VaME: Exploring Variational Latent Reasoning for Multimodal Embeddings

Anonymous authors
Paper under double-blind review

Abstract

Universal multimodal retrieval requires compact embeddings that preserve task-relevant semantic information across diverse modalities. Prior works have incorporated latent reasoning into multimodal embedding learning to refine this information before embedding extraction. However, most existing approaches remain confined to deterministic latent paths, without exploring alternative trajectories to discover better embeddings. Thus, we propose **VaME** (***Va****riational* ***M****ultimodal* ***E****mbeddings*), a framework that models latent reasoning as a learnable distribution over trajectories. Specifically, we first introduce Variational Latent Reasoning (VLR) to enable autoregressive exploration in latent space, guided by answer reconstruction through a lightweight decoder. Meanwhile, we augment the original embedding-token readout with a latent-fused embedding to facilitate exploration during subsequent reinforcement learning. Finally, we optimize latent reasoning over stochastic variational trajectories through reinforcement learning, using Semantic Decoding Reward (SDR) to favor semantically meaningful trajectories with interpretable decoded outcomes. On the 78-task MMEB-V2 benchmark, spanning image, video, and visual-document retrieval, VaME outperforms most explicit CoT-based models and all latent-reasoning baselines. VaME also demonstrates robust performance on reasoning-intensive benchmarks such as MRMR, with substantial gains after reinforcement learning. Importantly, VaME achieves these gains with at least a $\mathbf{4.25}\times$ inference speedup over the deterministic latent autoregressive baselines. *The code will be made publicly available.*

1 Introduction

Universal Multimodal Embedding (UME) aims to map text, images, videos, visual documents, and their mixtures into a shared embedding space, allowing a single model to serve diverse retrieval tasks (Radford et al., 2021; Wei et al., 2023). Recent vision-language embedding models (Liu et al., 2025; Chen et al., 2025) have made substantial progress (Jiang et al., 2025; Zhang et al., 2025b) by learning instruction-aware representations (Lin et al., 2025; Jiang et al., 2024) across these modalities (Meng et al., 2026; Lan et al., 2025a). Yet universal retrieval is not simply a matter of compressing more modalities into one vector (Xiao et al., 2026). Depending on the instruction, the model must identify and compose task-relevant evidence into a compact representation.

Accordingly, a growing line of work augments embedding formation with intermediate reasoning. Explicit approaches make this computation human-readable through retrieval-oriented rewrites (Wu et al., 2026a; Wang et al., 2026) or textual reasoning traces (Lan et al., 2025b; Jiang et al., 2026; Cui et al., 2025). However, the required autoregressive generation before embedding extraction (Cui et al., 2025; Cheng et al., 2026) is far slower than direct embedding extraction, making these approaches impractical for large-scale, real-time applications. Latent reasoning offers a compact alternative (Shen et al., 2025; He et al., 2026): methods such as LaME (Wu et al., 2026b) and TTE-Flash (Cheng et al., 2026) perform additional computation in hidden space while preserving an efficient retrieval interface. Most latent approaches, however, remain deterministic (Hao et al., 2025; Shen et al., 2025; He et al., 2026): a given input follows one fixed computational path. Such a path can refine the representation, but cannot explore alternative trajectories (Xu et al., 2025b). Consequently, deterministic latent reasoning cannot fully benefit from exploration-based learning. This gap raises our central question: *Can an embedding model explore diverse latent reasoning paths while retaining efficient inference?*

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.

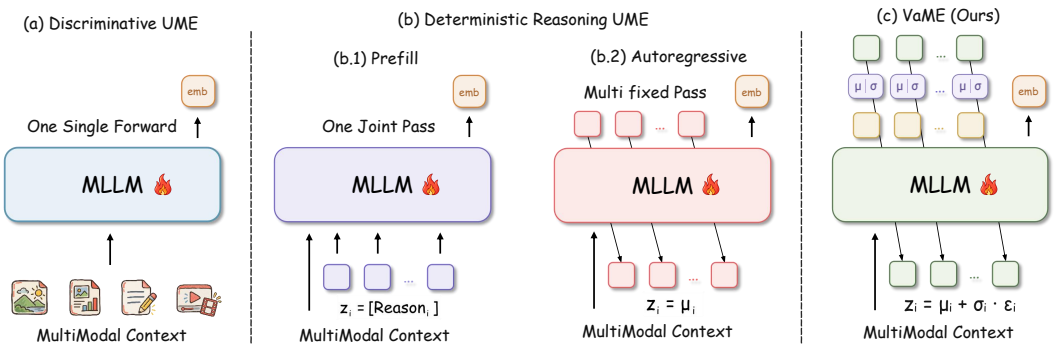

Figure 1: **From deterministic embedding to variational latent reasoning.** (a) Discriminative embedders map each input directly to a single embedding. (b) Deterministic reasoning adds latent computation but follows one fixed path. (c) VaME samples and compares alternative latent trajectories during training and follows the conditional means for deterministic inference.

To answer this question, we propose **VaME** (**Va**riational **M**ultimodal **E**mbeddings), a framework that models latent reasoning as a learnable distribution over trajectories (Figure 1). Its first component, Variational Latent Reasoning (VLR), replaces each deterministic latent transition with a conditional Gaussian (Kingma & Welling, 2013; Rezende et al., 2014). At every autoregressive step, the model samples a continuous latent action and feeds it back into the backbone; each sampled action therefore changes the context from which subsequent actions are drawn (Hao et al., 2025; Shen et al., 2025). During latent supervised fine-tuning, retrieval objectives align these trajectories with relevance, while a lightweight decoder reconstructs answer annotations from their hidden states (Yu et al., 2025). Together with variational regularization (Alemi et al., 2017; Li et al., 2025), this answer supervision encourages exploratory trajectories to preserve retrieval-relevant semantics without requiring intermediate chain-of-thought supervision.

Exploration and retrieval stability impose different demands on the final representation. We therefore retain the original embedding-token readout and introduce a complementary latent-fused embedding that exposes sampled trajectory information to retrieval supervision. The former provides a stable single-vector retrieval interface, whereas the latter supplies a trajectory-sensitive signal for subsequent reinforcement learning. In the second stage, we sample multiple variational trajectories (Tan et al., 2025) and refine the continuous latent policy through group-relative optimization (Shao et al., 2024). Retrieval feedback rewards discriminative trajectories (Wu et al., 2026a; Jiang et al., 2026), while our Semantic Decoding Reward (SDR) evaluates whether the frozen lightweight decoder can decode an answer semantically consistent with the reference. SDR thereby discourages semantically empty exploration and makes the outcome of latent reasoning inspectable through decoding. At inference, VaME follows a single mean trajectory and emits one embedding per item, requiring neither text generation nor trajectory search.

We evaluate VaME on broad and reasoning-intensive multimodal retrieval benchmarks. Across the 78 tasks of MMEB-V2 (Meng et al., 2026), covering image, video, and visual document retrieval, VaME outperforms most explicit CoT-based models (Lan et al., 2025b; Cui et al., 2025; Jiang et al., 2026) and all comparable prior latent-reasoning baselines (Wu et al., 2026b; Cheng et al., 2026; He et al., 2026). It further demonstrates robust performance on reasoning-intensive benchmarks such as MRMR (Zhang et al., 2025a), with substantial gains after reinforcement learning. Importantly, VaME achieves these gains with at least a $\mathbf{4.25}\times$ inference speedup over the deterministic latent autoregressive baselines. Our contributions are as follows:

- We introduce Variational Latent Reasoning, which formulates multimodal latent reasoning as an autoregressive distribution over continuous trajectories, enabling training-time exploration with deterministic single-trajectory inference.
- We develop a two-stage framework in which supervised fine-tuning aligns variational trajectories with retrieval relevance and answer semantics, and reinforcement learning then refines the latent policy using retrieval feedback and semantic decoding reward.

- Extensive experiments on MMEB-V2 and MRMR demonstrate that VaME delivers strong retrieval performance across diverse multimodal retrieval tasks while achieving substantially faster inference than deterministic latent autoregressive baselines.

2 Related Work

Universal multimodal embeddings. Prior works such as VLM2Vec (Jiang et al., 2025) and GME (Zhang et al., 2025b) train instruction-aware representations (Jiang et al., 2024; Chen et al., 2025; Wu et al., 2026c; Gu et al., 2026) with vision–language backbones (Wang et al., 2024; Lan et al., 2025a). MM-Embed (Lin et al., 2025) studies universal retrieval (Wei et al., 2023; Gu et al., 2025) across multimodal inputs (Jian et al., 2025; Liu et al., 2025), while VLM2Vec-V2 (Meng et al., 2026) extends evaluation and training to video and visual document tasks. MetaEmbed (Xiao et al., 2026) explores flexible late interaction (Santhanam et al., 2022) and the allocation of multiple vectors at test time. Our focus is complementary: we change the computation that precedes extraction while retaining a single-vector retrieval interface.

