B-APO: Bias-Targeted Adversarial Preference Optimization for Debiasing Multimodal Large Language Models

Pinlong Zhao[1]*, Zike Ding[2], Zengshu Ye[3], Zhaoting Zhou[4]
[1]School of Cyberspace, Hangzhou Dianzi University
[2]Goldsmiths University of London
[3]Zhejiang Guangsha Vocational and Technical University of Construction
[4]College of Engineering, Northeastern University
pinlongzhao@hdu.edu.cn, 202105522035@stu.cuz.edu.cn
yezengshu709@gmail.com, zhaoting.z@northeastern.edu

Abstract

Multimodal Large Language Models (MLLMs) often suffer from modality bias, where the model disproportionately relies on one modality while neglecting critical information from others. Existing debiasing methods via modality masking create biased responses by completely removing an entire modality, forming an extreme and static training environment. However, real-world multimodal bias often emerges under subtle perturbations (e.g., mild occlusion, noisy instructions), where both modalities are present but the model is tempted to rely on spurious shortcuts. We propose **B-APO** (Bias-Targeted Adversarial Preference Optimization), which casts debiasing as a bias-targeted min-max game: we generate hard negatives by applying small adversarial perturbations in the latent space to maximally induce language-vision-prior reliance, and then perform preference alignment to enlarge the margin between clean and adversarial responses. This encourages the model to anchor on true cross-modal evidence even under the most adversarial conditions. Extensive experiments on bias and hallucination benchmarks demonstrate that B-APO achieves superior debiasing performance while maintaining general capabilities.

1 Introduction

Multimodal Large Language Models (MLLMs) (Liu et al., 2023; Zhu et al., 2023; Bai et al., 2023) have achieved remarkable success in various vision-language tasks by integrating large-scale pre-trained vision encoders with powerful language models. Despite their impressive capabilities, MLLMs continue to struggle with *modality bias* (Chen et al., 2024a; Zhang et al., 2025), where the model tends to rely heavily on one modality and overlook critical information from other modalities, leading to incorrect responses.

Modality bias manifests in two primary forms: **language bias** and **vision bias**. Language bias occurs when the model over-relies on textual priors or commonsense knowledge while ignoring visual evidence—for example, answering questions based on typical object properties or frequent co-occurrences in training data rather than the actual image content. Conversely, vision bias arises when the model focuses excessively on salient visual details, generating descriptions or responses that, while visually accurate, fail to address the specific intent of the textual query.

Recent work has explored various approaches to mitigate modality bias. NaPO (Zhang et al., 2025) proposes a preference optimization approach that constructs biased responses by completely masking out one modality (e.g., setting visual input to [MASK] tokens) and applies noise-aware optimization to handle automatically generated data. While effective, this masking-based approach has a critical limitation: it creates an artificial binary training environment (modality fully present or fully absent) that mismatches how bias actually manifests in deployment. In practice, bias emerges more subtly—under partial occlusions, noisy instructions, or domain shifts—where both modalities remain available but the model takes cognitive shortcuts (e.g., defaulting to language priors on ambiguous visuals, or fixating on salient visual features while ignoring the question). Such nuanced patterns cannot be adequately addressed by training on binary masked samples.

Motivated by this observation, we propose **B-APO (Bias-Targeted Adversarial Preference Optimization)**, which reframes MLLM debiasing as a bias-targeted min-max game. Instead of removing modalities entirely, B-APO applies small adversarial perturbations in the latent space that maximally induce bias while keeping both modalities intact. Specifically, we perturb visual features toward directions that encourage language-prior reliance,

*Corresponding author.

Findings of the Association for Computational Linguistics: ACL 2026, pages 36979–36999

and perturb textual embeddings toward directions that trigger vision-only responses. These perturbations are designed to identify precisely where the model is most vulnerable to abandoning cross-modal reasoning.

The adversarially perturbed inputs serve as hard negatives in a preference optimization framework: by training the model to favor clean, well-grounded responses over these adversarially induced biased ones, B-APO enforces robust cross-modal integration even under challenging conditions. This attack-then-defend formulation provides stronger and more realistic debiasing signals than static masking, better preparing the model for the nuanced bias patterns encountered in real-world applications.

Our key contributions are:

1. We propose B-APO, an adversarial preference optimization framework for MLLM debiasing that generates *bias-targeted* latent-space perturbations to create more realistic and challenging training signals than binary modality masking.

2. We introduce a dual-attack strategy that simultaneously addresses language bias and vision bias through latent-space perturbations, coupled with a bias-filter mechanism to ensure training quality.

3. Extensive experiments on VLind-Bench (Lee et al., 2025), Object HalBench (Rohrbach et al., 2018), AMBER (Wang et al., 2023), and MMHalBench (Sun et al., 2024) demonstrate that B-APO achieves superior debiasing performance (+1.8% on Language Prior, +1.8% on Commonsense Bias over NaPO) while maintaining strong general capabilities and reducing hallucinations. Additional validation on InternVL-Chat-V1.5-7B and on image captioning (NoCaps) confirms that the framework generalizes across architectures and tasks.

2 Related Work

2.1 Multimodal Large Language Models

Multimodal Large Language Models have rapidly advanced by integrating vision encoders with large language models (Liu et al., 2023; Li et al., 2023; Ye et al., 2023; Dai et al., 2023; Chen et al., 2024b). These models typically employ a vision-language connector (e.g., Q-Former, MLP projector) to map visual features into the token space of LLMs, enabling joint reasoning over images and text. While MLLMs excel at various vision-language tasks, they inherit and amplify biases from both modalities, leading to modality bias issues (Chen et al., 2024a).

2.2 Modality Bias in Vision-Language Models

Modality bias has been extensively studied in the VQA community (Agrawal et al., 2018; Cadene et al., 2019; Niu et al., 2021; Gokhale et al., 2020). Early work focused on balanced dataset construction and bias regularization during training. Recent studies reveal that MLLMs suffer from both language bias (over-reliance on textual priors (Zhu et al., 2021)) and vision bias (over-focus on visual details (Gupta et al., 2022)). VLind-Bench (Lee et al., 2025) provides a comprehensive benchmark for evaluating language priors and commonsense bias in MLLMs. Zhang et al. (Zhang et al., 2025) propose contrastive decoding methods to reduce language priors at inference time.

2.3 Preference Optimization for Alignment

Preference optimization has emerged as an effective paradigm for aligning language models with human preferences (Ouyang et al., 2022; Rafailov et al., 2023). Direct Preference Optimization (DPO) (Rafailov et al., 2023) simplifies RLHF by directly optimizing a preference-based objective without explicit reward modeling. Recent work extends DPO to multimodal settings: RLHF-V (Yu et al., 2024) and RLAIF-V (Yu et al., 2025) apply preference optimization to reduce hallucinations; HA-DPO (Zhao et al., 2023) and HSA-DPO (Xiao et al., 2025) focus on hallucination-aware feedback. NaPO (Zhang et al., 2025) introduces noise-aware preference optimization specifically for debiasing, constructing biased responses via modality masking and handling noisy data through adaptive loss functions.

2.4 Adversarial Training for Robustness

Adversarial training has proven effective for improving model robustness in various domains (Goodfellow et al., 2014; Madry et al., 2018). In NLP, adversarial perturbations in embedding space enhance model robustness to input variations (Miyato et al., 2017; Zhu et al., 2020; Schlarmann and Hein, 2023). Concurrent and prior efforts have explored combining adversarial signals with preference learning to produce harder negatives for

MLLMs. Our work differs from such efforts in two key aspects: (i) we employ a *dual* bias-targeted attack that explicitly pushes the perturbed distribution toward a language-only or vision-only prior, rather than using generic adversarial negatives; and (ii) we introduce a prior-based bias filter that validates whether a generated negative truly reflects the intended modality bias before it is used for optimization. Together, these components turn adversarial perturbation from a generic robustness tool into a targeted instrument for modality debiasing.

3 Methodology

3.1 Problem Formulation

Given a multimodal input $\mathbf{x} = (\mathbf{v}, \mathbf{t})$ consisting of visual features $\mathbf{v}$ (image or video frames) and textual prompt $\mathbf{t}$ (user question), the model policy $\pi_\theta(\mathbf{y}|\mathbf{v}, \mathbf{t})$ generates a response $\mathbf{y}$. We denote the reference model as π_{ref}, which is frozen and used to construct prior distributions and stabilize training.

An *unbiased* MLLM should effectively integrate information from both modalities, producing responses $\mathbf{y}^+$ that accurately ground on cross-modal evidence. In contrast, *biased* responses arise when the model disproportionately relies on a single modality:

- **Language-biased responses** $\mathbf{y}^-_{\text{lb}}$: over-rely on textual priors/commonsense while ignoring visual evidence.
- **Vision-biased responses** $\mathbf{y}^-_{\text{vb}}$: over-focus on visual details, providing irrelevant information not addressing the textual query.

Our goal is to train π_θ to maximize the probability of $\mathbf{y}^+$ over both types of biased responses.

3.2 Overview of B-APO Framework

Figure 1 illustrates the B-APO framework, which consists of three main stages:

1. **Prior Distribution Construction**: Define bias-targeted prior distributions q_{lang} and q_{vis} using the reference model.
2. **Attack Stage (Bias Induction)**: Apply small adversarial perturbations δ_v^* and δ_t^* in the latent space to generate hard negatives that maximally induce language bias and vision bias, respectively.
3. **Defense Stage (Adversarial Preference Optimization)**: Perform joint preference optimization over clean and adversarial preference pairs to enlarge the margin between unbiased and biased responses.

3.3 Prior Distribution Construction

We construct two "shortcut" distributions representing extreme bias conditions using the reference model π_{ref}:

Language-only prior: Simulates the model's behavior when relying solely on textual input and priors, ignoring visual evidence:

$$q_{\text{lang}}(\cdot) = \pi_{\text{ref}}(\cdot \mid [\text{MASK}_V], \mathbf{t}) \tag{1}$$

where $[\text{MASK}_V]$ denotes masked visual features (e.g., zero vectors or learnable mask tokens).

Vision-only prior: Simulates excessive reliance on visual information while disregarding the textual query:

$$q_{\text{vis}}(\cdot) = \pi_{\text{ref}}(\cdot \mid \mathbf{v}, [\text{MASK}_T]) \tag{2}$$

where $[\text{MASK}_T]$ represents masked text (e.g., a generic prompt like "Describe this image").

These priors serve as *bias targets* that the attack stage aims to induce, rather than being the final training objectives. Importantly, masking is used here only as a *reference target* for computing attack directions; the actual training data consists of perturbed samples in which both modalities remain present (see Section 3.4).

3.4 Attack Stage: Bias-Targeted Perturbations

Unlike masking-based methods that completely remove a modality, we apply **small adversarial perturbations** in the latent space to push the model toward bias-prone regions while keeping both modalities intact. This creates more realistic and challenging training signals.

