Technical Summary:
VaPR: Vision-language Preference Alignment for Reasoning

Kai-Yu Lu

1 Research Problem and Motivation

Large Vision-Language Models (LVLMs) aim to generate text responses conditioned on both images and natural language instructions. Although recent LVLMs achieve strong performance on open-ended vision-language tasks, their outputs frequently exhibit two critical issues: misalignment with visual content and unreliable reasoning. Typical failure modes include hallucinating objects that are not present, ignoring spatial relations, or providing linguistically fluent but logically incorrect explanations.

Preference finetuning methods, such as Reinforcement Learning from Human Feedback (RLHF) and Direct Preference Optimization (DPO), align model outputs with desired behaviors through pairwise comparisons between better and worse responses. Recently, synthetic preference datasets generated by large models have become popular due to the high cost of human annotation. However, many existing synthetic preference datasets contain systematic noise in the form of stylistic and length biases between preferred and rejected responses. DPO is known to exploit such biases, which encourages the model to learn superficial patterns, such as preferring longer or more elaborate sentences, instead of truly learning content-level distinctions related to visual grounding or reasoning quality.

The research problem addressed in this paper is therefore twofold. First, LVLMs require better preference signals that target visio-linguistic reasoning and alignment, especially for perception and reasoning tasks. Second, existing synthetic preference datasets tend to encode spurious stylistic and length cues that cause reward hacking and degrade generalization. The paper proposes VaPR as a dataset construction framework and as a training recipe to mitigate these issues.

2 Related Work

2.1 Preference Optimization for LVLMs

LVLMs extend Large Language Models (LLMs) by connecting a vision encoder to a language backbone through a projection module. They are commonly trained in multiple stages: multimodal pretraining on large image-text corpora, supervised instruction tuning on curated datasets, and optional preference optimization to better match human judgments.

Preference optimization methods encompass RLHF, Reinforcement Learning from AI Feedback (RLAIF), and DPO. RLHF trains an explicit reward model and applies policy optimization, which is effective but computationally expensive and unstable. DPO directly optimizes the policy using pairwise preferences without training a separate reward model, providing a simpler and more stable alternative.

In LVLMs, several works have constructed preference datasets using human annotators, closed-source LVLMs such as GPT-4V, or self-preference techniques where the model generates multiple responses and selects preferred ones using its own scoring or auxiliary heuristics. Although these methods improve alignment and hallucination robustness, they largely ignore stylistic and length biases between chosen and rejected responses.

2.2 Synthetic Preference Datasets and Their Limitations

Existing synthetic datasets for LVLM preference finetuning include those based on hallucination-aware DPO, GPT-4V based rejection generation, and self-rewarding strategies. Many of these pipelines generate rejected outputs by sampling from large LVLMs or by instructing them to produce worse answers. This often leads to rejected responses that differ in style and length from the chosen ones, for example overly short or verbose sentences, even when the semantic content is similar.

Empirical analysis in the paper shows that such datasets exhibit large word-level Levenshtein distances and large token length gaps between chosen and rejected responses. Under DPO, these discrepancies can be exploited by the model as easy shortcuts. For some datasets, reward accuracy saturates very quickly, indicating that the model learns to distinguish pairs using superficial stylistic cues instead of genuine content differences.

Self-preference methods face an additional challenge. When preferred and rejected responses are generated by the same model and scored with similar heuristics, many pairs become nearly identical or even exact duplicates. In this regime the reference model plays a limited regularizing role, and DPO may overfit to noise in the comparisons.

2.3 Positioning of VaPR

VaPR is proposed as a response-editing based preference dataset tailored for LVLMs. Instead of generating both sides of the pair from a LVLM, VaPR treats a high-quality supervised response as the chosen output, then uses an LLM editor to produce a hard-negative rejected response. The editor is guided to introduce targeted semantic perturbations that flip correctness while preserving style and length. This design directly addresses two limitations of prior synthetic datasets:

- It removes spurious stylistic and length biases by constraining edits to minimal, task-relevant spans.
- It produces genuinely hard negatives that are close to the ground truth in form but incorrect in content, thereby providing stronger learning signals for DPO.