Thinking before embedding. A growing line of work introduces an intermediate reasoning stage before deriving the final retrieval embedding (Shen et al., 2025; Deng et al., 2023). Some methods enrich the context for embedding extraction through textual CoT (Wu et al., 2026a; Wang et al., 2026; Lan et al., 2025b; Jiang et al., 2026), but require multi-step autoregressive decoding. Other methods process learned tokens in a single prefill pass (Wu et al., 2026b; Cheng et al., 2026) or autoregressively feed hidden states back into the backbone (He et al., 2026; Cheng et al., 2026). CoT generation can further supervise these latent representations (Cheng et al., 2026). These methods rely on explicit rewrites or deterministic latent computation. We instead use a stochastic policy over vector transitions, allowing the model to explore different latent reasoning paths.

Continuous reasoning and variational representations. Continuous reasoning shifts intermediate computation from text to hidden states. Coconut (Hao et al., 2025) feeds hidden states back recurrently, while CoDI (Shen et al., 2025), CoLaR (Tan et al., 2025), and SoftCoT (Xu et al., 2025a;b) learn or compress reasoning in continuous space. LaSER (Jin et al., 2026) and PLUME (He et al., 2026) apply latent reasoning to dense and multimodal retrieval. In parallel, variational autoencoders model reparameterized latent variables (Kingma & Welling, 2013; Rezende et al., 2014), while information-bottleneck methods learn stochastic representations (Tishby et al., 2000; Alemi et al., 2017), including multimodal and contrastive variants (Wu et al., 2025; Li et al., 2025). Our method seeks to introduce exploratory latent reasoning for multimodal retrieval.

3 From Prefill to Autoregressive Latent Thinking

Multimodal retrieval formulation. Universal multimodal retrieval seeks to identify the most relevant target from a candidate pool for an instruction-conditioned query. Following standard practice, we obtain a multimodal embedder by contrastively fine-tuning a pretrained MLLM. Given a mini-batch $\mathcal{B} = \{(q_i, t_i)\}_{i=1}^{B}$ containing B matched pairs, let the MLLM backbone F_θ encode q_i and t_i into query and target embeddings $F_\theta(q_i), F_\theta(t_i) \in \mathbb{R}^d$, respectively, with each given by the normalized final-layer hidden state of the last token. The corresponding InfoNCE objective is

$$\mathcal{L}_{\mathrm{NCE}} = -\frac{1}{2B}\sum_{i=1}^{B}\left[\log\frac{e^{F_\theta(q_i)^\top F_\theta(t_i)/\tau}}{\sum_j e^{F_\theta(q_i)^\top F_\theta(t_j)/\tau}} + \log\frac{e^{F_\theta(t_i)^\top F_\theta(q_i)/\tau}}{\sum_j e^{F_\theta(t_i)^\top F_\theta(q_j)/\tau}}\right], \tag{1}$$

where τ denotes the temperature; $\{t_j\}_{j\neq i}$ serve as in-batch negatives for q_i, and vice versa.

From prefill to latent recurrence. Reasoning-enhanced embedding models differ in the computation performed before representation extraction. LaME and register-based TTE-Flash place learnable latent slots $\{r_k\}_{k=1}^{K}$ in a single prefill pass, preserving parallelism (Wu et al., 2026b; Cheng et al., 2026). Their hidden states attend to earlier positions, but each input embedding r_k is shared across examples and does not depend on the preceding latent state h_{k-1}. Recurrence instead takes h_{k-1} as its step-k input, enabling input-dependent refinement:

$$(h_0, c_0) = F_\theta(x), \qquad (h_k, c_k) = F_\theta(h_{k-1}, c_{k-1}), \quad k = 1, \ldots, K, \tag{2}$$

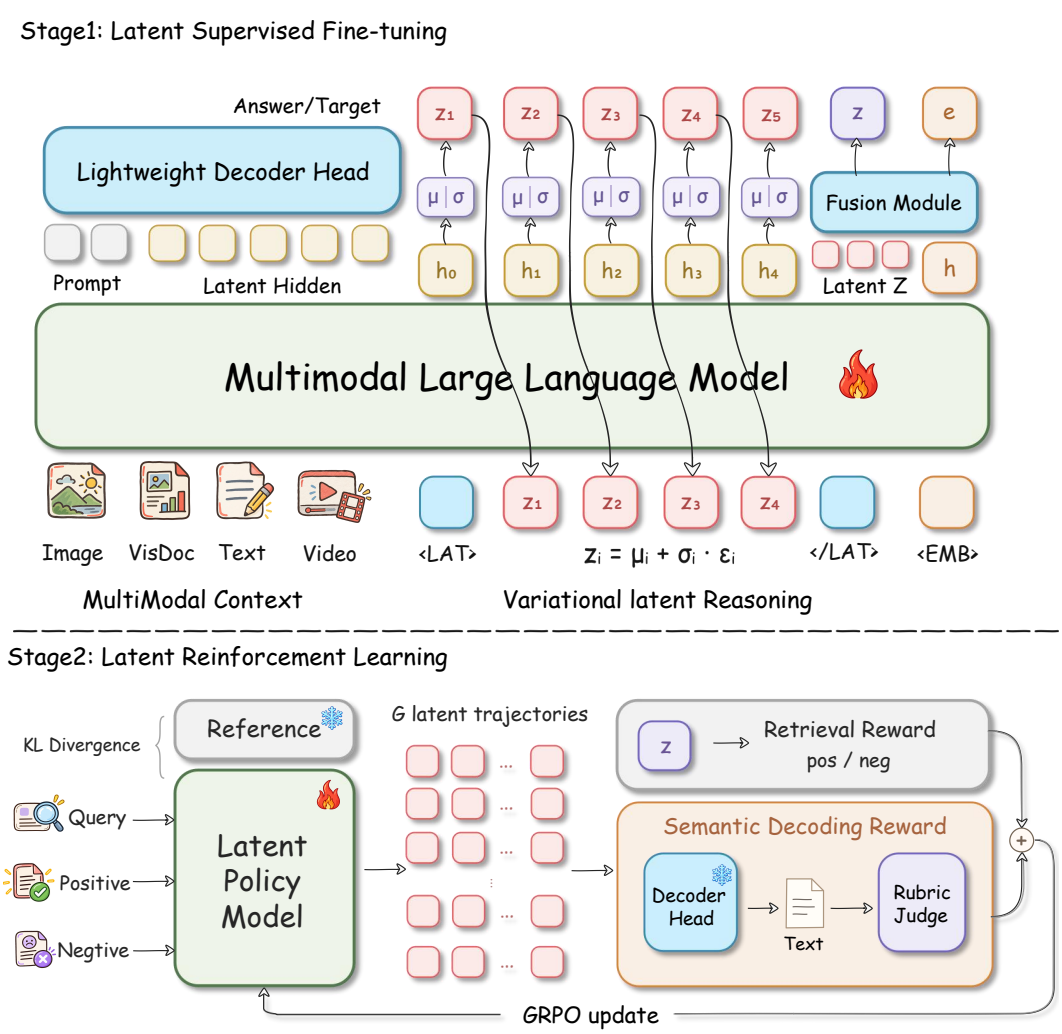

Figure 2: Overview of the VaME framework. Latent SFT aligns variational trajectories with retrieval and answer supervision. Latent RL compares sampled trajectories and refines the policy using retrieval feedback and the semantic decoding reward.

where F_θ is the MLLM backbone, h_k the hidden state at step k, and c_k the running KV cache. After K transitions, a final embedding token reads out the representation. Recurrence enables input-dependent refinement but remains deterministic: each input follows a single fixed trajectory. VaME instead learns a conditional distribution over recurrent trajectories, enabling exploration during training while following the conditional means at inference.

4 Variational Multimodal Embeddings

As shown in Figure 2, VaME establishes a two-stage training framework for learning multimodal embeddings through latent computation. Its central challenge is not merely to introduce noise or stochasticity, but to make sampled trajectories represent distinct computations, align their variation with retrieval, and confine exploration to training. Accordingly, Section 4.1 formulates autoregressive computation as a regularized conditional trajectory distribution; Section 4.2 aligns its samples with retrieval relevance and answer semantics through latent SFT and complementary readouts; Section 4.3 compares alternative trajectories using retrieval feedback and SDR, then improves the latent policy with group-relative RL. At inference, the learned policy follows its conditional means, retaining deterministic single-vector retrieval.

4.1 VARIATIONAL LATENT REASONING

Stochastic latent recurrence. The latent recurrence in Equation 2 enables input-dependent refinement, but maps each input to only one deterministic trajectory. This leaves no alternative computations for exploration or outcome-based comparison. We therefore introduce stochasticity into the recurrent transitions, allowing multiple latent trajectories to be sampled from the same multimodal input. VLR realizes this idea by replacing each deterministic transition with a conditional diagonal Gaussian. As shown in Figure 2, we introduce two special tokens, <LAT> and </LAT>, to mark the start and end of latent reasoning, respectively. Given an input x, a multimodal prefill produces an initial cache c_0 and hidden state h_0. At step k, a variational head predicts the distribution parameters from the current hidden state, draws noise $\epsilon_k \sim \mathcal{N}(0, I)$ to construct a continuous latent action, and feeds it back into the backbone:

$$\begin{aligned} (\mu_k, \log \sigma_k) &= g_\phi(h_{k-1}), \qquad z_k = \mu_k + \sigma_k \odot \epsilon_k, \\ (h_k, c_k) &= F_\theta(z_k, c_{k-1}), \qquad q_{\boldsymbol{\theta}}(Z \mid x) = \prod_{k=1}^{K} q_{\boldsymbol{\theta}}(z_k \mid x, z_{<k}), \end{aligned} \tag{3}$$

where $\boldsymbol{\theta} = (\theta, \phi)$ comprises the backbone parameters θ and variational-head parameters ϕ, and $Z = (z_1, \ldots, z_K)$ is the sampled latent trajectory. Since each z_k is processed before the next distribution is produced, early samples affect the cache, hidden states, and subsequent action distributions. Thus, $q_{\boldsymbol{\theta}}(Z \mid x)$ defines a distribution over complete latent computations rather than independent noise added to the final embedding. Sampling from this distribution yields alternative latent paths for the same multimodal input, providing candidate trajectories for latent RL. The reparameterization in Equation 3 allows gradients from both training objectives to pass through the sampled latent actions, enabling joint optimization of the backbone and variational head.

