Beyond One-Geometry-Fits-All: State-Conditioned Dual-Path Reconstruction for Missing-Modality PEFT

Anonymous authors
Paper under double-blind review

Abstract

Parameter-efficient fine-tuning (PEFT) for multimodal missing-modality adaptation typically reuses a single low-rank adaptation structure across all availability states. However, independently optimized adapters exhibit systematically larger cross-state than same-state, cross-seed subspace separation. Assigning independent adapters per state enhances state-specific expressiveness but scales linearly with the number of states and may weaken cross-state knowledge transfer through parameter isolation; a fixed shared structure is more efficient, yet its limited rank budget cannot simultaneously cover all state-preferred update directions. We formalize this Capacity-Transfer Tension geometrically, establishing a constructive representational separation between CODAR and fixed shared rank-r geometry via subspace coverage and reconstruction analyses. To address it, we propose Conditional Dual-Path Adapter Reconstruction (CODAR), which reconstructs both low-rank factors from reusable dictionaries while conditioning their composition on modality availability. A modality-specific pathway captures state-dependent variation, whereas a transferable pathway reuses shared input-side directions across towers. Across MM-IMDb, UPMC-Food101, and Hateful Memes, CODAR consistently improves over representative PEFT baselines with 101,116 trainable adapter parameters. Controlled comparisons and component ablations further show that the gains cannot be explained by availability conditioning or parameter count alone, while geometric analyses show that CODAR realizes state-dependent update subspaces while retaining shared primitives. Code is available at https://anonymous.4open.science/r/anonymous_code_-48D6/.

1 Introduction

Parameter-efficient fine-tuning (PEFT) is widely used to adapt multimodal foundation models such as CLIP (Radford et al., 2021) to downstream tasks (Hu et al., 2022; Zhou et al., 2024). In deployment, however, sensor failures, transmission interruptions, or privacy constraints (Ma et al., 2022; Choi & Lee, 2019) may render modalities unavailable, substantially degrading performance (Wang et al., 2024b). This raises a natural question: **should the PEFT parameterization itself vary with modality availability?** Rather than asking only how much of a PEFT module to share, we ask what geometry to share: **should complete, image-missing, and text-missing inputs be constrained to the same low-rank row and column subspaces?**

Relevant PEFT methods broadly follow two directions. State-aware methods encode missing patterns: MMP (Lee et al., 2023) uses learnable prompts without changing the weight-update parameterization, whereas MoRA (Zhao et al., 2026) combines modality-common low-rank parameters shared across vision and text encoders with modality-specific components. Parameter-sharing methods instead emphasize compact reconstruction through state-invariant shared structures (Renduchintala et al., 2024): VeRA (Kopiczko et al., 2024) shares frozen random matrices across layers; VB-LoRA (Li et al., 2024b) uses a shared vector bank with differentiable Top-K routing; and Uni-LoRA (Li et al., 2025) reconstructs network-wide updates from a shared basis. Neither direction directly examines whether modality availability should alter the update subspaces themselves.

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.

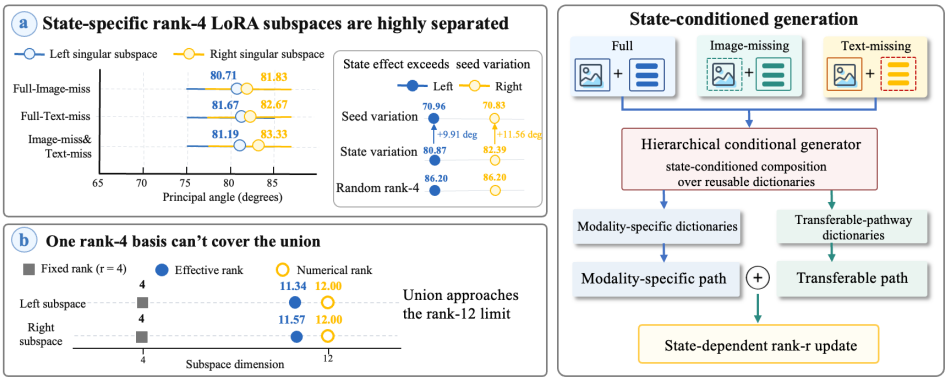

Figure 1: **Motivation and overview of CODAR. Left:** Independently trained rank-4 LoRA updates show cross-state subspace separation beyond seed variation, with effective union ranks of 11.34/11.57 (left/right). **Right:** CODAR reconstructs state-dependent updates through modality-specific and transferable pathways over reusable dictionaries.

We examine this question geometrically (Fig. 1). On Hateful Memes, independently trained rank-4 LoRA adapters exhibit larger left/right principal angles across states than across seeds within the same state ($80.87°/82.39°$ vs. $70.96°/70.83°$). The corresponding effective union ranks are $11.34/11.57$ vs. $10.70/10.75$. These results show systematic availability-dependent separation beyond seed variability, with the updates jointly spanning substantially more than a single rank-4 subspace pair can cover. In the mixed-trained diagnostic, StateSpecificLoRA uses more parameters yet underperforms state-agnostic FixedBasis on complete and missing-modality inputs (Fig. 4(a)). Since the adapters observe different state subsets, this serves as empirical motivation for the Capacity-Transfer Tension rather than a strict causal proof.

To address this tension, we propose **Conditional Dual-Path Adapter Reconstruction (CODAR)**, which separates reusable low-rank directions from their state-dependent composition. A hierarchical generator conditions routing weights for both low-rank factors on layer, tower, factor role, and modality availability. A modality-specific pathway captures state-dependent variation, while a transferable pathway reuses shared input-side directions across towers. Together, they enable state-dependent update geometry without per-state adapter copies.

Contributions: (1) we provide an update-subspace characterization of availability dependence in image–text missing-modality PEFT, revealing systematic cross-state subspace separation beyond seed-to-seed variation, deriving a reconstruction lower bound for fixed shared rank-r geometry, and establishing a constructive representational separation for CODAR; (2) we introduce CODAR, which conditions both low-rank factors on modality availability while reconstructing them through reusable modality-specific and cross-tower primitives; and (3) through parameter-matched controls, component ablations, and geometric diagnostics across three benchmarks, together with cross-scenario evaluation on Hateful Memes, we show that structured state-conditioned reconstruction provides benefits beyond availability conditioning alone.

2 Problem Setup and Geometric Analysis

Problem Formulation. Let $\mathbf{W}_l^\mu \in \mathbb{R}^{d_{\text{out}}^\mu \times d_{\text{in}}^\mu}$ be a frozen projection at layer $l \in \{1, \dots, L_\mu\}$ of tower $\mu \in \{\text{vis}, \text{txt}\}$. We suppress the projection index; module-local parameters remain distinct for each adapted layer–projection pair. For availability state $m \in \mathcal{M} = \{\text{full}, \text{vis-miss}, \text{txt-miss}\}$, parameter-efficient adaptation learns a rank-constrained update (Zhang et al., 2023; Liu et al., 2024): $\mathbf{W}_l^{\mu,m} = \mathbf{W}_l^\mu + \frac{\alpha}{r}\Delta\mathbf{W}_l^{\mu,m}, \text{rank}(\Delta\mathbf{W}_l^{\mu,m}) \le r$. Here, r is the update-rank budget and α its fixed scaling factor. A state-agnostic adapter sets $\Delta\mathbf{W}_l^{\mu,m} = \Delta\mathbf{W}_l^\mu$ for all m, forcing a shared update subspace.

Cross-State Subspace Separation. The separation observed in Fig. 1 motivates a narrower representational question: given state-specific reference updates, how much of their row and column geometry can the best shared rank-r subspace pair cover?

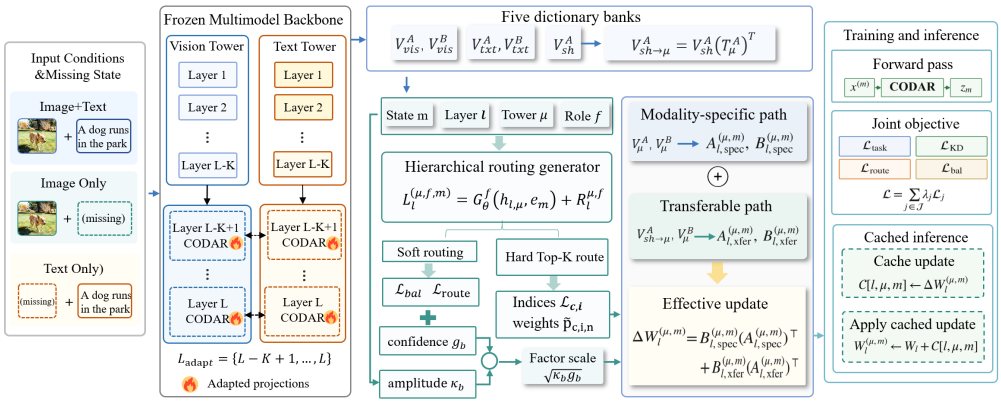

Figure 2: **CODAR architecture.** A hierarchical generator conditions Top-K routing on layer, tower, factor role, and modality availability. Modality-specific and transferable pathways reconstruct each rank-r update from reusable dictionaries, with bounded layer-specific residuals providing local corrections.

Capacity-Transfer Quantification. Let $r^{\mu,m}_l = \text{rank}(\Delta\mathbf{W}^{\mu,m}_l) \le r$, and let $\mathbf{Q}^{(\mu,m)}_l \in \mathbb{R}^{d^\mu_{\text{out}} \times r^{\mu,m}_l}$ and $\mathbf{U}^{(\mu,m)}_l \in \mathbb{R}^{d^\mu_{\text{in}} \times r^{\mu,m}_l}$ be orthonormal bases for its column and row spaces. Concatenate them over states as $\mathbf{Q}^\mu_{l,\text{all}}$ and $\mathbf{U}^\mu_{l,\text{all}}$. With $s^\mu_l = \sum_m r^{\mu,m}_l$, define $q^\mu_{l,\text{col}} = \min(d^\mu_{\text{out}}, s^\mu_l)$ and $q^\mu_{l,\text{row}} = \min(d^\mu_{\text{in}}, s^\mu_l)$.

By the Eckart–Young theorem (Eckart & Young, 1936), the minimum column-space coverage error over rank-r shared subspaces is

$$\varepsilon^{\mu,\text{col}}_l = \min_{\mathbf{Z}^\top\mathbf{Z}=\mathbf{I}_r} \sum_m \left\|(\mathbf{I} - \mathbf{Z}\mathbf{Z}^\top)\mathbf{Q}^{(\mu,m)}_l\right\|^2_F = \sum_{j=r+1}^{q^\mu_{l,\text{col}}} \sigma^2_j\left(\mathbf{Q}^\mu_{l,\text{all}}\right). \tag{1}$$

The analogous row-space error is $\varepsilon^{\mu,\text{row}}_l$. The normalized coverage deficit

$$\rho^\mu_l = \frac{\varepsilon^{\mu,\text{col}}_l + \varepsilon^{\mu,\text{row}}_l}{\sum_{j=1}^{q^\mu_{l,\text{col}}} \sigma^2_j(\mathbf{Q}^\mu_{l,\text{all}}) + \sum_{j=1}^{q^\mu_{l,\text{row}}} \sigma^2_j(\mathbf{U}^\mu_{l,\text{all}})} \tag{2}$$

measures the fraction of concatenated-basis energy uncovered by the best shared rank-r subspace pair. We set $\rho^\mu_l = 0$ when all target updates are zero and interpret empty sums as zero. The near-maximal observed union ranks further show that the state-specific updates collectively span substantially more than a single rank-4 geometry.

Energy-weighted reconstruction consequence. The coverage deficit treats subspace directions uniformly. To account for the actual update energy, let $\mathbf{D}_m = \Delta\mathbf{W}^{(\mu,m)}_l$ and consider the broader fixed-geometry family $\widehat{\mathbf{D}}_m = \mathbf{Z}\mathbf{\Gamma}_m\mathbf{Y}^\top$, where $\mathbf{Z}^\top\mathbf{Z} = \mathbf{Y}^\top\mathbf{Y} = \mathbf{I}_r$ and $\mathbf{\Gamma}_m \in \mathbb{R}^{r\times r}$ may vary freely with m. Define

$$\mathbf{S}_{\text{col}} = \sum_m \mathbf{D}_m\mathbf{D}^\top_m, \quad \mathbf{S}_{\text{row}} = \sum_m \mathbf{D}^\top_m\mathbf{D}_m. \tag{3}$$

Then the minimum aggregate reconstruction error under fixed shared rank-r row/column geometry satisfies

$$\mathcal{E}_{\text{fix}} := \min_{\substack{\mathbf{Z}^\top\mathbf{Z}=\mathbf{I}_r,\ \mathbf{Y}^\top\mathbf{Y}=\mathbf{I}_r \\ \{\mathbf{\Gamma}_m\}}} \sum_m \left\|\mathbf{D}_m - \mathbf{Z}\mathbf{\Gamma}_m\mathbf{Y}^\top\right\|^2_F \ge \max\left\{\sum_{j>r}\lambda_j(\mathbf{S}_{\text{col}}), \sum_{j>r}\lambda_j(\mathbf{S}_{\text{row}})\right\}. \tag{4}$$

The eigenvalues are ordered non-increasingly. Thus, nonzero spectral tail energy implies unavoidable reconstruction error even with arbitrary state-specific coefficients inside the shared geometry. The derivation is given in Appendix B.

Observation 1 (Capacity–Transfer Tension)**.** A nonzero coverage deficit means that no shared rank-r subspace pair can exactly cover all state-specific update directions. Whenever the corresponding spectral tail is nonzero, even a broader fixed-geometry family with state-specific coefficients

incurs unavoidable reconstruction error. Conversely, our ablations show that fully independent state adapters may weaken cross-state transfer. Together, these effects define the **Capacity–Transfer Tension** between geometric expressiveness and cross-state sharing.

These are representational statements about the observed target updates and do not imply a lower bound on downstream task performance.

CODAR therefore requires **(R1)** state-conditioned update geometry whose union across states can exceed rank r, **(R2)** shared learnable building blocks for cross-state reuse, and **(R3)** state-dependent composition without storing a separate full adapter for each state.

3 Method

3.1 Architectural Overview

Figure 2 summarizes CODAR, which separates reusable low-rank directions, represented by dictionary atoms, from their state-dependent composition through routing tensors. This realizes R1–R3 by allowing the union of state-specific subspaces to exceed rank r while reusing shared atoms without duplicating full adapters. The hierarchical generator, sparse routing, bounded amplitudes, and regularizers provide a parameter-efficient and stable realization; implementation details appear in Appendix A.

Dictionary banks. CODAR uses modality-specific banks $\mathbf{V}_\mu^A \in \mathbb{R}^{N\times d_\mu^A}$ and $\mathbf{V}_\mu^B \in \mathbb{R}^{N\times d_\mu^B}$ for $\mu \in \{\mathrm{vis}, \mathrm{txt}\}$, together with a transferable bank $\mathbf{V}_{\mathrm{sh}}^A \in \mathbb{R}^{N\times d_s^A}$ shared across towers. Tower-specific projections $\mathbf{T}_\mu^A \in \mathbb{R}^{d_\mu^A \times d_s^A}$ produce $\mathbf{V}_{\mathrm{sh}\to\mu}^A = \mathbf{V}_{\mathrm{sh}}^A(\mathbf{T}_\mu^A)^\top$. Here, N is the bank size, $d_\mu^{A/B}$ the chunk widths, and d_s^A the shared-atom width; $N \gg r$ decouples bank capacity from update rank.