The framework is instantiated with GPT-4o as the primary editor, and an ablation study demonstrates that a strong open-source LLM, Qwen3-32B, can also serve as an effective editor, yielding VaPR-OS with comparable training benefits.

3 Dataset Construction

3.1 Source Dataset and Overall Pipeline

VaPR is constructed from the LLaVA-665K supervised finetuning dataset, which is a large-scale collection of instruction-image-response triples covering a wide range of multimodal capabilities. The VaPR pipeline operates in three main stages:

1. **Filtering** of unsuitable samples from the SFT corpus.
2. **Task-specific categorization and sampling** to cover diverse perception and reasoning skills.
3. **Hard-negative generation** via LLM-guided response editing that targets semantic errors while preserving style and length.

The final VaPR dataset consists of 30k preference pairs. Each pair includes an instruction with an associated image, a ground-truth response from the SFT corpus designated as the chosen output, and a synthetically generated hard-negative response designated as the rejected output.

3.2 Filtering Rules

The filtering stage removes samples that are not suitable for preference optimization of visio-linguistic reasoning. Specifically, the pipeline excludes:

- Purely text-only instructions that do not require visual understanding.
- Simple response formats such as multiple-choice questions and bounding box prediction, which do not provide sufficiently rich supervision for open-ended reasoning.
- Optical Character Recognition (OCR) instructions, for which performance is strongly affected by image resolution and low-level visual detail that are better addressed by architectural changes and higher-resolution input rather than preference finetuning.

Binary yes/no questions about existence, attributes, counting, or spatial relations are retained, because they directly probe key vision-language capabilities. These are converted into extended natural language responses, such as short explanations, in order to create more informative preference pairs.

3.3 Task Categories and Distribution

After filtering, the remaining samples are categorized into ten task types using task-specific keyword rules applied to instructions. The categories are:

- Object (type, material, action)
- Color
- Size
- Background (weather, time of day, surrounding context)
- Counting
- Spatial reasoning
- Existence
- Referential visual question answering (region-level properties or relations)
- General reasoning (abstract or knowledge-based)
- Image captioning

The goal is to cover both pure perception, pure reasoning, and composite perception plus reasoning skills. The resulting task distribution in VaPR is approximately balanced across core perception and reasoning tasks, with a modest fraction devoted to general reasoning and captioning.

Task Category	Approx. Share (%)	Description
Object	13.3	Object type, material, or action recognition.
Color	13.3	Fine-grained color attributes of objects or regions.
Size	3.3	Relative or absolute size of objects.
Background	3.3	Scene context such as weather or environment.
Spatial relation	13.3	Relative positions, such as left, right, in front of.
Counting	16.7	Number of objects or entities, often fine-grained.
Existence	13.3	Presence or absence of objects or attributes.
Referential VQA	13.3	Properties of localized regions or referred objects.
General reasoning	3.3	Abstract or world-knowledge based reasoning.
Captioning	6.7	Holistic descriptions of the image.

Table 1: Task categories and approximate percentage distribution in the 30k-sample VaPR dataset.

3.4 LLM-guided Hard-negative Generation

The core of VaPR is the generation of rejected responses by editing high-quality ground-truth outputs. For each selected supervised sample, the editor LLM receives:

- The instruction and a textual description of the visual context.
- The ground-truth response.
- Optional task-specific conditioning information, such as that the task is counting, color recognition, or spatial reasoning.
- For certain tasks, a penalty list of perturbation values, which prevents repetitive changes such as always changing a count from three to five.

The editor is instructed to minimally modify spans that are critical for correctness while keeping the rest of the sentence, style, and length as close as possible to the original. For example, for a counting question, the editor may only change the number while preserving wording, and for a spatial relation question, it may flip “next to” into “far away from” without altering surrounding phrasing.

This design leads to hard-negative responses that are fluent and stylistically similar but semantically incorrect for the given task. Quantitative analysis based on word-level Levenshtein distance and token length difference shows that VaPR has substantially lower stylistic and length gaps between chosen and rejected responses than several existing preference datasets.

