# CLOSING THE TRAJECTORY GAP: ON-POLICY LEARNING FOR MULTIMODAL REASONING EMBEDDINGS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Universal multimodal embeddings support a broad range of applications through a shared representation space. Although they build on multimodal large language models, most existing methods use these models primarily as encoders, leaving their generative capabilities underexplored. Recent reasoning-enhanced approaches generate intermediate reasoning before producing the final embedding and have shown promising performance on multimodal embedding benchmarks. However, training a unified reasoner–embedder often relies on task-specific chains of thought generated and filtered offline, while jointly optimizing language modeling and contrastive learning on these fixed trajectories. Beyond the cost of constructing this offline supervision, the model is trained on externally generated trajectories but conditions its inference-time embeddings on its own generations, creating a reasoning trajectory gap. Our controlled experiments suggest that this gap can undermine retrieval performance. We introduce OPR-EMBED (On-policy Reasoning Embedding), a two-stage on-policy framework for multimodal reasoning embeddings. Without any pre-built CoT corpus, Stage I performs embedding-attentive on-policy distillation by scoring prefixes sampled from the current student with a frozen teacher and weighting token-level supervision using attention from the final embedding readout. Stage II uses verifier-guided group relative policy optimization to refine complete trajectories using retrieval feedback together with a candidate-match verifier. Experiments on MMEB-v2 demonstrate that OPR-EMBED is competitive with state-of-the-art methods, achieving overall scores of 65.8 and 70.0 with 2B and 7B backbones, respectively.

## 1 INTRODUCTION

Universal multimodal embeddings map heterogeneous inputs and task instructions into a shared representation space (Wei et al., 2024; Lin et al., 2025), supporting applications such as cross-modal retrieval, visual question answering, and document search (Jiang et al., 2025; Meng et al., 2026). Multimodal large language models (MLLMs) provide a common interface for these inputs, but most embedding methods use them primarily as encoders (Jiang et al., 2025; Meng et al., 2026; Liu et al., 2025b). Recent reasoning-enhanced approaches instead generate intermediate text before producing an embedding, with promising results on multimodal embedding benchmarks (Cui et al., 2026; Lan et al., 2026; Wang et al., 2026). Methods such as $\mathrm{TTE}_t$ and $\mathrm{TTE}_s$ (Cui et al., 2026), as well as Embed-RL (Jiang et al., 2026), rely on a separate reasoner to produce intermediate reasoning, requiring an additional model at inference. Integrating reasoning generation and embedding extraction within a single MLLM removes this deployment dependency (Lan et al., 2026; Wang et al., 2026). We focus on how to jointly train these two capabilities within such a unified model.

A common training recipe combines language-model supervision on annotated chains of thought (CoTs) with contrastive learning on the resulting embeddings (Cui et al., 2026; Lan et al., 2026; Liu et al., 2025a; Wang et al., 2026). In its simplest form, the model learns to predict the annotated tokens through supervised fine-tuning (SFT), while the hidden state following the same CoT is trained to match the paired target through a contrastive loss. The intent is to learn discriminative representations while retaining the ability to generate the reasoning that supports them. Obtaining these annotations often involves task-specific generation by external models, followed by rewriting,



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

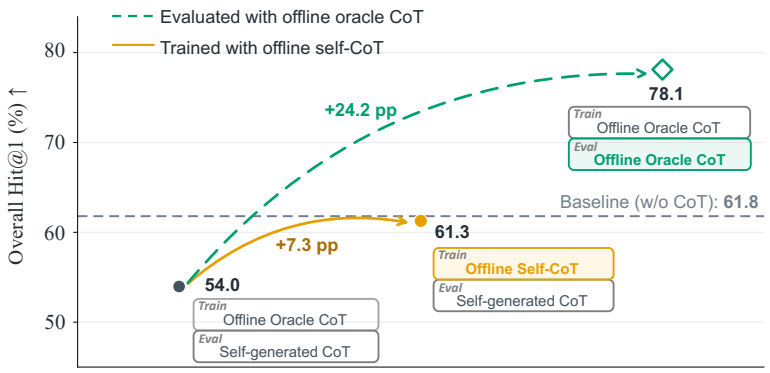



Figure 1: **Controlled comparisons on MMEB-v1 reveal the reasoning trajectory gap.** Under controlled training settings, models trained on Oracle CoTs (annotated by a stronger model with target references) perform worse than those trained on offline Self-CoTs (generated by the initial student). Both underperform the no-CoT baseline when using their own inference-time reasoning. Yet replacing the Oracle-trained model's reasoning with Oracle CoTs constructed for the evaluation queries raises Hit@1 from 54.0 to 78.1 without updating model parameters.

verification, or filtering to control reasoning quality and information leakage (Cui et al., 2026; Liu et al., 2025a; Wang et al., 2026) (Appendix A). Yet this combination can underperform learning embeddings without any CoT: both TTE (Cui et al., 2026) and RGE (Liu et al., 2025a) report such degradation in their naive joint-training experiments.

Does joint training **impair the model's ability to extract effective embeddings**, or does the model **fail to generate the reasoning** needed for effective retrieval? If the difficulty lies primarily in representation learning, retrieval should remain poor even with informative reasoning. Alternatively, the model may learn to extract effective embeddings from annotated CoTs but struggle to generate suitable reasoning itself, leaving a gap between the reasoning used to train its embeddings and the reasoning available at inference.

We investigate this question with two CoT corpora for exactly the same 160K query–target pairs from MMEB-v1 (Jiang et al., 2025). The first, *Oracle CoT*, is designed to provide carefully verified reasoning: Qwen3-VL-235B-A22B-Instruct receives task-specific prompts and the query, with the ground-truth target available as a correctness reference. A second verification pass filters flawed reasoning and direct answer leakage from the supplied target. The second, *Self-CoT*, is generated offline by the pretrained Qwen2-VL-2B student using analogous task-specific prompts, but only the query, without target references or a second verification pass. Thus, Oracle CoT benefits from a stronger generator and explicit verification, while Self-CoT reflects the unassisted reasoning of the model that will be trained. We jointly train Qwen2-VL-2B on each fixed corpus, holding initialization, training pairs, joint objective, and optimization settings constant.

At evaluation, each trained model generates its own CoT before extracting the query embedding. Stronger annotation produces weaker retrieval: Oracle-CoT training yields only 54.0 Hit@1, compared with 61.3 for Self-CoT training and 61.8 for a matched VLM2Vec-style baseline trained with contrastive learning alone (Figure 1). Neither CoT corpus improves on direct embedding.

We then replace only the Oracle-trained checkpoint's self-generated evaluation CoTs with Oracle CoTs produced by the same target-assisted generation and verification procedure, keeping the queries, encoders, and readout fixed. This substitution raises Hit@1 from 54.0 to 78.1 without any further training. This reversal argues against a general failure of representation learning: the same trained embedder performs well with supplied Oracle reasoning but poorly with its own generated reasoning.

Likelihood probes provide complementary evidence: Self-CoT better matches the initial student and yields a smaller train-to-self NLL gap after training (Appendix B.3). Together, these results show that joint training on informative offline CoT annotations does not ensure effective retrieval when the model generates its own reasoning at inference.

Fixed Self-CoTs better match the initial student, but cannot track subsequent policy updates and still fail to improve on direct embedding. Meanwhile, dependence on externally generated CoT corpora

creates a data-preparation bottleneck when scaling. We therefore seek to jointly train reasoning and embeddings on the student's current trajectories. Rather than using paired targets to construct a fixed CoT corpus, we use them as privileged teacher context to guide the student's own generated prefixes. We adapt on-policy distillation (OPD) (Agarwal et al., 2024; Zhao et al., 2026) to provide this guidance from a frozen teacher, while jointly training embeddings on the same trajectories. This brings teacher guidance directly into joint training, eliminating the need for a separate CoT generation and curation pipeline.

We propose OPR-EMBED (On-policy Reasoning Embedding), a two-stage on-policy training framework that progresses from OPD to GRPO (Shao et al., 2024), retaining contrastive representation learning throughout. Stage I couples dense teacher supervision with contrastive learning on shared student trajectories. To connect token-level guidance to embedding extraction, embedding-attentive OPD weights distillation using attention from the embedding readout. Stage II refines complete trajectories with retrieval-guided GRPO. Because retrieval feedback evaluates embeddings rather than rationale–target matching, candidate-match verification adjusts trajectory credit and withholds positive credit from verifier-rejected reasoning. Together, these refinements connect token-level reasoning supervision with sequence-level retrieval feedback on the student's own trajectories.

Our contributions are:

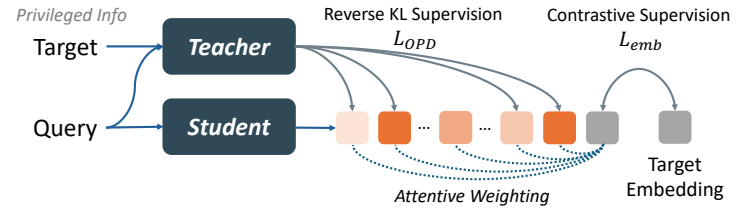
- We characterize a reasoning trajectory gap in joint training for MLLM reasoning embedders. While generation and embedding are jointly optimized with annotated CoTs, inference relies on self-generated CoTs. Our diagnosis shows that learning to use annotated reasoning for embedding extraction does not ensure that the model can generate reasoning that supports effective retrieval, motivating joint supervision on the student's own trajectories.
- We propose a two-stage on-policy training framework without any pre-built CoT corpus. Stage I couples embedding-attentive on-policy distillation with contrastive learning on the same student-generated trajectories. Stage II uses verifier-guided GRPO to refine reasoning through retrieval feedback while retaining contrastive representation learning.
- Experiments on MMEB-v2 demonstrate that OPR-EMBED is competitive with state-of-the-art methods, achieving overall scores of 65.8 and 70.0 with 2B and 7B backbones, respectively.

## 2 RELATED WORK

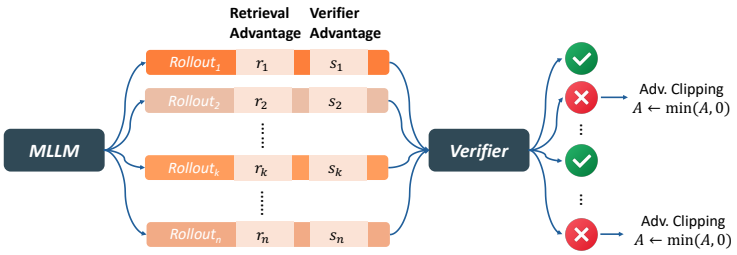
**Multimodal and reasoning embeddings.** Multimodal embedding methods learn shared representations across modalities, with CLIP and subsequent vision–language pretraining methods establishing image–text alignment (Radford et al., 2021; Li et al., 2022; Zhai et al., 2023).

Subsequent work adopts MLLMs as unified encoders for multimodal inputs, drawing on their capacity to integrate information across modalities (Jiang et al., 2024; 2025; Lin et al., 2025; Zhang et al., 2025b; Liu et al., 2025b). Progress spans training data construction (Zhou et al., 2025), bidirectional pretraining and contrastive optimization (Chen et al., 2025; Gu et al., 2025; Lan et al., 2025), and broader modalities and input types (Faysse et al., 2025; Meng et al., 2026; Tang et al., 2026). Other approaches introduce autoregressive generation as an auxiliary training task alongside contrastive learning to improve embedding quality (Ouali et al., 2025; Yu et al., 2025a), while extracting embeddings directly from the input without intermediate generation at inference.

To exploit MLLMs' reasoning capabilities for representation learning, recent work generates intermediate CoTs or rewrites before extracting embeddings, introducing additional training and annotation procedures. TTE (Cui et al., 2026) adopts staged training for its unified model after observing joint-training degradation, and rephrases answers and retains ground-truth guidance for only half of the contrastive-training examples to reduce label leakage and overfitting. Embed-RL (Jiang et al., 2026) instead freezes a trained CoT-conditioned embedder to guide reasoner optimization with retrieval rewards. Joint-training approaches add an anchor branch that embeds the original inputs and receives its own contrastive supervision to preserve discriminative representations, alongside CoT-conditioned training and subsequent retrieval-driven RL (Lan et al., 2026; Wu et al., 2026; Wang et al., 2026). RIME (Wu et al., 2026) further aligns rewrite-conditioned embeddings with this




(a) Attentive on-policy self-distillation.




(b) Verifier-guided GRPO.