Why input-side sharing? Only input-side atoms are shared, while output dictionaries remain tower-specific. Thus, the transferable pathway uses $\mathbf{V}_{\mathrm{sh}\to\mu}^A$ for A_{xfer} and $\mathbf{V}_\mu^B$ for B_{xfer}, rather than sharing both factor banks.

Role-to-dictionary map. For role $f \in \{A_{\mathrm{spec}}, B_{\mathrm{spec}}, A_{\mathrm{xfer}}, B_{\mathrm{xfer}}\}$ and tower μ:

$$\mathcal{V}(\mu, f) = \begin{cases} \mathbf{V}_\mu^A, & f = A_{\mathrm{spec}}, \\ \mathbf{V}_\mu^B, & f \in \{B_{\mathrm{spec}}, B_{\mathrm{xfer}}\}, \\ \mathbf{V}_{\mathrm{sh}\to\mu}^A, & f = A_{\mathrm{xfer}}. \end{cases} \tag{5}$$

Each atom is row-wise ℓ_2-normalized before routing: $\widehat{\mathcal{V}}_n(\mu, f) = \mathcal{V}_n(\mu, f)/\|\mathcal{V}_n(\mu, f)\|_2$, separating direction from magnitude.

3.2 State-Conditioned Dual-Path Update

For module l in tower μ and availability state m,

$$\Delta\mathbf{W}_l^{(\mu,m)} = \mathbf{B}_{l,\mathrm{spec}}^{(\mu,m)}\big(\mathbf{A}_{l,\mathrm{spec}}^{(\mu,m)}\big)^\top + \mathbf{B}_{l,\mathrm{xfer}}^{(\mu,m)}\big(\mathbf{A}_{l,\mathrm{xfer}}^{(\mu,m)}\big)^\top. \tag{6}$$

For $b \in \{\mathrm{spec}, \mathrm{xfer}\}$, $\mathbf{A}_{l,b}^{(\mu,m)} \in \mathbb{R}^{d_{\mathrm{in}}^\mu \times r_b}$ and $\mathbf{B}_{l,b}^{(\mu,m)} \in \mathbb{R}^{d_{\mathrm{out}}^\mu \times r_b}$, with $r_{\mathrm{spec}} + r_{\mathrm{xfer}} = r$.

Chunk-wise factorization. We factorize $d_{\mathrm{in}}^\mu = Pd_\mu^A$ and $d_{\mathrm{out}}^\mu = Qd_\mu^B$, where P, Q are the input/output chunk counts. This reduces dictionary parameters from $\mathcal{O}(Nd_{\mathrm{in/out}}^\mu)$ to $\mathcal{O}(Nd_\mu^{A/B})$ while retaining chunk-specific atom selection.

Sparse top-K routing. For factor role f, we reconstruct unscaled directions as $\widetilde{\mathbf{F}}_l^{(\mu,f,m)} = \mathrm{Route}(\mathbf{L}_l^{\mu,f,m}, \widehat{\mathcal{V}}(\mu, f))$, where $\mathbf{L}_l^{\mu,f,m} \in \mathbb{R}^{C_f \times r_f \times N}$. Set $(C_f, d_f) = (P, d_\mu^A)$ for A roles and (Q, d_μ^B) for B roles, with $r_f = r_b$ for $f \in \{A_b, B_b\}$. Route stacks chunk-wise dictionary mixtures into $\widetilde{\mathbf{F}}_l^{(\mu,f,m)} \in \mathbb{R}^{(C_f d_f)\times r_f}$.

For each chunk c and rank coordinate i, we select the K atoms with the largest routing logits and mix them using a softmax over the selected logits at temperature $\tau_r > 0$. During training, the routing

weights $\widetilde{\mathbf{p}}_{c,i}$ use this sparse distribution in the forward pass and dense-softmax gradients through a straight-through estimator (Bengio et al., 2013). Our implementation uses dense dictionary contraction and retains dense soft weights for STE, so sparse support alone provides no computational or training-memory savings. Full routing definitions and implementation details appear in Appendix A.

Pathway amplitudes. To restore magnitude after atom normalization, pathway b uses bounded rank-wise amplitudes conditioned on $\mathbf{c}_{l,\mu,m} = [\mathbf{e}_l^{\mu}; \mathbf{e}_{\mu}; \mathbf{e}_m] \in \mathbb{R}^{d_c}$. The layer, tower, and state embeddings have dimensions d_h, d_h, d_s, respectively, with $d_c = 2d_h + d_s$. A two-layer MLP produces $\kappa_{l,i}^{\mu,b,m} \in [e^{-\delta_\kappa}, e^{\delta_\kappa}]$ for fixed $\delta_\kappa > 0$; its parameterization and initialization are given in Appendix A. For factor role $f \in \{A_b, B_b\}$, each chunk is reconstructed as

$$\mathbf{b}_{c,i} = \sqrt{\kappa_{l,i}^{\mu,b,m}\, g_{l,i}^{\mu,b,m}}\; \widehat{\mathcal{V}}(\mu, f)^{\top} \widetilde{\mathbf{p}}_{c,i}. \tag{7}$$

Stacking $\mathbf{b}_{c,i} \in \mathbb{R}^{d_f}$ over (c, i) yields $\mathbf{F}_l^{(\mu,f,m)}$. The confidence gate $g_{l,i}^{\mu,b,m}$ is defined in §3.4; applying $\sqrt{\kappa g}$ to both factors scales each rank-one component by exactly κg.

Theorem 1 (Constructive Separation from Fixed Geometry)**.** *Fix a layer l and tower μ, and let $M = |\mathcal{M}| \geq 2$, $t = r_{\text{xfer}}$, $s = r_{\text{spec}}$, and $r = t + s$. Suppose*

$$N \geq K(t + Ms), \qquad R_g \geq \max(t, s), \qquad d_s \geq M,$$

and $d_s^A \geq t$ and $d_\mu^A, d_\mu^B \geq t + Ms$. Then there exist positive constants $c_{\text{xfer}}, c_{\text{spec}} > 0$ and a family of rank-r target updates

$$\mathbf{D}_m = c_{\text{xfer}} \mathbf{U}_0 \mathbf{V}_0^{\top} + c_{\text{spec}} \mathbf{U}_m \mathbf{V}_m^{\top}, \tag{8}$$

where $\mathbf{U}_0, \mathbf{V}_0$ have t columns, $\mathbf{U}_m, \mathbf{V}_m$ have s columns, and the columns of $[\mathbf{U}_0, \mathbf{U}_1, \ldots, \mathbf{U}_M]$ and $[\mathbf{V}_0, \mathbf{V}_1, \ldots, \mathbf{V}_M]$ are respectively orthonormal, such that CODAR realizes every $\mathbf{D}_m$ exactly.

For the fixed-geometry family in Eq. (4), let $\nu_1 \geq \cdots \geq \nu_{t+Ms} > 0$ be the sorted multiset containing Mc_{xfer}^2 repeated t times and c_{spec}^2 repeated Ms times. Then

$$\mathcal{E}_{\text{fix}} = \sum_{j=r+1}^{t+Ms} \nu_j \geq (M-1)s \min\{Mc_{\text{xfer}}^2, c_{\text{spec}}^2\} > 0. \tag{9}$$

Hence, CODAR admits target-update families that no fixed shared rank-r row/column geometry can represent exactly.

Proof sketch. The construction contains t directions shared across states and s mutually orthogonal state-specific directions per state. Its row- and column-energy operators therefore have eigenvalues Mc_{xfer}^2 on the shared directions and c_{spec}^2 on the state-specific ones. Since $r = t + s$, any fixed rank-r geometry must omit $(M-1)s$ of the $t + Ms$ directions, yielding Eq. (9). CODAR realizes all these directions through its two pathways; the finite dictionary, Top-K, and CP-routing construction is given in Appendix B.

Remark (Scope). Theorem 1 establishes strict representational separation from fixed shared rank-r geometry for a nonempty family of state-dependent targets. It does not imply a downstream-risk lower bound, guarantee optimization convergence, or claim superiority over all conditional low-rank parameterizations.

Design implication. Together, these results motivate separating reusable shared structure from availability-dependent variation: the transferable pathway models the former, while the modality-specific pathway captures the latter, both through reusable dictionary atoms.

3.3 Amortized CP-Based Routing Generation

Explicitly storing all routing tensors requires $\Theta_{\text{table}} = |\mathcal{M}| \sum_{\mu,l,f} C_f r_f N$ parameters. Instead, CODAR generates

$$\mathbf{L}_l^{\mu,f,m} = \mathcal{G}_\theta^f(\mathbf{h}_{l,\mu}, \mathbf{e}_m) + \mathbf{R}_l^{\mu,f}, \tag{10}$$

where $\mathbf{h}_{l,\mu} = \mathbf{e}_l^{\mu} + \mathbf{e}_{\mu} \in \mathbb{R}^{d_h}$, distinct from the concatenated context $\mathbf{c}_{l,\mu,m}$.

Each role-specific generator is shared across layers, towers, and states. A state-independent module-local residual $\mathbf{R}_l^{\mu,f} \in \mathbb{R}^{C_f \times r_f \times N}$, constrained to CP rank at most R_{res} with bounded amplitudes, provides local corrections to the routing logits. Its parameterization, parameter count, norm bound, and initialization are given in Appendix A.

We use a rank-R_g CP decomposition (Kolda & Bader, 2009) over the chunk, rank, and dictionary modes. For $\bullet \in \{C_f, r_f, N\}$ and $t = 1, \ldots, R_g$,

$$\mathbf{z}_{\bullet,t}^{l,\mu,f,m} = \left[\mathbf{\Phi}_{\bullet,t}^{f} + \mathbf{C}_{\bullet,t}^{f} \operatorname{diag}(\mathbf{e}_m)(\mathbf{D}_{\bullet,t}^{f})^{\top}\right] \mathbf{h}_{l,\mu}. \tag{11}$$

The base projection captures layer–tower regularities, while the correction introduces availability dependence. Using normalized mode vectors,

$$\mathcal{G}_{\theta}^{f}(\mathbf{h}_{l,\mu}, \mathbf{e}_m) = \sum_{t=1}^{R_g} s_t^{l,\mu,f,m}\, \widehat{\mathbf{z}}_{C_f,t} \otimes \widehat{\mathbf{z}}_{r_f,t} \otimes \widehat{\mathbf{z}}_{N,t}. \tag{12}$$

Here, $s_t^{l,\mu,f,m}$ controls each component's amplitude; its parameterization, together with residual construction, dimensions, and initialization, is detailed in Appendix A.

Parameter count and caching. CODAR introduces 101,116 trainable adapter parameters in our CLIP configuration, excluding the frozen backbone and classifier; Appendix A provides the breakdown. Because reconstructed factors depend only on layer, tower, factor role, availability state, and the frozen entropy references, they can be cached after training, eliminating runtime routing and factor generation. Uncached inference generates logits and composes factors on demand, whereas cached inference directly applies the stored low-rank updates.

3.4 Training Objective and Regularization

Training protocol. Following MoRA (Zhao et al., 2026), missing-state views replace images with white RGB inputs or text with empty strings, while both towers remain active. Let $\eta \in [0, 1]$ denote the incomplete-sample probability and $p_{\mathcal{S}}(m)$ the availability-state distribution for scenario $\mathcal{S}$. A single-missing scenario assigns probability η to designated missing state and $1-\eta$ to complete state; Both scenario uses $p_{\mathcal{S}}(\text{full}) = 1 - \eta$ and $p_{\mathcal{S}}(\text{vis-miss}) = p_{\mathcal{S}}(\text{txt-miss}) = \eta/2$. The task loss is

$$\mathcal{L}_{\text{task}} = \mathbb{E}_{(x,y)\sim\mathcal{D},\, m\sim p_{\mathcal{S}}}\, \ell(f(x^{(m)}; m), y). \tag{13}$$

Confidence-aware gating. For each rank coordinate, let $\mathcal{H}_{l,i}^{\mu,b,m}$ denote the normalized soft-routing entropy averaged over chunks and both factor roles of pathway b, and let $\overline{\mathcal{H}}^{\,b}$ be its pathway-level exponential moving average (EMA) reference. We attenuate diffuse routes using

$$g_{l,i}^{\mu,b,m} = 1 - \lambda_g \sigma\big((\mathcal{H}_{l,i}^{\mu,b,m} - \overline{\mathcal{H}}^{\,b})/\gamma\big), \tag{14}$$

where $\lambda_g \in [0, 1]$ controls suppression, $\gamma > 0$ is the temperature, and σ is the sigmoid. Entropy remains differentiable through g; the EMA reference is updated without gradients during factor reconstruction and frozen at inference. Full definitions and update rules appear in Appendix C.

Auxiliary regularizers. For load balancing, we penalize the KL divergence between normalized aggregate soft-routing usage and a uniform distribution within each physical dictionary bank, discouraging persistent atom starvation (Fedus et al., 2022). Shared-bank usage is aggregated across both towers; confidence gating acts on per-route uncertainty, whereas load balancing regularizes aggregate utilization.

Consistency distillation. Complete-input predictions serve as stop-gradient teachers for missing inputs (Hinton et al., 2015; Huo et al., 2024; Li et al., 2024a; Hoffman et al., 2016). We use temperature-scaled categorical KL for single-label tasks and per-class Bernoulli KL for multi-label tasks; Appendix C provides the full definitions.

Transferable routing consistency. For matched modules $l \in \mathcal{L}_{\text{match}}$ (e.g., same relative depth), let $\overline{\mathbf{P}}_l^{(\mu,m)} \in \mathbb{R}^{N \times r_{\text{xfer}}}$ contain chunk-averaged soft routes over shared input-side atoms. We align their Gram matrices $\overline{\mathbf{P}}_l^{(\mu,m)}(\overline{\mathbf{P}}_l^{(\mu,m)})^{\top}$ across towers, weighting each state's discrepancy by ω_m. The

Table 1: Performance at $\eta = 70\%$. Image/text availability (%): 100/30 (text-missing), 30/100 (image-missing), and 65/65 (Both). MoRA, StateCondLoRA, and CODAR report mean±std over three matched seeds; † denotes sourced results.

Dataset	Image	Text	MMP†	MaPLe†	DePT†	DCP†	MoRA	StateCondLoRA	CODAR
MM-IMDb	100%	30%	45.64	45.52	46.38	48.52	52.81±0.73	53.08±0.66	**54.01±0.71**
	30%	100%	50.52	50.64	52.13	53.14	56.34±0.41	56.41±0.50	**57.16±0.68**
	65%	65%	48.12	49.16	50.32	51.42	53.12±0.33	53.29±0.63	**54.41±0.65**
	Average		48.09	48.44	49.61	51.03	54.09±0.45	54.26±0.59	**55.19±0.64**
UPMC-Food101	100%	30%	76.52	77.02	77.34	78.87	81.04±0.42	81.49±0.40	**82.32±0.38**
	30%	100%	85.64	85.89	86.12	87.32	88.15±0.31	88.42±0.30	**88.95±0.29**
	65%	65%	79.12	79.84	81.46	81.87	83.76±0.36	83.80±0.34	**84.94±0.33**
	Average		80.43	80.92	81.64	82.69	84.32±0.35	84.57±0.33	**85.40±0.32**
Hateful Memes	100%	30%	61.12	61.26	61.56	62.82	68.27±0.54	69.83±0.59	**71.68±0.49**
	30%	100%	63.24	63.14	63.48	64.12	68.59±0.51	70.15±0.56	**71.26±0.47**
	65%	65%	65.04	65.23	65.48	66.08	69.74±0.48	71.49±0.53	**72.51±0.52**
	Average		63.13	63.21	63.51	64.34	68.87±0.46	70.49±0.56	**71.82±0.53**

resulting $\mathcal{L}_{\text{route}}$ aligns route co-usage rather than representations and is invariant to rank-coordinate permutations; dictionary-atom indices remain aligned across towers. The full loss definition is given in Appendix C.