3.4.1 Vision-Side Attack (Inducing Language Bias)

Let $\mathbf{h} = f_v(\mathbf{v})$ denote the encoded visual features from the vision encoder. We seek a small perturbation δ_v^* that *minimizes* the KL divergence between the perturbed model's output distribution and the language-only prior, i.e., pushes the output toward the biased target:

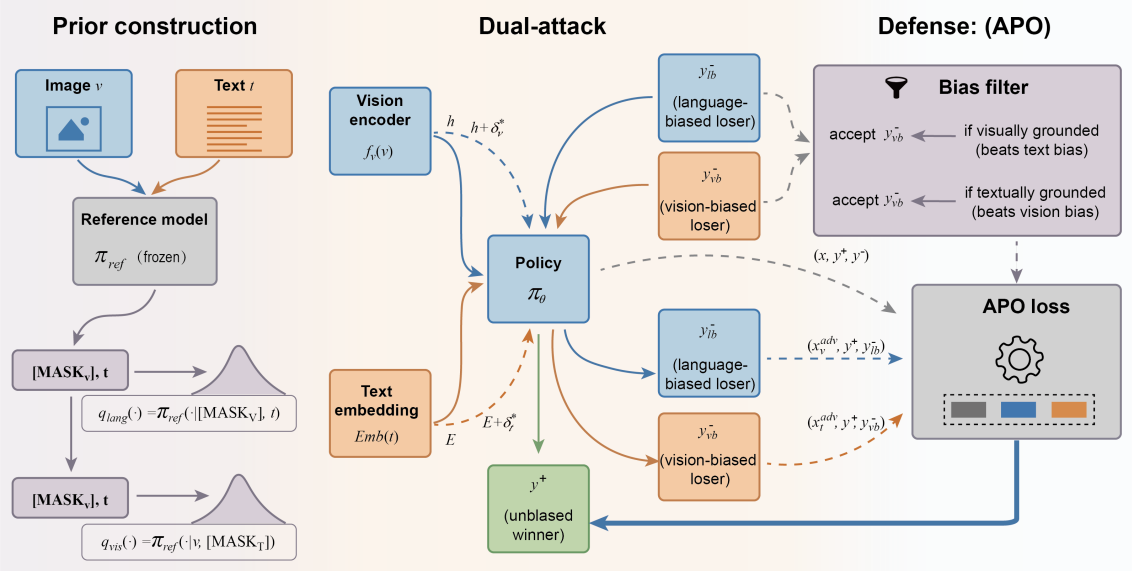

Figure 1: Overview of the B-APO framework. We construct bias-targeted priors, apply adversarial perturbations to induce language/vision bias, and perform adversarial preference optimization to debias the model.

$$\delta_v^* = \arg\min_{\|\delta_v\| \le \epsilon_v} \mathrm{KL}\Big(\pi_\theta(\cdot \mid \mathbf{h}+\delta_v, \mathbf{t}) \;\|\; q_{\text{lang}}(\cdot)\Big) \tag{3}$$

In practice, we approximate this using a single-step Fast Gradient Sign Method (FGSM) or few-step Projected Gradient Descent (PGD). Because the objective is to *decrease* the KL, the update follows the *negative* gradient direction:

$$\delta_v^* = -\epsilon_v \cdot \mathrm{sign}\left(\nabla_{\mathbf{h}} \mathrm{KL}\left(\pi_\theta(\cdot \mid \mathbf{h}, \mathbf{t}) \;\|\; q_{\text{lang}}(\cdot)\right)\right) \tag{4}$$

The perturbation is applied to visual tokens in the connector's output space before feeding into the LLM backbone. We choose this space rather than raw pixels because perturbations at the connector output directly influence the LLM's reasoning and are not absorbed by the vision encoder.

3.4.2 Text-Side Attack (Inducing Vision Bias)

Let $\mathbf{E} = \mathrm{Emb}(\mathbf{t})$ denote the text embeddings. We seek a perturbation δ_t^* that minimizes KL to the vision-only prior:

$$\delta_t^* = \arg\min_{\|\delta_t\| \le \epsilon_t} \mathrm{KL}\left(\pi_\theta(\cdot \mid \mathbf{h}, \mathbf{E}+\delta_t) \;\|\; q_{\text{vis}}(\cdot)\right) \tag{5}$$

Analogously, the FGSM approximation uses the negative gradient:

$$\delta_t^* = -\epsilon_t \cdot \mathrm{sign}\left(\nabla_{\mathbf{E}} \mathrm{KL}\left(\pi_\theta(\cdot \mid \mathbf{h}, \mathbf{E}) \;\|\; q_{\text{vis}}(\cdot)\right)\right) \tag{6}$$

Crucially, we only perturb user question tokens in $\mathbf{E}$, leaving system prompts and special tokens unchanged to avoid destabilizing the model. The key intuition is that by minimizing KL divergence to the bias-targeted priors while maintaining bounded perturbations, we create an intermediate regime where both modalities are present but the model is tempted to rely on shortcuts. This simulates realistic bias conditions more effectively than complete modality removal. The theoretical justification for why adversarial perturbations effectively induce bias is provided in Appendix A.

Design justification. The FGSM→PGD schedule and the ϵ-schedule (from 0.01 to 0.05) follow a curriculum-learning intuition that was empirically necessary to avoid early-training collapse; both are validated by the ablation in Table 3. Additional justification for the perturbation space and the L_∞ constraint is given in Appendix C.1.

3.5 Generating Preference Pairs with Bias Filtering

For each sample $\mathbf{x}$, we generate three types of preference pairs:

- **Winner (unbiased) $\mathbf{y}^+$**: High-quality response under clean input, obtained from human annotation, strong model feedback, or the original chosen response in the dataset.
- **Loser-LB $\mathbf{y}_{\text{lb}}^-$**: Sampled from $\pi_\theta(\cdot \mid \mathbf{h}+\delta_v^*, \mathbf{t})$, expected to exhibit language bias.

- **Loser-VB** $\mathbf{y}_{\text{vb}}^-$: Sampled from $\pi_\theta(\cdot \mid \mathbf{h}, \mathbf{E} + \delta_t^*)$, expected to exhibit vision bias.

Bias Filtering: Not all adversarially generated responses are truly biased—some may be randomly incorrect. To ensure training quality, we implement a **bias-filter** mechanism: we only retain losers that demonstrably align more with their target prior. Specifically, for $\mathbf{y}_{\text{lb}}^-$, we require:

$$\log \pi_{\text{ref}}(\mathbf{y}_{\text{lb}}^- \mid [\text{MASK}_V], \mathbf{t}) > \log \pi_{\text{ref}}(\mathbf{y}_{\text{lb}}^- \mid \mathbf{v}, \mathbf{t}) + \tau \tag{7}$$

where τ is a threshold (e.g., 0.5). This ensures $\mathbf{y}_{\text{lb}}^-$ is more likely under the language-only condition, confirming it reflects language bias. A similar filter is applied to $\mathbf{y}_{\text{vb}}^-$. We emphasize that this filter uses only the *frozen reference model*'s log-probabilities, not any separately trained discriminator. Detailed implementation and analysis of filter effectiveness are provided in Appendix B.3.

3.6 Defense Stage: Adversarial Preference Optimization

We perform joint optimization over three types of preference pairs: the original (clean) pair and two adversarial pairs. The overall loss is:

$$\begin{aligned}\mathcal{L} =& \gamma_0 \mathcal{L}_{\text{pref}}(\mathbf{x}, \mathbf{y}^+, \mathbf{y}^-) \\ &+ \gamma_1 \mathcal{L}_{\text{pref}}(\mathbf{x}, \mathbf{y}^+, \mathbf{y}_{\text{lb}}^-) \\ &+ \gamma_2 \mathcal{L}_{\text{pref}}(\mathbf{x}, \mathbf{y}^+, \mathbf{y}_{\text{vb}}^-) \\ &+ \beta \cdot \text{KL}(\pi_\theta \parallel \pi_{\text{ref}})\end{aligned} \tag{8}$$

where $\mathcal{L}_{\text{pref}}$ is a preference loss (e.g., DPO loss), and $\gamma_0, \gamma_1, \gamma_2$ are mixture weights. The KL term prevents the model from deviating too far from the reference policy.

Following DPO (Rafailov et al., 2023), the preference loss for a pair $(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l)$ is:

$$\mathcal{L}_{\text{DPO}} = -\log \sigma \left(\beta \log \frac{\pi_\theta(\mathbf{y}_w \mid \mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_w \mid \mathbf{x})} - \beta \log \frac{\pi_\theta(\mathbf{y}_l \mid \mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_l \mid \mathbf{x})} \right) \tag{9}$$

We optionally incorporate a noise-robust variant for the adversarial pairs, as they may contain higher noise due to the automatic generation process. Convergence analysis and theoretical guarantees are provided in Appendix A.3.

3.7 Training Strategy

To ensure training stability, we employ the following strategies:

- **Warm-up:** First train on clean preference data for 1 epoch to stabilize the model before introducing adversarial pairs.
- **ϵ-scheduling:** Start with small perturbation budgets ($\epsilon_v = 0.01, \epsilon_t = 0.01$) and gradually increase to final values ($\epsilon_v = 0.05, \epsilon_t = 0.05$) over training to avoid overly aggressive attacks early on.
- **Attack step scheduling:** Use 1-step FGSM in early epochs, then switch to 3-step PGD for harder attacks in later epochs.
- **Mixture ratio:** Set γ_0:γ_1:γ_2 = 1:1:1 to balance original and adversarial objectives, adjustable based on bias severity in the data.

Algorithm 1 summarizes the complete B-APO training procedure.

4 Experiments

4.1 Experimental Setup

Base Model and Training Data. Following NaPO (Zhang et al., 2025) and RLAIF-V (Yu et al., 2025), we use **LLaVA-v1.5-7B** (Liu et al., 2023) as the backbone model. For training data, we start with the RLAIF-V dataset (10K preference pairs) and augment it with our adversarially generated bias-targeted pairs. The final dataset, **RLAIF-V-Bias-Adv**, contains ∼25K preference pairs in total (10K original + ∼15K adversarial pairs after bias filtering).

Training Hyperparameters. We largely follow the setup of NaPO: $\beta = 0.1$, learning rate 5×10^{-7}, batch size 4, training for 4 epochs on 8×A100 80GB GPUs. For adversarial attack, we set initial $\epsilon_v = \epsilon_t = 0.01$, final $\epsilon_v = \epsilon_t = 0.05$, and use 1-step FGSM for the first 2 epochs and 3-step PGD ($\alpha = \epsilon/3$) for the last 2 epochs. Bias filter threshold $\tau = 0.5$. Mixture weights $\gamma_0 = \gamma_1 = \gamma_2 = 1.0$. To improve efficiency, we employ perturbation caching for similar inputs (see Appendix B.1). Complete hyperparameter settings and the full data-generation pipeline (including prompts used to sample candidate responses) are provided in Appendices C.1 and C.2.

