactual context.

Table 7: Faithfulness reference results (%) under the same evaluation protocol.

Model	ConFiQA	RAGTruth	FinLLM-Eval
Qwen3-8B (Base)	68.2	40.4	34.6
Qwen3-8B (BALTO)	**97.5**	**98.4**	92.5
DeepSeek-v3.2-thinking	84.9	88.8	94.4
Claude-Opus-4.8	92.9	96.3	**96.8**

F.4 Training Compute Overhead

Hallucination detection runs on the fly during RL training: each batch generates $K = 8$ responses per prompt, verifies their claims with Qwen3.5-35B-A3B, projects the judgments to token labels, and computes balanced advantages for the policy update. GRPO$_D$ and BALTO use the same claim extraction and verification call. BALTO additionally requests the specific erroneous spans through the `error_spans` field.

We measure the marginal detection cost with and without `error_spans` using the same evaluator on five random samples from each of the three datasets. Token localization increases detection time by 3.0% on average, or 0.12 seconds per sample. With reward computation accounting for approximately 32% of total step time in the measured runs, this corresponds to an estimated 1% increase in total training step time. The small marginal cost reflects that the evaluator already reasons about the errors during claim verification. Figure 5 further shows that BALTO reaches higher faithfulness with fewer training steps.

G Discussions

G.1 Possible Extension to PPO

BALTO targets the credit mismatch caused by broadcasting a response-level advantage to all tokens. Its localized corrective design can also be explored in PPO by treating the balanced credits in Eq. 5 as token-level factuality rewards, $r_t^{\text{fact}} = A_t^{\text{bal}}$: hallucinated tokens receive -1, faithful factual tokens receive N^-/N^+ when both sets are nonempty, and neutral tokens receive zero. The boundary cases follow the same assignment rule. Standard GAE can then compute advantages from these rewards and the value estimates, while the value model learns factuality-aware returns and KL penalties enter as additive reward terms. This preserves the localized reward construction. Bootstrapping, GAE, and KL terms need not preserve its zero-sum property in the resulting advantages. This extension provides a path to actor–critic training, while its interaction with value learning remains to be empirically evaluated.

G.2 Extension without Pre-existing Reference Documents

BALTO's token-level credit assignment can also be explored when reference documents are not provided in advance. FActScore [27] and SAFE [42] provide claim-level factuality verification using retrieved evidence. A possible extension is to extract atomic claims, retrieve supporting evidence, and ask the verifier to return both factuality judgments and minimal erroneous text spans. The token alignment procedure in Appendix E.2 then projects these spans to rollout tokens, after which the same balanced advantage rule can be applied. This integration would enable fully online RL training without pre-existing reference documents.