Full objective. We minimize

$$\mathcal{L} = \mathcal{L}_{\text{task}} + \lambda_{\text{KD}}\mathcal{L}_{\text{KD}} + \lambda_{\text{route}}\mathcal{L}_{\text{route}} + \lambda_{\text{bal}}\mathcal{L}_{\text{bal}}, \tag{15}$$

where $\lambda_{\text{KD}}, \lambda_{\text{route}}, \lambda_{\text{bal}} \geq 0$ weight distillation, route alignment, and atom balancing. The model initializes by relying solely on the shared generator (rank amplitudes at 1, local residuals at 0), with full training details and CODAR algorithms deferred to the Appendix.

4 Experiments

Datasets and metrics. We evaluate on MM-IMDb (Arevalo et al., 2017), UPMC-Food101 (Wang et al., 2015), and Hateful Memes (Kiela et al., 2021), using Macro-F1, top-1 accuracy, and AUROC (%), respectively. Dataset and split details are in Appendix C.

Missing-modality protocol. Following prior works (Lee et al., 2023; Zhao et al., 2026), modalities may be absent during training and evaluation. At a missing rate η (default 70%), single-missing settings drop the designated modality, while the Both setting splits η equally between missing images and text. Missing inputs are replaced by white images or empty text with both towers active and known availability. Unless stated otherwise, each setting is trained independently.

Baselines and implementation. We compare CODAR with prompt-based MMP (Lee et al., 2023), MaPLe (Khattak et al., 2023), DePT (Zhang et al., 2024), and DCP (Hu et al., 2024), and low-rank PEFT methods including LoRA (Hu et al., 2022) and MoRA (Zhao et al., 2026). Prompt-based results follow the MoRA study; controlled low-rank comparisons match adapted projections, missing masks, training budgets, and checkpoint selection. We freeze CLIP ViT-B/16 (Radford et al., 2021), adapt the query/value projections of Layers 11–12 in both towers, and split CODAR's total rank $r = 4$ equally between its two pathways. New experiments at $\eta = 70\%$ average seeds $\{42, 43, 44\}$ unless noted otherwise. Full configurations and additional low-rank comparisons are provided in Appendices C and D.

4.1 Main Results

Table 1 shows that CODAR achieves the highest mean performance across all nine dataset–availability combinations. Relative to MoRA, CODAR improves the dataset averages by 1.10, 1.08, and 2.95 points on MM-IMDb, UPMC-Food101, and Hateful Memes, respectively, while using fewer trainable adapter parameters. It also exceeds the parameter-matched StateCondLoRA by 0.93, 0.83, and 1.33 points, supporting the benefit of structured state-conditioned dual-path reconstruction beyond availability conditioning alone.

Table 2: Core architectural ablations at $\eta = 70\%$, averaged over three missing-modality settings. All variants use matched seeds, masks, budgets, and checkpoint selection; single-pathway variants retain rank $r = 4$.

Variant	MM-IMDb	Food101	Hateful
CODAR	**55.19**	**85.40**	**71.82**
State-invariant routing	53.62	83.91	70.21
Input-side cond. only	54.28	84.61	71.02
Output-side cond. only	53.97	84.33	70.74
Spec. pathway only	54.00	84.19	70.55
Xfer. pathway only	53.71	83.98	70.30
w/o Cross-Tower Sharing	54.72	84.86	71.20

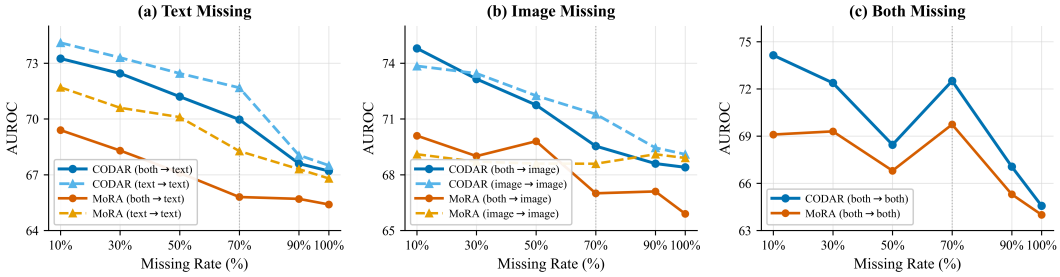

Figure 3: Cross-scenario generalization on Hateful Memes. Models are trained at $\eta = 70\%$ and evaluated without retraining over $\eta_{\text{test}} \in \{10, 30, 50, 70, 90, 100\}\%$. "Both→text/image" denotes Both-trained models evaluated on the corresponding missing state; scenario-specific models are trained on that state. Right: Both→Both.

4.2 Cross-Scenario Generalization

Figure 3 evaluates distribution shift by applying models trained at $\eta = 70\%$ across test missing rates of 10–100% without retraining. CODAR narrows the specialization gap relative to MoRA: the mean absolute gap is 0.90 vs. 2.18 points for text-missing and 0.84 vs. 1.52 for image-missing evaluation. It also leads across all Both-to-Both ratios, including 100% incomplete data. The non-monotonic trend is consistent with mismatch from the fixed 70% training distribution, with performance recovering near the matched condition. Overall, the results support cross-state sharing while retaining availability-dependent adaptation.

4.3 Ablation Study

State-conditioned geometry. State-invariant routing causes the largest drop in Table 2 (1.57/1.49/1.61 points across the three datasets). Conditioning only one factor partially recovers performance but remains weaker than CODAR, supporting conditioning of both low-rank factors.

Dual-path reconstruction. Using either pathway alone degrades performance, while replacing the transferable pathway's shared input-side dictionary with tower-private dictionaries also hurts, supporting complementary pathways and cross-tower reuse of input-side primitives. Additional generator, training-objective, and routing ablations are reported in Appendix F.

4.4 State-Dependent Update Geometry

Table 3 shows that independently trained rank-4 LoRA updates from different availability states have larger left/right principal angles than same-state, cross-seed updates ($80.87°/82.39°$ vs. $70.96°/70.83°$), while their union ranks increase from 10.70/10.75 to 11.34/11.57. The separation remains below random rank-4 subspaces, indicating distinct but partially shared update directions across states.

CODAR likewise realizes state-dependent geometry under the Both setting, with union ranks of 10.86/11.05, whereas state-invariant routing collapses both sides to rank 4. One-sided conditioning diagnostics are provided in Appendix F.

Table 3: Cross-state geometry of rank-4 updates (mean±std over 8 projections and 9 state–seed pairs). The upper panel analyzes independently optimized state-specific LoRA; the lower panel contrasts CODAR with state-invariant routing under the Both setting.

Comparison	Left angle	Right angle	Eff. union rank (L/R)	Tail ratio (L/R)
Diagnostic: independently trained rank-4 LoRA per state				
Same state, diff. seeds	$70.96^\circ \pm 4.27^\circ$	$70.83^\circ \pm 3.00^\circ$	10.70 ± 0.63 / 10.75 ± 0.45	0.52 ± 0.05 / 0.52 ± 0.03
Diff. states, same seed	$80.87^\circ \pm 0.51^\circ$	$82.39^\circ \pm 1.24^\circ$	11.34 ± 0.05 / 11.57 ± 0.04	0.57 ± 0.00 / 0.58 ± 0.00
Random rank-4 subspaces	$86.20^\circ \pm 0.26^\circ$	$86.20^\circ \pm 0.24^\circ$	11.92 ± 0.01 / 11.92 ± 0.01	0.62 ± 0.00 / 0.62 ± 0.00
Realized CODAR geometry across availability states				
CODAR	$78.14^\circ \pm 0.86^\circ$	$79.63^\circ \pm 0.74^\circ$	10.86 ± 0.17 / 11.05 ± 0.14	0.53 ± 0.01 / 0.54 ± 0.01
State-invariant routing	$0.00^\circ \pm 0.00^\circ$	$0.00^\circ \pm 0.00^\circ$	4.00 ± 0.00 / 4.00 ± 0.00	0.00 ± 0.00 / 0.00 ± 0.00

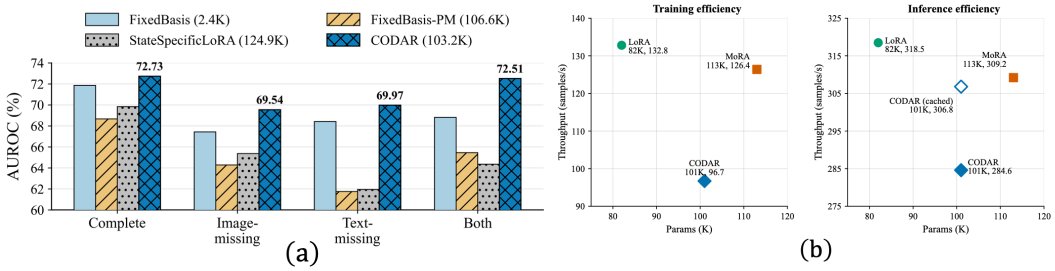

Figure 4: Capacity–transfer analysis and efficiency on Hateful Memes at $\eta = 70\%$. **(a)** Both-trained models evaluated across four availability settings; results average three seeds and parameter counts include the classifier. **(b)** Training/inference throughput on one RTX 4090 (batch size 16, FP16); parameter counts exclude the classifier, and "Cached" precomputes CODAR factors.

The same pattern is reproduced on MM-IMDb and UPMC-Food101, with energy-weighted analyses further showing non-negligible deviation from the best shared rank-4 geometry (Appendix F).

4.5 Capacity–Transfer Analysis

Figure 4(a) is a diagnostic of the two extremes. FixedBasis imposes state-invariant rank-4 geometry and FixedBasis-PM only increases static capacity. StateSpecificLoRA removes cross-state sharing, but its three adapters also see different state subsets (30/35/35% under the Both protocol), so its lower score is not a balanced causal estimate of transfer loss. A single Both-trained CODAR checkpoint outperforms FixedBasis across all four conditions with fewer parameters than StateSpecificLoRA; the parameter-matched StateCondLoRA comparison provides the main evidence for the claimed architectural gain. All CODAR–FixedBasis gains remain significant after Holm-corrected paired-bootstrap testing ($p < 0.01$; Appendix E).

4.6 Efficiency

CODAR uses 101,116 trainable adapter parameters, 10.2% fewer than MoRA's 112,640, at the cost of additional training overhead. Since reconstructed factors depend only on layer, tower, projection role, and availability state, they can be cached after training. Cached CODAR reaches 306.8 samples/s, recovering most of the inference gap to MoRA; for a fixed state, the resulting update can further be merged into the frozen backbone.

5 Conclusion

CODAR resolves the Capacity–Transfer Tension in missing-modality PEFT using state-conditioned dual-path low-rank factor reconstruction over reusable dictionaries. Geometrically and theoretically, fixed rank-r geometry incurs unavoidable reconstruction error, while CODAR represents both shared and state-specific structures. Across three vision-language benchmarks, CODAR consistently improves performance, validated by parameter-matched controls, ablations, and retrieval tasks. Currently evaluated on CLIP ViT-B/16 under known discrete states, future work will explore broader backbones, tasks, and open-set settings.

AI Use Statement

Large language models were used to assist with writing and editing, literature search and discovery, and research ideation and execution. Their use included improving grammar, clarity, and organization, supporting the discovery of relevant literature, and assisting with research brainstorming and parts of the research workflow. All methodological decisions, cited literature, implementation, experiments, results, and analyses were independently reviewed and validated by the authors. The authors take full responsibility for the accuracy, originality, and integrity of the work.

Reproducibility Statement

Method details and algorithms are provided in Appendix A, and the assumptions and proofs are given in Appendix B. Datasets, missing-modality protocols, baseline configurations, and optimization settings are described in Appendix C. Additional results and statistical testing are reported in Appendices D–F, and the software and hardware environment is listed in Appendix G.

References

John Arevalo, Thamar Solorio, Manuel Montes-y Gómez, and Fabio A. González. Gated multimodal units for information fusion, 2017. URL https://arxiv.org/abs/1702.01992.

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation, 2013. URL https://arxiv.org/abs/1308.3432.

Eric L. Buehler and Markus J. Buehler. X-lora: Mixture of low-rank adapter experts, a flexible framework for large language models with applications in protein mechanics and molecular design. *APL Machine Learning*, 2(2):026119, 05 2024. ISSN 2770-9019. doi: 10.1063/5.0203126. URL https://doi.org/10.1063/5.0203126.

Jun-Ho Choi and Jong-Seok Lee. Embracenet: A robust deep learning architecture for multimodal classification. *Inf. Fusion*, 51(C):259–270, November 2019. ISSN 1566-2535. doi: 10.1016/j.inffus.2019.02.010. URL https://doi.org/10.1016/j.inffus.2019.02.010.

Carl Eckart and Gale Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936. doi: 10.1007/BF02288367.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022. URL http://jmlr.org/papers/v23/21-0998.html.

Yunhao Gou, Zhili Liu, Kai Chen, Lanqing Hong, Hang Xu, Zhenguo Li, Dit-Yan Yeung, James T. Kwok, and Yu Zhang. Mixture of cluster-conditional lora experts for vision-language instruction tuning. *IEEE Transactions on Image Processing*, 35:3881–3892, 2026. doi: 10.1109/TIP.2026.3678087.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1026–1034, 2015. doi: 10.1109/ICCV.2015.123.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015. URL https://arxiv.org/abs/1503.02531.

Judy Hoffman, Saurabh Gupta, and Trevor Darrell. Learning with side information through modality hallucination. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 826–834, 2016. doi: 10.1109/CVPR.2016.96.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Lianyu Hu, Tongkai Shi, Wei Feng, Fanhua Shang, and Liang Wan. Deep correlated prompting for visual recognition with missing modalities. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=zO55ovdLJw.

Fushuo Huo, Wenchao Xu, Jingcai Guo, Haozhao Wang, and Song Guo. C2kd: Bridging the modality gap for cross-modal knowledge distillation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16006–16015, 2024. doi: 10.1109/CVPR52733.2024.01515.

Sanaz Karimijafarbigloo, Reza Azad, Amirhossein Kazerouni, Saeed Ebadollahi, and Dorit Merhof. Mmcformer: Missing modality compensation transformer for brain tumor segmentation. In *Medical Imaging with Deep Learning*, pp. 1144–1162, 2024. URL https://proceedings.mlr.press/v227/karimijafarbigloo24b.html.

Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19113–19122, 2023.

Douwe Kiela, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Casey A. Fitzpatrick, Peter Bull, Greg Lipstein, Tony Nelli, Ron Zhu, Niklas Muennighoff, Riza Velioglu, Jewgeni Rose, Phillip Lippe, Nithin Holla, Shantanu Chandra, Santhosh Rajamanickam, Georgios Antoniou, Ekaterina Shutova, Helen Yannakoudakis, Vlad Sandulescu, Umut Ozertem, Patrick Pantel, Lucia Specia, and Devi Parikh. The hateful memes challenge: Competition report. In Hugo Jair Escalante and Katja Hofmann (eds.), *Proceedings of the NeurIPS 2020 Competition and Demonstration Track*, volume 133 of *Proceedings of Machine Learning Research*, pp. 344–360. PMLR, 06–12 Dec 2021. URL https://proceedings.mlr.press/v133/kiela21a.html.