3.5 Quality Assessment

Human evaluation is conducted on a stratified sample of 500 VaPR pairs across task categories. Annotators judge whether the rejected response is indeed incorrect and whether it remains stylistically close to the chosen response. The study reports that 97% of the samples satisfy the hard-negative criteria, and the inter-annotator agreement measured by Fleiss’ kappa is 0.86, which indicates high reliability of the synthetic annotations.

3.6 Open-source Variant VaPR-OS

To test generalizability, the same pipeline is applied using Qwen3-32B as the editor LLM, producing VaPR-OS on the same subset as a 10k-sample slice of VaPR. VaPR-OS exhibits slightly larger but still controlled stylistic and length differences, and models trained on VaPR-OS reach approximately 99% of the performance of models trained on the GPT-4o based VaPR, indicating that the framework does not rely on a specific proprietary editor.

4 Query Protocol and Task Definitions

4.1 LVLM Input and Output Representation

An LVLM takes as input a multimodal prompt denoted by

$$x = \langle x_v, x_t \rangle,$$

where x_v is the image and x_t is the instruction text describing the task or query. The model produces a textual response y that attempts to answer the instruction conditioned on the visual content.

For preference optimization, a dataset of preference pairs is defined as

$$\mathcal{D} = \{(x^{(i)}, y_w^{(i)}, y_l^{(i)})\}_{i=1}^{N},$$

where $x^{(i)}$ is the multimodal input in the i-th example, $y_w^{(i)}$ is the more preferred response (the chosen response), and $y_l^{(i)}$ is the less preferred response (the hard-negative rejection). In the VaPR setting, $y_w^{(i)}$ is the supervised ground-truth response and $y_l^{(i)}$ is the output of the LLM editor.

4.2 Task Definitions

Each sample in VaPR corresponds to one of the previously described task categories. The query protocol can be summarized as follows:

- **Perception tasks** such as object, color, size, background, and existence. The instruction asks about specific visual attributes or object presence, and correctness depends on accurate perception of the image.
- **Reasoning tasks** such as counting and spatial reasoning. The instruction requires combining perception with logical reasoning about number, order, or relationships.
- **Composite tasks** such as referential VQA, general reasoning, and captioning. These tasks require both fine-grained perception and higher-level reasoning or summarization.

Correctness for each task is defined with respect to the visual content and the instruction. The hard-negative responses are constructed to violate this correctness while remaining otherwise plausible, which is crucial for effective preference optimization.

5 Modeling Approach

5.1 Direct Preference Optimization Objective

Let $\pi_\theta(y \mid x)$ denote the conditional probability of generating response y given input x under the trainable LVLM with parameters θ, and let $\pi_{\text{ref}}(y \mid x)$ denote a fixed reference policy, typically the base instruct model after supervised finetuning.

A latent reward function $r(x, y)$ assigns a scalar reward to each response. The probability that the preferred response y_w is judged better than y_l is modeled as

$$p(y_w \succ y_l) = \sigma\big(r(x, y_w) - r(x, y_l)\big), \tag{1}$$

where $\sigma(\cdot)$ is the sigmoid function mapping real-valued differences to a probability in $[0, 1]$. The term $r(x, y_w) - r(x, y_l)$ measures how much better the preferred response is compared to the rejected one, and the sigmoid converts this difference into a probability of preference.

Direct Preference Optimization removes the need to explicitly learn $r(x, y)$ by expressing the reward differences in terms of log-probabilities under the trainable and reference policies. The DPO loss is defined as

$$\mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log\sigma\left(\alpha\log\frac{\pi_\theta(y_w \mid x)}{\pi_{\text{ref}}(y_w \mid x)} - \alpha\log\frac{\pi_\theta(y_l \mid x)}{\pi_{\text{ref}}(y_l \mid x)}\right)\right], \tag{2}$$

where α is a temperature scaling factor that controls the sharpness of the preference signal. In this expression:

- $\pi_\theta(y_w \mid x)$ and $\pi_\theta(y_l \mid x)$ are the probabilities assigned by the current model to the chosen and rejected responses.
- $\pi_{\text{ref}}(y_w \mid x)$ and $\pi_{\text{ref}}(y_l \mid x)$ are the corresponding probabilities under the reference model.
- The logarithms and ratios compare how much more the current model favors one response relative to the reference model.
- The sigmoid and negative log turn this into a loss that is minimized when the model consistently gives higher relative probability to the chosen responses.