Latent transition regularization. Without a constraint on the conditional distributions, the policy could encode training instances through arbitrarily displaced means or variances. We anchor each transition to an isotropic Gaussian using

$$\mathcal{L}_{\mathrm{KL}} = \frac{1}{K} \sum_{k=1}^{K} D_{\mathrm{KL}}(q_{\boldsymbol{\theta}}(z_k \mid x, z_{<k}) \,\|\, \mathcal{N}(0, I)) \,. \tag{4}$$

The KL term keeps the trajectory space smooth enough for sampling, while the task losses below determine which variations preserve relevance and answer semantics. We use the term *variational* for this conditional latent distribution, reparameterized sampling, and prior regularization; the overall retrieval objective is not an evidence lower bound for reconstructing the raw multimodal input. Appendix B.1 gives the exact Gaussian parameterization and closed-form KL.

4.2 LATENT SUPERVISED FINE-TUNING

The variational distribution above provides alternative latent computations, but does not by itself identify which trajectories support retrieval. To supervise these trajectories for retrieval, we complete the K latent transitions and let the embedding token read from the resulting cache, yielding the hidden state h. To balance exploration with retrieval stability, we use complementary readouts: the original embedding-token readout preserves a stable single-vector interface, while the latent-fused embedding provides a trajectory-sensitive signal for subsequent reinforcement learning:

$$e = \mathrm{Norm}(h) \,, \qquad z = \mathrm{Norm}(F_\omega(h, Z)) \,, \tag{5}$$

where F_ω aggregates the trajectory Z via attention pooling and residually combines it with h, parameterized by ω. For a paired batch $\mathcal{B} = \{(x_i^q, x_i^t)\}_{i=1}^{B}$, their joint retrieval objective is

$$\mathcal{L}_{\mathrm{ret}} = \mathcal{L}_{\mathrm{NCE}}^{e} + \lambda_z \mathcal{L}_{\mathrm{NCE}}^{z} + \lambda_Z \mathcal{L}_{\mathrm{NCE}}^{Z}, \tag{6}$$

where e and z identify the two readouts, while Z denotes position-aligned pairwise scoring over latent actions. The first term learns the deployed geometry; the latter two supervise trajectory-dependent retrieval, with z also scoring latent-RL rollouts. At inference, only e is used to build the single-vector retrieval index. Meanwhile, to align the trajectories with retrieval-relevant semantics, a lightweight decoder p_ψ reconstructs query-side answers and target-side explanations or descriptions from the projected post-action states $H = [h_1, \ldots, h_K]$:

$$\mathcal{L}_{\mathrm{ans}} = -\mathbb{E}_{(x,y),\, Z \sim q_{\boldsymbol{\theta}}(\cdot \mid x)} \log p_\psi(y \mid \mathcal{I}_{\mathrm{dec}}, H) \,, \tag{7}$$

where $\mathcal{I}_{\text{dec}}$ denotes the fixed decoding prompt. $\mathcal{L}_{\text{div}}$ penalizes pairwise cosine similarity between posterior means to prevent latent collapse. Together with KL regularization, the retrieval and reconstruction signals yield the complete SFT objective:

$$\mathcal{L}_{\text{SFT}} = \mathcal{L}_{\text{ret}} + \lambda_{\text{ans}}\mathcal{L}_{\text{ans}} + \lambda_{\text{KL}}\mathcal{L}_{\text{KL}} + \lambda_{\text{div}}\mathcal{L}_{\text{div}}, \tag{8}$$

where λ_z, λ_Z, λ_{ans}, λ_{KL}, and λ_{div} weight the corresponding objectives; exact settings are provided in Appendix B.3.

4.3 Latent Reinforcement Learning

As illustrated in Equation 8, latent SFT turns the variational trajectory distribution into an aligned policy whose samples are grounded in retrieval relevance and answer semantics. However, it does not express a preference among the alternative computations available to the same query. Latent RL supplies this missing comparison by refining the SFT policy under group-relative retrieval and semantic feedback. Specifically, given a query q, we draw G trajectories $Z_g \sim \pi_{\text{old}}(Z \mid q)$, $g = 1, \ldots, G$, and assign each trajectory two complementary rewards:

$$R(Z_g) = R_{\text{ret}} + \lambda_{\text{SDR}} R_{\text{SDR}}, \tag{9}$$

where R_{ret} and R_{SDR} denote the retrieval and semantic decoding rewards, respectively; λ_{SDR} balances them, and $R(Z_g)$ is the trajectory-level reward used by GRPO. Specifically, inspired by the similarity-gap reward in UME-R1 (Lan et al., 2025b), we make a simple refinement to R_{ret} by softly weighting harder negatives. We complement this retrieval signal with SDR, which evaluates whether a sampled trajectory retains retrieval-relevant semantics:

$$R_{\text{ret}} = s_g^+ - \sum_{j \in \mathcal{N}} w_{g,j} s_{g,j}^-, \qquad R_{\text{SDR}} = J(\mathcal{I}_{\text{eval}}, y, F_\psi(\mathcal{I}_{\text{dec}}, H_g)), \tag{10}$$

where $\mathcal{N}$ is the negative candidate set, s_g^+ and $s_{g,j}^-$ are positive and negative similarities for trajectory Z_g, and $w_{g,j}$ is computed by applying softmax to its negative similarities. For SDR, the frozen decoder F_ψ greedily generates an answer from H_g under $\mathcal{I}_{\text{dec}}$, and the rubric judge J in Figure 2 scores its agreement with reference y under the rubric-based evaluation instruction $\mathcal{I}_{\text{eval}}$. Together, the two rewards favor retrieval-discriminative and semantically grounded trajectories. The rubric and scoring details are provided in Appendix B.2. Given these trajectory-level rewards, we first standardize them within each group as $A_g = (R(Z_g) - \bar{R})/(\text{std}_{g=1}^G[R(Z_g)] + \epsilon)$ and then minimize the following clipped group-relative loss for the continuous Gaussian policy:

$$\begin{aligned}\mathcal{L}_{\text{GRPO}} = -\mathbb{E}_{q\sim\mathcal{D},\,\{Z_g\}\sim\pi_{\text{old}}(\cdot|q)}\Bigg[\frac{1}{G\cdot K}\sum_{g=1}^{G}\sum_{k=1}^{K}\Bigg(\min\Bigg\{\frac{\pi_{\boldsymbol{\theta}}(z_{g,k})}{\pi_{\text{old}}(z_{g,k})}A_g,\\ \text{clip}\Bigg(\frac{\pi_{\boldsymbol{\theta}}(z_{g,k})}{\pi_{\text{old}}(z_{g,k})}, 1-\epsilon_-, 1+\epsilon_+\Bigg)A_g\Bigg\} - \beta D_{\text{KL}}(\pi_{\boldsymbol{\theta}}\|\pi_{\text{ref}})\Bigg)\Bigg],\end{aligned} \tag{11}$$

where $\mathcal{D}$ is the training query distribution and K is the number of latent steps. The current and rollout policies are $\pi_{\boldsymbol{\theta}}$ and π_{old}, respectively. For the step-k action $z_{g,k}$ of trajectory g, we compute its joint Gaussian probability density under each policy, conditioned on the same query q and preceding actions $z_{g,<k}$, and take the current-to-rollout density ratio. The KL penalty uses the same conditioning, with β controlling regularization toward the frozen reference policy π_{ref}. Thus, the rewards rank the sampled computations, and GRPO reinforces better trajectories while limiting policy drift.

5 Experiments

5.1 Experimental Setup

Implementation. Following prior works (Meng et al., 2026; He et al., 2026; Lan et al., 2025b; Wu et al., 2026b), we initialize the 2B and 7B VaME variants from Qwen2-VL-2B-Instruct and Qwen2-VL-7B-Instruct, respectively. Latent SFT trains on approximately 1.5 million image, video, and visual-document retrieval examples for 3,200 steps with $K = 8$ latent steps, temperature $\tau = 0.02$, and effective batch size 512. The backbone and newly initialized modules use learning rates of 10^{-5}

Table 1: Results on the MMEB-V2 benchmark. Bold and underline indicate the best and second-best scores within each size group, respectively. CLS: classification, QA: question answer, RET: retrieval, GD: grounding, MRET: moment retrieval, VDR: ViDoRe, VR: VisRAG, OOD: out-of-distribution. Reported metrics adhere to the settings of VLM2Vec-V2.