Tamara G. Kolda and Brett W. Bader. Tensor decompositions and applications. *SIAM Review*, 51 (3):455–500, 2009. doi: 10.1137/07070111X.

Dawid Jan Kopiczko, Tijmen Blankevoort, and Yuki M Asano. VeRA: Vector-based random matrix adaptation. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=NjNfLdxr3A.

Yi-Lun Lee, Yi-Hsuan Tsai, Wei-Chen Chiu, and Chen-Yu Lee. Multimodal prompting with missing modalities for visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14943–14952, June 2023.

Kaiyang Li, Shaobo Han, Qing Su, Wei Li, Zhipeng Cai, and Shihao Ji. Uni-loRA: One vector is all you need. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=hzBqQZK2iV.

Mingcheng Li, Dingkang Yang, Xiao Zhao, Shuaibing Wang, Yan Wang, Kun Yang, Mingyang Sun, Dongliang Kou, Ziyun Qian, and Lihua Zhang. Correlation-decoupled knowledge distillation for multimodal sentiment analysis with incomplete modalities. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12458–12468, 2024a. doi: 10.1109/CVPR52733.2024.01184.

Yang Li, Shaobo Han, and Shihao Ji. VB-loRA: Extreme parameter efficient fine-tuning with vector banks. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024b. URL https://openreview.net/forum?id=kuCY0mW4Q3.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. Dora: Weight-decomposed low-rank adaptation. In *International Conference on Machine Learning*, 2024. URL https://api.semanticscholar.org/CorpusID:267657886.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Mengmeng Ma, Jian Ren, Long Zhao, Davide Testuggine, and Xi Peng. Are multimodal transformers robust to missing modality? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18177–18186, June 2022.

Vittorio Pipoli, Federico Bolelli, Sara Sarto, Marcella Cornia, Lorenzo Baraldi, Costantino Grana, Rita Cucchiara, and Elisa Ficarra. Semantically conditioned prompts for visual recognition under missing modality scenarios. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 4968–4977, 2025. doi: 10.1109/WACV61041.2025.00486.

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*, pp. 8748–8763, 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Adithya Renduchintala, Tugrul Konuk, and Oleksii Kuchaiev. Tied-LoRA: Enhancing parameter efficiency of LoRA with weight tying. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 8694–8705, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.481. URL https://aclanthology.org/2024.naacl-long.481/.

Hanqing Wang, Bowen Ping, Shuo Wang, Xu Han, Yun Chen, Zhiyuan Liu, and Maosong Sun. LoRA-flow: Dynamic LoRA fusion for large language models in generative tasks. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12871–12882, Bangkok, Thailand, August 2024a. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.695. URL https://aclanthology.org/2024.acl-long.695/.

Hao Wang, Shengda Luo, Guosheng Hu, and Jianguo Zhang. Gradient-guided modality decoupling for missing-modality robustness. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'24/IAAI'24/EAAI'24. AAAI Press, 2024b. ISBN 978-1-57735-887-9. doi: 10.1609/aaai.v38i14.29474. URL https://doi.org/10.1609/aaai.v38i14.29474.

Xin Wang, Devinder Kumar, Nicolas Thome, Matthieu Cord, and Frédéric Precioso. Recipe recognition with large multimodal food dataset. In *2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pp. 1–6, 2015. doi: 10.1109/ICMEW.2015.7169757.

Xun Wu, Shaohan Huang, and Furu Wei. Mixture of loRA experts. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=uWvKBCYh4S.

Ji Zhang, Shihan Wu, Lianli Gao, Heng Tao Shen, and Jingkuan Song. Dept: Decoupled prompt tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12924–12933, 2024.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. Adaptive budget allocation for parameter-efficient fine-tuning. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=lq62uWRJjiY.

Zhihui Zhang, Luanyuan Dai, Qika Lin, Yunfeng Diao, Guangyin Jin, Yufei Guo, Jing Zhang, and Xiaoshuai Hao. Synergistic prompting for robust visual recognition with missing modalities. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1881–1890, 2025. doi: 10.1109/ICCV51701.2025.00183.

Jinming Zhao, Ruichen Li, and Qin Jin. Missing modality imagination network for emotion recognition with uncertain missing modalities. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 2608–2618, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.203. URL https://aclanthology.org/2021.acl-long.203/.

Shu Zhao, Nilesh Ahuja, Tan Yu, Tianyi Shen, and Vijaykrishnan Narayanan. MoRA: Missing modality low-rank adaptation for visual recognition. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=ZgQnIPG4uV.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022.

Xiongtao Zhou, Jie He, Yuhua Ke, Guangyao Zhu, Victor Gutierrez Basulto, and Jeff Pan. An empirical study on parameter-efficient fine-tuning for MultiModal large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 10057–10084. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-acl.598. URL https://aclanthology.org/2024.findings-acl.598/.

Yan Zhuang, Minhao Liu, Wei Bai, Yanru Zhang, Xiaoyue Zhang, Jiawen Deng, and Fuji Ren. Cmad: Correlation-aware and modalities-aware distillation for multimodal sentiment analysis with missing modalities. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4626–4636, 2025. doi: 10.1109/ICCV51701.2025.00440.

Supplementary Material

Beyond One-Geometry-Fits-All: State-Conditioned Dual-Path Reconstruction for Missing-Modality PEFT

Abstract

This supplementary material provides the evidence and implementation details supporting CODAR. It specifies the architecture, initialization, routing algorithms, and training protocol; develops the shared-subspace coverage, fixed-geometry reconstruction, and constructive separation analyses; and documents datasets, matched baselines, efficiency measurement, and statistical testing. Additional results cover attachment position, low-rank controls, missing-ratio sensitivity, and semantic retrieval, followed by reproducibility details, related work, limitations, ethics, and LLM usage.

Contents

A. Method Details . . . 14
B. Theory and Coverage Analysis . . . 18
C. Experimental Setup . . . 25
D. Controlled Low-Rank Comparisons . . . 30
E. Additional Missing-Ratio Results . . . 31
F. Supplementary Results . . . 32
G. Reproducibility . . . 34
H. Extended Related Work . . . 35
I. Limitations . . . 36
J. Ethical Considerations . . . 36

A Method Details

Notation

Architectural Hyperparameters

CODAR is attached to the query and value projections of Layers 11–12 in both CLIP towers. The backbone and all unmodified projections remain frozen. The settings in Table A2 follow the final configuration used by the main text. The proof of Theorem 1 and the accompanying fixed-geometry and perturbation analyses are provided in Appendix B.

Parameter complexity and numerical instantiation. Let $\mathcal{A}_\mu$ denote the set of adapted projections in tower μ. Define the total dictionary width as

$$D_{\text{dict}} = d^A_{\text{vis}} + d^B_{\text{vis}} + d^A_{\text{txt}} + d^B_{\text{txt}} + d^A_s. \tag{A1}$$

Table A1: Notation used by CODAR.

Symbol	Description
$\mu \in \{\text{vis}, \text{txt}\}$	Visual or textual tower
$l \in \mathcal{L}_\mu$	Adapted layer; $\mathcal{L}_\mu = \{11, 12\}$ in the experiments
$p_{\text{proj}} \in \{\text{qry}, \text{val}\}$	Adapted query or value projection
$m \in \mathcal{M}$	Modality-availability state
$\mathcal{M}$	$\{\text{full}, \text{vis-miss}, \text{txt-miss}\}$
r	Total update rank
$r_{\text{spec}}, r_{\text{xfer}}$	Modality-specific and transferable pathway ranks
P, Q	Input- and output-side chunk counts
$d_\mu^\star$	$\max(d_\mu^A, d_\mu^B)$, the larger factor-side chunk width
N	Number of atoms in each dictionary bank
K	Number of selected atoms in Top-K routing
R_g	CP rank of the hierarchical conditional generator
R_{res}	CP rank of the layer-specific routing residual
τ_r	Routing temperature
d_h, d_s	Tower/layer and state embedding dimensions
d_c, d_s^A	Conditional-context and shared-atom dimensions
d_κ	Hidden width of the rank-amplitude MLP
$\varepsilon_{\text{norm}}$	Numerical stabilizer for routing-vector normalization
$\delta_\kappa, \delta_s, \delta_\rho$	Bounds for rank, CP, and residual amplitudes
λ_g, γ	Confidence-gate suppression and temperature
$\beta_{\mathcal{H}}$	EMA decay for the entropy reference
η	Proportion of incomplete samples
T	Consistency-distillation temperature
ω_m	State weight in transferable-routing consistency
$\lambda_{\text{KD}}, \lambda_{\text{route}}, \lambda_{\text{bal}}$	Auxiliary-loss weights

Excluding the frozen backbone and task-specific classifier, the trainable parameter groups are given below, where f ranges over the four factor roles:

$$
\begin{aligned}
\Theta_{\text{emb}} &= \big(|\mathcal{A}_{\text{vis}}| + |\mathcal{A}_{\text{txt}}| + 2\big) d_h + |\mathcal{M}| d_s, \\
\Theta_{\text{gen}} &= \sum_f R_g \sum_{\bullet \in \{C_f, r_f, N\}} \left(d_\bullet d_h + d_\bullet d_s + d_h d_s\right), \\
\Theta_{\text{res}} &= \sum_\mu \sum_{l \in \mathcal{A}_\mu} \sum_f R_{\text{res}} \left(C_f + r_f + N + 1\right), \\
\Theta_\kappa &= \sum_{b \in \{\text{spec}, \text{xfer}\}} \left(d_\kappa d_c + d_\kappa + r_b d_\kappa + r_b\right), \\
\Theta_s &= \sum_f R_g (d_c + 2), \\
\Theta_{\text{dict}} &= N D_{\text{dict}} + \sum_\mu d_\mu^A d_s^A, \\
\Theta_{\text{CODAR}} &= \Theta_{\text{emb}} + \Theta_{\text{gen}} + \Theta_{\text{res}} + \Theta_\kappa + \Theta_s + \Theta_{\text{dict}}.
\end{aligned}
\tag{A2}
$$

Substituting the settings in Table A2 into Eq. (A2) yields

$$
\begin{aligned}
(\Theta_{\text{emb}}, \Theta_{\text{gen}}, \Theta_{\text{res}}) &= (288, 65{,}792, 2{,}400), \\
(\Theta_\kappa, \Theta_s, \Theta_{\text{dict}}) &= (4{,}828, 1{,}056, 26{,}752), \\
\Theta_{\text{CODAR}} &= 101{,}116.
\end{aligned}
\tag{A3}
$$

CP Generator Parameterization

Mode dimensions and normalization. The generator operates on chunk, rank, and dictionary modes, with widths $d_{C_f} = C_f$, $d_{r_f} = r_f$, and $d_N = N$. For each mode $\bullet \in \{C_f, r_f, N\}$ and

Table A2: CODAR architecture and objective settings.

Hyperparameter	Value
Adapted layers	11–12 in both towers
Adapted projections	Query and value
Adapted projections per tower	4
Total rank r	4
Pathway ranks	$r_{\text{spec}} = r_{\text{xfer}} = 2$
Chunk counts (P, Q)	$(8, 8)$
Chunk widths (vision, text)	$(96, 64)$
Dictionary size N	64
Top-K sparsity K	4
Generator CP rank R_g	4
Residual CP rank R_{res}	1
Routing temperature τ_r	0.5
Tower/layer embedding d_h	24
State embedding d_s	16
Conditional context d_c	64
Shared-atom dimension d_s^A	28
Amplitude-MLP hidden width d_κ	36
Normalization stabilizer $\varepsilon_{\text{norm}}$	10^{-6}
Rank-amplitude bound δ_κ	0.5
Generator-amplitude bound δ_s	0.3
Residual-amplitude bound δ_ρ	0.2
Gate suppression λ_g	0.5
Gate temperature γ	0.1
Entropy EMA $\beta_{\mathcal{H}}$	0.99
Entropy-reference initialization	1
Incomplete-sample proportion η	0.7
Route weights (full / each missing state)	$1/0.3$
Distillation temperature T	2.0
$\lambda_{\text{KD}}, \lambda_{\text{route}}, \lambda_{\text{bal}}$	$1.0,\ 0.1,\ 0.01$
LoRA scaling α	r
Trainable adapter parameters	101,116

component $t \in \{1, \dots, R_g\}$, the generator parameters have dimensions

$$\begin{aligned} \mathbf{\Phi}^f_{\bullet,t} &\in \mathbb{R}^{d_\bullet \times d_h}, \\ \mathbf{C}^f_{\bullet,t} &\in \mathbb{R}^{d_\bullet \times d_s}, \\ \mathbf{D}^f_{\bullet,t} &\in \mathbb{R}^{d_h \times d_s}. \end{aligned} \tag{A4}$$

The base projection $\mathbf{\Phi}^f_{\bullet,t}$ captures layer–tower regularities, while $\mathbf{C}^f_{\bullet,t}$ and $\mathbf{D}^f_{\bullet,t}$ inject state dependence. Each generated mode vector is normalized as

$$\widehat{\mathbf{z}} = \frac{\mathbf{z}}{\|\mathbf{z}\|_2 + \varepsilon_{\text{norm}}}, \qquad \varepsilon_{\text{norm}} > 0. \tag{A5}$$

A rank-R_g CP representation uses $\mathcal{O}(R_g(C_f + r_f + N))$ mode coefficients. The conditional construction avoids explicitly enumerating layer–tower–state combinations and supports continuous availability encodings through linear mode combinations.

State-conditioned component amplitudes. Let $\mathbf{c}_{l,\mu,m} = [\mathbf{e}^\mu_l; \mathbf{e}_\mu; \mathbf{e}_m] \in \mathbb{R}^{d_c}$ be the concatenated context, with $d_c = 2d_h + d_s$. Each CP component uses the amplitude

$$s^{l,\mu,f,m}_t = s^f_{0,t} \exp\left[\delta_s \tanh\left((\mathbf{w}^f_{s,t})^\top \mathbf{c}_{l,\mu,m} + b^f_{s,t}\right)\right]. \tag{A6}$$

Here, $s^f_{0,t} > 0$ is a learnable base amplitude, $\mathbf{w}^f_{s,t} \in \mathbb{R}^{d_c}$, $b^f_{s,t} \in \mathbb{R}$, and $\delta_s > 0$ bounds relative state variation.

Layer-specific routing residuals. Local deviations from the shared generator are represented by

$$\mathbf{R}^{\mu,f}_l = \sum_{t=1}^{R_{\text{res}}} \rho^{\mu,f}_{l,t}\, \widehat{\mathbf{a}}^{\mu,f}_{l,t} \otimes \widehat{\mathbf{c}}^{\mu,f}_{l,t} \otimes \widehat{\mathbf{d}}^{\mu,f}_{l,t}. \tag{A7}$$

Here, R_{res} is the residual CP rank, and the normalized mode vectors have widths (C_f, r_f, N). Residual amplitudes are parameterized as

$$\rho_{l,t}^{\mu,f} = \delta_\rho \tanh\left(\widetilde{\rho}_{l,t}^{\mu,f}\right), \tag{A8}$$

where $\widetilde{\rho}_{l,t}^{\mu,f}$ is unconstrained and the fixed $\delta_\rho > 0$ bounds the amplitude.