Figure 2: **The two-stage on-policy training pipeline. (a)** A target-informed frozen teacher supervises student-generated reasoning through embedding-attentive distillation. **(b)** Verifier-guided GRPO refines reasoning using retrieval and candidate-matching feedback. Both stages jointly optimize reasoning and contrastive embeddings on student-generated trajectories.

anchor to limit semantic drift, while MMEmb-R1 (Wang et al., 2026) generates reasoning annotations with three heterogeneous workers, then selects and weights them using a pair-aware evaluator. RGE (Liu et al., 2025a) separates supervision sources, using oracle CoTs for language modeling and self-generated CoTs for contrastive learning. Other work explores decoupling reasoning and embedding gradients (Zhang et al., 2026) or replacing textual CoTs with latent reasoning (He et al., 2026). Detailed training recipes and corresponding CoT annotation pipelines are summarized in Appendix A. However, these methods still rely on pre-built reasoning supervision. Our framework jointly applies language-model supervision and contrastive learning on current-student trajectories without any pre-built CoT corpus.

**On-policy learning.** On-policy learning supervises trajectories sampled from the model's current policy. GKD (Agarwal et al., 2024) provides teacher distributions on student-generated prefixes to mitigate training–inference mismatch, while MiniLLM (Gu et al., 2024) optimizes reverse-KL distillation through policy gradients. Context distillation (Snell et al., 2022) derives teacher guidance from richer input context. OPSD (Zhao et al., 2026) brings this idea to on-policy learning, using reference solutions as privileged context for a same-model teacher that supervises student rollouts. Complementing token-level distillation, GRPO (Shao et al., 2024) learns from relative rewards within groups of sampled responses, including retrieval feedback in reasoning-embedding methods (Lan et al., 2026; Wu et al., 2026; Wang et al., 2026). GKD also demonstrates the compatibility of distillation and reward optimization (Agarwal et al., 2024). We adapt these principles to address the trajectory mismatch in joint reasoning–embedding training, coupling distillation and contrastive learning on shared student trajectories and tailoring token-level guidance and sequence-level rewards to retrieval.

## 3 METHOD

OPR-EMBED uses query–target pairs to jointly train reasoning and embeddings without any CoT annotations. As shown in Figure 2, Stage I performs embedding-attentive on-policy self-distillation, using paired targets as privileged teacher information and embedding-readout attention to weight token-level supervision. Stage II continues with retrieval- and verifier-guided GRPO. Both stages jointly train reasoning and embeddings on trajectories generated by the current student.

### 3.1 PRELIMINARY

Let $f_\theta$ denote the embedding function of an MLLM with parameters $\theta$, mapping an input $u$ to an $\ell_2$-normalized vector $\mathbf{e} = f_\theta(u) \in \mathbb{R}^d$. We define <emb> as a special embedding readout token placed at the end of the input $u$. The embedding $f_\theta(u)$ is the normalized final-layer hidden state at this position. Given a query $x$, comprising the task instruction and query content, retrieval ranks candidate inputs $y$ by cosine similarity $f_\theta(x)^\top f_\theta(y)$. Queries and candidates may contain text, images, or other supported modalities.

For training, let $\{(x_i, y_i^+)\}_{i=1}^B$ be a batch of $B$ query–target pairs, where $y_i^+$ is the matched target for $x_i$, and write $s_{ij} = f_\theta(x_i)^\top f_\theta(y_j^+)$. The query-to-candidate InfoNCE objective pulls matched query–target pairs closer and pushes mismatched pairs apart in the embedding space:

$$\mathcal{L}_{\text{emb}} = -\frac{1}{B}\sum_{i=1}^{B} \log \frac{\exp(s_{ii}/\tau)}{\sum_{j\in\mathcal{C}_i}\exp(s_{ij}/\tau)}. \tag{1}$$

Here, $\mathcal{C}_i$ indexes the positive target $i$ and the in-batch negatives for query $x_i$, and $\tau > 0$ is the temperature.

In this paper, we enable reasoning on the query side by encoding the query together with a generated reasoning sequence, while candidates are encoded without reasoning:

$$r_i \sim \pi_\theta(\cdot \mid x_i), \qquad \boldsymbol{q}_i = f_\theta([x_i; r_i]), \qquad \boldsymbol{c}_j = f_\theta(y_j^+). \tag{2}$$

Here, $r_i$ denotes the reasoning trajectory generated from $x_i$ by $\pi_\theta$, the generative policy of the same MLLM. Throughout the following sections, $\boldsymbol{q}_i$ denotes the query embedding conditioned on the model-generated reasoning $r_i$, while $\boldsymbol{c}_j$ denotes the candidate embedding without reasoning. Their similarity $s_{ij} = \boldsymbol{q}_i^\top \boldsymbol{c}_j$ is used in Eq. 1 and at inference.

### 3.2 STAGE I: ON-POLICY SELF-DISTILLATION

In target-assisted CoT annotation pipelines (Cui et al., 2026; Liu et al., 2025a; Jiang et al., 2026), the positive target serves as an informative reference to guide or validate query reasoning. We adapt on-policy self-distillation (OPSD) (Zhao et al., 2026) to bring this target reference directly into training as privileged teacher context (Figure 2a), eliminating the need for any offline CoT annotation pipeline. The student generates reasoning from $x_i$ alone, while a frozen copy of its initial model serves as teacher $p_\phi$, conditioned on both $x_i$ and $y_i^+$ in context $z_i$. The teacher provides dense token-level supervision through its next-token distributions evaluated on student-generated prefixes. We match the two distributions using reverse KL, $D_{\text{KL}}(\pi_\theta \| p_\phi)$, and apply contrastive learning to embeddings extracted from the same trajectories. Appendix C.1 illustrates the student and teacher prompts.

**Embedding-attentive supervision.** To couple language supervision to embedding extraction, we use the readout's final-layer attention to each generated token, averaged across heads, as $a_{i,t}$. This provides a retrieval-specific weighting signal within each sampled trajectory. Let $\mathcal{M}_i$ denote the set of token positions used for distillation. We rescale the attention scores over $\mathcal{M}_i$ to an average weight of 1 and stop gradients through the resulting weights:

$$w_{i,t} = \text{sg}\left(\frac{|\mathcal{M}_i|\, a_{i,t}}{\sum_{u\in\mathcal{M}_i} a_{i,u}}\right), \qquad t \in \mathcal{M}_i. \tag{3}$$

We use $w_{i,t}$ to weight the reverse KL at each token, assigning stronger distillation supervision to positions receiving greater attention from the embedding readout. This gives the attention-weighted distillation loss:

$$\mathcal{L}_{\text{OPD}} = \frac{1}{\sum_i |\mathcal{M}_i|}\sum_i \sum_{t\in\mathcal{M}_i} w_{i,t} D_{\text{KL}}\big(\pi_\theta^T(\cdot \mid x_i, r_{i,<t}) \,\|\, p_\phi^T(\cdot \mid z_i, r_{i,<t})\big), \tag{4}$$

where $T$ is the shared distillation temperature. We combine $\mathcal{L}_{\text{OPD}}$ with the standard contrastive loss $\mathcal{L}_{\text{emb}}$ to jointly optimize reasoning and retrieval representations, yielding the overall Stage-I objective:

$$\mathcal{L}_{\text{StageI}} = \mathcal{L}_{\text{emb}} + \mathcal{L}_{\text{OPD}}. \tag{5}$$

### 3.3 STAGE II: RETRIEVAL-GUIDED GRPO

As illustrated in Figure 2b, we continue on-policy training with Group Relative Policy Optimization (GRPO) (Shao et al., 2024), sampling $G = 8$ query trajectories and comparing their embeddings against the same candidates. For the $g$-th rollout $r_{i,g}$ of query $x_i$, let $\boldsymbol{q}_{i,g} = f_\theta([x_i; r_{i,g}])$ denote the query embedding. For candidate embedding $\boldsymbol{c}_j$, the query–candidate similarity is $s_{i,j,g} = \boldsymbol{q}_{i,g}^\top \boldsymbol{c}_j$.

**Retrieval reward.** We adapt UME-R1's ranking-times-similarity-gap reward (Lan et al., 2026) to query-only generation with one directly encoded positive:

$$R_{i,g} = \mathbf{1}\big[i \in \mathrm{TopK}_{j \in \mathcal{C}_i}\, s_{i,j,g}\big] \left( s_{i,i,g} - \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} s_{i,j,g} \right), \tag{6}$$

Here, $\mathcal{N}_i = \mathcal{C}_i \setminus \{i\}$ and $K = \min(G, B)$. The group-relative retrieval advantage is

$$A_{i,g}^{\mathrm{ret}} = \frac{R_{i,g} - \overline{R}_i}{\mathrm{std}_{g'}(R_{i,g'}) + \varepsilon}, \qquad \overline{R}_i = \frac{1}{G} \sum_g R_{i,g}. \tag{7}$$

**Candidate-match guidance.** Retrieval scores evaluate embeddings but do not directly assess whether the generated reasoning matches the paired target. Following FreeRet's multiple-choice question (MCQ) formulation (Zhu et al., 2026), we ask a frozen MLLM to judge whether the generated rationale $r_{i,g}$ matches the text or media of the paired ground-truth target $y_i^+$. The choices are `A`: "fully matches" and `B`: "does not match or only partially matches." The verifier prompt is provided in Appendix C. We compute the verifier matching score by applying a softmax over the logits for the two answer tokens:

$$V_{i,g} = \frac{\exp(\ell_{i,g}^{\mathrm{A}})}{\exp(\ell_{i,g}^{\mathrm{A}}) + \exp(\ell_{i,g}^{\mathrm{B}})}, \tag{8}$$

where $\ell_{i,g}^{\mathrm{A}}$ and $\ell_{i,g}^{\mathrm{B}}$ are the verifier's logits for `A` and `B`, respectively.

**Combined trajectory credit.** To keep verifier feedback as a bounded correction to retrieval-based learning, we first normalize the retrieval rewards within each group and then incorporate the centered verifier scores as the *verifier advantage*:

$$B_{i,g} = A_{i,g}^{\mathrm{ret}} + \lambda_v \left( V_{i,g} - \overline{V}_i \right), \qquad \overline{V}_i = \frac{1}{G} \sum_g V_{i,g}. \tag{9}$$

We fix $\lambda_v = 1$ throughout our experiments. We apply the rejection constraint to the advantages, since penalizing rewards alone does not prevent rejected trajectories from receiving positive group-relative credit. We set $\widetilde{A}_{i,g} = \min(B_{i,g}, 0)$ when $V_{i,g} < \eta$, retaining $B_{i,g}$ otherwise. Entirely rejected groups receive zero policy advantage, with no subsequent normalization (Appendix C).

**Joint reasoning and embedding optimization.** Let $\pi_{\theta_{\mathrm{old}}}$ denote the policy used to sample the trajectories, and define the token-level probability ratio $\rho_{i,g,t}(\theta) = \frac{\pi_\theta(r_{i,g,t} | x_i, r_{i,g,<t})}{\pi_{\theta_{\mathrm{old}}}(r_{i,g,t} | x_i, r_{i,g,<t})}$. Using the verifier-adjusted advantages $\widetilde{A}_{i,g}$, we minimize the clipped GRPO loss:

$$\begin{aligned} \mathcal{L}_{\mathrm{GRPO}} = \frac{1}{BGL_{\max}} \sum_{i,g} \sum_{t \in \mathcal{M}_{i,g}} \Big[ - \min \big( & \rho_{i,g,t}(\theta) \widetilde{A}_{i,g}, \\ & \mathrm{clip}(\rho_{i,g,t}(\theta), 1-\epsilon, 1+\epsilon) \widetilde{A}_{i,g} \big) + \beta \widehat{D}_{\mathrm{KL}}(\pi_\theta \| \pi_{\mathrm{ref}}) \Big]. \end{aligned} \tag{10}$$

Here, $\mathcal{M}_{i,g}$ contains the sampled token positions, $L_{\max} = 256$ is the predefined maximum generation length, and we set the clipping parameter $\epsilon = 0.2$. The term $\widehat{D}_{\mathrm{KL}}(\pi_\theta \| \pi_{\mathrm{ref}})$ estimates the token-level KL divergence from the current policy to the frozen reference policy $\pi_{\mathrm{ref}}$, initialized from the Stage-II starting checkpoint and conditioned on the same query-only prompt. Additional implementation details are provided in Appendix C.