Model	Image					Video					VisDoc					All
	CLS	QA	RET	GD	Avg.	CLS	QA	RET	MRET	Avg.	VDRv1	VDRv2	VR	OOD	Avg.	
# of Datasets	10	10	12	4	36	5	5	5	3	18	10	4	6	4	24	78
~2B Model Size																
VLM2Vec	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
DUME	59.3	55.0	66.3	78.0	62.5	37.7	46.6	17.1	30.0	33.2	67.6	43.3	47.1	33.8	52.8	52.7
GME	54.4	29.9	66.9	55.5	51.9	34.9	42.0	25.6	32.4	33.9	**86.1**	54.0	82.5	43.1	72.7	54.1
VLM2Vec-V2	62.0	56.3	69.5	77.3	64.9	39.3	34.3	28.8	38.5	34.9	75.5	44.9	79.4	39.4	65.4	58.0
UME-R1	64.8	62.8	67.6	77.2	66.6	44.3	51.0	32.9	39.7	42.2	72.4	46.2	79.2	28.9	62.5	59.7
PLUME	66.5	59.2	67.6	79.7	66.3	45.0	52.3	33.5	46.7	44.1	72.1	49.8	78.1	57.4	67.5	61.6
Ops-MM-Embed	**68.1**	65.1	69.2	80.9	69.0	53.6	**55.7**	41.8	33.7	47.6	76.4	53.2	77.6	32.6	65.5	63.0
TTE-Flash	67.1	61.7	**70.7**	**81.3**	68.3	**54.4**	51.5	**45.4**	**51.5**	**50.6**	75.6	52.7	83.8	41.1	68.1	64.1
LaME	67.6	66.2	70.5	81.2	69.3	45.6	52.3	36.3	43.5	44.5	78.2	48.4	82.8	55.4	72.1	64.4
VaME (Ours)	67.6	**66.6**	70.0	80.3	**69.5**	46.2	54.8	38.2	45.5	46.3	79.4	**55.2**	**85.6**	**64.1**	**74.4**	**65.7**
7–8B Model Size																
LamRA	51.7	34.1	66.9	56.7	52.4	32.9	42.6	23.2	37.6	33.7	56.3	33.3	58.2	40.1	50.2	47.4
VLM2Vec	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
DUME	64.2	57.0	70.8	81.8	66.4	32.9	47.4	8.6	28.0	29.4	67.1	35.2	82.6	34.9	60.3	55.9
GME	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
CAFe	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	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
TWN	70.2	**74.3**	70.8	87.1	**73.4**	50.1	64.0	34.8	40.8	48.2	82.5	56.7	86.2	**70.0**	77.0	68.7
LaME	70.0	71.0	73.1	85.8	73.0	51.0	**64.8**	40.3	44.6	50.8	81.5	58.2	85.0	68.2	75.9	68.8
Ops-MM-Embed	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
VaME (Ours)	**71.3**	70.3	**73.7**	84.1	73.2	49.9	58.6	42.0	**49.3**	50.0	83.2	59.0	**89.4**	67.9	**78.2**	**69.4**

and 2×10^{-5}, respectively. Latent RL trains for 500 steps on approximately 10K records sampled from a separate pool of approximately 127K candidates, with effective batch size 64 and learning rate 10^{-6}. We use $G = 8$ trajectories per query, GRPO clipping bounds $[\epsilon_-, \epsilon_+] = [0.8, 1.28]$, per-dimension KL weight $\beta = 0.01$, and semantic-reward weight $\lambda_{\text{SDR}} = 0.2$. We use Qwen3.5-122B-A10B via API as the judge model for SDR.

Evaluation. We evaluate broad multimodal retrieval on MMEB-V2 (Meng et al., 2026), which has 36 image and 18 video datasets covering retrieval, classification, moment retrieval, and visual QA, all evaluated with Hit@1. Its 24 VisDoc datasets involve document images such as complex charts and paper figures and use nDCG@5. To assess reasoning-intensive retrieval, we also evaluate on MRMR (Zhang et al., 2025a), which comprises 11 expert-level tasks. We report Hit@1 for Negation and nDCG@10 for the remaining tasks, following the official MRMR protocol.

Baselines. We compare VaME with representative MLLM-based multimodal embedders spanning three paradigms: direct-embedding models such as VLM2Vec, GME, and Ops-MM-Embed; explicit-reasoning models TWN and UME-R1; and latent-reasoning models PLUME, TTE-Flash, and LaME. Most baselines follow the training and evaluation protocols of VLM2Vec-V2 (Meng et al., 2026), making their performance broadly comparable. We note that Ops-MM-Embed, GME, and LamRA use additional training sources, while UME-R1 and TWN uses CoT from external reasoners to enhance its embeddings.

5.2 Main Results

Results on MMEB-V2. As shown in Table 1, VaME-2B and VaME-7B score 65.7 and 69.4 on the 78-task MMEB-V2 benchmark, leading their respective size groups. At 2B, VaME outperforms LaME and PLUME by 1.3 and 4.1 points; relative to LaME, it gains 0.2 on Image, 1.8 on Video, and 2.3 on VisDoc. It also surpasses UME-R1 by 6.0 points at 2B and the explicit-reasoning baselines TWN and UME-R1 by 0.7 and 4.9 points at 7B, respectively. Thus, variational latent reasoning can improve multimodal embeddings across modalities without generated reasoning text.

Results on MRMR. As shown in Table 2, the 7B variant of VaME scores 51.4 on MRMR, a benchmark comprising 11 retrieval tasks that require reasoning. It outperforms RIME by 1.2 points and LaME, the deterministic baseline for latent reasoning, by 1.6 points. It ranks first on six subtasks, including Medicine and Science in Knowledge and all four Theorem domains, and improves over

Table 2: Results on the reasoning-intensive MRMR benchmark including Art, Medicine (Med.), Science (Sci.), Humanities (Hum.), Math, Physics (Phy.), Engineering (Eng.), Business (Bus.), Negation (Neg.), Design (Des.), and Traffic (Tra.). Bold and underlined scores denote the best and second-best performance, respectively.

Model	Backbone	Knowledge				Theorem				Contradiction			All
		Art	Med.	Sci.	Hum.	Math	Phy.	Eng.	Bus.	Neg.	Des.	Tra.	
E5-V	LLaVA-Next-8B	25.1	11.7	16.6	10.8	2.1	3.4	2.5	5.2	<u>11.5</u>	3.7	2.1	8.6
EVA-CLIP	EVA-ViT-0.4B	10.2	13.5	26.1	12.9	6.2	10.5	9.3	11.7	8.5	4.4	5.4	10.8
OpenCLIP	ViT-G/14-1B	56.0	17.9	33.2	22.0	5.7	5.0	7.0	9.7	13.0	8.1	12.4	17.3
VLM2Vec	Qwen2-VL-7B	53.5	22.4	36.7	24.0	2.1	2.8	2.8	2.9	<u>11.5</u>	5.6	18.3	18.1
VISTA	Qwen2-VL-2B	21.3	27.8	32.6	17.0	18.8	17.1	17.3	28.6	20.0	20.2	9.4	20.9
ColPali	PaliGemma-3B	36.1	29.9	42.7	29.2	5.7	14.8	12.0	24.6	**28.5**	19.4	18.2	23.7
GME	Qwen2-VL-7B	54.3	40.1	46.8	45.6	28.8	36.0	30.2	45.1	15.0	26.3	29.6	36.2
MM-Embed	NV-Embed-8B	65.6	53.0	63.5	62.8	23.6	30.8	27.4	44.9	7.0	23.8	34.9	39.8
UME-R1	Qwen2-VL-7B	<u>77.8</u>	55.7	72.9	64.1	27.2	39.2	32.2	47.8	7.5	61.9	<u>41.7</u>	48.0
Ops-MM-Embed	Qwen2-VL-7B	**79.3**	52.5	70.0	67.8	27.7	39.5	30.1	<u>52.3</u>	8.0	55.9	**45.8**	48.1
LaME	Qwen2-VL-7B	73.4	58.2	<u>73.8</u>	65.6	<u>29.5</u>	<u>44.4</u>	<u>36.4</u>	52.2	8.5	**64.9**	40.9	49.8
RIME	Qwen2-VL-7B	76.8	<u>58.5</u>	73.6	**71.4**	29.2	43.0	35.6	51.8	8.5	<u>64.1</u>	39.5	<u>50.2</u>
VaME (Ours)	Qwen2-VL-7B	75.6	**60.7**	**76.3**	<u>68.9</u>	**32.5**	**47.2**	**37.8**	**52.9**	10.1	63.7	39.7	**51.4**

LaME on nine tasks, including gains of 3.3, 3.0, and 2.8 points on Humanities, Math, and Physics. VaME also exceeds the explicit reasoning model UME-R1 by 3.4 points overall. These broad gains demonstrate robust retrieval on tasks that require reasoning while retaining a latent interface based on a single vector without generated reasoning text.