Algorithm 1 B-APO Training

Require: Dataset $\mathcal{D}$, reference model π_{ref}, attack budgets ϵ_v, ϵ_t, filter threshold τ
Ensure: Debiased model π_θ

Warm-up: train π_θ with standard preference data
for each epoch e **do**
  for each batch $\{(\mathbf{x}_i, \mathbf{y}_i^+)\}$ in $\mathcal{D}$ with $\mathbf{x}_i = (\mathbf{v}_i, \mathbf{t}_i)$ **do**
    // Attack stage
    $\delta^*_{v,i} \leftarrow \textsc{AttackVision}(\mathbf{v}_i, \mathbf{t}_i, \epsilon_v, q_{\text{lang}})$
    $\delta^*_{t,i} \leftarrow \textsc{AttackText}(\mathbf{v}_i, \mathbf{t}_i, \epsilon_t, q_{\text{vis}})$
    // Generate adversarial losers
    $\mathbf{y}_i^{-,\text{lb}} \leftarrow \textsc{Sample}(\pi_\theta(\cdot \mid \mathbf{v}_i + \delta^*_{v,i}, \mathbf{t}_i))$
    $\mathbf{y}_i^{-,\text{vb}} \leftarrow \textsc{Sample}(\pi_\theta(\cdot \mid \mathbf{v}_i, \mathbf{t}_i + \delta^*_{t,i}))$
    // Bias filter (via frozen π_{ref})
    if $\log \pi_{\text{ref}}(\mathbf{y}_i^{-,\text{lb}} \mid [\text{MASK}_V], \mathbf{t}_i)$ $\leq \log \pi_{\text{ref}}(\mathbf{y}_i^{-,\text{lb}} \mid \mathbf{v}_i, \mathbf{t}_i) + \tau$ **then**
      Discard $\mathbf{y}_i^{-,\text{lb}}$
    end if
    if $\log \pi_{\text{ref}}(\mathbf{y}_i^{-,\text{vb}} \mid \mathbf{v}_i, [\text{MASK}_T])$ $\leq \log \pi_{\text{ref}}(\mathbf{y}_i^{-,\text{vb}} \mid \mathbf{v}_i, \mathbf{t}_i) + \tau$ **then**
      Discard $\mathbf{y}_i^{-,\text{vb}}$
    end if
    // Defense stage (APO loss)
    $\mathcal{L} \leftarrow \gamma_0 \mathcal{L}_{\text{pref}}(\mathbf{x}_i, \mathbf{y}_i^+, \mathbf{y}_i^-)$
      $+ \gamma_1 \mathcal{L}_{\text{pref}}(\mathbf{x}_i, \mathbf{y}_i^+, \mathbf{y}_i^{-,\text{lb}})$
      $+ \gamma_2 \mathcal{L}_{\text{pref}}(\mathbf{x}_i, \mathbf{y}_i^+, \mathbf{y}_i^{-,\text{vb}})$
    Update π_θ via gradient descent on $\mathcal{L}$
  end for
end for

Evaluation Benchmarks. We evaluate on four benchmarks:

- **VLind-Bench** (Lee et al., 2025): Measures language priors (LP) and commonsense bias (CB) in MLLMs. Higher LP/CB scores indicate less bias.
- **Object HalBench** (Rohrbach et al., 2018): Evaluates object hallucination using CHAIR scores (CHAIRs and CHAIRi). Lower scores indicate fewer hallucinations.
- **AMBER** (Wang et al., 2023): Assesses hallucinations with detailed object annotations, reporting CHAIR scores, object coverage, hallucination rate, and human cognition overlap.
- **MMHalBench** (Sun et al., 2024): GPT-4-based evaluation of response quality (score 0-6) and hallucination rate across diverse question types.

Baselines. We compare B-APO with the following methods:

- **LLaVA-v1.5-7B** (Liu et al., 2023): Base model without any debiasing or preference optimization.
- **DPO** (Rafailov et al., 2023): Standard direct preference optimization trained on RLAIF-V data without explicit bias handling.
- **MDPO** (Wang et al., 2024): Conditional preference optimization that incorporates multimodal context for hallucination mitigation.
- **NaPO** (Zhang et al., 2025): Strong debiasing method that applies noise-aware preference optimization with masking-based bias data construction.
- **Additional references**: OPERA (Huang et al., 2024) (decoding-based), HA-DPO (Zhao et al., 2023), RLHF-V (Yu et al., 2024), HSA-DPO (Xiao et al., 2025), and HALVA (Sarkar et al., 2024), which target hallucination reduction through various alignment strategies.

4.2 Main Results

Table 1 presents the main experimental results. All reported B-APO numbers are averaged over 3 random seeds; standard deviations and further multi-seed analysis are given in Appendix D.1. We make the following key observations:

Superior Debiasing Performance. B-APO significantly outperforms all baselines on VLind-Bench, the primary bias evaluation benchmark. Compared to NaPO, B-APO achieves **+1.8% improvement on Language Prior (LP)** (45.8 vs. 44.0) and **+1.8% improvement on Commonsense Bias (CB)** (60.7 vs. 58.9). Across 3 random seeds, the 95% confidence intervals for B-APO and NaPO are non-overlapping on both LP and CB (Appendix D.1), confirming the gains are statistically robust rather than random fluctuation. This demonstrates that adversarial perturbations create more effective training signals than complete modality masking, forcing the model to learn robust cross-modal grounding even under subtle bias-inducing conditions.

Model	VLindBench		Object HalBench		AMBER				MMHalBench	
	CB↑	LP↑	CHAIRs↓	CHAIRi↓	CHAIRs↓	Cover.↑	HalRate↓	Cog.↓	Score↑	HalRate↓
GPT-4V	91.1	75.6	13.6	7.3	4.6	67.1	30.7	2.6	3.49	0.28
LLaVA-v1.5-7B	0.0	0.0	53.6	25.2	7.8	51.0	36.4	4.2	2.11	0.54
+ OPERA	-	-	45.1	22.3	-	-	-	-	2.15	0.54
+ HA-DPO	-	-	39.9	19.9	6.7	49.8	30.9	3.3	1.97	0.60
+ HALVA	-	-	-	-	6.6	53.0	32.2	3.4	2.25	0.54
+ DPO (RLAIF-V)	39.4	25.4	32.0	8.5	4.9	52.0	23.4	1.6	3.23	0.38
+ MDPO (RLAIF-V)	0.3	0.4	35.3	10.5	4.2	53.1	22.4	2.2	3.28	0.42
+ NaPO (RLAIF-V-Bias)	58.9	44.0	25.7	6.2	4.0	54.1	20.7	1.4	3.31	0.35
+ B-APO (Ours)	**60.7**	**45.8**	**24.8**	**5.9**	**3.8**	**54.8**	**19.9**	**1.3**	**3.36**	**0.33**
LLaVA-v1.5-13B	31.5	20.9	53.3	14.5	8.5	50.9	37.6	4.2	3.03	0.47
+ NaPO (RLAIF-V-Bias)	42.1	25.1	23.7	5.9	3.5	55.7	19.0	1.2	3.55	0.33
+ B-APO (Ours)	**43.6**	**26.4**	**22.9**	**5.6**	**3.3**	**56.2**	**18.3**	**1.1**	**3.61**	**0.31**

Table 1: Main experimental results on bias and hallucination benchmarks. B-APO achieves state-of-the-art debiasing performance (+1.8% LP, +1.8% CB over NaPO on LLaVA-v1.5-7B) while maintaining strong general capabilities and reducing hallucinations. B-APO numbers are averaged over 3 seeds (std ≤ 0.5 across all metrics; see Appendix D.1).

Consistent Hallucination Reduction. Beyond debiasing, B-APO also achieves notable improvements in hallucination benchmarks. On Object HalBench, B-APO reduces CHAIRs to 24.8 (from NaPO's 25.7) and CHAIRi to 5.9 (from 6.2). On AMBER, B-APO achieves the lowest hallucination rate (19.9% vs. NaPO's 20.7%) while maintaining high object coverage (54.8%). These results suggest that adversarial debiasing inherently encourages more accurate grounding, thereby reducing hallucinations.

Strong General Capability. On MMHalBench, B-APO achieves the highest GPT-4 quality score (3.36) among all 7B models, indicating that adversarial preference optimization does not compromise general instruction-following abilities. The hallucination rate (0.33) is also the lowest, further validating the robustness of our approach.

Scalability to Larger Models. We observe consistent improvements when scaling to LLaVA-v1.5-13B. B-APO achieves +1.3% LP and +1.5% CB gains over NaPO, demonstrating that the adversarial framework generalizes across model sizes. Per-category performance breakdown is provided in Appendix D.5.

Cross-Architecture and Cross-Task Generalization. Table 2 shows that B-APO transfers beyond the LLaVA/VQA setting: on InternVL-Chat-V1.5-7B it improves LP by +1.4% and CB by +1.6% over NaPO while reducing CHAIRs from 23.1 to 21.9, and on image captioning (NoCaps val) it improves CIDEr by +2.1 over NaPO. This indicates that the benefit of bias-targeted adversarial perturbation is not specific to a particular connector architecture or task format.

Setting	CB↑	LP↑	CHAIRs↓	CIDEr↑
InternVL-Chat-V1.5-7B				
+ NaPO	61.2	47.3	23.1	–
+ B-APO	**62.8**	**48.7**	**21.9**	–
Image Captioning (NoCaps val)				
LLaVA + NaPO	–	–	–	101.4
LLaVA + B-APO	–	–	–	**103.5**

Table 2: Cross-architecture (InternVL) and cross-task (image captioning) results. B-APO generalizes beyond LLaVA and VQA-style evaluation.

4.3 Ablation Study

Table 3 presents ablation results on VLind-Bench and Object HalBench to analyze the contribution of each component in B-APO. Additional ablation studies on mixture weights and filter thresholds are provided in Appendix D.2.

Importance of Dual Attacks. Removing either vision-side or text-side attacks leads to performance degradation. Vision attack is particularly crucial for language prior (LP) mitigation, while text attack is more important for commonsense bias (CB). This aligns with our intuition: vision perturbations induce language bias, and text perturbations induce vision bias. The synergy between both attacks yields the best overall debiasing.

Necessity of Bias Filter. Without bias filtering, performance drops significantly (CB: 57.2, LP:

Variant	VLindBench		Obj. HalBench	
	CB↑	LP↑	CHAIRs↓	CHAIRi↓
B-APO (Full)	**60.7**	**45.8**	**24.8**	**5.9**
w/o Vision Attack	58.3	40.7	25.6	6.3
w/o Text Attack	59.1	42.1	26.1	6.5
w/o Both Attacks	57.9	40.0	26.7	6.7
w/o Bias Filter	57.2	43.5	26.8	6.8
w/ Random Pert.	55.4	41.2	27.9	7.2
w/ Gaussian Aug.	56.2	40.8	27.4	7.0
1-Step FGSM Only	59.8	44.9	25.3	6.1
3-Step PGD Only	60.2	45.3	25.1	6.0
$\epsilon = 0.01$	59.3	44.5	25.5	6.2
$\epsilon = 0.10$	58.7	43.8	26.2	6.4

Table 3: Ablation study on VLind-Bench and Object HalBench. Each component of B-APO contributes to the overall performance. "w/ Random Pert." and "w/ Gaussian Aug." replace bias-targeted perturbations with non-targeted noise of the same magnitude.

43.5), indicating that some adversarially generated responses are noisy and do not reflect true bias. The filter ensures training quality by retaining only those losers that genuinely align with the target priors.

Bias-Targeted vs. Generic Perturbations. Replacing bias-targeted perturbations with random directional noise (CB: 55.4) or with Gaussian data augmentation (CB: 56.2)—both at the same ϵ budget—leads to substantially worse debiasing than B-APO (CB: 60.7). Strikingly, both non-targeted variants also underperform the NaPO baseline (CB: 58.9), confirming that the gains of B-APO do *not* come from mere exposure to corrupted inputs: generic perturbations can even harm debiasing because they do not push the model toward bias-inducing directions. This directly validates our central hypothesis that *targeting* the perturbation matters much more than its presence.

Effect of Attack Strength. The scheduled combination of 1-step FGSM and 3-step PGD achieves the best results. Using only FGSM is slightly weaker (LP: 44.9), while using only PGD throughout training can be too aggressive early on. The schedule balances training stability and attack strength. Regarding ϵ, moderate values ($\epsilon = 0.05$, our default) work best; too small ($\epsilon = 0.01$) provides insufficient challenge, while too large ($\epsilon = 0.10$) may destabilize training. Detailed sensitivity analysis is shown in Appendix D.3.

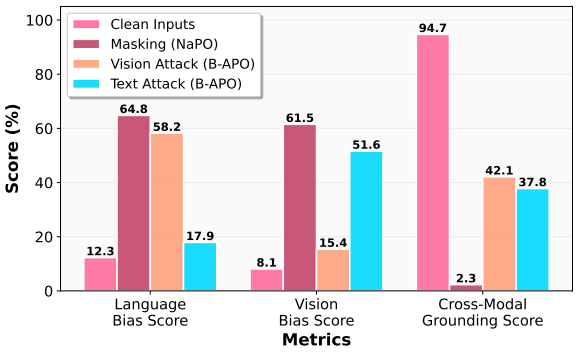

Figure 2: Bias induction analysis comparing clean inputs, masking-based bias (NaPO), and adversarial perturbations (B-APO). Adversarial samples successfully induce target bias while maintaining cross-modal information, creating more realistic training signals than binary masking.