This state-independent residual is added directly to the conditional routing logits for each adapted module and factor role. Its parameter count is

$$R_{\text{res}}(C_f + r_f + N + 1)$$

per adapted module and role. Because the mode vectors are normalized and $|\rho_{l,t}^{\mu,f}| \le \delta_\rho$, the residual satisfies

$$\|\mathbf{R}_l^{\mu,f}\|_F \le R_{\text{res}}\delta_\rho.$$

Zero initialization of $\widetilde{\rho}_{l,t}^{\mu,f}$ disables the residual at initialization, so optimization starts from the shared conditional generator.

Pathway Amplitudes

Atom normalization separates dictionary directions from their magnitudes. To restore scale, each pathway $b \in \{\text{spec}, \text{xfer}\}$ uses rank-wise amplitudes conditioned on $\mathbf{c}_{l,\mu,m} = [\mathbf{e}_l^\mu; \mathbf{e}_\mu; \mathbf{e}_m] \in \mathbb{R}^{d_c}$, where the layer, tower, and state embeddings have dimensions d_h, d_h, d_s, respectively, and $d_c = 2d_h + d_s$.

A two-layer GELU MLP $h_\kappa^b : \mathbb{R}^{d_c} \to \mathbb{R}^{r_b}$ with hidden dimension d_κ produces

$$\boldsymbol{\kappa}_l^{\mu,b,m} = \exp\left[\delta_\kappa \tanh(h_\kappa^b(\mathbf{c}_{l,\mu,m}))\right]. \tag{A9}$$

The exponential and hyperbolic tangent are applied elementwise. The fixed constant $\delta_\kappa > 0$ bounds each amplitude:

$$\kappa_{l,i}^{\mu,b,m} \in [e^{-\delta_\kappa}, e^{\delta_\kappa}], \qquad i = 1, \ldots, r_b.$$

The final layer of each amplitude MLP is zero initialized, yielding $\boldsymbol{\kappa}_l^{\mu,b,m} = \mathbf{1}$ initially and preventing initial amplitude bias.

For each rank coordinate i, the same amplitude $\kappa_{l,i}^{\mu,b,m}$ and confidence gate $g_{l,i}^{\mu,b,m}$ are applied to both factor roles A_b and B_b. Multiplying both reconstructed factors by $\sqrt{\kappa_{l,i}^{\mu,b,m} g_{l,i}^{\mu,b,m}}$ therefore scales the corresponding rank-one update by exactly $\kappa_{l,i}^{\mu,b,m} g_{l,i}^{\mu,b,m}$. The confidence gate and its update rules are detailed in Appendix C.

Initialization

Dictionary banks. Each learnable dictionary atom is sampled from $\mathcal{N}(0, 0.02^2\mathbf{I})$ and normalized to unit ℓ_2 norm. Tower-specific projections for the transferable input dictionary use Kaiming uniform initialization (He et al., 2015).

Embeddings and generator. Layer and tower embeddings are sampled from $\mathcal{N}(0, 0.02^2\mathbf{I})$. State embeddings are initialized to zero so that the initial generator is state neutral. Base generator projections use Kaiming uniform initialization (He et al., 2015). Both state-correction matrices $\mathbf{C}$ and $\mathbf{D}$ are initialized using Xavier uniform initialization, while state embeddings are initialized to zero. This yields zero initial state corrections while allowing gradients to update the state embeddings. The base CP amplitudes are initialized to one.

Residuals and amplitudes. Layer-specific residual amplitudes are initialized to zero, so optimization starts from the amortized generator. The final layer of each rank-amplitude MLP is zero initialized, yielding $\boldsymbol{\kappa} = \mathbf{1}$ initially. The classification head follows the baseline MoRA initialization.

Inference Caching

The generated factors depend on the adapted layer, tower, projection role, and availability state, but not on the individual input sample. After training, we therefore generate and cache all factors for $(\mu, l, p_{\text{proj}}, m) \in \{\text{vis}, \text{txt}\} \times \{11, 12\} \times \{\text{qry}, \text{val}\} \times \mathcal{M}$. At inference, the factors associated with the known availability state are retrieved and applied as ordinary LoRA-style updates. For state-homogeneous deployment, the corresponding updates may also be merged into the frozen weights offline.

Straight-Through Top-K Routing

For factor role f, the routing tensor $\mathbf{L}_l^{\mu,f,m} \in \mathbb{R}^{C_f \times r_f \times N}$ assigns logits to dictionary atoms for each chunk and rank coordinate. Set $(C_f, d_f) = (P, d_\mu^A)$ for input-factor roles and (Q, d_μ^B) for output-factor roles, with $r_f = r_b$ for $f \in \{A_b, B_b\}$.

Sparse forward routing. Suppressing module, tower, factor-role, and state indices for readability, we select the K largest logits for chunk c and rank coordinate i:

$$\mathcal{I}_{c,i} = \text{TopK}\big(\mathbf{L}(c, i, :)/\tau_r, K\big), \tag{A10}$$

where $\tau_r > 0$ is the routing temperature. The sparse forward distribution is

$$p_{\text{hard}}(n) = \begin{cases} \dfrac{\exp(\mathbf{L}(c,i,n)/\tau_r)}{\sum_{j \in \mathcal{I}_{c,i}} \exp(\mathbf{L}(c,i,j)/\tau_r)}, & n \in \mathcal{I}_{c,i}, \\ 0, & \text{otherwise.} \end{cases} \tag{A11}$$

Straight-through gradients. Let $\mathbf{p}_{\text{soft}} = \text{Softmax}(\mathbf{L}(c, i, :)/\tau_r)$ be the dense soft-routing distribution. We use a straight-through estimator (Bengio et al., 2013):

$$\widetilde{\mathbf{p}}_{c,i} = \mathbf{p}_{\text{soft}} + \text{sg}\big(\mathbf{p}_{\text{hard}} - \mathbf{p}_{\text{soft}}\big), \tag{A12}$$

where sg denotes stop-gradient. The forward values equal $\mathbf{p}_{\text{hard}}$, while routing gradients follow $\mathbf{p}_{\text{soft}}$. A reference implementation is:

```
p_soft = F.softmax(L / tau_r, dim=-1)
topk_vals, topk_idx = torch.topk(L / tau_r, K, dim=-1)
p_hard = torch.zeros_like(p_soft)
p_hard.scatter_(-1, topk_idx, F.softmax(topk_vals, dim=-1))
p_tilde = p_soft + (p_hard - p_soft).detach()
```

Factor reconstruction. The normalized dictionary $\widehat{\mathcal{V}}(\mu, f) \in \mathbb{R}^{N \times d_f}$ maps routing weights to chunk-wise mixtures. Stacking these mixtures gives the unscaled factor:

$$\widetilde{\mathbf{F}}_l^{(\mu,f,m)} = \text{Stack}_{c,i}\left[\widehat{\mathcal{V}}(\mu, f)^\top \widetilde{\mathbf{p}}_{c,i}\right] \in \mathbb{R}^{(C_f d_f) \times r_f}. \tag{A13}$$

Pathway amplitudes and confidence gates are applied subsequently, as described in Appendix A.

Computational scope. Dense logit generation, Top-K selection, and the training softmax still depend on N. After selection, gathered dictionary composition would cost $\mathcal{O}(C_f r_f K d_f)$ rather than $\mathcal{O}(C_f r_f N d_f)$. Our implementation instead uses dense contraction and retains dense soft weights for STE, so sparse forward support alone does not provide this computational or training-memory reduction.

CODAR Algorithms

Algorithms A1 and A2 summarize the forward pass and optimization procedure. For incomplete-state examples, complete-input teacher outputs are computed once per minibatch and reused by sample identifier, avoiding duplicated teacher computation within an optimization step.

B Theory and Coverage Analysis

Shared-Subspace Coverage Identity

Fix a layer l and tower μ, and suppress these indices for clarity. For each availability state $m \in \mathcal{M}$, let $r_m = \text{rank}(\Delta \mathbf{W}^{(m)}) \le r$, and let $\mathbf{Q}_m \in \mathbb{R}^{d_{\text{out}} \times r_m}$ be an orthonormal basis for the column

Algorithm A1 CODAR Forward Pass

Require: Sample x, availability state m, layer l, tower μ
Ensure: Adapted output h_{out}

$\mathbf{c}_{l,\mu,m} \leftarrow [\mathbf{e}_l^{\mu}; \mathbf{e}_{\mu}; \mathbf{e}_m]$
for $b \in \{\text{spec}, \text{xfer}\}$ **do**
  $\boldsymbol{\kappa}_l^{\mu,b,m} \leftarrow \exp[\delta_\kappa \tanh(h_\kappa^b(\mathbf{c}_{l,\mu,m}))]$
  {First compute routing, mixtures, and entropy for both factor roles}
  for $f \in \{A_b, B_b\}$ **do**
    $\mathbf{L}_l^{\mu,f,m} \leftarrow G_\theta^f(\mathbf{e}_l^{\mu} + \mathbf{e}_{\mu}, \mathbf{e}_m) + \mathbf{R}_l^{\mu,f}$
    $\widehat{\mathbf{V}}(\mu, f) \leftarrow \text{RowNorm}(\mathbf{V}(\mu, f))$
    for chunk c and rank coordinate i **do**
      $\mathcal{I}_{c,i}^f \leftarrow \text{TopK}(\mathbf{L}_l^{\mu,f,m}(c, i, :)/\tau_r, K)$
      $\widetilde{\mathbf{p}}_{c,i}^f \leftarrow \text{STE-TopK}(\mathbf{L}_l^{\mu,f,m}(c, i, :), \mathcal{I}_{c,i}^f)$
      $\mathcal{H}_{l,c,i}^{\mu,f,m} \leftarrow -\sum_n p_{\text{soft}}^f(n) \log p_{\text{soft}}^f(n)/\log N$
      $\mathbf{u}_{c,i}^f \leftarrow \widehat{\mathbf{V}}(\mu, f)^\top \widetilde{\mathbf{p}}_{c,i}^f$
    end for
  end for
  {Aggregate both factor sides before computing the confidence gate}
  for rank coordinate i **do**
    $\mathcal{H}_{l,i}^{\mu,b,m} \leftarrow \frac{1}{2}\left(\frac{1}{P}\sum_c \mathcal{H}_{l,c,i}^{\mu,A_b,m} + \frac{1}{Q}\sum_c \mathcal{H}_{l,c,i}^{\mu,B_b,m}\right)$
    $g_{l,i}^{\mu,b,m} \leftarrow 1 - \lambda_g \sigma((\mathcal{H}_{l,i}^{\mu,b,m} - \overline{\mathcal{H}}^{\,b})/\gamma)$
  end for
  {Apply the shared pathway gate to both reconstructed factors}
  for $f \in \{A_b, B_b\}$ **do**
    $\mathbf{F}_l^{(\mu,f,m)} \leftarrow \text{Stack}_{c,i}\left[\sqrt{\kappa_{l,i}^{\mu,b,m} g_{l,i}^{\mu,b,m}}\, \mathbf{u}_{c,i}^f\right]$
  end for
end for
$\Delta\mathbf{W}_l^{(\mu,m)} \leftarrow \mathbf{B}_{l,\text{spec}}\mathbf{A}_{l,\text{spec}}^\top + \mathbf{B}_{l,\text{xfer}}\mathbf{A}_{l,\text{xfer}}^\top$
return $(\mathbf{W}_l^{\mu} + \frac{\alpha}{r}\Delta\mathbf{W}_l^{(\mu,m)})h_{\text{in}}$

space of $\Delta\mathbf{W}^{(m)}$. Define

$$\mathbf{Q}_{\text{all}} = [\mathbf{Q}_1, \ldots, \mathbf{Q}_{|\mathcal{M}|}]$$

and

$$s = \sum_{m \in \mathcal{M}} r_m, \qquad q = \min(d_{\text{out}}, s).$$

For a rank-r orthogonal projector $\mathbf{Z}\mathbf{Z}^\top$ with $\mathbf{Z}^\top\mathbf{Z} = \mathbf{I}_r$, the total column-space coverage error is

$$\mathcal{E}(\mathbf{Z}) = \sum_{m \in \mathcal{M}} \left\|(\mathbf{I} - \mathbf{Z}\mathbf{Z}^\top)\mathbf{Q}_m\right\|_F^2. \tag{A14}$$

Since $\mathbf{Q}_m^\top\mathbf{Q}_m = \mathbf{I}_{r_m}$,

$$\left\|(\mathbf{I} - \mathbf{Z}\mathbf{Z}^\top)\mathbf{Q}_m\right\|_F^2 = r_m - \|\mathbf{Z}^\top\mathbf{Q}_m\|_F^2. \tag{A15}$$

Summing over states yields

$$\mathcal{E}(\mathbf{Z}) = s - \|\mathbf{Z}^\top\mathbf{Q}_{\text{all}}\|_F^2. \tag{A16}$$

Let $\mathbf{Q}_{\text{all}} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$, with singular values $\sigma_1 \geq \cdots \geq \sigma_q \geq 0$. By the variational characterization of the leading singular subspace,

$$\max_{\mathbf{Z}^\top\mathbf{Z} = \mathbf{I}_r} \|\mathbf{Z}^\top\mathbf{Q}_{\text{all}}\|_F^2 = \sum_{j=1}^{r} \sigma_j^2(\mathbf{Q}_{\text{all}}), \tag{A17}$$

Algorithm A2 CODAR Training Loop

Require: Dataset $\mathcal{D}$, training-state distribution $p_{\mathcal{S}}(m)$
Ensure: Trained CODAR parameters

```
Initialize dictionaries, generator, amplitudes, and residuals
$\overline{\mathcal{H}}^{\mathrm{spec}} \leftarrow 1, \overline{\mathcal{H}}^{\mathrm{xfer}} \leftarrow 1$
for epoch $e$ do
  for minibatch $\{(x_i, y_i)\}_{i=1}^{B}$ do
    for $i = 1, \ldots, B$ do
      $m_i \sim p_{\mathcal{S}}(m)$
      $x_i^{(m_i)} \leftarrow \mathrm{ApplyMissing}(x_i, m_i)$
      $\mathbf{z}_i^{m_i} \leftarrow f(x_i^{(m_i)}; m_i)$
    end for
    $\mathcal{L}_{\mathrm{task}} \leftarrow B^{-1} \sum_{i=1}^{B} \ell(\mathbf{z}_i^{m_i}, y_i)$
    $\mathcal{I}_{\mathrm{miss}} \leftarrow \{i : m_i \neq \mathrm{full}\}$
    if $\mathcal{I}_{\mathrm{miss}} \neq \varnothing$ then
      Compute each $\mathbf{z}_i^{\mathrm{full}} \leftarrow \mathrm{sg}(f(x_i; \mathrm{full}))$ once for $i \in \mathcal{I}_{\mathrm{miss}}$
      Compute $\mathcal{L}_{\mathrm{KD}}$ over $\mathcal{I}_{\mathrm{miss}}$ with temperature $T$
    else
      $\mathcal{L}_{\mathrm{KD}} \leftarrow 0$
    end if
    Aggregate soft atom usage and compute $\mathcal{L}_{\mathrm{bal}}$
    Compute transferable-route loss $\mathcal{L}_{\mathrm{route}}$
    $\mathcal{L} \leftarrow \mathcal{L}_{\mathrm{task}} + \lambda_{\mathrm{KD}}\mathcal{L}_{\mathrm{KD}} + \lambda_{\mathrm{route}}\mathcal{L}_{\mathrm{route}} + \lambda_{\mathrm{bal}}\mathcal{L}_{\mathrm{bal}}$
    Backpropagate and update parameters with AdamW
    Update pathway entropy references by EMA
  end for
end for
```

where singular values beyond q are interpreted as zero. Because $\|\mathbf{Q}_{\mathrm{all}}\|_F^2 = s$, the minimum shared-subspace coverage error is

$$\min_{\mathbf{Z}^\top \mathbf{Z} = \mathbf{I}_r} \mathcal{E}(\mathbf{Z}) = \sum_{j=r+1}^{q} \sigma_j^2(\mathbf{Q}_{\mathrm{all}}). \tag{A18}$$

The same argument applied to orthonormal row-space bases $\mathbf{U}_m$ gives the row-space identity used in the main text. Thus, whenever either tail is nonzero, no single shared rank-r column/row subspace pair can exactly cover all state-specific update directions.