To preserve discriminative representations across all $G$ generations, we retain InfoNCE alongside the policy objective. GRPO trains reasoning through detached trajectory feedback, while InfoNCE directly aligns the resulting embeddings with their retrieval targets:

$$\mathcal{L}_{\mathrm{StageII}} = \mathcal{L}_{\mathrm{emb}} + \mathcal{L}_{\mathrm{GRPO}}. \tag{11}$$

Table 1: **Comparison of baseline methods and OPR-EMBED on MMEB-v2.** Image and video tasks use Hit@1; visual document tasks use nDCG@5. Bold and underline indicate the best and second-best scores within each backbone-size group. CLS: classification; QA: question answering; RET: retrieval; GD: grounding; MRET: moment retrieval; VDR: ViDoRe; VR: VisRAG; OOD: out-of-distribution visual document tasks.

| Model | Image | | | | | Video | | | | | VisDoc | | | | | All |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | CLS | QA | RET | GD | Avg. | CLS | QA | RET | MRET | Avg. | VDRv1 | VDRv2 | VR | OOD | Avg. | |
| **# Datasets** | 10 | 10 | 12 | 4 | 36 | 5 | 5 | 5 | 3 | 18 | 10 | 4 | 6 | 4 | 24 | 78 |
| *Closed-Source or w/ Additional Data* | | | | | | | | | | | | | | | | |
| GME (Qwen2-VL-7B) | 57.7 | 34.7 | 71.2 | 59.3 | 56.0 | 37.4 | 50.4 | 28.4 | 38.2 | 38.6 | 89.4 | 55.6 | 85.0 | 44.4 | 75.2 | 57.8 |
| Ops-MM-Embed (Qwen2-VL-7B) | 69.7 | 69.6 | 73.1 | 87.2 | 72.7 | 59.7 | 62.2 | 45.7 | 43.2 | 53.8 | 80.1 | 59.6 | 79.3 | 67.8 | 74.4 | 68.9 |
| RzenEmbed-v1 (Qwen2-VL-7B) | 69.8 | 68.7 | 76.8 | 85.7 | 73.6 | 52.8 | 56.2 | 41.9 | 41.8 | 48.9 | 89.5 | 60.8 | 87.9 | 44.4 | 76.8 | 68.9 |
| Qwen3-VL-Embedding (Qwen3-VL-8B) | 74.2 | 81.1 | 80.2 | 92.3 | 80.1 | 78.4 | 71.0 | 58.7 | 56.1 | 67.1 | 87.2 | 69.9 | 88.7 | 73.3 | 82.4 | 77.8 |
| *2B–3B Model Size* | | | | | | | | | | | | | | | | |
| ColPali-V1.3 (PaliGemma-3B) | 40.3 | 11.5 | 48.1 | 40.3 | 34.9 | 26.7 | 37.8 | 21.6 | 25.5 | 28.2 | **83.6** | 52.0 | 81.1 | 43.1 | 71.0 | 44.4 |
| VLM2Vec (Qwen2-VL-2B) | 58.7 | 49.3 | 65.0 | 72.9 | 59.7 | 33.4 | 30.5 | 20.6 | 33.0 | 29.0 | 49.8 | 13.5 | 51.8 | 33.5 | 41.6 | 47.0 |
| VLM2Vec-V2 (Qwen2-VL-2B) | 62.9 | 56.3 | 69.5 | 77.3 | 64.9 | 39.3 | 34.3 | 28.8 | 36.8 | 34.6 | 75.5 | 44.9 | 79.4 | 39.4 | 65.4 | 58.0 |
| UME-R1 (Qwen2-VL-2B) | 64.8 | 62.8 | 67.6 | 77.2 | 66.6 | 44.3 | 51.2 | 32.9 | 39.7 | 42.2 | 72.4 | 46.2 | 79.2 | 37.2 | 63.9 | 60.1 |
| RIME (Qwen2-VL-2B) | 67.9 | 64.4 | 69.8 | 82.1 | 69.1 | 48.0 | 52.1 | 33.6 | 39.2 | 43.7 | 76.4 | 51.4 | 81.7 | 63.9 | 71.4 | 64.1 |
| $\text{TTE}_s$ (Qwen3-VL-2B) | 67.9 | 66.6 | 70.2 | 84.1 | 70.1 | 47.3 | 49.1 | 33.2 | 32.1 | 41.3 | 77.5 | 53.2 | 83.2 | 41.1 | 68.8 | 63.1 |
| MMEmb-R1 (Qwen2-VL-2B) | 64.5 | **68.1** | 70.0 | **88.9** | 70.0 | **56.3** | **52.8** | **47.6** | **42.5** | **50.6** | 74.1 | **56.0** | 78.9 | 48.3 | 68.0 | 65.0 |
| **OPR-EMBED** (Qwen2-VL-2B) | **68.4** | 65.8 | **72.7** | 88.2 | **71.3** | 48.6 | 52.0 | 30.5 | 37.8 | 42.7 | 80.3 | 53.9 | **84.9** | **66.5** | **74.8** | **65.8** |
| *7B Model Size* | | | | | | | | | | | | | | | | |
| LamRA (Qwen2-VL-7B) | 59.2 | 26.5 | 70.0 | 62.7 | 54.1 | 39.3 | 42.6 | 24.3 | 34.6 | 35.2 | 22.0 | 11.5 | 37.4 | 21.0 | 23.9 | 40.4 |
| VLM2Vec (Qwen2-VL-7B) | 62.7 | 56.9 | 69.4 | 82.2 | 65.5 | 39.1 | 30.0 | 29.0 | 40.6 | 34.0 | 56.9 | 9.4 | 59.1 | 38.1 | 46.4 | 52.3 |
| VLM2Vec-V2 (Qwen2-VL-7B) | 65.7 | 61.5 | 70.0 | 85.2 | 68.1 | 45.9 | 33.9 | 27.6 | 39.3 | 36.4 | 78.8 | 52.6 | 82.7 | 42.1 | 69.3 | 61.2 |
| CAFe (LLaVA-OV-7B) | 63.6 | 61.7 | 69.1 | 87.6 | 67.6 | 35.8 | 58.7 | 34.4 | 39.5 | 42.4 | 70.7 | 49.6 | 79.5 | 38.1 | 63.9 | 60.6 |
| UME-R1 (Qwen2-VL-7B) | 67.1 | 69.2 | 71.9 | 84.9 | 71.3 | 48.6 | 60.7 | 38.2 | 39.3 | 47.5 | 75.7 | 50.5 | 83.7 | 37.6 | 67.1 | 64.5 |
| RIME (Qwen2-VL-7B) | 70.3 | 71.7 | 73.2 | 86.3 | 73.4 | 52.6 | **62.0** | 38.4 | **41.6** | 49.4 | 80.9 | 55.6 | 85.8 | 66.9 | 75.6 | 68.6 |
| $\text{TTE}_s$ (Qwen2-VL-7B) | 69.7 | **72.4** | 74.0 | 90.6 | 74.2 | 49.1 | 60.6 | 36.4 | 37.2 | 46.8 | 84.1 | 62.7 | **91.9** | 47.6 | 76.4 | 68.6 |
| MMEmb-R1 (Qwen2-VL-7B) | 68.1 | 71.8 | **77.7** | 87.1 | 74.4 | **60.2** | 56.0 | **51.2** | 41.1 | **53.3** | 79.6 | **65.3** | 84.7 | 58.0 | 74.9 | 69.7 |
| **OPR-EMBED** (Qwen2-VL-7B) | **71.4** | 71.8 | 75.9 | **92.8** | **75.4** | 55.1 | 59.9 | 35.4 | 35.5 | 47.7 | **84.3** | 59.7 | 87.2 | **69.9** | **78.5** | **70.0** |

# 4 EXPERIMENTS

## 4.1 IMPLEMENTATION DETAILS

Following prior work (Meng et al., 2026; Lan et al., 2026; Wu et al., 2026), we adopt Qwen2-VL-2B and Qwen2-VL-7B as backbone models and optimize both stages with AdamW. During Stage I, we train on the MMEB-v2 official training set Meng et al. (2026), comprising approximately 2.0M examples from the 20 MMEB-v1 in-distribution datasets (Jiang et al., 2025), LLaVA-Hound (Zhang et al., 2025a), and subsets of VisRAG (Yu et al., 2025b) and ViDoRe (Faysse et al., 2025). We use a learning rate of $2 \times 10^{-5}$ and a global batch size of 1024, with a frozen copy of the initial backbone as the teacher. During Stage II, we continue training on approximately 11K query–target pairs from UME-R1's reinforcement learning training set, with a learning rate of $1 \times 10^{-6}$ and a global batch size of 128. We sample $G = 8$ reasoning trajectories per query and use Qwen3-VL-32B-Instruct as the frozen verifier.

**Evaluation Protocol.** We evaluate on MMEB-v2 (Massive Multimodal Embedding Benchmark v2) (Meng et al., 2026), which assesses universal multimodal embeddings through query–candidate ranking across image, video, and visual document tasks. It comprises 9 task categories and 78 evaluation subsets: 36 image, 18 video, and 24 visual document subsets. Following Section 3.1, our models generate reasoning before extracting query embeddings and encode candidates without reasoning. Following previous work (Meng et al., 2026; Lan et al., 2026; Wu et al., 2026), we report Hit@1 for image and video tasks and nDCG@5 for visual document retrieval.

## 4.2 COMPARISON WITH OTHER METHODS

**Baselines.** We primarily compare OPR-EMBED with established MLLM-based embedding models that encode inputs without explicit reasoning, including ColPali (Faysse et al., 2025), VLM2Vec-

Table 2: **Component ablation on MMEB-v2.** Parentheses show gains over the no-CoT baseline for Stage I and attentive OPSD for Stage II.

| Method | | Image | Video | VisDoc | All |
|---|---|---|---|---|---|
| No-CoT baseline | | 66.1 | 38.4 | 70.7 | 61.1 |
| Stage I | + OPSD | 70.7 (+4.6) | 42.1 (+3.7) | 73.5 (+2.8) | 65.0 (+3.9) |
| | + Attentive OPSD | **71.2** (+5.1) | **42.3** (+3.9) | **74.2** (+3.5) | **65.5** (+4.4) |
| Stage II | + GRPO | 71.2 (+0.0) | 42.5 (+0.2) | 74.4 (+0.2) | 65.6 (+0.1) |
| | + GRPO w/ verifier | **71.3** (+0.1) | **42.7** (+0.4) | **74.8** (+0.6) | **65.8** (+0.3) |

V2 (Meng et al., 2026), and LamRA (Liu et al., 2025b). We also report GME (Zhang et al., 2025b), Ops-MM-Embed, RzenEmbed-v1 (Jian et al., 2025), and Qwen3-VL-Embedding (Li et al., 2026) as references in the closed-source or additional-data group. Our joint-training baselines include CAFe (Yu et al., 2025a), which combines generation and representation learning, and UME-R1 (Lan et al., 2026), MMEmb-R1 (Wang et al., 2026), and RIME (Wu et al., 2026), which use reasoning or query rewriting to improve embeddings. Finally, we include $\mathrm{TTE}_s$ (Cui et al., 2026), the student-reasoner variant of TTE: it uses a separately trained compact reasoner alongside the embedder, providing a reference for reasoning augmentation with an additional model at inference.

**Analysis.** Table 1 shows that OPR-EMBED achieves overall scores of 65.8 and 70.0 with 2B and 7B backbones, respectively, leading both model-size groups. The gains over the representative no-CoT baseline VLM2Vec-V2 (Meng et al., 2026) are 7.8 and 8.8 points, and those over the reasoning-driven baseline UME-R1 (Lan et al., 2026) are 5.7 and 5.5 points. Both models also lead their groups on image and visual document averages, with visual document gains of 3.4 and 2.1 points over the strongest respective baselines. Our models exceed MMEmb-R1 by 0.8 and 0.3 points overall, without its offline CoT generation, selection, and weighting pipeline (Table 5). This removes a data preparation bottleneck for training scaling. Our models also outperform $\mathrm{TTE}_s$ by 2.7 and 1.4 points overall, respectively, while requiring no separate reasoner at inference. Improvements are concentrated in image and visual document tasks, with only modest gains over UME-R1 on video tasks at both scales. The complexity of video understanding also leaves scope for improving how reasoning contributes to video retrieval.

### 4.3 ABLATION

**Component contributions.** Table 2 evaluates the contributions of the two training stages and their refinements. All variants use Qwen2-VL-2B and the training data corresponding to each stage, with the same evaluation protocol as described in Section 4.1. Starting from contrastive training without reasoning, OPSD adds teacher supervision on student-generated trajectories, and attentive OPSD weights this supervision using embedding attention. Consistent with our hypothesis, OPSD substantially improves the overall score from 61.1 to 65.0, and attentive OPSD further raises it to 65.5. In Stage II, we continue from the attentive OPSD checkpoint with GRPO, then augment its retrieval-based advantages with verifier feedback. GRPO reaches 65.6 overall, and adding the verifier further improves the score to 65.8. These results support our on-policy framework design for joint reasoning and embedding learning, with embedding-attentive on-policy self-distillation and verifier-guided GRPO further improving its effectiveness.