5.3 Ablation Study

To validate the effectiveness of variational latent reasoning, latent supervision, and reward-guided refinement, we conduct ablations on MMEB-V2 using the Qwen2-VL-2B embedder.

Ablation on Core Components. As shown in Table 3, we isolate VLR, Latent SFT, and Latent RL under the same 2B setting. The deterministic AR baseline scores 63.8, while VLR alone drops to 63.1, showing that unguided exploration introduces noise. Latent SFT raises AR and VLR to 64.6 and 64.9, respectively, demonstrating that direct supervision of latent trajectories is essential for effective exploration. Latent RL further reaches 65.7, 1.9 points above the AR baseline, validating the two-stage design in which supervision first grounds latent exploration and reward guidance then refines the resulting trajectories.

Table 3: Ablation on core components on MMEB-V2. All crosses ($\times\ \times\ \times$) denote the deterministic AR baseline. All reports the overall MMEB-V2 score.

VLR	Latent SFT	Latent RL	Image	Video	VisDoc	All
×	×	×	68.5	43.9	71.3	63.8
×	✓	×	68.9	45.1	72.8	64.6
✓	×	×	67.8	43.3	70.9	63.1
✓	✓	×	69.0	45.5	73.3	64.9
✓	✓	✓	69.5	46.3	74.4	65.7

Table 4: Ablation on training objectives on MMEB-V2. VaME-2B uses the full objective in Equation 8. All reports the overall MMEB-V2 score.

#	Model	Image	Video	VisDoc	All
1	VaME-2B	69.5	46.3	74.4	65.7
2	w/o $\mathcal{L}^{z}_{\mathrm{NCE}}$	68.8	45.1	73.6	64.8
3	w/o $\mathcal{L}^{Z}_{\mathrm{NCE}}$	69.3	46.2	73.6	65.3
4	w/o $\mathcal{L}_{\mathrm{ans}}$	68.8	45.7	73.1	64.8
5	w/o $\mathcal{L}_{\mathrm{KL}}$	68.4	45.2	72.7	64.4
6	w/o $\mathcal{L}_{\mathrm{div}}$	68.7	45.5	73.1	64.7

Ablation on Training Objectives. As shown in Table 4, we ablate each training objective under the same 2B setting while retaining $\mathcal{L}^{e}_{\mathrm{NCE}}$. Removing $\mathcal{L}_{\mathrm{KL}}$ causes the largest drop, from 65.7 to 64.4, highlighting the importance of variational regularization for smooth sampling. Removing $\mathcal{L}_{\mathrm{div}}$ and $\mathcal{L}_{\mathrm{ans}}$ costs 1.0 and 0.9 points, consistent with preventing latent collapse and anchoring retrieval semantics. Removing $\mathcal{L}^{z}_{\mathrm{NCE}}$ or $\mathcal{L}^{Z}_{\mathrm{NCE}}$ costs 0.9 or 0.4 points, showing complementary supervision of the fused latent readout and per-step latent actions. All removals reduce the three modality averages, indicating that each objective contributes to overall performance.

Ablation on Latent RL Rewards. As shown in Table 5, we ablate retrieval reward and SDR under a matched 2B setup. Latent SFT scores 64.9 and 38.3 on MMEB-V2 and MRMR, while retrieval reward raises these scores to 65.2 and 38.5, while SDR reaches 65.3 and 40.7. Combining both rewards improves Image, Video, and VisDoc by 0.5, 0.8, and 1.1 points, respectively, and yields the best MMEB-V2 score of 65.7. Its MRMR score reaches 40.5, 2.2 points above Latent SFT but 0.2 below SDR alone. Overall, SDR substantially improves reasoning-intensive retrieval on MRMR.

Table 5: Ablation on latent RL rewards on MMEB-V2. Latent RL combines the retrieval reward and SDR. All reports the overall MMEB-V2 score.

#	Model	Image	Video	VisDoc	All	MRMR
1	Latent SFT	69.0	45.5	73.3	64.9	38.3
2	w/ ret reward	69.0	45.8	73.8	65.2	38.5
3	w/ SDR	69.1	46.2	74.1	65.3	40.7
4	VaME-2B	69.5	46.3	74.4	65.7	40.5

6 Deep Analysis

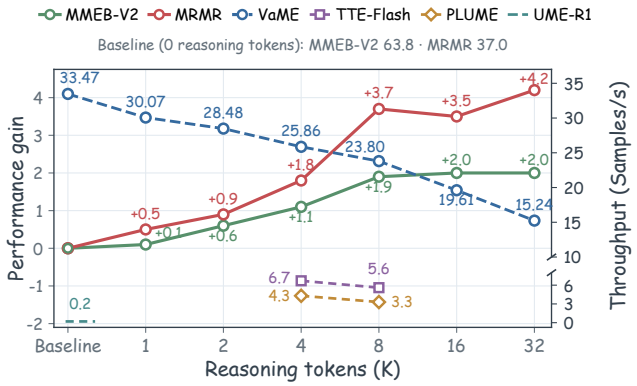

Figure 3: Effect of latent tokens on gains (solid lines) and throughput (dashed lines).



Figure 4: Qualitative latent-rollout visualization.

Analysis of Latent Token Budget. Figure 3 compares performance and throughput across latent token budgets, with throughput measured on a single H800 GPU at batch size 1. At $K = 8$, VaME improves MMEB-V2 and MRMR by 1.9 and 3.7 points over the baseline, respectively, while achieving at least a $4.25\times$ speedup over TTE-Flash and PLUME at the same budget. Larger budgets yield limited additional gains while reducing throughput, supporting $K = 8$ for the main experiments.

Analysis of Trajectory Diversity. As shown in Figure 4, SDR rewards recovering the queried fact. "Tracks" answers what the train is no longer on; "platform" is only contextually related, while "station" gives the wrong location. Likewise, "umbrella" identifies the unusual object omitted by the street description. Yet "platform" earns higher retrieval reward than both correct "tracks" trajectories. Retrieval reward measures embedding separation rather than answer fidelity, explaining this mismatch and motivating SDR as complementary semantic supervision.

7 Conclusion

We present VaME, a variational multimodal embedding framework that models latent reasoning as autoregressive trajectory distributions. Latent SFT grounds trajectories in retrieval relevance and answer semantics, while Latent RL refines them with retrieval feedback and SDR. At inference, one conditional-mean trajectory yields a single embedding without textual reasoning or search. Results on MMEB-V2 and MRMR show that VaME outperforms prior methods in both broad and reasoning-intensive retrieval with substantially faster inference. Future work will explore adaptive computation to further reduce overhead and align stochastic training with deterministic inference.

AI USE STATEMENT

An AI coding assistant was used to draft and revise the manuscript, inspect the local implementation and evaluation summaries, generate vector figures and tables, and check the LaTeX build. An additional AI agent was used to review the draft. No missing experimental results were generated or imputed. Human author review and verification are required before submission.

REFERENCES

Alexander A. Alemi, Ian Fischer, Joshua V. Dillon, and Kevin Murphy. Deep variational information bottleneck. In *Proceedings of the 5th International Conference on Learning Representations (ICLR)*, 2017.

Haonan Chen, Liang Wang, Nan Yang, Yutao Zhu, Ziliang Zhao, Furu Wei, and Zhicheng Dou. mmE5: Improving multimodal multilingual embeddings via high-quality synthetic data. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 8254–8275, 2025. doi: 10.18653/v1/2025.findings-acl.433. URL https://aclanthology.org/2025.findings-acl.433/.

Jianpeng Cheng, Xian Wu, Jiangfan Zhang, Wentao Bao, Chaitanya Ahuja, Shlok Kumar Mishra, Hanchao Yu, Yang Gao, Fan Xia, Qi Guo, Shaodan Zhai, Xiangjun Fan, and Jun Xiao. TTE-Flash: Accelerating reasoning-based multimodal representations via think-then-embed tokens. *arXiv preprint arXiv:2605.16638*, 2026. URL https://arxiv.org/abs/2605.16638v1.

Xuanming Cui, Jianpeng Cheng, Hong-you Chen, Satya Narayan Shukla, Abhijeet Awasthi, Xichen Pan, Chaitanya Ahuja, Shlok Kumar Mishra, Yonghuan Yang, Jun Xiao, Qi Guo, Ser-Nam Lim, Aashu Singh, and Xiangjun Fan. Think then embed: Generative context improves multimodal embedding. *arXiv preprint arXiv:2510.05014*, 2025. URL https://arxiv.org/abs/2510.05014.

Yuntian Deng, Kiran Prasad, Roland Fernandez, Paul Smolensky, Vishrav Chaudhary, and Stuart Shieber. Implicit chain of thought reasoning via knowledge distillation. *arXiv preprint arXiv:2311.01460*, 2023. URL https://arxiv.org/abs/2311.01460.

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*, 2025.

Tiancheng Gu, Kaicheng Yang, Kaichen Zhang, Xiang An, Ziyong Feng, Yueyi Zhang, Weidong Cai, Jiankang Deng, and Lidong Bing. UniME-V2: MLLM-as-a-judge for universal multimodal embedding learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 21378–21386, 2026. doi: 10.1609/aaai.v40i26.39284. URL https://ojs.aaai.org/index.php/AAAI/article/view/39284.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space. In *The Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=Itxz7S4Ip3.