H Prompts

H.1 Hallucination Detection

Hallucination Detection

```
# Role
You are a strict auditor of response accuracy. Your core competency lies in
    meticulously cross-referencing statements within a response text against
    provided reference materials. Your objective is to verify whether every
    informational statement in the response text aligns with the reference
    materials, and to precisely pinpoint any specific erroneous segments.

# Task
I will provide a "Response Text" and a set of "Reference Materials." Your task is
     to verify, sentence by sentence, every informational statement within the
    Response Text -- specifically validating whether its assertions are
    consistent with the content found in the Reference Materials -- and to
    precisely identify any specific erroneous segments.

# Input Data
## [User Query & Reference Materials]
{{user_prompt}}

## [Response Text]
{{response}}

# Workflow
1. **Extraction:** Extract all informational sentences from the "Response Text."
- Extracted sentences typically contain specific data points -- such as figures (
    dates, times, and various other numerical values), entities (people, places,
    venues, etc.), and logical relationships (e.g., whether a cause-and-effect
    link holds true).
2. **Localization:** Locate the corresponding paragraphs within the "Reference
    Materials."
3. **Comparison:** Compare the statements made in the response against the
    original wording found in the reference materials.
4. **Verdict:**
- **Correct:** The statement aligns perfectly with the content found in the
    reference materials.
- **Incorrect:** A corresponding statement exists in the reference materials, but
     the response's phrasing does not match it; *or* the statement cannot be
    found within the specified reference materials (i.e., it is fabricated/
    hallucinated).
5. **Error Localization:** For every incorrect item, further pinpoint the
    specific "erroneous segment" (‘error_spans‘) within the "Response Text"
    itself, to facilitate subsequent token-level penalties.
- For each incorrect item, you must identify the minimal erroneous segment within
     the [Response Text] and output it to the ‘error_spans‘ list.
- Each entry in ‘error_spans‘ must be a contiguous substring taken directly from
    the original Response Text.
- If a single error spans multiple non-contiguous segments, split them into
    separate elements and add each to the ‘error_spans‘ list.
- Avoid marking entire sentences as erroneous segments; instead, identify only
    the minimal, specific error fragments.
- If no errors are detected, the ‘error_spans‘ list must be empty. # Possible
    Errors All possible data accuracy error types:

- conflict: Factual contradiction (the model generates information that
    contradicts facts in the references, such as misattribution of dates, metrics
    , entities, etc.)
```

```
- fabrication: Fabrication (the model generates information that is not mentioned
     in the original text/context)

- no_error: No error

# Output Format Please analyze carefully. The output should be in the following
    format. Strictly adhere to the format and do not output any extra symbols:

### Information Accuracy Analysis

{
"details": [

{
"claim_text": "The extracted sentence containing the information",

"source_text": "The corresponding sentence in the references",

"analysis": "Analyze whether there is a match or whether the calculation result
    is correct",

"error_type": "One of the error types in Possible Errors: conflict/fabrication/
    no_error",

"judgment_result": "Correct/Incorrect",

"error_spans": ["Error Span 1", "Error Span 2", ...]

},

{ "claim_text": "The extracted sentence containing the information",

"source_text": "The corresponding sentence from the references",

"analysis": "Analysis of whether a match is found or whether the calculation
    result is correct",

"error_type": "One of the error types in Possible Errors: conflict/fabrication/
    no_error",

"judgment_result": "Correct/Incorrect",

"error_spans": ["Error Span 1", "Error Span 2", ...]

},...

]
}
### Summary of Response Accuracy The response contains X claims, of which Y are
    correct matches and Z are incorrect matches (Note that this summary must
    strictly adhere to the format; do not output any extra symbols). If illusions
     exist, summarize the illusion points.
```

H.2 Response Correction

Response Correction

```
# Role
You are a rigorous text proofreading and fact-checking expert. Your task is to
    verify and correct numerical or factual errors -- identified in the [
```

```
    Hallucination Detection Report] -- within the [Original Reasoning Process]
    and [Original Answer], based on the provided [Reference Materials].

# Rules
When making modifications, please strictly adhere to the following workflow and
    editing principles:

## Step 1: Verify the Accuracy of Hallucination Detection
Before making any changes, you must first cross-reference the [Reference
    Materials] to determine whether the errors flagged in the [Hallucination
    Detection Report] actually exist.
- If the [Hallucination Detection Report] is accurate (i.e., the original text is
     indeed incorrect), proceed to Step 2 to make the necessary corrections.
- If the [Hallucination Detection Report] is a false positive (i.e., the original
     text actually aligns with the reference materials), retain the original text
     and do not make that specific correction.

## Step 2: Apply the "Minimal Change" Principle
If you confirm that the original text contains an error, you must correct it in
    both the [Original Reasoning Process] and the [Original Answer]. When making
    corrections, strictly follow the priority order below (from highest to lowest
    ):
1. **Precise Numerical Replacement (Primary Choice)**: Modify only the incorrect
    number (e.g., ‘17.96%‘ -> ‘12.08%‘).
2. **Precise Entity Replacement (Secondary Choice)**: If changing only the number
     results in a semantic contradiction (e.g., a subject mismatch), make minor
    adjustments to the local subject entity name or qualifier (e.g., ‘Bond D‘ ->
    ‘Bond A‘).
3. **Local Refinement (Tertiary Choice)**: If simply replacing the number or
    entity leads to grammatical errors, semantic incoherence, or logical
    conflicts within the context, you are permitted to make minor additions,
    deletions, or stylistic tweaks to the specific sentence -- or its immediate
    context -- to ensure it reads smoothly.
4. **Full Sentence Deletion (Last Resort)**: If the erroneous information is a
    complete fabrication (a "hallucination") that cannot be factually corrected
    through simple edits -- or if correcting it would cause the logic of the
    entire paragraph to collapse -- directly delete the entire sentence
    containing the error.

## Step 3: Formatting Requirements
- Preserve the original text structure, tone, and formatting (e.g., Markdown
    headings, lists, etc.). - The citation markers in the original text (such as
    ‘[12](@ref)‘) must remain accurate after modification. If a sentence is
    deleted, please also delete the citation markers that belong to that sentence
    .

---

# Input Data

### [References]

${user_prompt}

### [Hallucination Detection Results]

${detect_result}

### [Original Reasoning Process]

${reasoning_content}

### [Original Answer]
```

```
${answer}

---

# Output Format
Please output your results in the following structure:

### 1. Verification and Modification Explanation

(Briefly explain your verification results of the [Hallucination Identification
    Report] and what level of modification strategy you adopted: replacement/fine
    -tuning/deletion.)

### 2. Modified Reasoning Process

(Output the corrected complete reasoning_content)

### 3. Modified Answer

(Output the corrected complete answer)
```