Fixed-Geometry Reconstruction Lower Bound

We now connect the basis-level coverage analysis to approximation of the actual state-specific updates. Fix a layer and tower and write

$$\mathbf{D}_m = \Delta \mathbf{W}^{(m)}.$$

Consider the broader fixed-geometry family

$$\widehat{\mathbf{D}}_m = \mathbf{Z} \mathbf{\Gamma}_m \mathbf{Y}^\top,$$

where $\mathbf{Z}^\top \mathbf{Z} = \mathbf{I}_r$, $\mathbf{Y}^\top \mathbf{Y} = \mathbf{I}_r$, and $\mathbf{\Gamma}_m \in \mathbb{R}^{r \times r}$ is allowed to vary freely with the availability state. Define

$$\mathbf{S}_{\mathrm{col}} = \sum_{m \in \mathcal{M}} \mathbf{D}_m \mathbf{D}_m^\top, \qquad \mathbf{S}_{\mathrm{row}} = \sum_{m \in \mathcal{M}} \mathbf{D}_m^\top \mathbf{D}_m. \tag{A19}$$

Let

$$\mathcal{E}_{\mathrm{fix}} = \min_{\substack{\mathbf{Z}^\top \mathbf{Z} = \mathbf{I}_r,\, \mathbf{Y}^\top \mathbf{Y} = \mathbf{I}_r \\ \{\mathbf{\Gamma}_m\}}} \sum_{m \in \mathcal{M}} \left\| \mathbf{D}_m - \mathbf{Z} \mathbf{\Gamma}_m \mathbf{Y}^\top \right\|_F^2. \tag{A20}$$

For any feasible $\mathbf{Z}, \mathbf{Y}, \{\mathbf{\Gamma}_m\}$, the column space of $\mathbf{Z}\mathbf{\Gamma}_m \mathbf{Y}^\top$ is contained in $\mathrm{span}(\mathbf{Z})$. Hence, orthogonal projection onto $\mathrm{span}(\mathbf{Z})$ gives a relaxation of the joint row/column-constrained approximation:

$$\left\| \mathbf{D}_m - \mathbf{Z} \mathbf{\Gamma}_m \mathbf{Y}^\top \right\|_F^2 \geq \left\| (\mathbf{I} - \mathbf{Z}\mathbf{Z}^\top) \mathbf{D}_m \right\|_F^2. \tag{A21}$$

Therefore,

$$\begin{aligned}\mathcal{E}_{\text{fix}} &\geq \min_{\mathbf{Z}^\top\mathbf{Z}=\mathbf{I}_r} \sum_m \left\|(\mathbf{I}-\mathbf{Z}\mathbf{Z}^\top)\mathbf{D}_m\right\|_F^2 \\ &= \operatorname{tr}(\mathbf{S}_{\text{col}}) - \max_{\mathbf{Z}^\top\mathbf{Z}=\mathbf{I}_r} \operatorname{tr}\left(\mathbf{Z}^\top\mathbf{S}_{\text{col}}\mathbf{Z}\right) \\ &= \sum_{j>r} \lambda_j(\mathbf{S}_{\text{col}}),\end{aligned} \tag{A22}$$

where the last equality follows from the variational characterization of the leading eigenspace.

Applying the same argument to the transposed updates gives

$$\mathcal{E}_{\text{fix}} \geq \sum_{j>r} \lambda_j(\mathbf{S}_{\text{row}}). \tag{A23}$$

Combining the two inequalities yields

$$\boxed{\mathcal{E}_{\text{fix}} \geq \max\left\{\sum_{j>r}\lambda_j(\mathbf{S}_{\text{col}}), \sum_{j>r}\lambda_j(\mathbf{S}_{\text{row}})\right\}.} \tag{A24}$$

Eigenvalues beyond the corresponding matrix dimension are interpreted as zero.

This result is deliberately stated for a family broader than a state-agnostic adapter: each state may use an arbitrary core $\mathbf{\Gamma}_m$, while only the row and column subspaces are shared. Thus, nonzero spectral tail energy reflects a limitation of the fixed geometry itself rather than of state-independent coefficients.

Constructive Separation for CODAR

We prove Theorem 1 by constructing a family of state-specific updates that satisfies the actual CODAR parameterization. Fix a layer and tower and write

$$t = r_{\text{xfer}}, \qquad s = r_{\text{spec}}, \qquad r = t + s, \qquad M = |\mathcal{M}|.$$

Step 1: Orthogonal shared and state-specific directions. Choose orthonormal chunk-space vectors

$$\mathbf{a}_1, \ldots, \mathbf{a}_{t+Ms} \in \mathbb{R}^{d_\mu^A}, \qquad \mathbf{b}_1, \ldots, \mathbf{b}_{t+Ms} \in \mathbb{R}^{d_\mu^B},$$

which exist under $d_\mu^A, d_\mu^B \geq t + Ms$. Define their full-dimensional repeated-chunk counterparts as

$$\mathbf{v}_j = \frac{1}{\sqrt{P}}[\mathbf{a}_j; \ldots; \mathbf{a}_j], \qquad \mathbf{u}_j = \frac{1}{\sqrt{Q}}[\mathbf{b}_j; \ldots; \mathbf{b}_j]. \tag{A25}$$

The vectors $\{\mathbf{v}_j\}$ and $\{\mathbf{u}_j\}$ are respectively orthonormal.

Let

$$\mathbf{V}_0 = [\mathbf{v}_1, \ldots, \mathbf{v}_t], \qquad \mathbf{U}_0 = [\mathbf{u}_1, \ldots, \mathbf{u}_t].$$

For state $m \in \{1, \ldots, M\}$, let

$$\mathbf{V}_m = [\mathbf{v}_{t+(m-1)s+1}, \ldots, \mathbf{v}_{t+ms}], \tag{A26}$$

$$\mathbf{U}_m = [\mathbf{u}_{t+(m-1)s+1}, \ldots, \mathbf{u}_{t+ms}]. \tag{A27}$$

Thus the shared and state-specific directions are mutually orthogonal on both factor sides.

Step 2: Dictionary realization. For each required chunk-space direction, allocate K identical normalized atoms. In the modality-specific input dictionary $\mathbf{V}_\mu^A$, allocate KMs atoms for the Ms private directions. In the output dictionary $\mathbf{V}_\mu^B$, allocate $K(t+Ms)$ atoms for the t transferable and Ms private directions.

For the transferable input bank, choose t orthonormal latent directions $\mathbf{e}_1, \ldots, \mathbf{e}_t \in \mathbb{R}^{d_s^A}$ and allocate K copies of each direction in $\mathbf{V}_{\mathrm{sh}}^A$. Because $d_s^A \geq t$, choose the tower-specific projection $\mathbf{T}_\mu^A$ such that

$$\mathbf{T}_\mu^A \mathbf{e}_i = \mathbf{a}_i, \qquad i = 1, \ldots, t.$$

After row normalization, the projected transferable atoms therefore coincide with the desired unit chunk directions. The condition

$$N \geq K(t + Ms)$$

is sufficient for every physical dictionary bank used in this construction.

Step 3: Exact Top-K mixtures. For a desired direction, let $\mathcal{G} \subset \{1, \ldots, N\}$ denote its associated group of K duplicate atoms. Choose the corresponding routing-logit vector to equal a constant $\ell > 0$ on $\mathcal{G}$ and zero elsewhere. Then Top-K selects exactly $\mathcal{G}$, and the selected logits are equal. Hence

$$p_{\mathrm{hard}}(n) = \begin{cases} 1/K, & n \in \mathcal{G}, \\ 0, & \text{otherwise.} \end{cases}$$

Since all atoms in $\mathcal{G}$ are identical, their routed mixture is exactly the desired normalized chunk direction. Thus finite $K > 1$ does not introduce approximation error in this construction.

Step 4: Realization by the conditional CP generator. For a role f with pathway rank r_f, define the desired routing tensor for state m as

$$\mathbf{L}^{f,m} = \ell \sum_{i=1}^{r_f} \mathbf{1}_{C_f} \otimes \mathbf{e}_i^{(r_f)} \otimes \mathbf{q}_{f,m,i}, \tag{A28}$$

where $\mathbf{e}_i^{(r_f)}$ is the ith rank-coordinate basis vector and $\mathbf{q}_{f,m,i} \in \{0,1\}^N$ indicates the K atoms assigned to that route. For transferable roles, the selected groups are independent of m; for modality-specific roles, different states use the corresponding private groups.

Equation (A28) has CP rank at most r_f. Since

$$R_g \geq \max(t, s),$$

the generator has sufficient CP rank for every factor role. Any unused CP components are made identically zero by setting one of their pre-normalized mode vectors to zero, so they contribute nothing despite the positive amplitude parameterization. It remains to verify that the state dependence in $\mathbf{q}_{f,m,i}$ is compatible with the generator in the main method. Choose linearly independent state embeddings supported on the first M coordinates of $\mathbb{R}^{d_s}$, which is possible because $d_s \geq M$. For the dictionary mode, the term

$$\left[\mathbf{\Phi}_{\bullet,t}^f + \mathbf{C}_{\bullet,t}^f \operatorname{diag}(\mathbf{e}_m)(\mathbf{D}_{\bullet,t}^f)^\top\right] \mathbf{h}_{l,\mu}$$

can then assign an arbitrary prescribed dictionary-mode vector to each of the M states by choosing the corresponding columns of $\mathbf{C}_{\bullet,t}^f$. The chunk and rank modes are made state independent by setting their state-correction terms to zero.

The mode normalization $\widehat{\mathbf{z}} = \mathbf{z}/(\|\mathbf{z}\|_2 + \varepsilon_{\mathrm{norm}})$ only introduces a positive scalar factor. Because the pre-normalized mode norms can be chosen identically across states and the learnable base CP amplitude $s_{0,t}^f > 0$ is unconstrained in scale, these fixed positive normalization factors can be absorbed into the CP amplitude. Therefore the generator realizes Eq. (A28) exactly. The layer-specific routing residuals are set to zero.

Step 5: Amplitudes and confidence gates. Choose the rank-amplitude MLP output to be zero, giving $\kappa_{l,i}^{\mu,b,m} = 1$. All constructed routing-logit vectors are permutations of the same two-level pattern: K entries equal ℓ and $N - K$ entries equal zero. Their dense-softmax entropies are therefore identical across states and rank coordinates within a pathway. Consequently, the confidence gate is a pathway-dependent positive constant, denoted

$$g_{\mathrm{xfer}} > 0, \qquad g_{\mathrm{spec}} > 0.$$

The positivity follows directly from $g = 1 - \lambda_g \sigma(\cdot)$ with $\lambda_g \in [0, 1]$ and finite logits.

Using Eq. (A25), each transferable rank-one factor product is therefore

$$\sqrt{PQ}\, g_{\text{xfer}}\, \mathbf{u}_i \mathbf{v}_i^\top,$$

while each modality-specific rank-one factor product is

$$\sqrt{PQ}\, g_{\text{spec}}\, \mathbf{u}_j \mathbf{v}_j^\top.$$

Define

$$c_{\text{xfer}} = \sqrt{PQ}\, g_{\text{xfer}}, \qquad c_{\text{spec}} = \sqrt{PQ}\, g_{\text{spec}}.$$

Both constants are strictly positive. Summing the corresponding rank-one terms yields exactly

$$\mathbf{D}_m = c_{\text{xfer}} \mathbf{U}_0 \mathbf{V}_0^\top + c_{\text{spec}} \mathbf{U}_m \mathbf{V}_m^\top. \tag{A29}$$

Hence CODAR realizes every target in the constructed family with zero reconstruction error.

Step 6: Strict limitation of fixed geometry. For the targets in Eq. (A29), orthogonality gives

$$\mathbf{S}_{\text{col}} = M c_{\text{xfer}}^2 \mathbf{U}_0 \mathbf{U}_0^\top + c_{\text{spec}}^2 \sum_{m=1}^{M} \mathbf{U}_m \mathbf{U}_m^\top, \tag{A30}$$

$$\mathbf{S}_{\text{row}} = M c_{\text{xfer}}^2 \mathbf{V}_0 \mathbf{V}_0^\top + c_{\text{spec}}^2 \sum_{m=1}^{M} \mathbf{V}_m \mathbf{V}_m^\top. \tag{A31}$$

Thus both operators have the same nonzero eigenvalue multiset:

$$\underbrace{M c_{\text{xfer}}^2, \ldots, M c_{\text{xfer}}^2}_{t \text{ times}}, \qquad \underbrace{c_{\text{spec}}^2, \ldots, c_{\text{spec}}^2}_{Ms \text{ times}}.$$

Let $\nu_1 \geq \cdots \geq \nu_{t+Ms} > 0$ denote these values in non-increasing order. Since the fixed geometry has rank $r = t + s$, the lower bound from Eq. (A24) gives

$$\mathcal{E}_{\text{fix}} \geq \sum_{j=r+1}^{t+Ms} \nu_j. \tag{A32}$$

This bound is attained by choosing the shared row and column subspaces to contain the same r matched directions associated with $\nu_1, \ldots, \nu_r$, and choosing each $\mathbf{\Gamma}_m$ to reproduce the retained rank-one terms. Therefore

$$\mathcal{E}_{\text{fix}} = \sum_{j=r+1}^{t+Ms} \nu_j. \tag{A33}$$

Because

$$t + Ms - r = (M - 1)s,$$

exactly $(M - 1)s$ positive eigenvalues remain in the tail, and hence

$$\mathcal{E}_{\text{fix}} \geq (M - 1)s \min\left\{ M c_{\text{xfer}}^2, c_{\text{spec}}^2 \right\} > 0. \tag{A34}$$

CODAR has zero reconstruction error on the same target family, which proves Theorem 1.

Factor-Realization Perturbation Bound

The constructive result above establishes exact realizability for a nonempty family of state-dependent targets. For a general target family, however, CODAR may only approximately realize a chosen transferable/private factorization. We next quantify this approximation error and relate it to the fixed-geometry lower bound. Fix a layer l and tower μ and suppress these indices. For each state m, write

$$\Delta \mathbf{W}^{(m)} = \Delta \mathbf{W}_{\text{xfer}}^{(m)} + \Delta \mathbf{W}_{\text{private}}^{(m)}, \tag{A35}$$

where $\operatorname{rank}(\Delta \mathbf{W}_{\text{xfer}}^{(m)}) \leq r_{\text{xfer}}$.