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. *arXiv preprint arXiv:2604.02073*, 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. URL https://arxiv.org/abs/2510.27350.

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. URL https://arxiv.org/abs/2602.13823.

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. URL https://arxiv.org/abs/2407.12580.

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.

Jiajie Jin, Yanzhao Zhang, Mingxin Li, Dingkun Long, Pengjun Xie, Yutao Zhu, and Zhicheng Dou. LaSER: Internalizing explicit reasoning into latent space for dense retrieval. *arXiv preprint arXiv:2603.01425*, 2026. URL https://arxiv.org/abs/2603.01425.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. URL https://arxiv.org/abs/1312.6114.

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*, 2025a. URL https://arxiv.org/abs/2503.04812.

Zhibin Lan, Liqiang Niu, Fandong Meng, Jie Zhou, and Jinsong Su. UME-R1: Exploring reasoning-driven generative multimodal embeddings. *arXiv preprint arXiv:2511.00405*, 2025b. URL https://arxiv.org/abs/2511.00405.

Jin Li, Yaoming Wang, Xiaopeng Zhang, Dongsheng Jiang, Wenrui Dai, Chenglin Li, Hongkai Xiong, and Qi Tian. Contrastive learning via variational information bottleneck. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(9):7410–7427, 2025. doi: 10.1109/TPAMI.2025.3571990.

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.

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 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4015–4025, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Liu_LamRA_Large_Multimodal_Model_as_Your_Advanced_Retrieval_Assistant_CVPR_2025_paper.html.

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. URL https://openreview.net/forum?id=TpU38jbKIJ.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pp. 1278–1286. PMLR, 2014. URL https://proceedings.mlr.press/v32/rezende14.html.

Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. ColBERTv2: Effective and efficient retrieval via lightweight late interaction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 3715–3734, 2022.

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

Zhenyi Shen, Hanqi Yan, Linhai Zhang, Zhanghao Hu, Yali Du, and Yulan He. CODI: Compressing chain-of-thought into continuous space via self-distillation. *arXiv preprint arXiv:2502.21074*, 2025. URL https://arxiv.org/abs/2502.21074.

Wenhui Tan, Jiaze Li, Jianzhong Ju, Zhenbo Luo, Ruihua Song, and Jian Luan. Think silently, think fast: Dynamic latent compression of LLM reasoning chains. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/0706261aedab63814a2b73c32564b4c4-Abstract-Conference.html.

Naftali Tishby, Fernando C. Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000. URL https://arxiv.org/abs/physics/0004057.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

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. URL https://arxiv.org/abs/2604.06156.

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. *arXiv preprint arXiv:2311.17136*, 2023. URL https://arxiv.org/abs/2311.17136.

Peixi Wu, Ke Mei, Feipeng Ma, Bosong Chai, Zhibin Lan, Chenxi Zhao, Shannan Yan, Jie Chen, Zhangchi Hu, Yansong Peng, Bo Lin, Junjie Zhou, Dacheng Yin, Tianyi Wang, Fengyun Rao, Jing Lyu, Hebei Li, and Xiaoyan Sun. Beyond Chain-of-Thought: Rewrite as a Universal Interface for Generative Multimodal Embeddings. *arXiv preprint arXiv:2604.22280*, 2026a. URL https://arxiv.org/abs/2604.22280.

Peixi Wu, Biao Yang, Feipeng Ma, Bosong Chai, Bo Lin, Wei Yuan, Fan Yang, Tingting Gao, Hebei Li, and Xiaoyan Sun. LaME: Learning to Think in Latent Space for Multimodal Embedding via Information Bottleneck. *arXiv preprint arXiv:2606.13061*, 2026b. URL https://arxiv.org/abs/2606.13061.

Qilong Wu, Yiyang Shao, Jun Wang, and Xiaobo Sun. Learning optimal multimodal information bottleneck representations. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025.

Yebo Wu, Feng Liu, Ziwei Xie, Zhiyuan Liu, Changwang Zhang, Jun Wang, and Li Li. TSEmbed: Unlocking task scaling in universal multimodal embeddings. *arXiv preprint arXiv:2603.04772*, 2026c. URL https://arxiv.org/abs/2603.04772.

Zilin Xiao, Qi Ma, Mengting Gu, Chun-cheng Jason Chen, Xintao Chen, Vicente Ordonez, and Vijai Mohan. MetaEmbed: Scaling Multimodal Retrieval at Test-Time with Flexible Late Interaction. *arXiv preprint arXiv:2509.18095*, 2026. URL https://arxiv.org/abs/2509.18095.

Yige Xu, Xu Guo, Zhiwei Zeng, and Chunyan Miao. SoftCoT: Soft chain-of-thought for efficient reasoning with LLMs. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, 2025a.

Yige Xu, Xu Guo, Zhiwei Zeng, and Chunyan Miao. SoftCoT++: Test-time scaling with soft chain-of-thought reasoning. *arXiv preprint arXiv:2505.11484*, 2025b. URL https://arxiv.org/abs/2505.11484.

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. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pp. 6345–6356, 2025. URL https://openaccess.thecvf.com/content/ICCV2025W/Findings/html/Yu_CAFE_Unifying_Representation_and_Generation_with_Contrastive-Autoregressive_Finetuning_ICCVW_2025_paper.html.

Siyue Zhang, Yuan Gao, Xiao Zhou, Yilun Zhao, Tingyu Song, Arman Cohan, Anh Tuan Luu, and Chen Zhao. MRMR: A realistic and expert-level multidisciplinary benchmark for reasoning-intensive multimodal retrieval. *arXiv preprint arXiv:2510.09510*, 2025a. URL https://arxiv.org/abs/2510.09510.

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 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9274–9285, 2025b. doi: 10.1109/CVPR52734.2025.00866. URL https://doi.org/10.1109/CVPR52734.2025.00866.

A Training Data and Setup

Latent SFT retains the public multimodal mixture used by the LaME training corpus (Wu et al., 2026b), itself derived from the VLM2Vec-V2 collection (Meng et al., 2026). Table 6 reports the modality-level composition rather than only the aggregate size. The mixture covers image–text matching, classification, visual question answering, composed retrieval, video QA and retrieval, and document-image retrieval. We reuse existing answer, description, or explanation fields as decoder targets; no intermediate chain-of-thought annotation is required. The initial count refers to records in the merged source pool. The final SFT count is measured after normalized-pair deduplication and an entity-level split that reserves approximately 8% of unique pairs as unseen RL candidates.

Table 6: Statistics of the latent-SFT training data composition.

Dataset	Initial	Final SFT	Ratio	Retrieval direction
Image-based (MMEB-train)				
A-OKVQA	37,929	22,685	59.81%	Image–text → text
CIRR	35,085	32,203	91.79%	Image–text → image–text
ChartQA	39,512	34,962	88.48%	Image–text → text
DocVQA	47,401	42,700	90.08%	Image–text → text
HatefulMemes	16,572	10,612	64.04%	Image–text → text
ImageNet-1K	44,409	40,940	92.19%	Image–text → text
InfographicsVQA	40,746	34,119	83.74%	Image–text → text
MSCOCO	26,429	19,111	72.31%	Image–text → image–text
MSCOCO-i2t	46,596	42,723	91.69%	Image–text → text
MSCOCO-t2i	43,173	39,698	91.95%	Text → image–text
N24News	30,320	26,072	85.99%	Image–text → text
NIGHTS	43,167	25,847	59.88%	Image–text → image–text
OK-VQA	19,900	12,641	63.52%	Image–text → text
SUN397	45,864	33,373	72.77%	Image–text → text
VOC2007	20,454	11,952	58.43%	Image–text → text
Visual7W	41,677	38,479	92.33%	Image–text → text
VisDial	34,652	31,915	92.10%	Text → image–text
VisualNews-i2t	34,364	31,568	91.86%	Image–text → text
VisualNews-t2i	28,684	26,383	91.98%	Text → image–text
WebQA	43,910	26,981	61.45%	Text → image–text
Video-based (LLaVA-Hound)				
Caption Retrieval	284,721	261,925	91.99%	Video → text
Video QA	274,006	252,242	92.06%	Video–text → text
Video Retrieval	260,410	239,462	91.96%	Text → video
Document-based				
ViDoRe	83,964	77,265	92.02%	Image–text → text
VisRAG	60,266	55,462	92.03%	Text → image
Image-based	720,844	584,964	81.15%	Image-centric
Video-based	819,137	753,629	92.00%	Video-centric
Document-based	144,230	132,727	92.02%	Document-centric
Total	**1,684,211**	**1,471,320**	**87.36%**	Multimodal

Latent RL draws from a separately curated pool of approximately 127K multimodal candidates. However, the reported RL run uses approximately 10K records sampled from this pool for policy optimization. Each selected record supplies a query, its paired positive, candidate targets, and a reference answer for semantic scoring. The candidate pool is used only for refinement; MMEB-V2 and MRMR remain held-out evaluation benchmarks.