H.3 Response Comparison

Response Comparison

```
You are a helpful assistant, that ranks models by the quality of their answers.

I want you to create a leaderboard of different of large-language models. To do
    so, I will give you the instructions (prompts) given to the models, and the
    responses of two models. Please rank the models based on which responses
    would be preferred by humans. All inputs and outputs should be python
    dictionaries.

Here is the prompt:
{
   "instruction": """{{instruction}}"""
}

Here are the outputs of the models:
[
    {
        "model": "model_1",
        "answer": """{{output_1}}"""
    },
    {
        "model": "model_2",
        "answer": """{{output_2}}"""
    }
]

Now please analyze the two models' answers and return the better model.

Output format:

### Analysis
xxx (your analysis)

### Better Model
model_1 / model_2 / tie (The model you think is better. If you think they are
    equally good, return "tie")
```

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state our contributions: a novel token-level RL framework (BALTO), a balanced credit assignment mechanism, theoretical analysis of gradient variance, and empirical SOTA results on three benchmarks (ConFiQA, RAGTruth, FinLLM-Eval).

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of our work in the Appendix.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Theoretical results (Theorem 1 and 2) regarding variance reduction and optimization efficiency are stated in Section 4, with detailed mathematical proofs and assumptions provided in Appendix.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provide detailed implementation details in Section 5.1 and Appendix E, including hyperparameters, dataset splits, optimizer settings, and the specific prompt templates used for the LLM evaluator.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We have provided a link to the code.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: As detailed in Section 5.1 and Appendix E.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: While we report mean performance across multiple benchmarks and model scales, we did not include explicit error bars for all main results due to computational constraints. However, we provide box plots for gradient norm distributions in Figure 3 to show optimization stability.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: As detailed in Section 5.1 and Appendix E.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our research follows standard machine learning practices and does not involve harmful data collection or unethical applications.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the potential societal impact in Appendix C.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: Our work focuses on algorithmic improvements for factuality and does not involve the release of models with high risks of dual-use or misuse beyond standard open-source weights.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cited all the original authors of the datasets (ConFiQA, RAGTruth, FinLLM-Eval) and models (Qwen3, DeepSeek) used in our study.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: No new datasets or software libraries were introduced; we focused on a new RL framework applied to existing benchmarks.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This research did not involve human subjects or crowdsourcing.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This research did not involve human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: The use of an LLM-based factuality evaluator is a core component of the BALTO framework and is described in Section 3.1 and Appendix E.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.