Intuitively, Equation (2) encourages the trainable model to increase the relative likelihood of preferred responses and decrease that of rejected responses, while remaining regularized by the reference model so that probability shifts are not arbitrary.

The loss can be rewritten to highlight two key log-probability gaps:

$$\mathcal{L}_{\text{DPO}} = -\log\sigma\big(\alpha(\Delta_\theta - \Delta_{\text{ref}})\big), \tag{3}$$

where

$$\Delta_\theta = \log\pi_\theta(y_w \mid x) - \log\pi_\theta(y_l \mid x), \quad \Delta_{\text{ref}} = \log\pi_{\text{ref}}(y_w \mid x) - \log\pi_{\text{ref}}(y_l \mid x).$$

Here:

- Δ_θ measures how strongly the current model prefers the chosen response over the rejected one.
- Δ_{ref} measures the same preference under the reference model.
- The difference $(\Delta_\theta - \Delta_{\text{ref}})$ captures how much the new model amplifies or reduces the reference model's preference.

Equation (3) shows that the loss is small when the trainable model prefers y_w over y_l at least as strongly as the reference model does. If the model starts to prefer the rejected response, the argument of the sigmoid becomes negative, the sigmoid output becomes small, and the negative log becomes large, which penalizes the model. This formulation is central to the analysis of how stylistic and length biases in preference datasets can lead to reward hacking: if one side of the pair is systematically easier to predict or has certain stylistic patterns, Δ_θ and Δ_{ref} can capture these artifacts instead of genuine semantic differences.

5.2 Model Families and Finetuning Configuration

VaPR is used to preference-tune three LVLM families:

- LLaVA-v1.5-Instruct with 7B and 13B parameters.
- Qwen2VL-Instruct with 2B and 7B parameters.
- Qwen2.5VL-Instruct with 3B and 7B parameters.

During preference learning, LoRA is applied to adapt the language backbone while keeping most parameters frozen. The main hyperparameters are shared across model families and include:

- Learning rate: 1×10^{-6}, selected by a shallow search. Higher values such as 1×10^{-5} cause forgetting of pretrained knowledge, while lower values such as 1×10^{-7} are too small for effective learning.
- Effective batch size: 32 preference pairs.
- Training epochs: 5 over the VaPR dataset.
- Warmup ratio: 0.03 for LLaVA and 0.1 for Qwen2VL and Qwen2.5VL.
- LoRA rank: 128 and LoRA scaling factor $\alpha_{\text{LoRA}} = 256$.
- DPO loss with sigmoid-based objective and temperature parameter $\beta = 0.1$.
- Maximum sequence length: 2048 tokens.

Training is conducted on two A100 GPUs. This setup ensures that preference finetuning is relatively lightweight compared to pretraining and large-scale supervised finetuning, while still allowing consistent comparison across model families.

6 Empirical Results

6.1 Evaluation Benchmarks and Metrics

VaPR models are evaluated on ten benchmarks that collectively probe open-ended generation, perception, reasoning, hallucination robustness, and academic reasoning:

- **LLaVA-in-the-wild (LLaVAW)**: Open-world visual reasoning and description. Responses are scored by GPT-4 as an automatic judge, reflecting overall quality and alignment with the image.
- **ConTextual (ConT)**: Text-rich image understanding that requires reasoning over both embedded text and visual elements. Scores are again obtained using an LLM-as-a-judge protocol.
- **MM-VET (MMV)**: Evaluation of integrated multimodal capabilities including OCR, spatial reasoning, and math. GPT-4 is used to grade the quality and correctness of responses.
- **SEED-Bench image split (SEEDI)**: A comprehensive benchmark for perception and reasoning over images. Performance is measured as overall accuracy.
- **CV-Bench (CV)**: Vision-centric reasoning focused on counting, spatial relations, and comparative depth or distance. The metric is accuracy over structured questions.