B Variational Latent Learning

Building on Section 4, this section details the dependency between the two optimization stages. Latent SFT first grounds the stochastic trajectory distribution in retrieval relevance and answer semantics; initialized from this aligned policy, latent RL then uses outcome feedback to favor more discriminative and semantically faithful trajectories.

B.1 Latent SFT Loss Formulation

As shown in Table 7, the latent-SFT objective comprises five groups of losses with complementary roles. The three retrieval losses share the contrastive objective but supervise different representations: $\mathcal{L}^{e}_{\mathrm{NCE}}$ shapes the deployed embedding-token geometry, $\mathcal{L}^{z}_{\mathrm{NCE}}$ makes complete rollouts comparable, and $\mathcal{L}^{Z}_{\mathrm{NCE}}$ directly supervises the step-wise latent actions. $\mathcal{L}_{\mathrm{ans}}$ preserves answer-level semantics, while $\mathcal{L}_{\mathrm{KL}}$ and $\mathcal{L}_{\mathrm{div}}$ regularize the conditional Gaussian trajectory. The main text already defines the embedding-token and fused-rollout objectives. Below, we give the additional implementation details for the position-aligned latent loss, answer reconstruction loss, and Gaussian regularization used throughout latent optimization.

For $\mathcal{L}^{Z}_{\mathrm{NCE}}$, the position-aligned similarity between query and target is

$$s^{Z}(q,t) = \frac{1}{K}\sum_{k=1}^{K}\left\langle \hat{z}^{q}_{k}, \hat{z}^{t}_{k} \right\rangle, \tag{12}$$

where $\hat{z}_k = \mathrm{Norm}(z_k)$ denotes the normalized latent action at step k.

Table 7: Functional role of the latent-SFT objectives.

Objective	Directly supervised object	Training role
$\mathcal{L}^{e}_{\mathrm{NCE}}$	embedding-token readout e	learns the single-vector query–target geometry used by the index
$\mathcal{L}^{z}_{\mathrm{NCE}}$	fused rollout readout z	makes complete sampled trajectories comparable for latent RL
$\mathcal{L}^{Z}_{\mathrm{NCE}}$	position-aligned actions $z_{1:K}$	supplies retrieval supervision throughout the recurrent trajectory
$\mathcal{L}_{\mathrm{ans}}$	post-action states $h_{1:K}$	anchors latent computation to answer-level semantics
$\mathcal{L}_{\mathrm{KL}} + \mathcal{L}_{\mathrm{div}}$	Gaussian parameters $\mu_{1:K}, \sigma_{1:K}$	controls sampling and discourages step-wise latent collapse

$\mathcal{L}^{Z}_{\mathrm{NCE}}$ applies Equation 1 using the pairwise scores $s^{Z}(q,t)$. Unlike $\mathcal{L}^{z}_{\mathrm{NCE}}$, which supervises the fused rollout readout, this term supplies a retrieval signal directly to every position in the latent trajectory. All three retrieval losses use temperature $\tau = 0.02$ and the same masked, hard-negative-weighted implementation.

For $\mathcal{L}_{\mathrm{ans}}$, a decoder p_ψ conditions on the fixed prompt $\mathcal{I}_{\mathrm{dec}}$ and post-action states H:

$$\mathcal{L}^{(s)}_{\mathrm{ans}} = -\frac{1}{N_s} \sum_{i \in \mathcal{B}_s} \sum_{t \in \mathcal{V}_i} \log p_\psi(y^i_t \mid y^i_{<t}, \mathcal{I}_{\mathrm{dec}}, H_i), \tag{13}$$

where $\mathcal{B}_s$ is the batch for logical side s, $\mathcal{V}_i$ contains valid answer-token positions, and $N_s = \sum_i |\mathcal{V}_i|$. The loss is averaged across supervised sides. Padding and prefix positions are excluded; retained end tokens are included. Training annotations provide the answer targets, not intermediate chain-of-thought text. The specific form of the decoder prompt $\mathcal{I}_{\mathrm{dec}}$ is shown in Figure 5.

Fixed instruction Decode the following latent representations into the answer.
Continuous prefix $P(h_1)\ P(h_2)\ \cdots\ P(h_K)$
Assistant target $y_1\ y_2\ \cdots\ y_T\ \langle\mathrm{eos}\rangle$

Figure 5: Illustration of the decoder prompt.

For $\mathcal{L}_{\mathrm{KL}}$, two learned projections parameterize the diagonal Gaussian at latent step k, and the closed-form standard-normal regularizer is

$$\mathcal{L}_{\mathrm{KL}} = \frac{1}{2Kd} \sum_{k=1}^{K} \sum_{j=1}^{d} \left(\mu^2_{kj} + \sigma^2_{kj} - 1 - 2\log \sigma_{kj}\right). \tag{14}$$

where K is the number of latent steps and d is the latent-state dimension. The KL term is averaged over examples and both retrieval sides. Its relation to the conditional answer-likelihood bound is derived in Appendix C. $\mathcal{L}_{\mathrm{div}}$ is the mean absolute off-diagonal cosine similarity among posterior means and discourages step-wise latent collapse.

B.2 Latent RL Training Details

We clarify several details of latent RL training below. For retrieval feedback, the target candidates follow deterministic mean trajectories and remain fixed within each rollout group. The retrieval reward can be written as

$$R_{\mathrm{ret}} = s^{+}_{g} - \sum_{j \in \mathcal{N}} \omega_{g,j} s^{-}_{g,j}. \tag{15}$$

The negative weights are $\omega_{g,j} = \mathrm{softmax}_{j \in \mathcal{N}}(s^{-}_{g,j}/\tau_r)$, with reward temperature τ_r. For SDR, the frozen decoder greedily generates one answer per trajectory, and the fixed evaluator scores reference-answer agreement in $[0,1]$. For each query, normalize the G rewards to obtain the advantage $A_g =$

$(R(Z_g) - \bar{R})/(\mathrm{std}_{g=1}^{G}[R(Z_g)] + 10^{-6})$. At latent step k, the policy ratio is

$$\rho_{g,k} = \exp\left[\frac{1}{d}\sum_{j=1}^{d}\log\frac{\pi_{\boldsymbol{\theta}}(z_{g,k,j} \mid q, z_{g,<k})}{\pi_{\mathrm{old}}(z_{g,k,j} \mid q, z_{g,<k})}\right], \tag{16}$$

where j indexes the d action dimensions. To control the numerical scale, we average the log-density ratio over the action dimensions. The RL objective is

$$\begin{aligned}\mathcal{L}_{\mathrm{RL}} = \mathbb{E}_{g,k}\Big[&-\min\big\{\rho_{g,k}A_g, \mathrm{clip}(\rho_{g,k}, 1-\epsilon_-, 1+\epsilon_+)A_g\big\} \\ &+ \beta D_{\mathrm{KL}}(\pi_{\boldsymbol{\theta}}(\cdot \mid q, z_{g,<k}) \| \pi_{\mathrm{ref}}(\cdot \mid q, z_{g,<k}))\Big].\end{aligned} \tag{17}$$

The clipped term favors higher-reward trajectories, while the KL term keeps the policy close to the frozen SFT reference π_{ref}. Specifically, our experiments use $G = 8$, $\lambda_{\mathrm{SDR}} = 0.2$, $\tau_r = 0.2$, $(\epsilon_-, \epsilon_+) = (0.2, 0.28)$, and $\beta = 0.01/d$ in Equation 17.

Semantic decoding reward. The SDR component uses a fixed grading contract. The rubric judge J receives the task question, reference answer y, and decoded candidate, and follows a fixed instruction $\mathcal{I}_{\mathrm{eval}}$: decompose the reference into question-relevant atomic facts and score their correctness in the candidate, accepting paraphrases but penalizing factual contradictions, reversed relations, and incorrect numerical values, without using external knowledge or rewarding style. The judge greedily emits a single token constrained to $\{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}, \mathtt{f}\}$, mapped to the scalar reward in Table 8; each latent trajectory is decoded once without sampling and receives one such score.

Table 8: SDR grade-to-reward mapping.

Grade	Reward	Criterion
a	0.00	No requested fact is correct, or the sole atomic answer is wrong, contradictory, irrelevant, or absent.
b	0.25	Only a peripheral fragment is correct; the requested answer is mostly wrong.
c	0.50	A meaningful subset of facts is correct, but another substantive fact is wrong or missing.
d	0.75	All core facts are correct, with only a minor non-core defect.
f	1.00	All answer facts are correct, with no contradiction.

B.3 Configuration Details

As shown in Table 9, latent SFT starts from Qwen2-VL and trains for 3,200 steps with an effective batch size of 512 pairs. Latent RL then refines the aligned checkpoint for 500 steps on approximately 10K records, using an effective batch size of 64 queries and a learning rate of 10^{-6}. Both stages use $K = 8$ latent steps, with RL sampling $G = 8$ trajectories per query.

Table 9: Optimization settings for the reported runs.