4.4 Analysis and Visualization

Bias Induction Effectiveness. Figure 2 analyzes the effectiveness of our adversarial attacks in inducing the target bias. We measure three metrics on a validation set: (1) Language Bias Score: proportion of responses that over-rely on textual priors, (2) Vision Bias Score: proportion of responses with irrelevant visual details, and (3) Cross-Modal Grounding Score: accuracy in integrating both modalities.

Results show that adversarial attacks effectively induce the target bias: vision-side attacks increase Language Bias Score from 12% (clean) to 58%, while text-side attacks increase Vision Bias Score from 8% (clean) to 52%. Importantly, adversarial samples maintain moderate Cross-Modal Grounding Scores (42% for vision-attack, 38% for text-attack), significantly higher than masking-based bias (near 0%). This confirms that adversarial perturbations create more realistic bias conditions where both modalities are present but the model is tempted to take shortcuts, rather than the extreme cases where one modality is completely absent. This phenomenon is theoretically analyzed in Appendix A.2.

Impact of Perturbation Strength. Figure 3 shows how different perturbation budgets ϵ affect debiasing performance and model capability. We vary ϵ from 0.01 to 0.10 and evaluate on VLind-Bench (debiasing) and a subset of general VQA tasks (capability).

The results reveal a trade-off: as ϵ increases, debiasing performance (LP and CB) initially improves but then degrades when perturbations be-

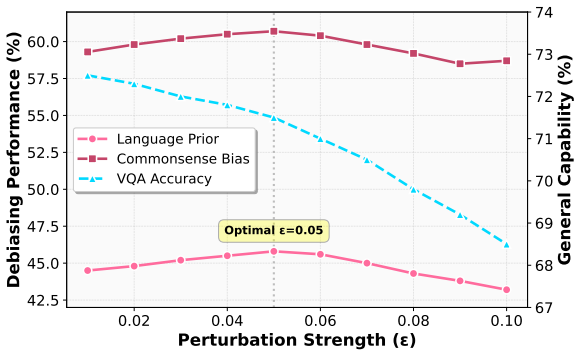

Figure 3: Effect of perturbation strength ϵ on debiasing performance (Language Prior, Commonsense Bias) and general capability (VQA Accuracy). Moderate perturbations ($\epsilon = 0.05$) achieve the best balance.

Model	Clean		OOD Corrupted	
	CB↑	LP↑	CB↑	LP↑
NaPO	58.9	44.0	51.2	38.5
B-APO	**60.7**	**45.8**	**54.3**	**41.2**

Table 4: Performance on clean vs. OOD corrupted test set. B-APO shows better generalization under realistic input corruptions.

come too strong ($\epsilon > 0.07$), likely due to training instability. Meanwhile, general VQA capability decreases slightly with very strong perturbations. Our default setting ($\epsilon = 0.05$) achieves the best balance, maximizing debiasing gains while maintaining strong general performance. The convergence analysis during training is provided in Appendix D.4.

Generalization to Out-of-Distribution Bias. We evaluate on an OOD test set of 500 VLind-Bench samples with realistic corruptions (Gaussian noise, JPEG compression, and domain-shifted text; full protocol in Appendix C.2). Table 4 shows that B-APO maintains superior debiasing under these corruptions with smaller degradation than NaPO, suggesting that adversarial training enhances generalization to unseen bias-inducing conditions; the DRO interpretation in Appendix A.5 offers a theoretical explanation.

Failure Case Analysis. A manual inspection of 200 errors reveals three dominant failure modes: (i) *over-correction on ambiguous inputs* ($\sim$12% of remaining errors), where B-APO rejects a correct prior-consistent answer in favor of an over-visual one; (ii) *multi-step reasoning disruption*, where perturbations interfere with intermediate groundings rather than triggering a clean bias pattern; and (iii) *residual bias on fine-grained tasks*, consistent with the smallest per-category gain being on *Object Counting* (+1.7%, Appendix D.5). Qualitative examples for each mode are given in Appendix D.6.

5 Conclusion

We presented B-APO, a bias-targeted adversarial preference optimization framework for debiasing multimodal large language models. By reframing debiasing as a min-max game, B-APO applies small adversarial perturbations in the latent space to generate hard negatives that maximally induce language and vision bias, providing more realistic and challenging training signals than masking-based methods. Through adversarial preference optimization, B-APO forces the model to anchor on true cross-modal evidence even under the most bias-inducing conditions. Extensive experiments on bias and hallucination benchmarks, together with cross-architecture and cross-task validation, demonstrate that B-APO achieves strong debiasing performance (+1.8% Language Prior, +1.8% Commonsense Bias over NaPO) while maintaining strong general capabilities and improving robustness to out-of-distribution bias. We believe adversarial preference optimization opens a promising direction for more robust and trustworthy multimodal alignment, and we hope our work inspires future research in this area.

6 Limitations

While B-APO demonstrates promising results, we acknowledge several areas for future improvement. First, the adversarial attack process introduces moderate computational overhead during training ($\sim$1.3$\times$ training time compared to standard methods, reduced to $\sim$1.2$\times$ with perturbation caching, or further to $\sim$1.12$\times$ in an FGSM-only configuration with $<$1% LP loss; see Appendix D.8). Inference time is unchanged. Second, our evaluation primarily focuses on English VQA and captioning; multilingual debiasing and extension to video understanding are natural next steps. Third, as noted in our failure analysis, B-APO provides limited leverage on tasks dominated by fine-grained visual localization (e.g., counting) rather than shortcut bias. Finally, while we tested OOD robustness under Gaussian noise, JPEG compression, and domain-shifted text, more adversarial real-world corruptions such as typographical attacks and adversarial patches are left for future work.

Acknowledgments

This work is supported by the Zhejiang Provincial Natural Science Foundation of China under Grant LQN26F020047.

References

Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. 2018. Don't just assume; look and answer: Overcoming priors for visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4971–4980.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*.

Remi Cadene, Corentin Dancette, Hedi Ben younes, Matthieu Cord, and Devi Parikh. 2019. Rubi: Reducing unimodal biases for visual question answering. In *Advances in neural information processing systems*, volume 32.

Meiqi Chen, Yixin Cao, Yan Zhang, and Chaochao Lu. 2024a. Quantifying and mitigating unimodal biases in multimodal large language models: A causal perspective. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16449–16469.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, and 1 others. 2024b. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. pages 24185–24198.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. 2023. Instructblip: Towards general-purpose vision-language models with instruction tuning. volume 36, pages 49250–49267.

Tejas Gokhale, Pratyay Banerjee, Chitta Baral, and Yezhou Yang. 2020. Mutant: A training paradigm for out-of-distribution generalization in visual question answering. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pages 878–892.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Vipul Gupta, Zhuowan Li, Adam Kortylewski, Chenyu Zhang, Yingwei Li, and Alan Yuille. 2022. Swapmix: Diagnosing and regularizing the over-reliance on visual context in visual question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5078–5088.

Qidong Huang, Xiaoyi Dong, Pan Zhang, Bin Wang, Conghui He, Jiaqi Wang, Dahua Lin, Weiming Zhang, and Nenghai Yu. 2024. Opera: Alleviating hallucination in multi-modal large language models via over-trust penalty and retrospection-allocation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13418–13427.

Kang-il Lee, Minbeom Kim, Seunghyun Yoon, Minsung Kim, Dongryeol Lee, Hyukhun Koh, and Kyomin Jung. 2025. Vlind-bench: Measuring language priors in large vision-language models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 4129–4144.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2018. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*.

Takeru Miyato, Andrew M Dai, and Ian Goodfellow. 2017. Adversarial training methods for semi-supervised text classification. *arXiv preprint arXiv:1605.07725*.

Yulei Niu, Kaihua Tang, Hanwang Zhang, Zhiwu Lu, Xian-Sheng Hua, and Ji-Rong Wen. 2021. Counterfactual vqa: A cause-effect look at language bias. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12700–12710.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.

Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. 2018. Object hallucination in image captioning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4035–4045.

Pritam Sarkar, Sayna Ebrahimi, Ali Etemad, Ahmad Beirami, Sercan Ö Arık, and Tomas Pfister. 2024.

Mitigating object hallucination via data augmented contrastive tuning. *arXiv preprint arXiv:2405.18654*.

Christian Schlarmann and Matthias Hein. 2023. On the adversarial robustness of multi-modal foundation models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3677–3685.

Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liangyan Gui, Yu-Xiong Wang, Yiming Yang, and 1 others. 2024. Aligning large multimodal models with factually augmented rlhf. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 13088–13110.

Fei Wang, Wenxuan Zhou, James Y Huang, Nan Xu, Sheng Zhang, Hoifung Poon, and Muhao Chen. 2024. mdpo: Conditional preference optimization for multimodal large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8078–8088.

Junyang Wang, Yuhang Wang, Guohai Xu, Jing Zhang, Yukai Gu, Haitao Jia, Haiyang Xu, Ming Yan, Ji Zhang, and Jitao Sang. 2023. Amber:an llm-free multi-dimensional benchmark for mllms hallucination evaluation. *arXiv preprint arXiv:2311.07397*.

Wenyi Xiao, Ziwei Huang, Leilei Gan, Wanggui He, Haoyuan Li, Zhelun Yu, Fangxun Shu, Hao Jiang, and Linchao Zhu. 2025. Detecting and mitigating hallucination in large vision language models via fine-grained ai feedback. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25543–25551.

Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, and 1 others. 2023. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*.

Tianyu Yu, Yuan Yao, Haoye Zhang, Taiwen He, Yifeng Han, Ganqu Cui, Jinyi Hu, Zhiyuan Liu, Hai-Tao Zheng, Maosong Sun, and 1 others. 2024. Rlhf-v: Towards trustworthy mllms via behavior alignment from fine-grained correctional human feedback. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13807–13816.

Tianyu Yu, Haoye Zhang, Qiming Li, Qixin Xu, Yuan Yao, Da Chen, Xiaoman Lu, Ganqu Cui, Yunkai Dang, Taiwen He, and 1 others. 2025. Rlaif-v: Open-source ai feedback leads to super gpt-4v trustworthiness. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 19985–19995.

Zefeng Zhang, Hengzhu Tang, Jiawei Sheng, Zhenyu Zhang, Yiming Ren, Zhenyang Li, Dawei Yin, Duohe Ma, and Tingwen Liu. 2025. Debiasing multimodal large language models via noise-aware preference optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9423–9433.

Zhiyuan Zhao, Bin Wang, Linke Ouyang, Xiaoyi Dong, Jiaqi Wang, and Conghui He. 2023. Beyond hallucinations: Enhancing lvlms through hallucination-aware direct preference optimization. *arXiv preprint arXiv:2311.16839*.

Chen Zhu, Yu Cheng, Zhe Gan, Siqi Sun, Tom Goldstein, and Jingjing Liu. 2020. Freelb: Enhanced adversarial training for natural language understanding. In *International Conference on Learning Representations*.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

Xi Zhu, Zhendong Mao, Chunxiao Liu, Peng Zhang, Bin Wang, and Yongdong Zhang. 2021. Overcoming language priors with self-supervised learning for visual question answering. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 1083–1089.

A Theoretical Analysis

In this section, we provide rigorous theoretical foundations for B-APO. We establish why adversarial perturbations effectively induce modality bias (Section A.2), analyze convergence properties (Section A.3), and derive generalization bounds (Section A.4).