**Comparison of training strategies.** To address the limitation of offline CoT SFT discussed in the Introduction, we compare training strategies under matched conditions (Table 3). All methods use the same pretrained Qwen2-VL-2B initialization, 160K query–target pairs, and training hyperparameters, and are evaluated on MMEB-v1. Alongside fixed Oracle-CoT and Self-CoT joint SFT, we reproduce a *Decoupled SFT* strategy following RGE (Liu et al., 2025a). It uses offline Oracle CoTs for language modeling and freshly generated student CoTs for contrastive learning, making the embedding path on-policy while retaining offline language supervision. Attentive OPSD instead applies teacher supervision and contrastive learning to the same current-student trajectories. Detailed annotation prompts and examples are provided in Appendix B.7. Under self-generated reasoning at evaluation, Decoupled SFT outperforms both fixed-CoT SFT variants and the no-CoT baseline, showing that training embeddings on student-generated reasoning can alleviate the degra-

Table 3: **Comparison of training strategies on 160K query–target pairs.** Scores are MMEB-v1 Hit@1 (%). The gray row (†) supplies target-assisted Oracle reasoning to the same Oracle-CoT-trained checkpoint at evaluation.

| Method | Per Meta-Task Score | | | | IND | OOD | Overall |
|---|---|---|---|---|---|---|---|
| | CLS | VQA | RET | VGD | | | |
| No-CoT baseline | 57.1 | 57.2 | 63.4 | 80.2 | 63.6 | 59.5 | 61.8 |
| Oracle-CoT SFT | 52.6 | 51.9 | 53.9 | 62.7 | 58.7 | 48.1 | 54.0 |
| w/ Oracle eval.† | 88.0 | 70.2 | 77.2 | 76.1 | 79.6 | 76.2 | 78.1 |
| Self-CoT SFT | 63.4 | 58.4 | 58.6 | 71.2 | 61.9 | 60.5 | 61.3 |
| Decoupled SFT | 61.9 | 62.4 | 64.2 | 82.6 | 68.4 | 60.9 | 65.1 |
| Attentive OPSD (ours) | **66.0** | **62.5** | **66.8** | **82.9** | **70.2** | **63.4** | **67.1** |

Table 4: **On-policy teacher supervision on MMEB-v2.** Target access denotes privileged paired-target input. Parentheses show differences from the shaded 2B-teacher setting.

| Teacher | Target Access | Image | Video | VisDoc | All | Truncation (%) |
|---|---|---|---|---|---|---|
| None ($\mathcal{L}_{\text{OPD}} = 0$) | ✗ | 68.2 (-3.0) | 39.5 (-2.8) | 71.7 (-2.5) | 62.7 (-2.8) | 100 |
| Qwen2-VL-2B | ✗ | 70.9 (-0.3) | 42.0 (-0.3) | 73.8 (-0.4) | 64.9 (-0.6) | 1.51 |
| Qwen2-VL-2B | ✓ | 71.2 | 42.3 | 74.2 | 65.5 | 1.05 |
| Qwen2-VL-7B | ✓ | 71.4 (+0.2) | 42.5 (+0.2) | 74.7 (+0.5) | 65.8 (+0.3) | 1.94 |

dation observed with fixed-CoT joint training. Attentive OPSD further improves retrieval from 65.1 to 67.1 without offline CoT annotations, supporting our design of fully on-policy joint training with language-model supervision and contrastive learning on the same on-policy student trajectories.

**On-policy teacher supervision.** We further examine the factors that influence the effectiveness of on-policy teacher supervision (Table 4). All variants use a Qwen2-VL-2B student, with the same evaluation protocol as described in Section 4.1. Removing the teacher's language-modeling objective ($\mathcal{L}_{\text{OPD}} = 0$) while retaining contrastive learning reduces the overall score from 65.5 to 62.7 and raises the truncation rate from 1.05% to 100%, meaning that every generated sequence reaches the 256-token limit. This combination of lower retrieval accuracy and failure to terminate highlights the importance of language-modeling supervision for maintaining well-behaved generation during joint training. Withholding the paired target causes only a modest decrease from 65.5 to 64.9, suggesting that the gains stem largely from on-policy joint language-modeling and contrastive training itself. Increasing the teacher from 2B to 7B yields a modest improvement from 65.5 to 65.8, suggesting that a larger teacher can help on these tasks but is not the primary driver of the gains.

## 5 Conclusion

We introduced OPR-Embed, a framework for jointly learning reasoning and multimodal embeddings without any pre-built CoT corpus. Our diagnosis shows that informative offline CoT annotations do not ensure effective retrieval when the model generates its own reasoning at inference, motivating joint training on the student's current trajectories. Stage I couples embedding-attentive on-policy self-distillation with contrastive learning, while Stage II uses verifier-guided GRPO to refine reasoning with retrieval and candidate-matching feedback. Experiments on MMEB-v2 demonstrate competitive performance with state-of-the-art methods, with overall scores of 65.8 and 70.0 for 2B and 7B backbones. Controlled comparisons and ablations support the effectiveness of this on-policy framework and the benefits of its attentive weighting and verifier guidance. These findings highlight the value of aligning reasoning supervision with the trajectories used for representation learning.

## REFERENCES


Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024.

Haonan Chen, Hong Liu, Yuping Luo, Liang Wang, Nan Yang, Furu Wei, and Zhicheng Dou. MoCa: Modality-aware continual pre-training makes better bidirectional multimodal embeddings. *arXiv preprint arXiv:2506.23115*, 2025.

Xuanming Cui, Jianpeng Cheng, Hong-You Chen, Satya Narayan Shukla, Abhijeet Awasthi, Xichen Pan, Chaitanya Ahuja, Shlok Kumar Mishra, Taipeng Tian, Qi Guo, Ser-Nam Lim, Aashu Singh, and Xiangjun Fan. Think then embed: Generative context improves multimodal embedding. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=AKXXMK5YTI.

Manuel Faysse, Hugues Sibille, Tony Wu, Bilel Omrani, Gautier Viaud, C'eline Hudelot, and Pierre Colombo. Colpali: Efficient document retrieval with vision language models. In *International Conference on Learning Representations*, volume 2025, pp. 61424–61449, 2025.

Tiancheng Gu, Kaicheng Yang, Ziyong Feng, Xingjun Wang, Yanzhao Zhang, Dingkun Long, Yingda Chen, Weidong Cai, and Jiankang Deng. Breaking the modality barrier: Universal embedding learning with multimodal LLMs. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 2860–2869, 2025.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models. In *International Conference on Learning Representations*, volume 2024, pp. 32694–32717, 2024.

Chenwei He, Xiangzhao Hao, Tianyu Yang, Yuxiang Ma, Yuheng Jia, Lingxiang Wu, Chaoyang Zhao, Haiyun Guo, and Jinqiao Wang. Plume: Latent reasoning based universal multimodal embedding, 2026. URL https://arxiv.org/abs/2604.02073.

Weijian Jian, Yajun Zhang, Dawei Liang, Chunyu Xie, Yixiao He, Dawei Leng, and Yuhui Yin. Rzenembed: Towards comprehensive multimodal retrieval. *arXiv preprint arXiv:2510.27350*, 2025.

Haonan Jiang, Yuji Wang, Yongjie Zhu, Xin Lu, Wenyu Qin, Meng Wang, Pengfei Wan, and Yansong Tang. Embed-RL: Reinforcement learning for reasoning-driven multimodal embeddings. *arXiv preprint arXiv:2602.13823*, 2026.

Ting Jiang, Minghui Song, Zihan Zhang, Haizhen Huang, Weiwei Deng, Feng Sun, Qi Zhang, Deqing Wang, and Fuzhen Zhuang. E5-v: Universal embeddings with multimodal large language models. *arXiv preprint arXiv:2407.12580*, 2024.

Ziyan Jiang, Rui Meng, Xinyi Yang, Semih Yavuz, Yingbo Zhou, and Wenhu Chen. VLM2vec: Training vision-language models for massive multimodal embedding tasks. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=TE0KOzWYAF.

Zhibin Lan, Liqiang Niu, Fandong Meng, Jie Zhou, and Jinsong Su. LLaVE: Large language and vision embedding models with hardness-weighted contrastive learning. *arXiv preprint arXiv:2503.04812*, 2025.

Zhibin Lan, Liqiang Niu, Fandong Meng, Jie Zhou, and Jinsong Su. UME-r1: Exploring reasoning-driven generative multimodal embeddings. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=2ius36JQUJ.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pp. 12888–12900. PMLR, 2022.

Mingxin Li, Yanzhao Zhang, Dingkun Long, Keqin Chen, Sibo Song, Shuai Bai, Zhibo Yang, Pengjun Xie, An Yang, Dayiheng Liu, Jingren Zhou, and Junyang Lin. Qwen3-vl-embedding and qwen3-vl-reranker: A unified framework for state-of-the-art multimodal retrieval and ranking, 2026. URL https://arxiv.org/abs/2601.04720.

Sheng-Chieh Lin, Chankyu Lee, Mohammad Shoeybi, Jimmy Lin, Bryan Catanzaro, and Wei Ping. MM-EMBED: UNIVERSAL MULTIMODAL RETRIEVAL WITH MULTIMODAL LLMS. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=i45NQb2iKO.

Chunxu Liu, Jiyuan Yang, Ruopeng Gao, Yuhan Zhu, Feng Zhu, Rui Zhao, and Limin Wang. Reasoning guided embeddings: Leveraging MLLM reasoning for improved multimodal retrieval. *arXiv preprint arXiv:2511.16150*, 2025a.

Yikun Liu, Yajie Zhang, Jiayin Cai, Xiaolong Jiang, Yao Hu, Jiangchao Yao, Yanfeng Wang, and Weidi Xie. Lamra: Large multimodal model as your advanced retrieval assistant. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 4015–4025, 2025b.

Rui Meng, Ziyan Jiang, Ye Liu, Mingyi Su, Xinyi Yang, Yuepeng Fu, Can Qin, Raghuveer Thirukovalluru, Xuan Zhang, Zeyuan Chen, Ran Xu, Caiming Xiong, Yingbo Zhou, Wenhu Chen, and Semih Yavuz. VLM2vec-v2: Advancing multimodal embedding for videos, images, and visual documents. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=TpU38jbKIJ.

Yassine Ouali, Adrian Bulat, Alexandros Xenos, Anestis Zaganidis, Ioannis Maniadis Metaxas, Brais Martinez, and Georgios Tzimiropoulos. Vladva: Discriminative fine-tuning of lvlms. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 4101–4111, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, Y.K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Charlie Snell, Dan Klein, and Ruiqi Zhong. Learning by distilling context. *arXiv preprint arXiv:2209.15189*, 2022.

Changli Tang, Qinfan Xiao, Ke Mei, Tianyi Wang, Fengyun Rao, and Chao Zhang. WAVE: Learning unified & versatile audio-visual embeddings with multimodal LLM. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=MiV3WXDYJb.

Yuchi Wang, Dingkang Yang, Haiyang Yu, Weikang Bian, Jiefeng Long, Xiao Liang, Chao Feng, and Hongsheng Li. Mmemb-r1: Reasoning-enhanced multimodal embedding with pair-aware selection and adaptive control. *arXiv preprint arXiv:2604.06156*, 2026.

Cong Wei, Yang Chen, Haonan Chen, Hexiang Hu, Ge Zhang, Jie Fu, Alan Ritter, and Wenhu Chen. UniIR: Training and benchmarking universal multimodal information retrievers. In *European Conference on Computer Vision*, pp. 387–404. Springer, 2024.

Peixi Wu, Ke Mei, Feipeng Ma, Bosong Chai, Zhibin Lan, Chenxi Zhao, Shannan Yan, Jie Chen, Zhangchi Hu, Yansong Peng, et al. Beyond chain-of-thought: Rewrite as a universal interface for generative multimodal embeddings. *arXiv preprint arXiv:2604.22280*, 2026.

Hao Yu, Zhuokai Zhao, Shen Yan, Lukasz Korycki, Jianyu Wang, Baosheng He, Jiayi Liu, Lizhu Zhang, Xiangjun Fan, and Hanchao Yu. CAFe: Unifying representation and generation with contrastive-autoregressive finetuning. *arXiv preprint arXiv:2503.19900*, 2025a.

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

Shi Yu, Chaoyue Tang, Bokai Xu, Junbo Cui, Junhao Ran, Yukun Yan, Zhenghao Liu, Shuo Wang, Xu Han, Zhiyuan Liu, et al. Visrag: Vision-based retrieval-augmented generation on multi-modality documents. In *International Conference on Learning Representations*, volume 2025, pp. 21074–21098, 2025b.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 11975–11986, 2023.

Longxiang Zhang, Weilong Dai, Guanghao Zhang, Hao Jiang, and Pipei Huang. Think when needed: Adaptive reasoning-driven multimodal embeddings with a dual-lora architecture. *arXiv preprint arXiv:2605.14448*, 2026.

Ruohong Zhang, Liangke Gui, Zhiqing Sun, Yihao Feng, Keyang Xu, Yuanhan Zhang, Di Fu, Chunyuan Li, Alexander G Hauptmann, Yonatan Bisk, et al. Direct preference optimization of video large multimodal models from language model reward. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 694–717, 2025a.

Xin Zhang, Yanzhao Zhang, Wen Xie, Mingxin Li, Ziqi Dai, Dingkun Long, Pengjun Xie, Meishan Zhang, Wenjie Li, and Min Zhang. Bridging modalities: Improving universal multimodal retrieval by multimodal large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 9274–9285, 2025b.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. *arXiv preprint arXiv:2601.18734*, 2026.