Setting	Latent SFT	Latent RL
Initialization	Qwen2-VL-2B/7B	aligned SFT checkpoint
Training pool	$\sim$1.55M pairs	$\sim$127K candidates
Records used for RL training	–	$\sim$10K records
Optimizer steps	3,200	500
Effective batch size	512 pairs	64 queries
Latent steps K	8	8
Trajectories per query G	1 sampled trajectory	8 sampled trajectories
Backbone learning rate	1×10^{-5}	1×10^{-6}
New-module learning rate	2×10^{-5}	1×10^{-6}
Contrastive/reward temperature	$\tau = 0.02$	$\tau_r = 0.2$
Other coefficients	$\lambda_z = \lambda_{\mathrm{ans}} = 1$, $\lambda_{\mathrm{div}} = 0.1$	$\lambda_{\mathrm{SDR}} = 0.2$
Policy clipping	–	$[0.80, 1.28]$
Precision / devices	bfloat16 / 8 GPUs	bfloat16 / 8 GPUs

C Proof of the Variational Lower Bound

This section proves the variational lower bound underlying the latent model, derives its autoregressive trajectory KL, and relates the result to the normalized training losses.

C.1 Conditional Model and Lower Bound

Fix an input–answer pair (x, y) and the parameters. Given $Z = (z_1, \dots, z_K)$, the recurrence deterministically produces $H_\theta(x, Z)$. Abbreviate the decoder likelihood as $r(y \mid x, Z) = p_\psi(y \mid \mathcal{I}_{\mathrm{dec}}, H_\theta(x, Z))$. It is the product of conditional probabilities of the supervised answer tokens, including retained end tokens; prompt and padding positions are excluded. For a truncated answer, it denotes the observed-prefix probability. With $p_0(Z) = \prod_{k=1}^{K} \mathcal{N}(z_k; 0, I)$, define

$$\begin{aligned} p_{\theta,\psi}(y, Z \mid x) &= p_0(Z)\, r(y \mid x, Z), \\ p_{\theta,\psi}(y \mid x) &= \int p_0(Z)\, r(y \mid x, Z)\, dZ. \end{aligned} \tag{18}$$

The decoder defines a normalized answer distribution and retains access to x through $H_\theta(x, Z)$. The prior p_0 is distinct from the learned rollout distribution $q(Z) = q_{\boldsymbol{\theta}}(Z \mid x)$ in Equation 3. Assume finite conditional Gaussian means, $0 < \sigma_{kj} < \infty$, positive evidence, and finite expected negative log-likelihood and trajectory KL. Then q has a positive density throughout $\mathbb{R}^{Kd}$.

Proof. Rewriting the marginal as an expectation under q and applying Jensen's inequality gives

$$\begin{aligned} \log p_{\theta,\psi}(y \mid x) &= \log \mathbb{E}_{Z\sim q}\left[\frac{p_0(Z) r(y \mid x, Z)}{q(Z)}\right] \\ &\geq \mathbb{E}_{Z\sim q}[\log r(y \mid x, Z) + \log p_0(Z) - \log q(Z)] \\ &= \mathbb{E}_{Z\sim q} \log r(y \mid x, Z) - D_{\mathrm{KL}}(q \| p_0) \;\equiv\; \mathcal{E}(x, y). \end{aligned} \tag{19}$$

By Bayes' rule, $p_{\theta,\psi}(Z \mid x, y) = p_0(Z) r(y \mid x, Z) / p_{\theta,\psi}(y \mid x)$. Substitution into the posterior KL gives the exact gap:

$$\log p_{\theta,\psi}(y \mid x) - \mathcal{E}(x, y) = D_{\mathrm{KL}}(q(Z) \,\|\, p_{\theta,\psi}(Z \mid x, y)) \geq 0. \tag{20}$$

Equality holds if and only if $q(Z) = p_{\theta,\psi}(Z \mid x, y)$ almost everywhere. This proves Equation 19. □

The identity holds at each fixed parameter value, including shared θ. Conditioning q only on x restricts the variational family and may loosen the bound, but does not invalidate it. No Jacobian of H_θ is needed: the integral is over actions Z, not a density over hidden states.

C.2 Autoregressive Trajectory KL

Let $q_k(\cdot \mid x, z_{<k}) = q_{\boldsymbol{\theta}}(\cdot \mid x, z_{<k})$ and $p_{0,k} = \mathcal{N}(0, I)$. Expanding the autoregressive density ratio and applying iterated expectation yields

$$\begin{aligned} D_{\mathrm{KL}}(q \| p_0) &= \mathbb{E}_{Z\sim q} \sum_{k=1}^{K} \log \frac{q_k(z_k \mid x, z_{<k})}{p_{0,k}(z_k)} \\ &= \sum_{k=1}^{K} \mathbb{E}_{z_{<k}\sim q}\left[\mathbb{E}_{z_k \sim q_k(\cdot \mid x, z_{<k})} \log \frac{q_k(z_k \mid x, z_{<k})}{p_{0,k}(z_k)}\right] \\ &= \sum_{k=1}^{K} \mathbb{E}_{z_{<k}\sim q} D_{\mathrm{KL}}(q_k(\cdot \mid x, z_{<k}) \| p_{0,k}) \,. \end{aligned} \tag{21}$$

Future actions integrate out to one; the prefix at $k = 1$ is empty. These are expected conditional KLs, not KLs of the marginal action distributions. At a fixed prefix, the Gaussian moments $\mathbb{E}[(z_{kj} - \mu_{kj})^2] = \sigma_{kj}^2$ and $\mathbb{E}[z_{kj}^2] = \mu_{kj}^2 + \sigma_{kj}^2$ give

$$D_{\mathrm{KL}}(q_k \| p_{0,k}) = \frac{1}{2} \sum_{j=1}^{d} \left(\mu_{kj}^2 + \sigma_{kj}^2 - 1 - 2 \log \sigma_{kj}\right). \tag{22}$$

Consequently, Equations 4 and 14 imply

$$D_{\mathrm{KL}}(q\|p_0) = Kd\,\mathbb{E}_{Z\sim q}\mathcal{L}_{\mathrm{KL}}(x,Z). \tag{23}$$

Although each conditional KL is analytic, its parameters depend on the sampled prefix. A single-rollout $\mathcal{L}_{\mathrm{KL}}$ is therefore an unbiased estimate of the trajectory KL divided by Kd. Likewise, $\log r(y \mid x, Z) - Kd\,\mathcal{L}_{\mathrm{KL}}(x,Z)$ has expectation $\mathcal{E}(x,y)$; an individual sampled value need not be a lower bound.

C.3 Relation to the Normalized Training Losses

Let $\mathcal{L}_{\mathrm{ans}}^{\mathrm{seq}}$ denote the expected negative log-likelihood per supervised answer sequence in Equation 7. Averaging the bound gives

$$-\,\mathbb{E}_{(x,y)}\mathcal{E}(x,y) = \mathcal{L}_{\mathrm{ans}}^{\mathrm{seq}} + Kd\,\mathbb{E}_{x,\,Z\sim q_{\boldsymbol{\theta}}(\cdot|x)}\mathcal{L}_{\mathrm{KL}}(x,Z). \tag{24}$$

For a fixed supervised-side batch $\mathcal{B}_s$, write $m_s = |\mathcal{B}_s| > 0$, $N_s > 0$ for its valid-token count, and $\ell_s = N_s/m_s$. Define the per-example averages $\overline{\mathcal{E}}_s = m_s^{-1}\sum_{i\in\mathcal{B}_s}\mathcal{E}(x_i,y_i)$ and $\mathcal{L}_{\mathrm{KL}}^{(s)} = m_s^{-1}\sum_{i\in\mathcal{B}_s}\mathcal{L}_{\mathrm{KL}}(x_i,Z_i)$. The token loss in Equation 13 divides the total answer negative log-likelihood by N_s rather than m_s, so

$$-\,\overline{\mathcal{E}}_s = \ell_s\,\mathbb{E}_Z\mathcal{L}_{\mathrm{ans}}^{(s)} + Kd\,\mathbb{E}_Z\mathcal{L}_{\mathrm{KL}}^{(s)}. \tag{25}$$

This identity is exact even for unequal answer lengths. For multiple supervised sides, average the right-hand side over sides, retaining each ℓ_s. When lengths vary across batches, this factor must remain inside the batch expectation. Answer and KL terms must cover the same examples and sides; KL on an unsupervised side is additional regularization.

For $\lambda_{\mathrm{ans}} > 0$, matching the answer-plus-KL objective to $-\overline{\mathcal{E}}_s$ up to a positive scale requires

$$\frac{\lambda_{\mathrm{KL}}}{\lambda_{\mathrm{ans}}} = \frac{Kd}{\ell_s}, \qquad \mathbb{E}_Z\left[\lambda_{\mathrm{ans}}\mathcal{L}_{\mathrm{ans}}^{(s)} + \lambda_{\mathrm{KL}}\mathcal{L}_{\mathrm{KL}}^{(s)}\right] = -\frac{\lambda_{\mathrm{ans}}}{\ell_s}\overline{\mathcal{E}}_s. \tag{26}$$

A fixed ratio cannot generally match this scaling across sides or batches with different ℓ_s. Equation 8 independently weights the normalized losses and adds retrieval and diversity supervision. Thus the proof supplies a variational interpretation of the answer and KL terms, without equating the complete SFT objective to the standard negative ELBO. The bound concerns stochastic trajectories; deterministic mean-path inference is not the marginalization in Equation 18.