Let $\Delta\mathbf{W}^{(m)}_{\text{private},r_{\text{spec}}}$ denote a best rank-r_{spec} Frobenius approximation of the private component. By the Eckart–Young theorem,

$$\tau_m = \left\|\Delta\mathbf{W}^{(m)}_{\text{private}} - \Delta\mathbf{W}^{(m)}_{\text{private},r_{\text{spec}}}\right\|_F = \left(\sum_{j>r_{\text{spec}}} \sigma_j^2\big(\Delta\mathbf{W}^{(m)}_{\text{private}}\big)\right)^{1/2}. \tag{A36}$$

Factor the transferable component and the truncated private component as

$$\Delta\mathbf{W}^{(m)}_b = \mathbf{B}^{(m)}_b\big(\mathbf{A}^{(m)}_b\big)^\top, \qquad b \in \{\text{spec}, \text{xfer}\}, \tag{A37}$$

where $\Delta\mathbf{W}^{(m)}_{\text{spec}} = \Delta\mathbf{W}^{(m)}_{\text{private},r_{\text{spec}}}$.

Let CODAR reconstruct these factors as $\widehat{\mathbf{A}}^{(m)}_b$ and $\widehat{\mathbf{B}}^{(m)}_b$, and define

$$e_{A,b,m} = \left\|\widehat{\mathbf{A}}^{(m)}_b - \mathbf{A}^{(m)}_b\right\|_F, \qquad e_{B,b,m} = \left\|\widehat{\mathbf{B}}^{(m)}_b - \mathbf{B}^{(m)}_b\right\|_F. \tag{A38}$$

For convenience, write

$$\mathbf{E}^{(m)}_{A,b} = \widehat{\mathbf{A}}^{(m)}_b - \mathbf{A}^{(m)}_b, \qquad \mathbf{E}^{(m)}_{B,b} = \widehat{\mathbf{B}}^{(m)}_b - \mathbf{B}^{(m)}_b.$$

Then

$$\begin{aligned}&\widehat{\mathbf{B}}^{(m)}_b\big(\widehat{\mathbf{A}}^{(m)}_b\big)^\top - \mathbf{B}^{(m)}_b\big(\mathbf{A}^{(m)}_b\big)^\top \\ &= \mathbf{B}^{(m)}_b\big(\mathbf{E}^{(m)}_{A,b}\big)^\top + \mathbf{E}^{(m)}_{B,b}\big(\mathbf{A}^{(m)}_b\big)^\top + \mathbf{E}^{(m)}_{B,b}\big(\mathbf{E}^{(m)}_{A,b}\big)^\top.\end{aligned} \tag{A39}$$

Using $\|\mathbf{X}\mathbf{Y}^\top\|_F \le \|\mathbf{X}\|_2\|\mathbf{Y}\|_F$ for the first two terms and $\|\mathbf{X}\mathbf{Y}^\top\|_F \le \|\mathbf{X}\|_F\|\mathbf{Y}\|_F$ for the last term gives

$$\begin{aligned}&\left\|\widehat{\mathbf{B}}^{(m)}_b\big(\widehat{\mathbf{A}}^{(m)}_b\big)^\top - \mathbf{B}^{(m)}_b\big(\mathbf{A}^{(m)}_b\big)^\top\right\|_F \\ &\le \|\mathbf{B}^{(m)}_b\|_2 e_{A,b,m} + \|\mathbf{A}^{(m)}_b\|_2 e_{B,b,m} + e_{A,b,m}e_{B,b,m}.\end{aligned} \tag{A40}$$

The CODAR update is

$$\widehat{\Delta\mathbf{W}}^{(m)} = \sum_{b\in\{\text{spec},\text{xfer}\}} \widehat{\mathbf{B}}^{(m)}_b\big(\widehat{\mathbf{A}}^{(m)}_b\big)^\top. \tag{A41}$$

Applying the triangle inequality to the two pathway errors and the private truncation residual gives

$$\begin{aligned}&\left\|\widehat{\Delta\mathbf{W}}^{(m)} - \Delta\mathbf{W}^{(m)}\right\|_F \\ &\le \tau_m + \sum_{b\in\{\text{spec},\text{xfer}\}} \Big(\|\mathbf{B}^{(m)}_b\|_2 e_{A,b,m} + \|\mathbf{A}^{(m)}_b\|_2 e_{B,b,m} + e_{A,b,m}e_{B,b,m}\Big).\end{aligned} \tag{A42}$$

For brevity, define

$$u_m = \tau_m + \sum_{b\in\{\text{spec},\text{xfer}\}} \Big(\|\mathbf{B}^{(m)}_b\|_2 e_{A,b,m} + \|\mathbf{A}^{(m)}_b\|_2 e_{B,b,m} + e_{A,b,m}e_{B,b,m}\Big). \tag{A43}$$

Then

$$\left\|\widehat{\Delta\mathbf{W}}^{(m)} - \Delta\mathbf{W}^{(m)}\right\|_F \le u_m.$$

Define CODAR's aggregate squared reconstruction error as

$$\mathcal{E}_{\text{CODAR}} = \sum_{m\in\mathcal{M}} \left\|\widehat{\Delta\mathbf{W}}^{(m)} - \Delta\mathbf{W}^{(m)}\right\|_F^2. \tag{A44}$$

Squaring the state-wise bound and summing over states gives

$$\boxed{\mathcal{E}_{\text{CODAR}} \le \sum_{m\in\mathcal{M}} u_m^2.} \tag{A45}$$

Table A3: Dataset statistics and evaluation metrics.

Dataset	Train	Validation	Test	Metric
MM-IMDb	15,552	2,608	7,799	Macro-F1
UPMC-Food101	61,127	6,845	22,716	Accuracy
Hateful Memes	8,500	500	1,000	AUROC

Corollary 2 (Sufficient Condition for Strict Approximation Advantage)**.** *Let*

$$L_{\mathrm{fix}} = \max\left\{\sum_{j>r}\lambda_j(\mathbf{S}_{\mathrm{col}}), \sum_{j>r}\lambda_j(\mathbf{S}_{\mathrm{row}})\right\}. \tag{A46}$$

For the same target family, if

$$\sum_{m\in\mathcal{M}} u_m^2 < L_{\mathrm{fix}}, \tag{A47}$$

then

$$\mathcal{E}_{\mathrm{CODAR}} < \mathcal{E}_{\mathrm{fix}}. \tag{A48}$$

Hence the realized CODAR updates approximate the target family more closely than any fixed shared rank-r row/column geometry.

Proof. Equation (A45) gives $\mathcal{E}_{\mathrm{CODAR}} \le \sum_m u_m^2$, while Eq. (A24) gives $\mathcal{E}_{\mathrm{fix}} \ge L_{\mathrm{fix}}$. The result follows immediately from Eq. (A47).

Interpretation and scope. The perturbation bound separates the intrinsic truncation error τ_m from factor-realization errors induced by CODAR's dictionaries, routing generator, amplitudes, and gates. It provides a sufficient, factorization-dependent condition for outperforming fixed shared geometry in target-update reconstruction, but does not establish a universal approximation rate in N or K, guarantee convergence of the non-convex STE optimization, or imply a downstream-task performance bound. The constructive family in Theorem 1 is the exact special case with $\tau_m = e_{A,b,m} = e_{B,b,m} = 0$.

C Experimental Setup

Datasets and Metrics

We follow the data splits and evaluation protocol of the ICLR 2026 MoRA study (Zhao et al., 2026).

MM-IMDb. MM-IMDb (Arevalo et al., 2017) contains paired movie posters and plot summaries with 23 genre labels. We use 15,552 training, 2,608 validation, and 7,799 test samples and report Macro-F1 (%).

UPMC-Food101. UPMC-Food101 (Wang et al., 2015) contains web-collected image–text pairs from 101 food categories. We use 61,127 training, 6,845 validation, and 22,716 test samples and report top-1 classification accuracy (%).

Hateful Memes. Hateful Memes (Kiela et al., 2021) contains meme images paired with overlaid text for binary hate-speech detection. We use 8,500 training, 500 validation, and 1,000 test samples and report AUROC (%).

Missing-Modality Protocol

Let η denote the proportion of samples with an incomplete modality. In a single missing-state experiment, an η fraction of the samples lacks the designated modality and the remaining $1-\eta$ fraction retains both modalities. In the Both setting, $\eta/2$ of all samples are image missing, another $\eta/2$ are text missing, and the remaining $1-\eta$ are complete. Image and text are never simultaneously absent.

Following MoRA, missing images are replaced with white RGB images before the standard CLIP preprocessing pipeline, while missing texts are replaced with empty strings and processed by the

standard CLIP tokenizer. The availability state is provided by the data construction procedure; CODAR does not use an additional state detector. For every dataset split, missing setting, missing ratio, and random seed, the missing-modality assignment is generated once before optimization and stored. The resulting mask remains fixed across all training epochs, checkpoint selection, and final evaluation; it is not resampled between epochs. CODAR and every reproduced baseline use the exact same stored mask under a matched seed. For the Both setting, the image-missing and text-missing subsets are likewise fixed and disjoint.

The main experiments use $\eta = 70\%$. At $\eta = 70\%$, the three reported configurations correspond to image/text availability of $(100\%, 30\%)$, $(30\%, 100\%)$, and $(65\%, 65\%)$ for text-missing, image-missing, and Both settings, respectively.

Checkpoint protocols. We distinguish the scenario-specific benchmark protocol from the capacity–transfer diagnostic. For the main results, the controlled low-rank comparison, and the additional missing-ratio experiments, the text-missing, image-missing, and Both settings are trained independently. Each reported row therefore uses its own checkpoint, selected on a validation split constructed with the same availability distribution as that row. In a text-missing or image-missing run, the incomplete portion contains only the designated missing state and the remaining samples are complete. In a Both run, the incomplete portion is divided equally between image-missing and text-missing samples. Checkpoints are not transferred across these three benchmark settings.

The capacity–transfer diagnostic uses a different protocol. Each method is trained once on the Both-setting distribution at $\eta = 70\%$. The selected checkpoint is then held fixed and evaluated, without retraining or further adaptation, on complete-only, image-missing, text-missing, and Both test distributions. Hence, all four bars reported for a method in the capacity–transfer figure are produced by exactly the same trained parameter set. The statement that one checkpoint handles all availability states refers to this Both-trained, state-conditioned deployment setting; it does not mean that the independently trained rows in the benchmark tables share a checkpoint. Under this distribution, the full, image-missing, and text-missing states occupy 30%, 35%, and 35% of training examples, respectively. StateSpecificLoRA therefore receives one subset per adapter; its capacity–transfer result is a descriptive diagnostic, not a balanced causal test of parameter isolation.

Baseline Methods

We compare CODAR with CoOp (Zhou et al., 2022), MMP (Lee et al., 2023), MaPLe (Khattak et al., 2023), DePT (Zhang et al., 2024), DCP (Hu et al., 2024), and the missing-modality MoRA (Zhao et al., 2026). The results of CoOp, MMP, MaPLe, DePT, and DCP are retained from the MoRA study. We reproduce LoRA, VeRA, and MoRA on all three datasets, including MM-IMDb, using the common CLIP ViT-B/16 backbone, classifier, query/value attachment positions, training budget, and validation-selection rule. At the main ratio $\eta = 70\%$, all reproduced low-rank results are averaged over seeds $\{42, 43, 44\}$; paired methods use identical missing masks under each seed. Only the prompt-based reported-source values are identified by † and are excluded from paired statistical tests. The controlled low-rank baselines are defined under the same adapted projections, training budget, validation-selection rule, missing masks, and parameter-count convention as CODAR. VB-LoRA (Li et al., 2024b) and Uni-LoRA (Li et al., 2025) retain their native global reconstruction mechanisms. StateCondLoRA is a parameter-matched rank-4 control in which availability embeddings condition both factors directly. It uses the same full-to-missing distillation objective as CODAR, but excludes CODAR's dictionary reconstruction, dual-path decomposition, and routing-consistency regularization. FixedBasis uses one state-invariant rank-4 basis pair, FixedBasis-PM increases its static capacity to approximately match CODAR's parameter budget, and StateSpecificLoRA assigns an independent rank-4 adapter to each availability state. These controls probe constrained sharing, static capacity, and state-wise isolation; the latter is interpreted with the state-specific exposure caveat above.

Controlled Baseline Configurations

Table A4 lists the configurations used for the reproduced low-rank methods and diagnostic controls. All methods adapt the query and value projections of Layers 11–12 in both towers. Unless specified

Table A4: Configurations of reproduced low-rank PEFT methods and diagnostic controls. Parameter counts exclude the dataset-specific classifier unless explicitly stated.

Method	Rank	Shared structure	Configuration	Adapter params
LoRA	8	None	Independent $\mathbf{A}/\mathbf{B}$ factors for every adapted projection; $\alpha = 8$ and zero adapter dropout.	81,920
VeRA	4	Frozen random factors	One frozen random factor pair is shared within each tower and projection role. Each adapted module learns an output scale $\mathbf{b} \in \mathbb{R}^{d_{\text{out}}}$ and a rank scale $\mathbf{d} \in \mathbb{R}^4$; random factors use Kaiming-uniform initialization.	5,152
VB-LoRA	4	Global vector bank	One vector bank is shared by all adapted projections with bank size $N_v = 64$, vector width 64, Top-K selection $K_v = 4$, and routing temperature $\tau_v = 0.5$. Selection logits and vector-bank entries are trainable.	23,808
Uni-LoRA	4	Network-wide basis	A shared set of 32 basis vectors reconstructs all adapted factors. The reconstruction width is 64 and the coefficient bottleneck is 16; layer- and tower-specific coefficient heads are learned.	12,544
MoRA	8	Common/private factors	The effective rank is split into four modality-common and four modality-private components. Common factors are shared across towers, while private factors follow the observed modality branch; $\alpha = 8$.	112,640
StateCondLoRA	4	Conditional factor generator	A 16-dimensional availability embedding is combined with 24-dimensional layer and tower embeddings into a 64-dimensional context. A shared two-layer GELU hypernetwork of hidden width 64 conditions both factor sides directly. The final projection is zero initialized. StateCondLoRA uses the same full-to-missing distillation objective as CODAR, but does not use dictionary reconstruction, dual-path decomposition, or routing-consistency regularization.	101,120
CODAR	4	Dual-path dictionaries	$r_{\text{spec}} = r_{\text{xfer}} = 2$, $N = 64$, $K = 4$, $R_g = 4$, and $R_{\text{res}} = 1$; complete settings appear in Table A2.	101,116
FixedBasis	4	Fixed basis pair	One frozen orthonormal input/output basis pair is used per projection. Only 384 state-independent mixing and rank-scaling coefficients are optimized; the effective row and column spaces do not vary by state.	384
FixedBasis-PM	10	Static learned basis	A state-invariant rank-10 factor pair is learned for every adapted projection, together with a 2,144-parameter global mixing/scaling table. This gives 104,544 adapter parameters, or 106,594 parameters including the 2,050-parameter Hateful Memes classifier (reported as 106.6K).	104,544
StateSpecificLoRA	3×4	None across states	Three independent rank-4 LoRA copies are stored for $\{\text{full}, \text{vis-miss}, \text{txt-miss}\}$. The availability state selects one copy without cross-state sharing.	122,880

otherwise, the LoRA scaling is set equal to the effective rank, adapter dropout is zero, and the same dataset-specific optimizer, scheduler, batch size, and checkpoint-selection rule as CODAR are used.

We retain each method's native or recommended effective-rank configuration rather than forcing an identical numerical rank across parameterizations. Fairness is controlled through identical adapted projections, data splits, missing masks, optimization budgets, and checkpoint selection, while the resulting trainable parameter counts are reported explicitly. We therefore pair native-configuration comparisons with rank-4 and parameter-matched controls; the conclusion does not rely on comparisons only against substantially smaller baselines.