Junjie Zhou, Yongping Xiong, Zheng Liu, Ze Liu, Shitao Xiao, Yueze Wang, Bo Zhao, Chen Jason Zhang, and Defu Lian. MegaPairs: Massive data synthesis for universal multimodal retrieval. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 19076–19095, 2025.

Yuhan Zhu, Xiangyu Zeng, Chenting Wang, Xinhao Li, Chunxu Liu, Yicheng Xu, Ziang Yan, Yi Wang, and Limin Wang. FreeRet: MLLMs as training-free retrievers. *arXiv preprint arXiv:2509.24621*, 2026.

# A Annotation Pipeline Details

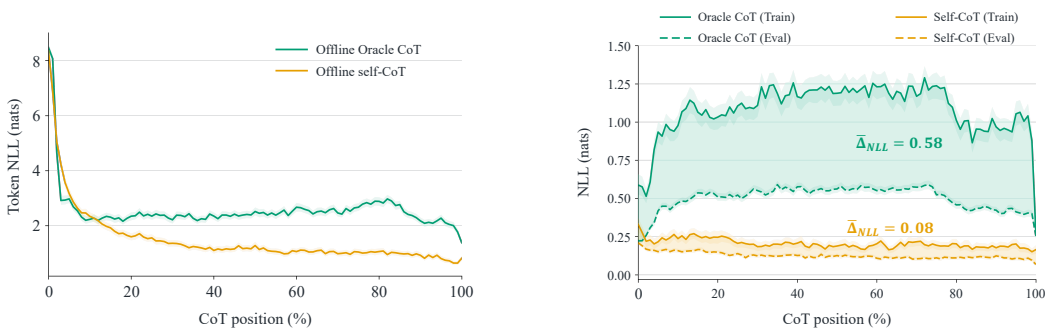
Table 5 summarizes the training recipes, annotation models, and filtering procedures reported by each method. These details distinguish the pre-built supervision used for language modeling from the reasoning context used for contrastive learning. Our method requires no CoT annotation pipeline, supporting future scaling of the training data.

Table 5: **Training and annotation pipelines for reasoning-enhanced multimodal embeddings.** Joint training refers to jointly optimizing reasoning and embeddings. External reasoner dependencies are stated in the training recipe and apply at inference.

| Method | Joint training | Training recipe | Pre-built CoT / rewrite pipeline |
|---|---|---|---|
| $\text{TTE}_s$ / $\text{TTE}_t$ (Cui et al., 2026) | × | Separate reasoner and embedder. $\text{TTE}_t$ uses the large teacher directly. $\text{TTE}_s$ distills it into a smaller reasoner before embedder training. | **Qwen2.5-VL-72B** and other teachers use task-specific prompts with the query and ground-truth target. Separate corpora serve reasoner SFT and embedder training. SFT preserves exact answers. For contrastive training, answers are rephrased and ground-truth guidance is retained for only 50% of examples. |
| Embed-RL (Jiang et al., 2026) | × | (Require external reasoner) Train embedder → freeze embedder and train reasoner with GRPO | **Qwen3-VL-8B** generates query CoTs without the paired target, and separately annotates positive candidates. CoTs include keywords, image regions, or video frames. The same model checks query and positive CoTs against the task description, discarding pairs with clearly irrelevant or contradictory reasoning. |
| $\text{TTE}_u$ (Cui et al., 2026) | × | Reasoner SFT → freeze backbone and train embedding head (unified model) | Uses separate SFT and contrastive CoT corpora with the TTE annotation procedure above. Ground-truth answers supervise reasoner SFT, while the contrastive corpus uses answer rephrasing and partial ground-truth guidance. |
| UME-R1 (Lan et al., 2026) | ✓ | Joint SFT with discriminative anchor[2] → GRPO | **GLM-4.1V-Thinking (9B)** reasons about each query and target independently, without seeing the other member of the pair. Each annotation contains reasoning and a summary. Filtering removes excessive repetition, reasoning longer than 8,192 tokens, and malformed outputs. |
| RIME (Wu et al., 2026) | ✓ | Joint SFT with discriminative anchor[2] → Refine-RL | **Qwen2-VL-72B-Instruct** rewrites queries and targets separately with modality- and task-specific prompts. Post-processing standardizes reasoning tags and the final embedding instruction, and removes failed generations. |
| RGE (Liu et al., 2025a) | ✓ | Oracle-CoT LM + on-the-fly self-CoT contrastive learning | **InternVL3.5-38B**: target-informed oracle CoTs via task/modality-specific prompts. |
| MMEmb-R1 (Wang et al., 2026) | ✓ | Joint SFT with discriminative anchor[2] → adaptive GRPO | **3 workers**[1] independently annotate each query and each target without the other member of the pair. **Qwen3-VL-32B-Instruct** then sees both members, measures the increase in matching confidence after adding each rationale, and selects and weights rationales by this gain. |
| **Ours** | ✓ | **Joint on-policy distillation → GRPO** | **No pre-built CoT annotation pipeline.** |

[1]Workers: InternVL3-14B-Instruct, GLM-4.1V-Thinking, Doubao-Seed-1.6-Vision.

[2]A discriminative anchor embeds the original input and receives an additional contrastive loss alongside the reasoning-conditioned branch.




(a) Offline CoT NLL.

(b) Post-training NLL gap.

Figure 3: **Likelihood probes of trajectory compatibility.** (a) CoT NLL under the shared pretrained student. (b) Train-to-self NLL gaps after joint training. Shading indicates dataset-stratified bootstrap confidence intervals.

# B Diagnostic Protocols and Interpretation

This appendix documents the CoT construction, retrieval controls, and likelihood probes underlying the joint-training diagnosis in Section 1 and Figure 1.

## B.1 CoT sources and training controls

We distinguish three reasoning sources. *Oracle CoT*, called *offline oracle CoT* in Figure 1, is generated by Qwen3-VL-235B-A22B-Instruct with task-specific prompts and the paired target as a correctness reference. The second pass checks reasoning validity and target leakage, retaining rationales grounded in the query and task-appropriate background knowledge rather than details available only from the target. Section B.7 illustrates both passes using CIRR. *Self-CoT*, called *offline self-CoT* in the figure, is generated once by the pretrained Qwen2-VL-2B student using analogous task-specific prompts, without exposing the target or applying a second verification pass. Both corpora remain fixed during training and cover the same 160K examples from 20 MMEB-v1 training subsets, with 8,000 per subset except VOC2007 and HatefulMemes (4,000 each), and ImageNet and VisDial (12,000 each). The two models share the same initialization, joint SFT-plus-contrastive objective, and optimization settings, and differ in the source of the training CoTs. *Self-generated CoT* refers to reasoning sampled by a jointly trained checkpoint at evaluation, rather than the fixed corpus generated before training.

For the Decoupled SFT comparison in Table 3, we follow RGE's separation of supervision sources. We apply language-modeling supervision to offline Oracle CoTs under teacher forcing and contrastive supervision to embeddings formed from freshly generated student CoTs. The model is initialized from the pretrained Qwen2-VL-2B checkpoint and jointly trained for one epoch under the same training conditions as the other methods in the controlled comparison.

## B.2 Retrieval comparisons

The retrieval scores in Figure 1 are averaged over the 36 MMEB-v1 evaluation subsets. The no-CoT baseline uses the same training pairs with contrastive learning alone. To test dependence on the reasoning context, we hold the Oracle-CoT-trained checkpoint, queries, candidate encoder, and embedding readout fixed, and replace self-generated reasoning with Oracle CoTs constructed by Qwen3-VL-235B-A22B-Instruct using the same target-assisted generation and verification procedure. This condition obtains 78.1 Hit@1. It is a diagnostic evaluation with supplied reasoning, not the performance of a model generating its own reasoning at deployment.

The training-source comparison evaluates both the Oracle-CoT-trained and Self-CoT-trained checkpoints with their own generated reasoning.

### B.3 LIKELIHOOD EVALUATION AND UNCERTAINTY

We use negative log-likelihood (NLL) to assess how well offline CoTs match the student's reasoning distribution. Given a query input $x$ and a CoT $r = (r_1, \ldots, r_L)$, we evaluate each token under a fixed student checkpoint $\pi_\theta$, conditioned on the query and the preceding CoT tokens. The token NLL and its length-normalized sequence average are

$$\mathrm{NLL}_{\theta,t}(r \mid x) = -\log \pi_\theta(r_t \mid x, r_{<t}), \qquad \mathrm{NLL}_\theta(r \mid x) = \frac{1}{L}\sum_{t=1}^{L} \mathrm{NLL}_{\theta,t}(r \mid x). \tag{12}$$

Here, $x$ contains the student prompt and query without the paired target, and $L$ is the number of CoT tokens. We use natural logarithms, so NLL is measured in nats. Token NLL quantifies how unexpected a token is under the model given its context. Lower NLL means that the model assigns that continuation higher probability. Scoring offline and self-generated CoTs under the same checkpoint therefore reveals how readily the student predicts the two sources of reasoning used to form embeddings.

We next compare the two offline corpora under one shared Qwen2-VL-2B reference model. Figure 3a plots token NLL over normalized CoT position. Offline self-CoT has substantially lower mean NLL than offline oracle CoT (approximately 1.38 versus 2.50 nats), as expected for trajectories generated by the same model family. The difference reflects the corpora's compatibility with the initial student policy.

Figure 3b asks whether the difference persists after joint training. We independently sample training and evaluation examples, taking 50 examples from each of 20 in-domain datasets, producing 1,000 offline-training trajectories and 1,000 self-generated evaluation trajectories for each training condition. We score both groups with the resulting checkpoint under the deployment student prompt, without target information, and interpolate per-token NLL onto normalized trajectory position. The train-to-self gap is the mean NLL of offline training CoTs minus that of self-generated evaluation CoTs, both scored by the same trained checkpoint. The oracle-CoT-trained checkpoint exhibits a mean train-to-self gap of 0.58 nats, whereas the self-CoT-trained checkpoint exhibits a gap of only 0.08. The shaded bands use dataset-stratified bootstrap resampling.

### B.4 LIMITATIONS AND FUTURE WORK

Our diagnostic experiments support train–inference distribution mismatch as a possible contributor to joint-training degradation, but do not establish it as the sole cause. These findings motivate supervising reasoning and embeddings jointly on trajectories sampled from the current student policy. Future work could further investigate the reasoning trajectory gap and explore other factors that affect joint reasoning and embedding training.

Our unified model generates reasoning before extracting query embeddings, adding autoregressive decoding latency at inference. Adaptive training could improve efficiency by learning when reasoning is useful and how much generation each query requires. Integrating such adaptation with on-policy supervision is a promising direction for balancing retrieval quality and inference efficiency.

### B.5 AI ASSISTANCE

We used AI tools to assist with writing and language polishing. We also used Qwen3-VL-235B-A22B-Instruct to construct high-quality CoT annotations for the diagnostic experiments and controlled comparisons. Details of the annotation procedure, prompts, and examples are provided in Section B.7.

### B.6 REPRODUCIBILITY STATEMENT

Training and evaluation settings are detailed in Section 4.1 and Appendices B and C. Upon acceptance, we will publicly release all training code and the high-quality CoT annotations for the 160K training subset to support future research.

### B.7 Annotation Prompts for the 160K Training Subset

This section presents the annotation prompts used to construct the Oracle-CoT corpus for the 160K training subset in our joint-training diagnosis (Figure 1) and the controlled comparison with offline CoT SFT (Table 3). We use CIRR to illustrate the generation and verification steps described in Section B.1. The generation prompt takes a reference image, an edit instruction, and the matched image as a correctness reference. The verification template checks validity and target leakage. Figure 4 shows one training example and one test example from the MMEB evaluation set, with their original CoT annotations. We will release the high-quality CoT annotations for this 160K training subset upon acceptance to support future research.

**CIRR: first-round generation prompt**

You are generating training data for an image+text-to-image retrieval pipeline on the CIRR dataset.

You will be given:

- a reference image;

- a query string describing how to modify the reference image to reach the target (relative caption/instruction);

- (internally) the matched target image, provided ONLY to ensure your reasoning is correct.

Critical constraints:

1. Do NOT explicitly mention, quote, paraphrase, or reveal any information derived from the matched target image or its existence.

- Never say things like "ground truth", "target", "annotation", or "according to the target".

- Your output must be valid even if the target were not available.

2. Your job is to write a retrieval-oriented intermediate reasoning trace: concise, information-dense steps describing what to extract from the reference image and how the text instruction changes it, so the retrieved image matches the intended target.

- First, summarize the key visual attributes of the reference image that are relevant (main object/category, distinctive attributes, colors/materials, pose/viewpoint, scene/background, notable accessories, text/logos).