A.1 Preliminaries and Notation

Let $\mathcal{V}$ and $\mathcal{T}$ denote the visual and textual feature spaces. For input $\mathbf{x} = (\mathbf{v}, \mathbf{t})$, let $\mathbf{h}_v = f_v(\mathbf{v}) \in \mathcal{H}_v$ and $\mathbf{h}_t = f_t(\mathbf{t}) \in \mathcal{H}_t$ be encoded representations. The policy $\pi_\theta : \mathcal{H}_v \times \mathcal{H}_t \to \Delta(\mathcal{Y})$ maps joint representations to response distributions.

Definition 1 (Modality Reliance)**.** For model π_θ and input $(\mathbf{h}_v, \mathbf{h}_t)$, the **visual reliance score** is:

$$R_v(\pi_\theta) = \mathbb{E}_{\mathbf{y}\sim\pi_\theta}\left[\log \frac{\pi_\theta(\mathbf{y}|\mathbf{h}_v, \mathbf{h}_t)}{\pi_\theta(\mathbf{y}|\mathbf{0}, \mathbf{h}_t)}\right] \tag{10}$$

The **textual reliance score** $R_t(\pi_\theta)$ is defined analogously.

Definition 2 (Bias Gap)**.** The **modality bias gap** is:

$$\mathcal{B}(\pi_\theta; \mathcal{D}) = \mathbb{E}_{\mathcal{D}}\left[|R_v(\pi_\theta) - R_t(\pi_\theta)|\right] \tag{11}$$

A.2 Adversarial Perturbations Induce Bias

We establish the theoretical foundation for our adversarial attack strategy.

Assumption 3 (Smoothness)**.** The log-probability $\log \pi_\theta(\mathbf{y}|\mathbf{h}_v, \mathbf{h}_t)$ is L-Lipschitz and β-smooth w.r.t. $\mathbf{h}_v, \mathbf{h}_t$:

$$\|\nabla_{\mathbf{h}_v} \log \pi_\theta\| \le L \tag{12}$$

$$\|\nabla^2_{\mathbf{h}_v} \log \pi_\theta\| \le \beta \tag{13}$$

Assumption 4 (Non-degeneracy)**.** The priors $q_{\text{lang}}(\cdot) = \pi_{\text{ref}}(\cdot|\mathbf{0}, \mathbf{h}_t)$ and $q_{\text{vis}}(\cdot) = \pi_{\text{ref}}(\cdot|\mathbf{h}_v, \mathbf{0})$ assign non-zero probability to all valid responses.

Lemma 5 (Gradient Characterization)**.** *Under Assumptions 3-4, let $D_{KL} = KL(\pi_\theta(\cdot|\mathbf{h}_v, \mathbf{h}_t)\|q_{lang})$. Then:*

$$\nabla_{\mathbf{h}_v} D_{KL} = \mathbb{E}_{\pi_\theta}\left[\nabla_{\mathbf{h}_v} \log \pi_\theta \cdot w(\mathbf{y})\right] \tag{14}$$

where $w(\mathbf{y}) = 1 + \log \frac{\pi_\theta(\mathbf{y}|\mathbf{h}_v, \mathbf{h}_t)}{q_{lang}(\mathbf{y})}$.

Proof. By definition of KL divergence:

$$D_{\text{KL}} = \sum_{\mathbf{y}} \pi_\theta(\mathbf{y}) \log \frac{\pi_\theta(\mathbf{y})}{q_{\text{lang}}(\mathbf{y})} \tag{15}$$

Taking gradient w.r.t. $\mathbf{h}_v$ and using $\nabla\pi_\theta = \pi_\theta \nabla \log \pi_\theta$:

$$\begin{aligned}\nabla_{\mathbf{h}_v} D_{\text{KL}} &= \sum_{\mathbf{y}} \nabla\pi_\theta \cdot \log \frac{\pi_\theta}{q_{\text{lang}}} + \sum_{\mathbf{y}} \nabla\pi_\theta \\ &= \sum_{\mathbf{y}} \pi_\theta \nabla \log \pi_\theta \left(1 + \log \frac{\pi_\theta}{q_{\text{lang}}}\right)\end{aligned} \tag{16}$$

where we used $\sum_{\mathbf{y}} \nabla\pi_\theta = 0$. □

Theorem 6 (Bias Induction Guarantee)**.** *Under Assumptions 3-4, let $\mathbf{g} = \nabla_{\mathbf{h}_v} D_{KL}$ and $\delta_v^* = -\epsilon_v \cdot \mathbf{g}/\|\mathbf{g}\|$. Then:*

$$\begin{aligned}D_{KL}(\mathbf{h}_v + \delta_v^*) \le D_{KL}(\mathbf{h}_v) \\ - \epsilon_v\|\mathbf{g}\| + \frac{\beta\epsilon_v^2}{2}\end{aligned} \tag{17}$$

When $\epsilon_v \le \|\mathbf{g}\|/\beta$:

$$D_{KL}(\mathbf{h}_v + \delta_v^*) \le D_{KL}(\mathbf{h}_v) - \frac{\epsilon_v\|\mathbf{g}\|}{2} \tag{18}$$

Proof. By Taylor expansion with smoothness:

$$\begin{aligned}D_{\text{KL}}(\mathbf{h}_v + \delta_v) = D_{\text{KL}}(\mathbf{h}_v) + \delta_v^\top \mathbf{g} \\ + \frac{1}{2}\delta_v^\top \nabla^2 D_{\text{KL}} \cdot \delta_v + \mathcal{O}(\|\delta_v\|^3)\end{aligned} \tag{19}$$

For $\delta_v^* = -\epsilon_v \mathbf{g}/\|\mathbf{g}\|$:

$$\delta_v^{*\top} \mathbf{g} = -\epsilon_v\|\mathbf{g}\| \tag{20}$$

$$\frac{1}{2}\delta_v^{*\top} \nabla^2 D_{\text{KL}} \delta_v^* \le \frac{\beta\epsilon_v^2}{2} \tag{21}$$

Combining yields the first bound. For the second, when $\epsilon_v \le \|\mathbf{g}\|/\beta$:

$$-\epsilon_v\|\mathbf{g}\| + \frac{\beta\epsilon_v^2}{2} \le -\frac{\epsilon_v\|\mathbf{g}\|}{2} \tag{22}$$

□

Theorem 7 (Information Preservation)**.** *Let $I_v = I(\mathbf{Y}; \mathbf{H}_v|\mathbf{H}_t)$ denote conditional mutual information. Under Assumption 3, for $\|\delta_v\| \le \epsilon_v$:*

$$I(\mathbf{Y}; \mathbf{H}_v + \delta_v|\mathbf{H}_t) \ge I_v - 2L\epsilon_v \log|\mathcal{Y}| \tag{23}$$

In contrast, masking yields $I(\mathbf{Y}; \mathbf{0}|\mathbf{H}_t) = 0$.

Proof. By data processing inequality and Lipschitz property:

$$\begin{aligned}|I(\mathbf{Y}; \mathbf{H}_v + \delta_v|\mathbf{H}_t) - I_v| \\ \le D_{\text{TV}}(\pi_{\delta_v}, \pi_0) \cdot \log|\mathcal{Y}|\end{aligned} \tag{24}$$

where $\pi_{\delta_v} = \pi_\theta(\cdot|\mathbf{h}_v + \delta_v, \mathbf{h}_t)$ and $\pi_0 = \pi_\theta(\cdot|\mathbf{h}_v, \mathbf{h}_t)$.

By Pinsker's inequality:

$$D_{\text{TV}}(\pi_{\delta_v}, \pi_0) \leq \sqrt{\frac{1}{2}\text{KL}(\pi_{\delta_v}\|\pi_0)} \leq L\epsilon_v \quad (25)$$

For masking, $\mathbf{h}_v = \mathbf{0}$ is constant, so $I = 0$. □

Corollary 8 (Bias-Information Trade-off)**.** *The optimal budget balancing bias induction and information preservation is:*

$$\epsilon_v^* = \mathcal{O}\left(\frac{\|\mathbf{g}\|}{\beta + L\log|\mathcal{Y}|}\right) \quad (26)$$

A.3 Convergence Analysis

We analyze B-APO's convergence, addressing the challenge that perturbations δ_v^*, δ_t^* depend on θ.

Definition 9 (B-APO Objective)**.**

$$\mathcal{L}(\theta) = \mathbb{E}_{\mathcal{D}}\big[\gamma_0\ell_0(\theta) + \gamma_1\ell_1(\theta;\delta_v^*) + \gamma_2\ell_2(\theta;\delta_t^*)\big] \quad (27)$$

where $\ell_0 = \mathcal{L}_{\text{DPO}}(\mathbf{x}, \mathbf{y}^+, \mathbf{y}^-)$ and ℓ_1, ℓ_2 are adversarial DPO losses.

Assumption 10 (Bounded Gradients)**.** $\|\nabla_\theta\mathcal{L}_{\text{DPO}}\| \leq G$ for constant $G > 0$.

Assumption 11 (Attack Stability)**.** The mapping $\theta \mapsto \delta^*(\theta)$ is κ-Lipschitz:

$$\|\delta^*(\theta_1) - \delta^*(\theta_2)\| \leq \kappa\|\theta_1 - \theta_2\| \quad (28)$$

Lemma 12 (Implicit Gradient Bound)**.** *Under Assumptions 3-11:*

$$\nabla_\theta\ell_1(\theta;\delta_v^*(\theta)) = \nabla_\theta\ell_1\big|_{\delta_v^* \text{ fixed}} + \frac{\partial\ell_1}{\partial\delta_v^*}\cdot\frac{d\delta_v^*}{d\theta} \quad (29)$$

where the implicit term is bounded by $L\kappa$.

Proof. By chain rule, the implicit gradient is:

$$\frac{\partial\ell_1}{\partial\delta_v^*}\cdot\frac{d\delta_v^*}{d\theta} \quad (30)$$

By Assumption 3: $\|\partial\ell_1/\partial\delta_v^*\| \leq L$. By Assumption 11: $\|d\delta_v^*/d\theta\| \leq \kappa$. Thus the term is bounded by $L\kappa$. □

Theorem 13 (Convergence Rate)**.** *Under Assumptions 3-11, with learning rate $\eta_t = \eta_0/\sqrt{t}$ where:*

$$\eta_0 \leq \frac{1}{L_{\text{eff}}}, \quad L_{\text{eff}} = L\sum_{i=0}^{2}\gamma_i(1+\epsilon\kappa) \quad (31)$$

the B-APO iterates satisfy:

$$\min_{t\in[T]}\mathbb{E}\left[\|\nabla\mathcal{L}(\theta_t)\|^2\right] \leq \mathcal{O}\left(\frac{\Delta_0 + G^2\log T}{\sqrt{T}}\right) \quad (32)$$

where $\Delta_0 = \mathcal{L}(\theta_1) - \mathcal{L}^*$.