- **MMStar (MMS)**: Another comprehensive benchmark for perception and reasoning that includes diverse tasks and scenes, evaluated with accuracy.
- **MathVista (MV)**: Visual mathematical reasoning over a variety of problem types. Performance is reported as accuracy.
- **MMMU**: Multi-discipline multimodal understanding at a college level, spanning physical sciences, social sciences, and finance. The metric is accuracy.
- **POPE**: Object hallucination benchmark that tests model robustness to presence and absence of objects. The metric is F1 score, capturing a balance between precision and recall.
- **NaturalBench (NB)**: Adversarial visio-linguistic reasoning benchmark where paired questions and images require distinct answers. The benchmark reports several metrics; the paper emphasizes overall accuracy, per-image and per-question accuracy, and a group accuracy that measures whether the model correctly answers all four combinations of paired images and questions.

For benchmarks that rely on LLM-as-a-judge scoring, GPT-4 is used with standardized prompts and a fixed model version to ensure consistent comparison across baselines and VaPR models. For classification-style benchmarks, success is measured through accuracy or F1, which directly reflect the proportion of correct predictions and the balance between false positives and false negatives.

6.2 Main Performance Trends

The paper compares three types of models for each LVLM family: the base instruct model, a supervised finetuned (SFT) variant on VaPR, and a DPO preference-tuned model on VaPR. Additional baselines are included for LLaVA-v1.5, such as human-feedback RLHF and other synthetic preference datasets.

For LLaVA-v1.5-7B, the VaPR DPO model improves LLaVAW from 64.8 to 76.2, ConTextual from 16.8 to 20.6, and MM-VET from 30.9 to 32.9. It also slightly improves SEED-Bench and MMStar and raises NaturalBench accuracy from 12.7 to 14.5. Averaged over the ten benchmarks, the improvement is around 6.5 points relative to the base model. The VaPR SFT model does not show comparable gains and can even slightly degrade performance, indicating that preference optimization is critical for extracting value from the VaPR dataset.

For LLaVA-v1.5-13B, VaPR DPO further raises LLaVAW from 72.3 to 80.5, improves ConTextual from 18.6 to 21.2, and increases MM-VET from 36.7 to 37.3. On SEED-Bench, CV-Bench, and MMStar there are consistent but moderate gains, and on NaturalBench the score rises from 14.9 to 18.2, indicating improved robustness in adversarial reasoning.

For Qwen2VL-2B and 7B, which are stronger base models, VaPR DPO still yields notable improvements. On the 2B model, ConTextual increases from 27.7 to 34.8, LLaVAW from 83.2 to 88.1, and NaturalBench from 24.3 to 25.7. On the 7B model, LLaVAW reaches 96.2 and MM-VET improves from 62.1 to 65.4, with NaturalBench increasing from 30.8 to 32.5.

For Qwen2.5VL-3B and 7B, the base models are already very strong. VaPR DPO still provides gains concentrated in vision-centric and adversarial benchmarks. For the 3B model, SEED-Bench increases from 75.0 to 75.5, CV-Bench from 71.5 to 72.7, and NaturalBench from 25.4 to 26.3. For the 7B model, SEED-Bench goes from 77.7 to 77.8, CV-Bench from 80.1 to 81.1, and NaturalBench from 32.0 to 32.8. Averaged across benchmarks, the improvements are about one to two points, which is meaningful given the strength of the baselines.

Overall, VaPR DPO outperforms prior preference-tuned models based on alternative datasets such as HA-DPO, POVID, SIMA, or CSR on eight out of ten benchmarks for LLaVA, and yields consistent gains across Qwen families. The improvements are especially pronounced in vision-centric reasoning, adversarial robustness, and text-rich reasoning benchmarks, even though VaPR does not explicitly train on OCR or pure math tasks.

6.3 Scaling Behavior

The paper studies the effect of training set size by constructing VaPR subsets of 3k, 10k, and 30k samples. Results indicate that:

- LLaVA models benefit significantly even from 3k samples, with diminishing returns at larger scales. This suggests that relatively modest amounts of carefully constructed preference data can substantially shift a weaker base model.
- Qwen2VL and Qwen2.5VL models show smaller gains at 3k samples but continue to improve more noticeably at 10k and 30k. Their stronger pretrained priors require more preference data to induce noticeable behavior changes.

This scaling pattern aligns with the intuition that weaker base models are easier to steer with small datasets, while stronger models demand more supervision to override entrenched behaviors.