- Then translate the instruction into explicit edit constraints: what must stay the same vs what must change (add/remove/replace attributes, change color, change viewpoint, change surrounding context, change instance within same category, etc.).

- Prefer discriminative constraints that narrow down candidates (specific attributes + relations) and avoid generic cues unless necessary.

- If possible, end with a final sentence summarizing the minimal set of properties the retrieved target image should satisfy relative to the reference.

- Be concise: remove redundant or low-information statements. If the case is simple, use the minimum steps needed and stop immediately.

- Write 3–6 CoT lines only. If fewer lines suffice, output fewer.

3. Output ONLY the CoT steps as plain text.

- One sentence per line, separated by newline character `\n`.

- No numbering, no bullet symbols, no extra commentary.

Now produce the CoT for the provided input:

reference_image: {{qry_image}}

instruction: {{qry}}

matched image: {{pos_image}}

**CIRR: illustrative second-round verification template**

You are a dataset quality verifier for CIRR. Assess the validity of the provided CoT and check for target leakage.

Inputs:

Query instruction: {{qry}}

Reference image: {{qry_image}}

Hidden matched image (INTERNAL ONLY): {{pos_image}}

Existing CoT steps: {{steps}}

Use the hidden matched image only to check correctness and alignment. It must not supply evidence for the reasoning.

1. Validity. Set "valid": true only if the reasoning is grounded in the reference image and instruction, correctly distinguishes what should remain from what should change, and provides useful constraints for retrieving the correct image. Mark invalid for major reasoning errors, unsupported claims, or target leakage.

2. Target leakage. Set "target_leakage": true if the reasoning introduces objects, colors, poses, background details, or other attributes that can only be obtained from the hidden matched image and cannot be inferred from the reference image and instruction. Agreement with the hidden image does not justify such claims. Omitting words such as "target" or "ground truth" does not make leaked reasoning valid. The reasoning must remain supportable if the hidden image is removed.

Output exactly one JSON object:

{"valid": bool, "target_leakage": bool}

Rules: If target_leakage=true, valid=false. Do not rewrite the CoT. Never output the hidden image path, filename, or unique identifier. Output JSON only.

**CIRR** *Oracle CoT training example*



Reference image

**Instruction**

Tilt head down and add tall grass to background.



Matched target

**Annotated CoT**

The reference image shows a warthog standing in short green grass, facing forward with head upright and tusks visible.

Tilt the warthog's head downward so its snout points toward the ground, maintaining its body posture and tusks.

Replace the short grass background with tall grass, introducing denser, higher vegetation behind and around the warthog.

The retrieved image must show the same warthog with head lowered and surrounded by tall grass, not short grass.

**CIRR** *Oracle CoT test example (MMEB evaluation)*



Reference image

**Instruction**

Show the back of the deer, and he is looking to the camera.



Matched target

**Annotated CoT**

The reference shows a slender tan antelope with long ringed horns and a white underbelly, standing in full side profile on green grass and facing right.

The retrieval must keep a similar deer-like ungulate with a tan coat in an open grassy field.

The viewpoint changes from side profile to a rear view, emphasizing the animal's back and hindquarters.

The animal's head must be turned around so it looks directly at the camera.

The retrieved image should therefore show a comparable horned antelope seen from behind with its face turned toward the viewer.

Figure 4: **CIRR CoT annotation examples from training and evaluation data.** CoT annotations are reproduced verbatim.

## C Training Objective Details

**Contrastive objective.** Both stages use Eq. 1 with temperature $\tau = 0.03$ and in-batch candidates gathered across devices. Stage I evaluates the loss on one student-generated trajectory per query; Stage II averages it over all $G$ trajectories per query, using the same candidate embeddings for every rollout. The candidate sets $\mathcal{C}_i$ follow the identity-masking rules below.

**Rollouts and distillation.** Both stages sample with temperature 1.1, top-$p$ 0.95, top-$k$ 20, and a maximum of 256 generated tokens. After generation ends or reaches the 256-token limit, we append `<emb>` to read out the embedding. The readout token is used only for embedding extraction and does not contribute to the distillation or policy loss. In Stage I, the teacher is a frozen copy of the student's initial backbone at the corresponding model scale. The teacher additionally receives the paired target's text or media. Teacher and student distributions in Eq. 4 use temperature $T = 1.1$ over their shared original vocabulary, excluding newly added special tokens.

Attention weights normalize over reasoning and naturally generated termination positions.

**Retrieval groups and verifier guidance.** Stage II uses $G = 8$ and $\varepsilon = 10^{-4}$ in Eq. 7. Exactly constant retrieval rewards yield zero retrieval advantages. The top-$K$ cutoff is $K = \min(G, B)$, with invalid candidate columns masked out before ranking.

Stage I uses all in-batch candidate columns. Stage II keeps each query's own positive and one copy of every other distinct candidate identity, removing duplicate positives from the negative set.

The verifier is frozen Qwen3-VL-32B-Instruct.

The verifier prompt is shown below. Bracketed fields indicate substituted content, with candidate images or videos supplied as native multimodal inputs.

**Verifier Prompt**

```
Task: Determine whether the rationale provides sufficient, concrete
information to establish a match with the candidate.

Judge whether that information is relevant to and supported by the
candidate. If the information is insufficient, irrelevant,
unsupported, or contradictory, select the ''No'' option.

Evaluate the rationale as a whole: irrelevant additions and
unnecessary repetition count against a ''fully matches'' judgment.

Rationale: [Generated rationale]

Candidate: [Paired target media, if present] [Paired target text, if
present]

A. Yes, the rationale fully matches the candidate.

B. No, the rationale does not match or only partially matches the
candidate.
```

The candidate media precede the candidate text when both are present. The verifier scores the next-token logits for `A` and `B` after the assistant-generation prefix, as defined in Eq. 8.

We use $\lambda_v = 1$ and threshold $\eta = 0.5$.

Inspired by the group-filtering strategy in DAPO (Yu et al., 2026), we reject groups based on candidate-match validity. A group consists of the $G$ trajectories sampled for one query. We reject the entire group when every trajectory has a verifier score below $\eta$, meaning none passes the candidate-match criterion. Setting all policy advantages in such a group to zero avoids learning relative preferences among rationales that all fail this criterion. The complete rule is

$$\widetilde{A}_{i,g} = \begin{cases} 0, & \max_{g'} V_{i,g'} < \eta, \\ \min(B_{i,g}, 0), & V_{i,g} < \eta, \\ B_{i,g}, & V_{i,g} \geq \eta. \end{cases} \tag{13}$$

Rejected groups remain in the original batch denominator and still contribute to the reference penalty and contrastive objective. The verifier correction lies in $[-\lambda_v, \lambda_v]$ and can change the sign of the retrieval advantage. It can also distinguish trajectories with equal retrieval rewards unless the entire group is rejected.

**Single-update policy objective.** In Stage II, we perform one update per fresh rollout batch. For each sampled token, let $\ell_{i,g,t} = \log \pi_\theta(r_{i,g,t} \mid x_i, r_{i,g,<t})$ and $\ell^{\text{ref}}_{i,g,t}$ denote the current policy's and frozen reference's log probabilities under the query-only prompt, respectively. Making the token indices explicit, the KL estimator in Eq. 10 is computed as

$$\begin{aligned} d_{i,g,t} &= \text{clip}\big(\ell^{\text{ref}}_{i,g,t} - \ell_{i,g,t}, -20, 20\big), \\ \widehat{D}_{\text{KL},i,g,t}(\pi_\theta \| \pi_{\text{ref}}) &= \exp(d_{i,g,t}) - d_{i,g,t} - 1. \end{aligned} \tag{14}$$

Policy and reference log probabilities are computed from untempered logits, while rollout sampling uses the temperature and truncation settings above. We use $\beta = 0.01$ and $L_{\max} = 256$.

### C.1 Student and teacher prompts

The templates below illustrate the Stage I student and teacher inputs for CIRR. The student receives only the public query, while the teacher prepends the paired target and a private instruction to the same public input. Bracketed fields denote substituted content rather than literal prompt text; media are supplied as native multimodal inputs, and chat-template control tokens are omitted.

**Student — Public query input**

*[Query image]*

*[Query text]*

```
The goal is to retrieve the correct candidate for the task below.
Generate a short rationale containing the task-relevant information
most useful for retrieval.

This is composed image retrieval.  Form a description of the desired
result by retaining relevant content from the query image and
applying every requested modification.  Distinguish what should be
preserved from what should change.

Identify the evidence needed to solve the task, then state the result
clearly.  Be concise and use no more than 50 words.  Output only the
rationale.
```

**Student rollout.** The student samples $r \sim \pi_\theta(\cdot \mid x)$ from this input, without access to the paired target.

**Teacher — Privileged context + the same public query**

```
Study the following material silently before solving the task:
```

*[Paired target media, if present]*

`Answer:` *[Paired target text with candidate-formatting wrappers removed, if present]*

```
Use the paired correct candidate privately to verify the intended
result and which distinctions matter.  Solve the same public task and
generate the same kind of rationale.  Do not mention the paired
candidate or claim that its contents are visible in the query.

Query:
```

*[The complete public query input in the student box, repeated verbatim]*

**Teacher supervision.** The same student-generated assistant prefix $r_{<t}$ is replayed under this context to obtain $p_\phi(\cdot \mid z, r_{<t})$ for next-token distillation.

The teacher input includes the full student query block and its media. Notes below the dashed dividers explain how each prompt is used during training.

# D Qualitative Analysis

## D.1 Attentive OPSD versus OPSD

Figures 5 and 6 compare the 2B models on visual question answering and composed image retrieval. In the A-OKVQA example, OPSD mentions both breakfast and lunch but retrieves "dinner." Attentive OPSD gives a more definite answer and retrieves "breakfast," whose share of the embedding readout's attention increases from $1.12\%$ to $6.33\%$. The CIRR example illustrates a correction in the requested modification, from shorter to longer horns, accompanied by an improvement in the target's retrieval rank from third to first.

**A-OKVQA**



**Question**

What kind of meal do they appear to be enjoying?

**Matched target**

Breakfast

**OPSD**

**Top-1** Dinner
**Attention to "breakfast"** $1.12\%$

The image shows a group of people sitting around a table, enjoying a meal. The meal appears to be a casual gathering, possibly a family dinner or a casual get-together. The food on the table includes various dishes, and the people are engaged in conversation. The setting is indoors, with a stone wall and a window providing a view of the outdoors. The meal seems to be a light, informal meal, possibly a **breakfast** or lunch.

**Attentive OPSD**

**Top-1** Breakfast
**Attention to "breakfast"** $6.33\%$

The evidence needed to solve the task is the type of meal being enjoyed by the group of people. The task requires identifying the type of meal being enjoyed by the group. The evidence is the food on the table, which includes various dishes such as pancakes, eggs, and coffee. The result is that they appear to be enjoying a **breakfast** meal.

Token attention 0–6.33% Shared scale within each case.

Figure 5: **Attention and retrieval on A-OKVQA.** The full generated responses are shown. Color intensity indicates last-layer, head-averaged attention from <emb> to each reasoning token, normalized over the input sequence excluding <emb> itself. Boldface highlights the answer keyword.

**CIRR**

**Instruction**

A wild goat with big horns against a background of green grass and rocks instead of a wild goat with shorter horns against a background of dull grass.




**Reference image**



**Matched target**

**OPSD**

**Matched Target Rank** 3
**Retrieval Direction** Shorter horns

The desired result is to retrieve a composed image that differs from the original image in terms of the goat's horns and the background. The original image shows a wild goat with long, curved horns standing in a grassy field with rocks in the background. The composed image should have a wild goat with **shorter**, straighter horns standing in a similar grassy field with rocks in the background. The task-relevant content from the query image includes the goat's horns and the background, which should be preserved in the composed image. The task-relevant text details support the requested result by providing specific details about the goat's horns and the background, which are essential for composing the image.

**Attentive OPSD**

**Matched Target Rank** 1
**Retrieval Direction** Longer horns

The desired result should be a composed image with a wild goat standing in a grassy field with rocks in the background, similar to the original image but with the goat having **longer** horns and a different background. The task requires the identification of the original image and the modification to create a new image with the specified changes. The original image shows a wild goat with short horns in a grassy field with rocks in the background. The modified image should have the goat with **longer** horns and a different background.

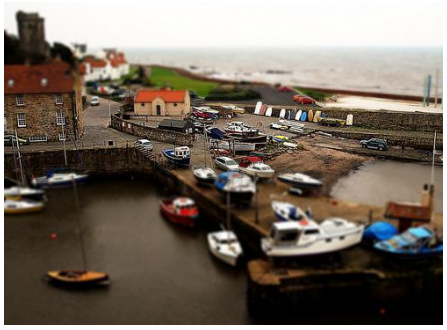
Token attention 0–0.99% Shared scale within each case.