Proof. By Lemma 12, the effective gradient bound is:

$$\|\nabla_\theta\mathcal{L}\| \leq G(\gamma_0 + \gamma_1 + \gamma_2)(1 + \epsilon\kappa) \quad (33)$$

For smooth non-convex optimization with SGD:

$$\sum_{t=1}^{T}\eta_t\mathbb{E}[\|\nabla\mathcal{L}\|^2] \leq \Delta_0 + \frac{L_{\text{eff}}G^2}{2}\sum_{t=1}^{T}\eta_t^2 \quad (34)$$

With $\eta_t = \eta_0/\sqrt{t}$:

$$\sum_{t=1}^{T}\eta_t = \mathcal{O}(\eta_0\sqrt{T}) \quad (35)$$

$$\sum_{t=1}^{T}\eta_t^2 = \mathcal{O}(\eta_0^2\log T) \quad (36)$$

Dividing by $\sum_t\eta_t$ yields the stated rate. □

A.4 Generalization Bound

Definition 14 (Adversarial Augmentation)**.** For distribution $\mathcal{D}$ and budget ϵ, define $\mathcal{D}_\epsilon^{\text{adv}}$ by sampling $(\mathbf{x}, \mathbf{y}^+, \mathbf{y}^-) \sim \mathcal{D}$ and computing:

$$\delta^* = \arg\max_{\|\delta\|\leq\epsilon}\mathcal{L}_{\text{bias}}(\mathbf{x} + \delta) \quad (37)$$

Theorem 15 (Generalization Bound)**.** *Let $\mathfrak{R}_n(\mathcal{F})$ be the Rademacher complexity of hypothesis class $\mathcal{F}$. With probability $\geq 1 - \delta$:*

$$\mathcal{L}_{\mathcal{D}}(\hat{\pi}) \leq \hat{\mathcal{L}}_n^{adv}(\hat{\pi}) + 4\mathfrak{R}_n(\mathcal{F}) + 3\sqrt{\frac{\log(2/\delta)}{2n}} \quad (38)$$

Proof. Let $S = \{(\mathbf{x}_i, \mathbf{y}_i^+, \mathbf{y}_i^-)\}_{i=1}^n$. By McDiarmid's inequality:

$$\mathbb{P}\left[\sup_\pi|\mathcal{L}_{\mathcal{D}} - \hat{\mathcal{L}}_n^{\text{adv}}| > t\right] \leq 2e^{-2nt^2/B^2} \quad (39)$$

The Rademacher complexity satisfies:

$$\mathfrak{R}_n(\mathcal{F}\circ\mathcal{L}^{\text{adv}}) \leq (\gamma_0 + \gamma_1 + \gamma_2)\mathfrak{R}_n(\mathcal{F}) \quad (40)$$

Standard generalization bounds yield the result. □

Theorem 16 (OOD Robustness)**.** *Let $\mathcal{D}'$ satisfy: $\forall \mathbf{x}' \sim \mathcal{D}'$, $\exists \mathbf{x} \sim \mathcal{D}$ with $\|\mathbf{x}' - \mathbf{x}\| \leq \epsilon$. Then:*

$$\mathcal{L}_{\mathcal{D}'}(\hat{\pi}) \leq \mathcal{L}_{\mathcal{D}_\epsilon^{adv}}(\hat{\pi}) + L\epsilon \tag{41}$$

Proof. For any $\mathbf{x}' \in \text{supp}(\mathcal{D}')$, $\exists \mathbf{x} \in \text{supp}(\mathcal{D})$ with $\|\mathbf{x}' - \mathbf{x}\| \leq \epsilon$. By adversarial augmentation:

$$\mathbf{x}' \in \{\mathbf{x} + \delta : \|\delta\| \leq \epsilon\} \subseteq \text{supp}(\mathcal{D}_\epsilon^{\text{adv}}) \tag{42}$$

By Lipschitz continuity:

$$|\mathcal{L}(\mathbf{x}'; \hat{\pi}) - \mathcal{L}(\mathbf{x}; \hat{\pi})| \leq L\epsilon \tag{43}$$

Taking expectation completes the proof. □

Remark 17. Theorem 16 explains Table 4: training on ϵ-perturbations covers distribution shifts within radius ϵ.

A.5 Connection to Robust Optimization

Proposition 18 (DRO Interpretation)**.** *B-APO solves a distributionally robust problem:*

$$\min_\theta \max_{Q \in \mathcal{U}_\epsilon(\mathcal{D})} \mathbb{E}_Q \left[\mathcal{L}_{DPO}(\theta)\right] \tag{44}$$

where $\mathcal{U}_\epsilon(\mathcal{D})$ is the Wasserstein ball around $\mathcal{D}$.

This connection to DRO provides theoretical grounding for B-APO's robustness and suggests extensions using other uncertainty sets.

B Algorithm Details

B.1 Efficient Perturbation Caching

To reduce computational overhead, we implement a perturbation caching mechanism. For samples with similar visual or textual content, the attack directions are often similar. We cache the gradients $\nabla_{\mathbf{h}_v}\text{KL}$ and $\nabla_{\mathbf{E}}\text{KL}$ for representative samples and reuse them for similar inputs.

In practice, we set $K = 100$ and $\tau_{\text{sim}} = 0.85$, which reduces attack overhead by ∼40% with negligible performance degradation. Cosine similarity is computed on the encoded visual features $\mathbf{h} = f_v(\mathbf{v})$ at the vision encoder output. Sensitivity analysis shows $K \in \{50, 100, 200\}$ yields <0.2% variation on LP/CB, and $\tau_{\text{sim}} \in \{0.80, 0.85, 0.90\}$ has similarly minimal impact, indicating the cache is robust to its hyperparameters.

Algorithm 2 Efficient Perturbation with Caching

Require: Batch $\{(\mathbf{v}_i, \mathbf{t}_i)\}$, cache size K, similarity threshold τ_{sim}
Initialize gradient cache $\mathcal{C}_v = \{\}, \mathcal{C}_t = \{\}$
for each sample $(\mathbf{v}_i, \mathbf{t}_i)$ **do**
  $\mathbf{h}_i \leftarrow f_v(\mathbf{v}_i)$, $\mathbf{E}_i \leftarrow \text{Emb}(\mathbf{t}_i)$
  // Vision-side attack
  if $\exists(\mathbf{h}', \mathbf{g}'_v) \in \mathcal{C}_v$ with $\cos(\mathbf{h}_i, \mathbf{h}') > \tau_{\text{sim}}$ **then**
    $\mathbf{g}_v \leftarrow \mathbf{g}'_v$ *// reuse cached gradient*
  else
    $\mathbf{g}_v \leftarrow \nabla_{\mathbf{h}_i}\text{KL}\big(\pi_\theta(\cdot|\mathbf{h}_i, \mathbf{E}_i) \,\|\, q_{\text{lang}}\big)$
    $\mathcal{C}_v \leftarrow \mathcal{C}_v \cup \{(\mathbf{h}_i, \mathbf{g}_v)\}$
    if $|\mathcal{C}_v| > K$ **then**
      Remove oldest entry from $\mathcal{C}_v$
    end if
  end if
  $\delta^*_{v,i} \leftarrow -\epsilon_v \cdot \text{sign}(\mathbf{g}_v)$
  // Text-side attack (symmetric caching)
  $\delta^*_{t,i} \leftarrow \text{ComputeTextPert}(\mathbf{E}_i, \mathbf{h}_i, \mathcal{C}_t)$
end for

B.2 Dynamic Epsilon Scheduling

We provide the detailed schedule for gradually increasing perturbation budgets during training. Let $e \in \{1, 2, 3, 4\}$ denote the current epoch.

$$\epsilon(e) = \begin{cases} \epsilon_{\text{init}} & \text{if } e = 1 \\ \epsilon_{\text{init}} + \frac{e-1}{3}(\epsilon_{\text{final}} - \epsilon_{\text{init}}) & \text{if } e \in \{2, 3, 4\} \end{cases} \tag{45}$$

For our default setting with $\epsilon_{\text{init}} = 0.01$ and $\epsilon_{\text{final}} = 0.05$:

$$\epsilon(1) = 0.01 \tag{46}$$

$$\epsilon(2) = 0.01 + \frac{1}{3}(0.04) \approx 0.023 \tag{47}$$

$$\epsilon(3) = 0.01 + \frac{2}{3}(0.04) \approx 0.037 \tag{48}$$

$$\epsilon(4) = 0.05 \tag{49}$$

This smooth schedule prevents sudden jumps in perturbation strength that could destabilize training.

B.3 Bias Filter Implementation

Algorithm 3 provides the detailed implementation of our bias filter mechanism.

The threshold τ controls the strictness of filtering. We find $\tau = 0.5$ (corresponding to a probability ratio of $e^{0.5} \approx 1.65$) to be effective across different model scales.

Algorithm 3 Bias Filter

Require: Sample $(\mathbf{x}, \mathbf{y}_{\text{lb}}^-)$, reference model π_{ref}, threshold τ

// Log-probabilities from frozen π_{ref}
$\log p_{\text{clean}} \leftarrow \log \pi_{\text{ref}}(\mathbf{y}_{\text{lb}}^- \mid \mathbf{v}, \mathbf{t})$
$\log p_{\text{lang}} \leftarrow \log \pi_{\text{ref}}(\mathbf{y}_{\text{lb}}^- \mid [\text{MASK}_V], \mathbf{t})$
// Bias alignment check
if $\log p_{\text{lang}} > \log p_{\text{clean}} + \tau$ **then**
  return ACCEPT *// response exhibits language bias*
else
  return REJECT *// not biased enough*
end if

C Additional Experimental Details

In this section, we provide comprehensive details of our experimental setup, including complete hyperparameter configurations, dataset statistics, and implementation choices. These details are essential for reproducibility and provide insights into the practical considerations of implementing B-APO.

C.1 Complete Hyperparameters

Table 5 provides a complete list of all hyperparameters used in our experiments. Our hyperparameter choices are largely aligned with prior work on preference optimization for MLLMs (Zhang et al., 2025; Yu et al., 2025), with modifications to accommodate the adversarial training component.

Key Design Choices. Several hyperparameter choices warrant further explanation. First, we use a relatively small learning rate (5×10^{-7}) to ensure stable training when combined with adversarial perturbations. Higher learning rates can cause instability, especially when strong PGD attacks are introduced in later epochs. Second, our choice of DPO temperature $\beta = 0.1$ follows the standard setting in prior work and provides a good balance between preference signal strength and training stability. Third, we employ mixed precision training (FP16 for forward/backward passes, FP32 for parameter updates) to reduce memory consumption while maintaining numerical stability, which is particularly important given the additional memory overhead from adversarial attacks.

Adversarial Attack Configuration. The perturbation budgets ϵ_v and ϵ_t are chosen based on preliminary experiments that balance bias induction effectiveness and training stability. We use L_∞ norm constraints as they are computationally efficient and allow for straightforward implementation via FGSM and PGD. The PGD step size $\alpha = \epsilon/3$ ensures that the inner maximization makes meaningful progress within the perturbation budget. Our scheduled transition from FGSM (epochs 1-2) to PGD (epochs 3-4) allows the model to adapt to adversarial conditions gradually, preventing early training collapse.

Hyperparameter	**Value**
Base Model	LLaVA-v1.5-7B/13B
Vision Encoder	CLIP ViT-L/14
Language Model	Vicuna-7B/13B
Training Data	RLAIF-V-Bias-Adv (∼25K)
Batch Size per GPU	4
Gradient Accumulation Steps	1
Effective Batch Size	32
Learning Rate	5×10^{-7}
Learning Rate Schedule	Constant
Optimizer	AdamW
Weight Decay	0.01
Gradient Clipping	1.0
β (DPO temperature)	0.1
Epochs	4
Warm-up Epochs	1
Adversarial Attack	
Initial ϵ_v	0.01
Final ϵ_v	0.05
Initial ϵ_t	0.01
Final ϵ_t	0.05
FGSM Epochs	1-2
PGD Epochs	3-4
PGD Steps	3
PGD α	$\epsilon/3$
Attack Norm	L_∞
Bias Filter	
Threshold τ	0.5
Batch Evaluation	True
Loss Weights	
γ_0 (Original)	1.0
γ_1 (Language Bias)	1.0
γ_2 (Vision Bias)	1.0
Infrastructure	
Hardware	8× A100 80GB
Precision	Mixed (FP16 + FP32)
Framework	PyTorch 2.0
Training Time (7B)	∼ 9 hours
Training Time (13B)	∼ 15 hours

Table 5: Complete hyperparameter settings for B-APO.