6.4 Comparison with Other Preference Datasets

Using the reparameterized DPO loss in Equation (3), the paper analyzes how different preference datasets influence the optimization process. For datasets like POVID, the reference model probability gap Δ_{ref} is observed to be large for many pairs, driven largely by stylistic and length differences in rejected responses. As a result, reward accuracy during training quickly saturates near one, indicating that the model can easily distinguish chosen from rejected responses using superficial cues. This rapid saturation suggests reward hacking, where the model learns to trust length or style rather than content.

For self-preference datasets such as SIMA, many pairs have $\Delta_{\text{ref}} \approx 0$, and around 20% of pairs are exact duplicates. In this regime, the reference model provides little regularization, and the loss depends almost entirely on Δ_θ. The model then tends to overfit to weak or noisy differences, leading to low reward accuracy and degraded downstream performance.

VaPR mitigates these issues by explicitly constraining stylistic and length similarity when generating hard negatives, which keeps both Δ_θ and Δ_{ref} focused on semantic content differences. Empirically, VaPR models achieve higher downstream scores and exhibit smoother reward accuracy curves that do not prematurely saturate.

6.5 Reduction of “Yes” Bias in Binary Questions

A common failure mode in LVLMs is an overuse of the answer “Yes” in binary questions, especially when there is uncertainty or when two visually similar images require different answers. On NaturalBench, base models for LLaVA, Qwen2VL, and Qwen2.5VL show a clear tendency to answer “Yes” too frequently, even when the correct label is “No”.

After VaPR preference finetuning, the distribution of predictions shifts. Models answer “No” more often when appropriate, especially on adversarial paired examples where one image requires a “Yes” and another requires a “No”. The effect is particularly strong for LLaVA-VaPR-13B. This shift indicates improved visio-linguistic compositionality and reduced reliance on a default affirmative bias.

6.6 Limitations and Future Directions

Despite its strong performance, VaPR has several limitations:

- VaPR relies on an existing high-quality SFT dataset as a source of ground-truth responses. In domains where such supervised data is scarce, applying the same pipeline becomes more challenging.
- The primary experiments focus on English-language, image-text tasks, so the applicability to multilingual or video-based settings is not directly established.
- The framework assumes access to a strong editor LLM, such as GPT-4o or Qwen3-32B. While VaPR-OS shows that open-source editors work well, extremely resource-constrained settings may still face practical barriers.
- Preference optimization primarily enhances alignment, truthfulness, and reasoning consistency rather than factual knowledge. Performance on knowledge-intensive benchmarks such as MMMU remains limited.

The paper suggests several future directions, including extending VaPR to larger and more diverse datasets, targeting more nuanced reasoning errors, combining VaPR with self-preference methods in low-resource settings, and integrating VaPR with online preference optimization algorithms such as PPO or GRPO.

7 Summary

This technical summary has outlined the core contributions and methodology of VaPR, a vision-language preference alignment framework for reasoning in LVLMs. The central idea is to generate hard-negative responses by minimally editing high-quality ground-truth outputs, preserving style and length while introducing targeted semantic errors. This strategy produces a 30k-sample preference dataset that is well-suited for Direct Preference Optimization and avoids many pitfalls of prior synthetic preference corpora.

VaPR-based DPO finetuning is applied to three major LVLM families, consistently improving performance on ten diverse benchmarks, with particularly strong gains in vision-centric reasoning, adversarial robustness, and text-rich reasoning tasks. Empirical analyses show that VaPR alleviates reward hacking induced by stylistic and length biases and reduces the overuse of “Yes” in binary questions. An open-source editor variant, VaPR-OS, demonstrates that the pipeline generalizes beyond proprietary LLMs.

Key Takeaways

- Carefully constructed hard-negative preference pairs that preserve style and length while altering task-critical semantics provide stronger and more reliable signals for DPO in LVLMs.
- VaPR significantly improves multimodal reasoning and adversarial robustness across multiple LVLM families compared with both base instruct models and alternative preference-tuned baselines.
- Controlling stylistic and length biases in preference datasets is crucial to avoid reward hacking and to ensure that preference optimization focuses on meaningful content-level differences rather than superficial cues.