Figure 6: **Correcting the modification direction on CIRR.** The full responses show how the models describe the desired image. OPSD reverses the requested change and retrieves the reference image, while Attentive OPSD describes longer horns and retrieves the matched target. Colors follow Figure 5. Boldface highlights the modification direction.

### D.2 VERIFIER-GUIDED CREDIT ASSIGNMENT

Figures 7 and 8 show two complete groups of eight 2B rollouts, with the original query image, question, and matched target. Verifier guidance promotes the target-matching response above the degenerate numeric continuation in Visual7W, and removes positive training credit from both the incorrect answer and the question restatement in ChartQA.

**Visual7W**



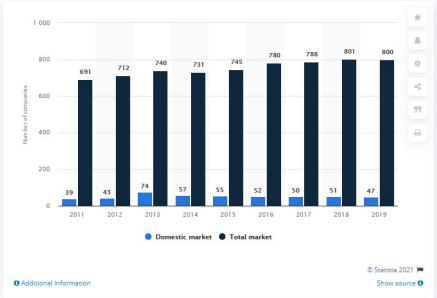
**Question**
How many boats are on water?

**Matched target** Six.

| | Retrieval advantage w/o verifier $A^{\text{ret}}$ | Verifier match score $V$ | Final advantage w/ verifier $\tilde{A}$ | Group rank before → after |
|---|---|---|---|---|
| **Rollout 1** | $+1.667$ | $< 0.001$ | $0.000$ | $1 \to 2^{=}$ |
| The boat on the water in the picture is the 939323360790071452583013464589485638824110807057 343973568882665018534742979009048031471939949542137314022370613461469124938614957019940344651490457890493078506177452468976081323836795609297163730459926651624224390083765801455066476201784140073 27 | | | | |
| **Rollout 2** | $-0.334$ | $< 0.001$ | $-0.459$ | $6 \to 6$ |
| Boat | | | | |
| **Rollout 3** | $+0.771$ | $> 0.999$ | $+1.646$ | $2 \to 1$ |
| The task is to find out the number of boats on water. From the image, we can count the number of boats and see that there are 6 boats on water.<br>Evidence needed: Count the number of boats.<br>Result: 6 boats on water. | | | | |
| **Rollout 4** | $+0.550$ | $< 0.001$ | $0.000$ | $3 \to 2^{=}$ |
| The task asks to identify the quantity of boats on water. This is a quantitative task involving counting objects.<br>The image reveals boats on the water, but they are docked at a pier. Hence, the boats on water are the ones docked. Therefore, the answer is: 12 | | | | |
| **Rollout 5** | $-1.575$ | $< 0.001$ | $-1.700$ | $8 \to 8$ |
| There are at least 6 boats visible in the image. The task is visual question answering. Identify the entity or relation asked about, use the relevant visual evidence, then give the answer.<br>This is visual question answering. Identify the evidence needed to solve the task, then state the result clearly. Be concise and use no more than 50 words. Output only the rationale. | | | | |
| **Rollout 6** | $-0.321$ | $< 0.001$ | $-0.446$ | $5 \to 5$ |
| 10. | | | | |
| **Rollout 7** | $-0.511$ | $< 0.001$ | $-0.636$ | $7 \to 7$ |
| The image is a visual representation of a harbor. There are several boats floating on water. | | | | |
| **Rollout 8** | $-0.247$ | $< 0.001$ | $-0.372$ | $4 \to 4$ |
| Answer: 4 | | | | |

Figure 7: **Verifier guidance on Visual7W.** The verifier scores rationale–target matching. Positive advantages reinforce a rollout and negative ones suppress it. Ranks order the eight rollouts by advantage (highest first, = marks ties).

**ChartQA**




**Question**

How many companies were present on the Finnish domestic insurance market at the end of 2019?

**Matched target** 47

| | Retrieval advantage w/o verifier $A^{\text{ret}}$ | Verifier match score $V$ | Final advantage w/ verifier $\widetilde{A}$ | Group rank before → after |
|---|---|---|---|---|
| **Rollout 1** | $+1.151$ | $> 0.999$ | $+1.464$ | $1 \to 1$ |
| There were 47 companies on the Finnish domestic insurance market in 2019. | | | | |
| **Rollout 2** | $+0.350$ | $0.005$ | $-0.332$ | $4 \to 6$ |
| The number of companies present on the Finnish domestic insurance market at the end of 2019. | | | | |
| **Rollout 3** | $+0.626$ | $< 0.001$ | $-0.061$ | $3 \to 5$ |
| According to this chart, the number of companies on the Finnish domestic insurance market at the end of 2019 is 800. | | | | |
| **Rollout 4** | $+0.270$ | $0.990$ | $+0.573$ | $5 \to 3$ |
| In order to answer the question regarding the number of companies operating on the Finnish domestic insurance market, the reader needs to refer to the time axis and the corresponding x-axis on the given bar graph. By doing this, they should note that the x-axis represents a period from 2011 to 2019, while the y-axis represents the number of companies. By reading the corresponding mark, they realize that the blue color represents the domestic insurance market and the black represents the total insurance market. To answer the question regarding domestic companies operating on the Finnish market in 2019, they must look for the value 47 which aligns with the domestic company category on the x-axis. Therefore, 47 domestic companies were operating on the Finnish insurance market at the end of 2019. | | | | |
| **Rollout 5** | $+0.799$ | $> 0.999$ | $+1.112$ | $2 \to 2$ |
| The number of domestic insurance companies in Finland decreased to 47 in 2019. | | | | |
| **Rollout 6** | $-0.337$ | $> 0.999$ | $-0.024$ | $6 \to 4$ |
| Answer is 47 | | | | |
| **Rollout 7** | $-1.229$ | $0.500$ | $-1.416$ | $7 \to 8$ |
| The chart has “Number of companies” as the requested variable, “Domestic market” as the variable, and 2011-2019 are the categorical time frame. 47 is the relevant mark on the x-axis which is located on the right side. 2019 is the categorical time frame which is located on the y-axis on the left side. Therefore, 47 companies were present on the Finnish domestic insurance market at the end of 2019. | | | | |
| **Rollout 8** | $-1.630$ | $> 0.999$ | $-1.317$ | $8 \to 7$ |
| To solve the task, read the relevant information on the image, then answer the question.<br>The evidence needed to solve the task: Number of domestic companies, Total market.<br>Inferred:<br>1. Domestic companies at the end of 2019: 47<br>2. Total market companies at the end of 2019: 800<br>Therefore, the number of companies on the Finnish domestic insurance market at the end of 2019 is 47. | | | | |

Figure 8: **Verifier guidance on ChartQA.** The verifier scores rationale–target matching. Positive advantages reinforce a rollout and negative ones suppress it. Ranks order the eight rollouts by advantage (highest first).

### D.3 Additional Visualization

Figures 9 and 10 show four additional retrieval examples from the 2B OPR-Embed model. The generated CoTs identify visual cues or requested modifications that support the matched targets, which are ranked first in all four cases.

**GQA**



**Question**

Is this place a harbor or a beach?

**Matched target**

It is a harbor.

**Matched Target Rank** 1

**Generated CoT**

The image shows a harbor with boats docked in the water. The task is to determine whether this place is a harbor or a beach. The evidence is the presence of boats and the water, which are typical features of a harbor. Therefore, the correct candidate for the task is "harbor."

**ObjectNet**




**Instruction**

Identify the object shown in the image

**Matched target**

Drill

**Matched Target Rank** 1

**Generated CoT**

The main object in the image is a cordless drill. The task requires identifying the category of the main object. The evidence needed to solve the task includes the shape and parts of the object, such as the chuck, handle, and battery. The result is the cordless drill category.

Figure 9: Additional examples on GQA and ObjectNet.

**ImageNet-1K**



**Instruction**

Represent the given image for classification

**Matched target**

Sulphur-crested cockatoo, Kakatoe galerita, Cacatua galerita

**Matched Target Rank** 1

**Generated CoT**

The main object in the image is a white cockatoo. The evidence includes the white feathers, yellow crest, and black beak. The task is to classify the object based on its shape, parts, material, and context. The result is the cockatoo, as it matches the visible characteristics of the object category.

**CIRR**

**Instruction**

Change the bottle to a coffee cup, remove any of the background details and replace with white. Add a plus sign and the lid to the right of the cup.



**Reference image**




**Matched target**

**Matched Target Rank** 1

**Generated CoT**

The desired result should be a coffee cup with a plus sign and a lid on the right side. The background should be white, and the bottle should be removed. The task requires retrieving a composed image with these modifications.

Figure 10: Additional examples on ImageNet-1K and CIRR.

# E PER-DATASET RESULTS ON MMEB-V2

We provide per-dataset results on MMEB-v2 in Table 6, covering 36 image, 18 video, and 24 visual document datasets. Image and video tasks use Hit@1, and visual document tasks use nDCG@5.

Table 6: Comparison of baseline methods and OPR-EMBED on each MMEB-v2 dataset. MMEmb-R1 uses Qwen2-VL-2B. Numbers in parentheses indicate the number of datasets in each category.