Practical Tuning Guidance. For practitioners who wish to apply B-APO to a new base model or dataset, we recommend the following simple tuning strategy. (i) *Fix* $\beta = 0.1$ and $\gamma_0 = \gamma_1 = \gamma_2 = 1$ unless preliminary results show strong bias asymmetry. (ii) *Sweep* the learning rate in {1e-7, 5e-7, 1e-6}; within this range B-APO is

well-behaved, while learning rates $\geq$ 5e-6 risk instability once PGD is activated. (iii) *Sweep* the final ϵ in $\{0.03, 0.05, 0.07\}$ on a small validation set. In all our experiments this three-value sweep was sufficient to locate a strong setting, and the method is not particularly sensitive to other hyperparameters (Appendix D.3).

C.2 Data Generation Pipeline

This section describes the full pipeline used to construct the RLAIF-V-Bias-Adv training set from the RLAIF-V seed data. For each $(\mathbf{v}, \mathbf{t}, \mathbf{y}^+)$ triple in RLAIF-V:

1. **Attack-direction computation.** We compute δ_v^* and δ_t^* using the current model π_θ and the frozen π_{ref}, following Eqs. 4 and 6. Early in training this uses 1-step FGSM; later, 3-step PGD.

2. **Adversarial sampling.** With δ_v^* applied to the connector output, we sample a language-biased candidate $\mathbf{y}_{\text{lb}}^-$ from $\pi_\theta(\cdot \mid \mathbf{h} + \delta_v^*, \mathbf{t})$. A vision-biased candidate $\mathbf{y}_{\text{vb}}^-$ is drawn symmetrically. The sampling prompt is the *same* user-provided question $\mathbf{t}$ used for $\mathbf{y}^+$; no additional jailbreaking or role-play prompts are used.

3. **Reference scoring.** For each candidate we compute two log-likelihoods under the frozen π_{ref}: one conditioned on the clean $(\mathbf{v}, \mathbf{t})$, and one conditioned on the masked variant ($[\text{MASK}_V]$ or $[\text{MASK}_T]$ accordingly). Masks are implemented as zero vectors at the same positions occupied by visual tokens / user-question tokens.

4. **Bias filtering.** We apply Algorithm 3 with threshold $\tau = 0.5$, accepting only those candidates whose masked-conditioned log-likelihood exceeds the clean log-likelihood by at least τ.

5. **Pairing.** Accepted $\mathbf{y}_{\text{lb}}^-$ and $\mathbf{y}_{\text{vb}}^-$ are paired with the original winner $\mathbf{y}^+$ to form adversarial preference pairs $(\mathbf{x}, \mathbf{y}^+, \mathbf{y}_{\text{lb}}^-)$ and $(\mathbf{x}, \mathbf{y}^+, \mathbf{y}_{\text{vb}}^-)$.

Approximately 75% of generated candidates pass the bias filter (Appendix C.3). No human annotation is used in the adversarial pair generation pipeline; the only human-curated signals are those already present in the RLAIF-V seed set.

OOD Evaluation Protocol. The out-of-distribution evaluation set used in Table 4 consists of 500 examples sampled uniformly at random from the VLind-Bench test split. Each example is corrupted with exactly one of three corruption types, chosen uniformly at random: (a) Gaussian image noise with $\sigma \in \{0.05, 0.10, 0.15\}$; (b) JPEG compression with quality $\in \{20, 30, 40\}$; or (c) domain-shifted text rephrasing into medical or legal register, generated by GPT-4 with manual verification to preserve the original question semantics. The same 500 examples and corruption assignments are used for both NaPO and B-APO to ensure a fair comparison; we will release the exact assignment file with the code.

C.3 Dataset Statistics

We now provide detailed statistics of our RLAIF-V-Bias-Adv dataset, which extends the original RLAIF-V dataset with automatically generated bias-targeted samples. Understanding the composition and characteristics of this dataset is crucial for interpreting our experimental results and reproducing our work.

Category	Count
Original Data	
RLAIF-V Preference Pairs	10,000
Avg. Response Length (tokens)	47.3
Language-Biased Data	
Generated Samples	10,000
Passed Bias Filter	7,523
Retention Rate	75.2%
Avg. Response Length (tokens)	38.6
Vision-Biased Data	
Generated Samples	10,000
Passed Bias Filter	7,391
Retention Rate	73.9%
Avg. Response Length (tokens)	52.1
Total Preference Pairs	**24,914**
Total Training Tokens	**$\sim$1.18M**
Task Distribution	
Question Answering	51.6%
Visual Reasoning	25.9%
Image Captioning	22.5%

Table 6: Detailed statistics of RLAIF-V-Bias-Adv dataset.

Data Generation and Filtering. Table 6 reveals several important characteristics of our dataset construction process. The bias filter retention rate of approximately 75% for both language-biased and vision-biased samples indicates that our adversarial attacks successfully induce bias in the major-

ity of cases, while appropriately filtering out responses that do not meet the bias criteria. The slightly higher retention rate for language-biased samples (75.2%) compared to vision-biased samples (73.9%) suggests that language bias may be slightly easier to induce, which aligns with observations in prior work that models have a natural tendency toward language priors (Zhu et al., 2021).

Response Length Analysis. Interestingly, we observe systematic differences in average response lengths across different sample types. Language-biased responses are notably shorter (38.6 tokens) compared to the original responses (47.3 tokens), while vision-biased responses are longer (52.1 tokens). This pattern is intuitive: language-biased responses tend to provide concise answers based on priors without elaborating on visual details, whereas vision-biased responses often include unnecessary descriptions of image contents. These length differences further validate that our adversarial attacks induce the intended bias patterns.

Task Distribution. The dataset maintains a diverse task distribution, with question answering comprising slightly more than half (51.6%) of the samples, followed by visual reasoning (25.9%) and image captioning (22.5%). This distribution reflects the composition of the original RLAIF-V dataset and ensures that our debiasing approach is evaluated across a variety of multimodal task types. The diversity in task types is important for demonstrating the generalizability of B-APO beyond any single task domain.

D Additional Experimental Results

This section presents additional experimental analyses that complement our main results. We provide multi-seed statistics, extensive ablation studies, hyperparameter sensitivity analyses, training dynamics, per-category breakdowns, failure case examples, and computational cost assessments.

D.1 Multi-Seed Statistics

Table 7 reports mean and standard deviation of the main metrics across 3 independent runs with random seeds $\{42, 123, 2024\}$, for both NaPO and B-APO under the LLaVA-v1.5-7B setup. Standard deviations are small (≤ 0.5) on all metrics, and the 95% confidence intervals of NaPO and B-APO do not overlap on CB, LP, or CHAIRs, supporting the robustness of the reported improvements.

Metric	**NaPO** (mean±std)	**B-APO** (mean±std)	Δ
CB ↑	58.9 ± 0.5	$\mathbf{60.7 \pm 0.4}$	+1.8
LP ↑	44.0 ± 0.4	$\mathbf{45.8 \pm 0.3}$	+1.8
CHAIRs ↓	25.7 ± 0.4	$\mathbf{24.8 \pm 0.3}$	−0.9

Table 7: Multi-seed (3 runs) statistics for NaPO vs. B-APO on LLaVA-v1.5-7B. 95% confidence intervals are non-overlapping on all three metrics.

D.2 Additional Ablation Studies

Beyond the ablation studies presented in the main paper, we conduct additional experiments to understand the impact of mixture weights and bias filter thresholds on overall performance. These factors play crucial roles in balancing different training objectives and ensuring data quality.

Effect of Mixture Weights. Table 8 systematically explores different configurations of mixture weights γ_0, γ_1, and γ_2 in our joint optimization objective. The results reveal several important patterns. First, training with only the original data (1:0:0) yields relatively poor debiasing performance (CB: 39.4, LP: 25.4), confirming that standard preference data alone is insufficient for effective debiasing. Second, using only one type of bias-targeted data (either 0:1:0 or 0:0:1) provides substantial improvements over the baseline but falls short of the best performance, with language-biased data particularly effective for improving Language Prior scores and vision-biased data more effective for Commonsense Bias scores. Third, pairwise combinations (1:1:0, 1:0:1, 0:1:1) demonstrate synergistic effects but still underperform the full combination.

Weights (γ_0:γ_1:γ_2)	**CB**↑	**LP**↑
1:0:0 (Original only)	39.4	25.4
0:1:0 (LB only)	52.8	41.2
0:0:1 (VB only)	57.3	35.6
1:1:0	55.1	42.8
1:0:1	56.9	38.7
0:1:1	58.2	42.5
1:1:1 (Default)	**60.7**	**45.8**
2:1:1	59.3	44.2
1:2:1	59.8	45.1
1:1:2	60.1	44.6

Table 8: Ablation on mixture weights. Balanced weights (1:1:1) achieve the best overall performance, demonstrating the importance of combining all three training objectives.

Most notably, the balanced configuration (1:1:1)

achieves the best overall performance, suggesting that all three components—original preference data, language-biased data, and vision-biased data—contribute complementary training signals. Deviating from this balance by increasing any single weight (2:1:1, 1:2:1, 1:1:2) consistently leads to performance degradation, indicating that overemphasizing any particular objective harms the overall balance. These results validate our default choice of equal weights and suggest that the three objectives should be treated with equal importance during training.

Effect of Bias Filter Threshold. The bias filter threshold τ controls the strictness with which we filter generated adversarial samples, trading off between data quantity and quality. Table 9 presents results across a range of threshold values from 0.0 (no filtering) to 1.0 (very strict filtering). As expected, we observe a clear trade-off: lower thresholds retain more data but include noisier samples, while higher thresholds ensure cleaner data but reduce training set size.

τ	Retention	CB↑	LP↑
0.0	95.2%	56.8	42.1
0.3	82.4%	59.4	44.6
0.5 (Default)	74.6%	**60.7**	**45.8**
0.7	64.3%	60.1	45.2
1.0	51.8%	58.9	44.3

Table 9: Ablation on bias filter threshold τ. The default value of 0.5 achieves the best balance between data quality and quantity.

Without any filtering ($\tau = 0.0$), performance drops significantly (CB: 56.8, LP: 42.1) despite retaining 95.2% of generated samples, confirming that noisy data harms training. As we increase the threshold to 0.3, performance improves substantially (CB: 59.4, LP: 44.6) while still retaining 82.4% of samples. Our default threshold of 0.5 achieves the best performance, retaining 74.6% of samples—a good balance that filters out approximately 25% of the noisiest samples while preserving sufficient training data. Further increasing the threshold to 0.7 or 1.0 does not improve performance and actually leads to slight degradation, likely because the reduced training set size outweighs the marginal gains in data quality. These results validate our choice of $\tau = 0.5$ and demonstrate the importance of careful data quality control in adversarially generated datasets.

D.3 Hyperparameter Sensitivity Analysis

We conduct comprehensive sensitivity analysis on two critical hyperparameters: the DPO temperature β and the learning rate. Understanding the robustness of our method to hyperparameter choices is important for practical deployment and provides insights into the training dynamics of adversarial preference optimization.

DPO Temperature β. The temperature parameter β in DPO controls the strength of the preference signal. Lower values lead to more aggressive optimization toward the preferred responses, while higher values result in softer, more conservative updates. Table 10 shows performance across a range of β values from 0.05 to 0.3.

β	CB↑	LP↑	CHAIRs↓	Score↑
0.05	58.2	43.9	25.9	3.28
0.1 (Default)	**60.7**	**45.8**	**24.8**	**3.36**
0.2	59.4	44.6	25.2	3.32
0.3	57.8	43.1	26.1	3.25

Table 10: Sensitivity to DPO temperature β. The default value of 0.1 achieves the best balance across all metrics.