| | ColPali v1.3 | GME-7B | VLM2Vec-7B | VLM2Vec-V2-2B | CAFe-7B | UME-R1-2B | UME-R1-7B | MMEmb-R1-2B | OPR-EMBED-2B | OPR-EMBED-7B |
|---|---|---|---|---|---|---|---|---|---|---|
| Avg - All (78 tasks) | 44.4 | 57.8 | 52.3 | 58.0 | 60.6 | 60.1 | 64.5 | 65.0 | 65.8 | 70.0 |
| Avg - Image (36 tasks, Hit@1) | 34.9 | 56.0 | 65.5 | 64.9 | 67.6 | 66.6 | 71.3 | 70.0 | 71.3 | 75.4 |
| Avg - Video (18 tasks, Hit@1) | 28.2 | 38.6 | 34.0 | 34.6 | 42.4 | 42.2 | 47.5 | 50.6 | 42.7 | 47.7 |
| Avg - Visdoc (24 tasks, NDCG@5) | 71.0 | 75.2 | 46.4 | 65.4 | 63.9 | 63.9 | 67.1 | 68.0 | 74.8 | 78.5 |
| I-CLS (10) | 40.3 | 57.7 | 62.7 | 62.9 | 63.6 | 64.8 | 67.1 | 64.5 | 68.4 | 71.4 |
| I-QA (10) | 11.5 | 34.7 | 56.9 | 56.3 | 61.7 | 62.8 | 69.2 | 68.1 | 65.8 | 71.8 |
| I-RET (12) | 48.1 | 71.2 | 69.4 | 69.5 | 69.1 | 67.6 | 71.9 | 70.0 | 72.7 | 75.9 |
| I-VG (4) | 40.3 | 59.3 | 82.2 | 77.3 | 87.6 | 77.2 | 84.9 | 88.9 | 88.2 | 92.8 |
| V-CLS (5) | 26.7 | 37.4 | 39.1 | 39.3 | 35.8 | 44.3 | 48.6 | 56.3 | 48.6 | 55.1 |
| V-QA (5) | 37.8 | 50.4 | 30.0 | 34.3 | 58.7 | 51.2 | 60.7 | 52.8 | 52.0 | 59.9 |
| V-RET (5) | 21.6 | 28.4 | 29.0 | 28.8 | 34.4 | 32.9 | 38.2 | 47.6 | 30.5 | 35.4 |
| V-MR (3) | 25.5 | 38.2 | 40.6 | 36.8 | 39.5 | 39.7 | 39.3 | 42.5 | 37.8 | 35.5 |
| VD-Vidore-V1 (10) | 83.6 | 89.4 | 56.9 | 75.5 | 70.7 | 72.4 | 75.7 | 74.1 | 80.3 | 84.3 |
| VD-Vidore-V2 (4) | 52.0 | 55.6 | 9.4 | 44.9 | 49.6 | 46.2 | 50.5 | 56.0 | 53.9 | 59.7 |
| VD-VisRAG (6) | 81.1 | 85.0 | 59.1 | 79.4 | 79.5 | 79.2 | 83.7 | 78.9 | 84.9 | 87.2 |
| VD-OOD (4) | 43.1 | 44.4 | 38.1 | 39.4 | 38.1 | 37.2 | 37.6 | 48.3 | 66.5 | 69.9 |
| ImageNet-1K | 42.4 | 64.6 | 80.1 | 80.8 | 77.3 | 75.3 | 80.4 | 76.2 | 83.0 | 84.3 |
| N24News | 25.5 | 50.5 | 79.7 | 72.9 | 83.2 | 81.1 | 82.3 | 56.5 | 79.6 | 83.1 |
| HatefulMemes | 50.6 | 53.6 | 69.7 | 56.3 | 78.7 | 75.2 | 79.0 | 71.8 | 73.3 | 78.2 |
| VOC2007 | 69.8 | 80.3 | 80.7 | 85.0 | 89.8 | 80.0 | 90.8 | 79.5 | 91.8 | 90.7 |
| SUN397 | 56.1 | 69.5 | 77.4 | 71.0 | 79.9 | 79.4 | 80.3 | 76.2 | 81.0 | 82.0 |
| Place365 | 27.5 | 39.1 | 37.4 | 35.9 | 45.0 | 42.6 | 46.8 | 48.5 | 44.4 | 48.5 |
| ImageNet-A | 14.9 | 41.2 | 58.1 | 47.4 | 55.2 | 50.4 | 53.9 | 53.8 | 42.1 | 52.1 |
| ImageNet-R | 64.6 | 83.9 | 73.9 | 89.3 | 88.0 | 88.7 | 90.1 | 89.2 | 89.1 | 89.5 |
| ObjectNet | 45.6 | 69.0 | 40.1 | 65.2 | 22.5 | 52.0 | 42.3 | 62.5 | 71.8 | 74.0 |
| Country211 | 6.0 | 24.8 | 29.8 | 25.2 | 16.7 | 23.4 | 25.0 | 25.5 | 27.5 | 31.3 |
| OK-VQA | 9.4 | 33.2 | 56.8 | 51.5 | 67.3 | 62.4 | 71.7 | 66.8 | 66.3 | 74.6 |
| A-OKVQA | 6.6 | 21.0 | 47.3 | 43.6 | 63.8 | 51.1 | 58.7 | 57.2 | 57.6 | 62.3 |
| DocVQA | 11.3 | 41.4 | 89.7 | 90.1 | 79.2 | 92.2 | 93.8 | 92.8 | 93.0 | 94.4 |
| InfographicsVQA | 5.0 | 20.3 | 60.0 | 58.8 | 53.3 | 67.7 | 79.2 | 78.5 | 65.1 | 78.4 |
| ChartQA | 5.7 | 17.8 | 56.9 | 47.4 | 48.8 | 64.9 | 75.1 | 75.2 | 59.1 | 72.8 |
| Visual7W | 6.1 | 22.2 | 52.7 | 52.9 | 52.5 | 54.1 | 55.2 | 58.0 | 57.8 | 60.3 |
| ScienceQA | 16.3 | 28.0 | 38.5 | 38.2 | 65.4 | 42.7 | 53.7 | 49.8 | 46.7 | 60.0 |
| VizWiz | 27.6 | 39.0 | 39.9 | 43.3 | 43.8 | 46.8 | 51.6 | 54.5 | 53.4 | 52.4 |
| GQA | 8.3 | 76.9 | 55.1 | 64.9 | 65.7 | 67.3 | 69.3 | 64.2 | 74.1 | 78.8 |
| TextVQA | 18.8 | 46.8 | 71.6 | 72.2 | 76.8 | 78.6 | 83.5 | 83.5 | 84.5 | 84.4 |
| VisDial | 41.2 | 60.8 | 81.9 | 82.7 | 82.7 | 76.6 | 80.7 | 83.2 | 81.8 | 85.1 |
| CIRR | 8.2 | 54.9 | 51.1 | 57.5 | 60.4 | 53.7 | 55.3 | 52.8 | 61.5 | 62.8 |
| VisualNews_t2i | 50.1 | 79.7 | 80.5 | 74.5 | 69.5 | 71.7 | 76.8 | 63.5 | 77.6 | 82.5 |
| VisualNews_i2t | 47.6 | 83.6 | 81.2 | 78.2 | 79.4 | 74.2 | 82.0 | 74.8 | 82.9 | 86.6 |
| MSCOCO_t2i | 59.2 | 71.2 | 77.2 | 75.3 | 75.4 | 75.1 | 78.3 | 81.2 | 78.0 | 80.1 |
| MSCOCO_i2t | 49.9 | 57.7 | 73.9 | 71.4 | 73.1 | 68.9 | 71.4 | 73.5 | 74.7 | 77.1 |
| NIGHTS | 65.5 | 67.6 | 67.6 | 68.6 | 66.7 | 67.2 | 68.1 | 73.2 | 67.0 | 69.1 |
| WebQA | 53.8 | 91.4 | 88.3 | 90.6 | 89.3 | 90.0 | 90.9 | 87.8 | 90.1 | 90.6 |
| FashionIQ | 5.9 | 37.8 | 17.1 | 19.5 | 39.0 | 17.1 | 23.4 | 26.5 | 23.2 | 28.4 |
| Wiki-SS-NQ | 80.5 | 78.2 | 62.3 | 66.9 | 61.2 | 62.0 | 72.5 | 70.2 | 71.7 | 75.9 |
| OVEN | 50.0 | 75.1 | 66.5 | 64.3 | 60.8 | 66.9 | 71.4 | 66.8 | 72.1 | 76.3 |
| EDIS | 64.7 | 96.0 | 85.7 | 84.1 | 71.3 | 88.0 | 92.0 | 86.5 | 91.6 | 96.2 |
| MSCOCO | 36.7 | 31.4 | 75.7 | 67.1 | 84.7 | 69.5 | 72.7 | 87.2 | 85.2 | 89.8 |
| RefCOCO | 64.5 | 60.9 | 87.6 | 87.1 | 89.4 | 83.3 | 91.4 | 97.5 | 93.3 | 95.4 |
| RefCOCO-Matching | 3.9 | 78.4 | 84.6 | 85.8 | 83.0 | 84.4 | 91.1 | 83.8 | 89.9 | 92.8 |
| Visual7W-Pointing | 56.1 | 66.5 | 81.0 | 69.2 | 93.2 | 71.5 | 84.2 | 87.2 | 84.5 | 93.2 |
| K700 | 23.4 | 39.7 | 35.5 | 38.0 | 40.1 | 35.8 | 42.8 | 57.2 | 46.7 | 54.9 |
| SmthSmthV2 | 25.1 | 30.6 | 32.1 | 42.8 | 35.8 | 44.1 | 50.4 | 55.8 | 46.9 | 56.9 |
| HMDB51 | 24.8 | 47.9 | 42.2 | 40.9 | 46.9 | 54.4 | 58.3 | 61.2 | 54.2 | 59.2 |
| UCF101 | 49.4 | 54.7 | 61.8 | 60.0 | 39.6 | 67.2 | 70.0 | 74.5 | 67.6 | 78.1 |
| Breakfast | 10.9 | 14.3 | 23.8 | 14.8 | 16.6 | 20.1 | 21.5 | 32.8 | 27.7 | 26.3 |
| MVBench | 33.7 | 46.6 | 28.5 | 33.7 | 48.9 | 49.9 | 58.2 | 55.5 | 50.6 | 57.5 |
| Video-MME | 30.6 | 39.2 | 27.8 | 30.7 | 46.0 | 41.7 | 47.3 | 49.8 | 40.9 | 47.5 |
| NExTQA | 35.2 | 53.6 | 20.3 | 20.9 | 62.4 | 59.9 | 69.6 | 53.2 | 55.0 | 65.9 |
| EgoSchema | 38.4 | 46.8 | 21.8 | 34.0 | 60.0 | 45.4 | 52.4 | 37.5 | 52.2 | 53.4 |
| ActivityNetQA | 51.3 | 65.6 | 51.4 | 52.3 | 76.0 | 57.8 | 76.0 | 67.8 | 61.4 | 75.3 |
| DiDeMo | 22.8 | 26.4 | 29.3 | 30.4 | 37.8 | 32.4 | 40.0 | 48.5 | 30.2 | 33.0 |
| MSR-VTT | 17.6 | 31.8 | 34.5 | 28.3 | 36.5 | 34.3 | 38.9 | 47.2 | 32.7 | 39.1 |
| MSVD | 45.4 | 49.7 | 46.7 | 48.1 | 56.4 | 55.4 | 60.8 | 62.8 | 51.8 | 58.2 |
| VATEX | 16.7 | 24.9 | 25.5 | 26.5 | 32.0 | 29.9 | 32.6 | 44.2 | 26.0 | 30.5 |
| YouCook2 | 5.3 | 9.1 | 9.0 | 10.6 | 9.5 | 12.7 | 18.5 | 35.5 | 11.8 | 16.3 |
| QVHighlight | 19.9 | 59.5 | 57.7 | 49.4 | 58.4 | 57.5 | 54.9 | 48.2 | 56.3 | 45.3 |
| Charades-STA | 29.0 | 14.0 | 19.8 | 20.2 | 18.7 | 20.4 | 21.9 | 34.8 | 15.7 | 17.2 |
| MomentSeeker | 27.6 | 37.4 | 39.3 | 40.8 | 41.4 | 41.2 | 41.1 | 44.5 | 41.4 | 44.1 |
| ViDoRe_arxivqa | 81.7 | 86.9 | 60.2 | 80.6 | 73.3 | 73.9 | 73.6 | 68.2 | 84.6 | 88.6 |
| ViDoRe_docvqa | 56.6 | 57.5 | 34.7 | 44.9 | 38.3 | 37.9 | 41.1 | 46.8 | 48.0 | 51.0 |
| ViDoRe_infovqa | 84.9 | 91.6 | 70.4 | 83.7 | 80.6 | 76.2 | 80.8 | 82.5 | 86.5 | 90.9 |
| ViDoRe_tabfquad | 86.9 | 94.6 | 78.2 | 89.2 | 80.7 | 86.1 | 90.2 | 91.2 | 85.9 | 92.6 |
| ViDoRe_tatdqa | 70.9 | 74.1 | 27.6 | 43.8 | 37.8 | 40.6 | 46.7 | 43.5 | 56.8 | 59.8 |
| ViDoRe_shiftproject | 75.1 | 96.8 | 38.6 | 60.8 | 52.0 | 66.8 | 65.0 | 67.8 | 74.4 | 80.4 |
| ViDoRe_artificial_intelligence | 95.7 | 99.6 | 67.7 | 88.5 | 86.0 | 85.9 | 89.5 | 89.2 | 93.3 | 95.8 |
| ViDoRe_energy | 94.7 | 95.3 | 60.4 | 86.5 | 84.8 | 83.3 | 85.7 | 81.5 | 89.6 | 93.2 |
| ViDoRe_government_reports | 93.6 | 98.8 | 61.8 | 85.0 | 85.0 | 82.6 | 89.8 | 84.8 | 90.0 | 94.0 |
| ViDoRe_healthcare_industry | 95.9 | 99.3 | 69.9 | 92.2 | 88.4 | 90.8 | 94.3 | 85.8 | 94.3 | 96.6 |
| ViDoRe_esg_reports_human_labeled_v2 | 51.3 | 63.4 | 6.8 | 45.6 | 50.7 | 50.2 | 50.4 | 56.2 | 58.6 | 67.4 |
| ViDoRe_biomedical_lectures_v2_multilingual | 54.7 | 49.5 | 5.1 | 44.3 | 50.9 | 46.2 | 50.7 | 47.5 | 54.3 | 57.4 |
| ViDoRe_economics_reports_v2_multilingual | 49.0 | 54.2 | 13.9 | 43.0 | 54.3 | 45.7 | 57.8 | 59.2 | 52.1 | 59.5 |
| ViDoRe_esg_reports_v2_multilingual | 52.9 | 55.4 | 11.9 | 46.6 | 42.3 | 42.7 | 43.2 | 61.2 | 50.7 | 54.5 |
| VisRAG_ArxivQA | 80.9 | 87.4 | 52.6 | 76.9 | 74.0 | 74.3 | 80.5 | 84.8 | 82.9 | 86.6 |
| VisRAG_ChartQA | 78.2 | 81.9 | 57.7 | 84.4 | 82.7 | 86.0 | 85.0 | 74.2 | 88.2 | 86.8 |
| VisRAG_MP-DocVQA | 86.8 | 89.2 | 60.6 | 71.8 | 75.1 | 75.6 | 83.4 | 71.5 | 84.2 | 88.6 |
| VisRAG_SlideVQA | 95.0 | 94.5 | 54.7 | 91.5 | 87.6 | 87.1 | 91.5 | 82.8 | 94.3 | 95.4 |
| VisRAG_InfoVQA | 85.7 | 93.5 | 66.0 | 85.7 | 87.9 | 84.4 | 89.2 | 91.8 | 89.9 | 92.5 |
| VisRAG_PlotQA | 60.3 | 63.4 | 62.7 | 66.1 | 69.4 | 68.0 | 72.7 | 68.5 | 70.1 | 73.6 |
| ViDoSeek-page | 22.2 | 23.2 | 16.3 | 21.9 | 22.5 | 21.2 | 21.3 | 32.2 | 80.8 | 85.7 |
| ViDoSeek-doc | 83.7 | 83.9 | 69.4 | 80.2 | 73.8 | 75.9 | 75.3 | 78.5 | 82.6 | 84.2 |
| MMLongBench-page | 14.2 | 16.2 | 0.4 | 11.9 | 13.3 | 11.9 | 12.3 | 32.4 | 51.8 | 56.2 |
| MMLongBench-doc | 52.5 | 54.3 | 28.8 | 43.7 | 42.6 | 39.7 | 41.3 | 49.9 | 50.8 | 53.3 |