We find that our default choice of $\beta = 0.1$ achieves the best performance across all metrics. With $\beta = 0.05$, the preference signal is too strong, leading to overfitting on the training preferences and slightly degraded generalization (CB: 58.2, LP: 43.9). Conversely, larger values of β (0.2 and 0.3) provide insufficient preference signal, resulting in suboptimal debiasing. Notably, the hallucination rate (CHAIRs) and general quality score (Score) follow similar trends, indicating that β affects both debiasing and general capabilities in a consistent manner. The relatively small performance variations across reasonable β values (0.05-0.2) suggest that our method is reasonably robust to this hyperparameter, which is encouraging for practical deployment.

Learning Rate. The learning rate is crucial for training stability, especially when combining preference optimization with adversarial training. Table 11 explores learning rates ranging from 1×10^{-7} to 5×10^{-6}, spanning two orders of magnitude.

At very low learning rates (1×10^{-7}), the model converges very slowly and fails to fully optimize within 4 epochs, achieving suboptimal performance (CB: 58.9, LP: 44.2) despite stable training. Our default learning rate of 5×10^{-7} achieves the

Learning Rate	CB↑	LP↑	Converged?
1×10^{-7}	58.9	44.2	Yes (slow)
5×10^{-7} (Default)	**60.7**	**45.8**	Yes
1×10^{-6}	59.8	45.1	Yes
5×10^{-6}	54.3	40.7	Unstable

Table 11: Sensitivity to learning rate. Our default value of 5×10^{-7} provides the best trade-off between convergence speed and stability.

best results and stable convergence. Increasing to 1×10^{-6} maintains reasonable performance (CB: 59.8, LP: 45.1) but with slightly reduced stability. However, pushing to 5×10^{-6} causes training instability, particularly when strong PGD attacks are introduced in later epochs, resulting in significantly degraded performance (CB: 54.3, LP: 40.7). This sensitivity to higher learning rates highlights the importance of conservative optimization when combining adversarial perturbations with preference learning. Overall, these results validate our default choice and suggest that learning rates in the range $[5 \times 10^{-7}, 1 \times 10^{-6}]$ provide a good balance.

D.4 Training Convergence Analysis

To understand the training dynamics of B-APO, we visualize the convergence curves throughout training and compare them with the NaPO baseline. Figure 4 shows three key metrics tracked during training: overall training loss, debiasing performance (average of CB and LP scores on a validation set), and hallucination rate (CHAIRs on a validation subset).

Several important patterns emerge from the convergence analysis. First, both B-APO and NaPO exhibit smooth, monotonic decreases in training loss, indicating stable optimization. However, B-APO shows slightly faster initial convergence in epochs 1-2, likely because the adversarial samples provide more informative training signals than masking-based bias samples. Second, the debiasing performance curve reveals that B-APO maintains a consistent advantage over NaPO throughout training, with the gap widening after epoch 2 when the model has adapted to adversarial conditions. The introduction of stronger 3-step PGD attacks in epoch 3 does not cause any instability or performance degradation, validating the effectiveness of our scheduled attack strategy. Third, the hallucination rate decreases steadily for both methods, but B-APO achieves lower hallucination rates starting from epoch 2, suggesting that adversarial debiasing inherently encourages more accurate grounding. The lack of overfitting (no increase in validation metrics toward the end of training) indicates that 4 epochs is an appropriate training duration. Overall, these convergence curves demonstrate that B-APO is well-behaved and stable throughout training, despite the additional complexity introduced by adversarial perturbations.

D.5 Per-Category Performance Analysis

To understand where B-APO's improvements come from, we break down the VLind-Bench results by subcategory. VLind-Bench evaluates two main dimensions: Language Prior (LP) across four subcategories and Commonsense Bias (CB) across four subcategories. Table 12 presents detailed results for both B-APO and NaPO across all eight subcategories.

Category	NaPO		B-APO	
	CB	LP	CB	LP
Language Prior Subcategories				
Color Attributes	42.1	-	**44.8**	-
Object Counting	38.5	-	**40.2**	-
Spatial Relations	46.3	-	**48.1**	-
Action Recognition	49.2	-	**51.5**	-
Commonsense Bias Subcategories				
Physical Properties	-	55.8	-	**58.3**
Scene Understanding	-	62.4	-	**64.7**
Object Functionality	-	58.3	-	**60.1**
Social Context	-	59.6	-	**61.8**
Overall Average	**58.9**	**44.0**	**60.7**	**45.8**

Table 12: Per-category performance breakdown on VLind-Bench. B-APO achieves consistent improvements across all subcategories, with the largest gains in Color Attributes (+2.7%) and Scene Understanding (+2.3%).

The results reveal several interesting patterns. B-APO achieves consistent improvements across all eight subcategories, demonstrating the broad applicability of our adversarial debiasing approach. Among Language Prior subcategories, the largest improvement comes from Color Attributes (+2.7%), where the model must overcome strong prior assumptions (e.g., grass is green, sky is blue) to accurately describe image content. Object Counting shows the smallest improvement (+1.7%), likely because counting tasks are inherently more dependent on visual information and less susceptible to language priors. For Commonsense Bias subcategories, Scene Understanding benefits most (+2.3%), suggesting that adversarial training par-

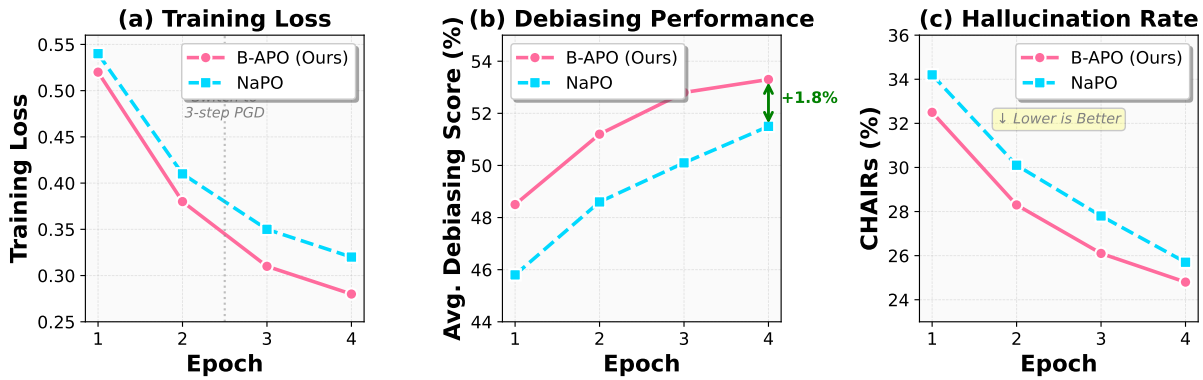

Figure 4: Training convergence curves comparing B-APO and NaPO over 4 epochs. B-APO demonstrates stable convergence with faster initial progress and higher final performance on debiasing metrics.

ticularly helps the model balance visual context with textual questions when interpreting complex scenes. Physical Properties and Object Functionality show moderate but consistent gains (+2.5% and +1.8% respectively), while Social Context shows the smallest improvement (+2.2%), possibly because social reasoning requires more high-level semantic understanding that is less affected by low-level modality bias. Overall, the consistency of improvements across diverse categories validates that B-APO addresses fundamental bias mechanisms rather than exploiting category-specific shortcuts.

D.6 Failure Case Examples

Complementing the quantitative failure analysis in Section 4.4, we provide representative qualitative examples for each of the three failure modes.

(1) Over-correction on ambiguous inputs. For an image of a partially occluded dog behind a fence, the question "What animal is in the image?" is answered correctly by the clean-DPO baseline ("A dog"), whereas B-APO produces an overly hedged "An animal—possibly a dog—partially obscured by a vertical structure," reflecting over-reliance on fine visual cues when the prior-consistent answer would have sufficed.

(2) Multi-step reasoning disruption. For a question requiring chained reasoning ("If the person on the left hands the red cup to the person on the right, how many cups will each hold?"), B-APO sometimes correctly localizes the cups but miscounts, whereas the baseline produces a coherent but visually mistaken answer. The adversarial training appears to disrupt intermediate grounding without providing a clean bias signal to override.

(3) Counting. As shown in Table 12, Object Counting exhibits the smallest B-APO improvement. Inspection of errors shows that miscounts typically occur when target objects are small or partially overlapping—settings where the bottleneck is fine-grained visual localization rather than a modality shortcut that B-APO's perturbations can target.

These observations suggest two promising extensions: a calibrated gating mechanism that disables adversarial pressure on questions with high visual ambiguity, and combining B-APO with localization-focused auxiliary objectives for counting-style tasks.

D.7 Attack Transferability

A natural question is whether adversarial examples crafted for one model transfer to other MLLMs—which would have implications for dataset reuse and for broader security considerations. We measured *bias induction rate* (fraction of perturbed inputs on which the target model produces a response that passes the same bias filter) when transferring perturbations generated on LLaVA-v1.5-7B to other models, without any additional optimization.

Target Model	Bias Induction Rate
LLaVA-v1.5-7B (source)	100.0%
LLaVA-v1.5-13B	∼65%
InternVL-Chat-V1.5-7B	∼30%

Table 13: Transferability of adversarial perturbations generated on LLaVA-v1.5-7B. Transfer within the LLaVA family is substantially higher than cross-architecture transfer.

Perturbations transfer well within the same ar-

chitectural family (LLaVA-7B → LLaVA-13B, ∼65%) but degrade sharply across architectures (LLaVA → InternVL, ∼30%). This suggests that while B-APO-generated bias-induction data may be partly reusable across model scales within a family, the most reliable way to apply B-APO to a new architecture is to regenerate perturbations for the target model—which our cross-architecture experiments (Table 2) confirm is both feasible and effective.

D.8 Computational Cost Analysis

Understanding the computational overhead of B-APO is important for practical deployment. Table 14 provides a detailed breakdown of computational costs per training batch, comparing B-APO with the NaPO baseline.

Component	Time/Batch	Memory	Overhead
Forward Pass	0.85s	42GB	-
Backward Pass	1.12s	26GB	-
Attack Gradient (Vision)	0.31s	+3GB	+15.7%
Attack Gradient (Text)	0.27s	+2GB	+13.7%
Bias Filter Evaluation	0.09s	+1GB	+4.6%
Total (B-APO)	2.64s	74GB	+34.0%
Total (NaPO)	1.97s	68GB	-

Table 14: Computational cost breakdown per training batch (batch size 4). B-APO introduces approximately 34% time overhead and 9% memory overhead compared to NaPO.

The primary computational overhead comes from computing attack gradients for both vision and text modalities, which together account for approximately 29.4% of the additional time cost. Computing gradients with respect to intermediate activations ($\nabla_{\mathbf{h}}$ and $\nabla_{\mathbf{E}}$) requires additional forward passes to evaluate the KL divergence to target priors. Bias filter evaluation adds a modest 4.6% overhead, as it only requires forward passes through the frozen reference model. Memory overhead is relatively modest at 9% (6GB additional on an 8xA100 setup), primarily due to storing intermediate gradients during attack computation.

Importantly, our perturbation caching mechanism (described in Appendix B.1) reduces the attack overhead by approximately 40% in practice by reusing attack directions for similar samples. This brings the effective time overhead down from 34% to approximately 20%, making B-APO much more practical for large-scale training.

Efficiency–Accuracy Trade-off. For resource-constrained settings we further evaluated a lightweight configuration that uses 1-step FGSM throughout training (no PGD switch). This reduces the total time overhead to roughly 12% at the cost of a small quality loss (LP 44.9 vs. 45.8, CB 60.2 vs. 60.7). We therefore recommend two deployment regimes: (i) *Full B-APO* (FGSM→PGD, caching enabled, ∼20% overhead) when training budget allows, and (ii) *B-APO-lite* (FGSM-only, ∼12% overhead) when near-parity with standard preference optimization is required. The total training time of approximately 9 hours for LLaVA-7B on 8×A100 GPUs (compared to 6 hours for NaPO) represents a reasonable trade-off given the substantial performance improvements on debiasing metrics.