Implementation and Optimization

We use CLIP ViT-B/16 as the frozen image–text backbone. The 512-dimensional image and text representations are concatenated and passed to a linear classification head. We use binary cross-entropy with logits for MM-IMDb and cross-entropy for UPMC-Food101 and Hateful Memes. CODAR is inserted into the query and value projections of Layers 11–12 in both the visual and textual encoders. All other CLIP parameters remain frozen. Complete adapter, routing, and objective settings are listed in Table A2.

We use AdamW (Loshchilov & Hutter, 2019) with $\beta_1 = 0.9$, $\beta_2 = 0.98$, and $\epsilon = 10^{-8}$. The learning rates are 10^{-3} for dictionary banks, 10^{-2} for routing parameters, 10^{-3} for state-dependent parame-

Table A5: Dataset-specific optimization settings.

Dataset	Batch	Epochs	Weight decay	Warmup ratio
MM-IMDb	256	15	0.02	0.05
UPMC-Food101	256	20	0.02	0.05
Hateful Memes	128	20	0.01	0.10

ters, and 10^{-2} for the classification head. We apply linear warmup followed by cosine learning-rate decay. We use mixed precision without gradient clipping.

The checkpoint with the best validation Macro-F1, accuracy, or AUROC is selected for MM-IMDb, UPMC-Food101, and Hateful Memes, respectively. All newly conducted experiments are averaged over three independent runs with random seeds $\{42, 43, 44\}$. Experiments are conducted on a single NVIDIA GeForce RTX 4090 GPU.

Efficiency Measurement Protocol

The efficiency results in Figure 4 of the main paper are measured on a single NVIDIA GeForce RTX 4090 with batch size 16. Inputs are preloaded to host memory, and dataloader and disk-I/O time are excluded. The same fixed sequence of Both-setting batches is used for every method, with state proportions $0.30/0.35/0.35$ for full, image-missing, and text-missing samples, respectively. All methods use PyTorch automatic mixed precision with FP16 autocast; trainable parameters and optimizer states remain in FP32.

For training throughput, each measurement includes the forward pass, task and auxiliary-loss computation, backward pass, and AdamW update. Gradients are cleared with `set_to_none=True`. We discard 50 warm-up iterations and time the following 200 iterations. For inference, models are evaluated under `torch.no_grad()` with the same FP16 autocast setting; 50 warm-up iterations are followed by 500 timed forward passes. We call `torch.cuda.synchronize()` immediately before and after every timed region. Each experiment is repeated five times, and Figure 4 reports the mean throughput in samples per second.

For cached CODAR inference, all factors for $(\mu, l, p_{\mathrm{proj}}, m) \in \{\mathrm{vis}, \mathrm{txt}\} \times \{11, 12\} \times \{\mathrm{qry}, \mathrm{val}\} \times \mathcal{M}$ are generated once before timing and retained on the GPU. Cache construction is excluded from inference throughput because it is a one-time operation for a trained checkpoint. During a timed batch, the known availability label selects the corresponding cached factors; no routing generator or Top-K operation is executed. The uncached CODAR result includes on-demand factor generation in every forward pass. Peak-memory allocation is reset after warm-up, although Figure 4 reports throughput and parameter count rather than memory.

Training Objective

The complete CODAR objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{task}} + \lambda_{\mathrm{KD}}\mathcal{L}_{\mathrm{KD}} + \lambda_{\mathrm{route}}\mathcal{L}_{\mathrm{route}} + \lambda_{\mathrm{bal}}\mathcal{L}_{\mathrm{bal}}. \tag{A49}$$

We use $\lambda_{\mathrm{KD}} = 1.0$, $\lambda_{\mathrm{route}} = 0.1$, and $\lambda_{\mathrm{bal}} = 0.01$.

Confidence-aware gating. For chunk c and rank coordinate i of factor role f, let $\mathbf{p}_{\mathrm{soft}} = \mathrm{Softmax}(\mathbf{L}_l^{\mu,f,m}(c, i, :)/\tau_r)$ be the dense soft-routing distribution. Its normalized entropy is

$$\mathcal{H}_{l,c,i}^{\mu,f,m} = -\frac{1}{\log N}\sum_{n=1}^{N} p_{\mathrm{soft}}(n)\log p_{\mathrm{soft}}(n). \tag{A50}$$

For pathway $b \in \{\mathrm{spec}, \mathrm{xfer}\}$, let $f_A(b) = A_b$ and $f_B(b) = B_b$ denote its input and output factor roles. We first average over the chunks on each side, then average the two sides:

$$\mathcal{H}_{l,i}^{\mu,b,m} = \frac{1}{2}\left(\frac{1}{P}\sum_{c}\mathcal{H}_{l,c,i}^{\mu,f_A(b),m} + \frac{1}{Q}\sum_{c}\mathcal{H}_{l,c,i}^{\mu,f_B(b),m}\right). \tag{A51}$$

We maintain a pathway-level exponential moving average reference, initialized to one and updated without gradients when factors are reconstructed during training:

$$\overline{\mathcal{H}}^{\,b} \leftarrow \beta_{\mathcal{H}} \overline{\mathcal{H}}^{\,b} + (1-\beta_{\mathcal{H}}) \left\langle \mathcal{H}_{l,i}^{\mu,b,m} \right\rangle_i . \tag{A52}$$

Here, $\beta_{\mathcal{H}}$ is the EMA decay, and $\langle\cdot\rangle_i$ averages rank coordinates for the current module and availability state. The reference is fixed at inference.

The confidence gate is

$$g_{l,i}^{\mu,b,m} = 1 - \lambda_g \sigma\big((\mathcal{H}_{l,i}^{\mu,b,m} - \overline{\mathcal{H}}^{\,b})/\gamma\big), \tag{A53}$$

where $\lambda_g \in [0,1]$ controls suppression, $\gamma > 0$ is the temperature, and σ is the sigmoid. Concentrated routes receive larger gates, whereas diffuse routes are attenuated. Entropy is not detached when computing g, so the task loss can affect routing through the gate; only the EMA reference is detached. Applying $\sqrt{\kappa_{l,i}^{\mu,b,m} g_{l,i}^{\mu,b,m}}$ to both factors scales the corresponding rank-one update by $\kappa_{l,i}^{\mu,b,m} g_{l,i}^{\mu,b,m}$.

Load balancing. Let

$$\mathfrak{B} = \{\mathbf{V}_{\text{vis}}^A, \mathbf{V}_{\text{vis}}^B, \mathbf{V}_{\text{txt}}^A, \mathbf{V}_{\text{txt}}^B, \mathbf{V}_{\text{sh}}^A\}$$

denote the physical dictionary banks. For each bank $\beta \in \mathfrak{B}$, aggregate soft-routing weights over all routes using that bank:

$$u_n^\beta = \sum_{(x,m,\mu,l,f,c,i)\in\mathcal{R}(\beta)} p_{\text{soft}}(n), \qquad \widetilde{u}_n^\beta = \frac{u_n^\beta}{\sum_{n'} u_{n'}^\beta}. \tag{A54}$$

Here, $\mathcal{R}(\beta)$ contains all routes that use bank β. For the shared bank, routes from both towers are merged by atom index. The load-balancing penalty is

$$\mathcal{L}_{\text{bal}} = \sum_{\beta\in\mathfrak{B}} \mathrm{KL}\left(\widetilde{\mathbf{u}}^\beta \,\middle\|\, \mathbf{1}/N\right). \tag{A55}$$

This weak regularizer discourages persistent dictionary-atom starvation.

Consistency distillation. Complete-input predictions serve as stop-gradient teachers for missing states (Hinton et al., 2015). Let z^m denote the logits under availability state m, and let sg denote stop-gradient. For single-label classification, we use

$$\mathcal{L}_{\text{KD}} = \mathbb{E}_{m\neq\text{full}} T^2 \,\mathrm{KL}\left(\mathrm{Softmax}\left(\mathrm{sg}(z^{\text{full}})/T\right) \,\middle\|\, \mathrm{Softmax}\left(z^m/T\right)\right). \tag{A56}$$

For multi-label classification on MM-IMDb, we use the corresponding per-class Bernoulli KL. The distillation temperature is $T = 2$.

Transferable routing consistency. For matched modules $l \in \mathcal{L}_{\text{match}}$ (e.g., same relative depth), let $p_{l,c,i,n}^{(\mu,A_{\text{xfer}},m)}$ denote the soft routing weight of shared atom n for input chunk c and rank coordinate i. We average the transferable input routes over chunks:

$$\overline{p}_{l,n,i}^{(\mu,m)} = \frac{1}{P}\sum_c p_{l,c,i,n}^{(\mu,A_{\text{xfer}},m)}, \qquad \overline{\mathbf{P}}_l^{(\mu,m)} \in \mathbb{R}^{N\times r_{\text{xfer}}}. \tag{A57}$$

The rows of $\overline{\mathbf{P}}_l^{(\mu,m)}$ index shared dictionary atoms, and its columns index rank coordinates. We align visual and textual co-usage matrices using

$$\mathcal{L}_{\text{route}} = \frac{1}{\sum_m \omega_m} \sum_m \frac{\omega_m}{|\mathcal{L}_{\text{match}}|} \sum_{l\in\mathcal{L}_{\text{match}}} \left\| \overline{\mathbf{P}}_l^{(\text{vis},m)} \big(\overline{\mathbf{P}}_l^{(\text{vis},m)}\big)^\top - \overline{\mathbf{P}}_l^{(\text{txt},m)} \big(\overline{\mathbf{P}}_l^{(\text{txt},m)}\big)^\top \right\|_F^2. \tag{A58}$$

Here, $\omega_m \geq 0$ weights availability state m. Complete-state routes receive weight one, while each missing state receives weight $\xi = 0.3$.

This objective aligns route co-usage in shared-dictionary coordinates, rather than feature representations. It is invariant to permutations of rank coordinates: for any permutation matrix $\mathbf{\Pi} \in \mathbb{R}^{r_{\text{xfer}}\times r_{\text{xfer}}}$,

$$(\overline{\mathbf{P}}\mathbf{\Pi})(\overline{\mathbf{P}}\mathbf{\Pi})^\top = \overline{\mathbf{P}}\mathbf{\Pi}\mathbf{\Pi}^\top\overline{\mathbf{P}}^\top = \overline{\mathbf{P}}\overline{\mathbf{P}}^\top.$$

Dictionary-atom indices remain aligned across towers.

Table A6: CODAR attachment position on UPMC-Food101 at $\eta = 70\%$ under the Both setting (accuracy, mean±std over seeds $\{42, 43, 44\}$).

Position	Front	Middle	Rear (main)
Layers	1–2	6–7	11–12
Accuracy	83.88±0.41	84.22±0.37	**84.94±0.33**

Attachment Position Analysis

We evaluate CODAR at three network depths in both CLIP encoders. The front, middle, and rear variants attach CODAR to the query and value projections of Layers 1–2, 6–7, and 11–12, respectively. All architectural and optimization settings other than the attachment position are unchanged. The experiment is conducted on UPMC-Food101 under the 70% Both setting.

Following the MoRA protocol, all main experiments attach CODAR to the query and value projections of Layers 11–12 in both CLIP encoders. As a post-hoc sensitivity analysis, we additionally evaluate front, middle, and rear attachment at Layers 1–2, 6–7, and 11–12, respectively, while keeping all other settings unchanged. On UPMC-Food101 under the 70% Both setting, rear attachment achieves 84.94±0.33 accuracy, compared with 83.88±0.41 and 84.22±0.37 for front and middle attachment, respectively. This result shows that the MoRA-matched rear-layer configuration is also favorable among the tested positions.

D Controlled Low-Rank Comparisons

Averaged Controlled PEFT Results

Table A7: Controlled low-rank PEFT comparison at $\eta = 70\%$. Each value averages the text-missing, image-missing, and Both settings. All reproduced methods use matched seeds, missing masks, adapted projections, training budgets, and checkpoint selection.

Method	MM-IMDb	Food101	Hateful
LoRA	51.35±0.61	82.14±0.39	67.97±0.58
VeRA	51.62±0.65	82.38±0.39	68.10±0.60
VB-LoRA	52.41±0.60	83.06±0.36	68.36±0.57
Uni-LoRA	52.88±0.58	83.51±0.34	68.61±0.55
MoRA	54.09±0.45	84.32±0.35	68.87±0.46
StateCondLoRA	54.26±0.59	84.57±0.33	70.49±0.56
CODAR	**55.19±0.64**	**85.40±0.32**	**71.82±0.53**

Table A7 compares CODAR with representative low-rank parameterizations under the same backbone, adapted projections, missing masks, optimization budget, and validation protocol. CODAR consistently improves over globally shared reconstruction methods such as VeRA, VB-LoRA, and Uni-LoRA, indicating that compact parameter sharing alone does not capture the availability-dependent structure required by the task.

StateCondLoRA provides a stronger parameter-matched control. It directly conditions both low-rank factors on the availability state and contains essentially the same number of trainable parameters as CODAR. It uses the same full-to-missing distillation objective as CODAR, but does not employ dictionary reconstruction, separate modality-specific and transferable pathways, or routing consistency. CODAR exceeds this control by 0.93, 0.83, and 1.33 points on MM-IMDb, UPMC-Food101, and Hateful Memes, respectively. These results indicate that availability conditioning and distillation alone do not account for CODAR's gains; structuring conditional updates through reusable dual-path dictionaries provides an additional benefit.

Results by Missing-Modality Setting

We compare representative low-rank PEFT parameterizations under the matched $\eta = 70\%$ protocol. All methods use the same frozen CLIP ViT-B/16 backbone, query/value attachment positions,

Table A8: Controlled low-rank PEFT at $\eta = 70\%$ (mean±std over seeds $\{42, 43, 44\}$). Average spans the three missing settings.

Dataset	Image	Text	LoRA	VeRA	VB-LoRA	Uni-LoRA	MoRA	StateCondLoRA	CODAR
MM-IMDb	100%	30%	49.96±0.67	50.18±0.70	51.02±0.65	51.47±0.63	52.81±0.73	53.08±0.66	54.01±0.71
	30%	100%	53.77±0.55	54.01±0.59	54.89±0.54	55.36±0.52	56.34±0.41	56.41±0.50	57.16±0.68
	65%	65%	50.32±0.62	50.67±0.66	51.32±0.61	51.81±0.59	53.12±0.33	53.29±0.63	54.41±0.65
	Average		51.35±0.61	51.62±0.65	52.41±0.60	52.88±0.58	54.09±0.45	54.26±0.59	**55.19±0.64**
UPMC-Food101	100%	30%	79.23±0.45	79.52±0.44	80.32±0.42	80.78±0.40	81.04±0.42	81.49±0.40	82.32±0.38
	30%	100%	87.02±0.36	87.18±0.35	87.84±0.32	88.11±0.31	88.15±0.31	88.42±0.30	88.95±0.29
	65%	65%	80.17±0.41	80.44±0.42	81.02±0.38	81.64±0.36	83.76±0.36	83.80±0.34	84.94±0.33
	Average		82.14±0.39	82.38±0.39	83.06±0.36	83.51±0.34	84.32±0.35	84.57±0.33	**85.40±0.32**
Hateful Memes	100%	30%	67.41±0.61	67.55±0.63	67.82±0.60	68.04±0.58	68.27±0.54	69.83±0.59	71.68±0.49
	30%	100%	67.78±0.58	67.92±0.60	68.13±0.58	68.39±0.56	68.59±0.51	70.15±0.56	71.26±0.47
	65%	65%	68.72±0.56	68.83±0.57	69.13±0.55	69.40±0.53	69.74±0.48	71.49±0.53	72.51±0.52
	Average		67.97±0.58	68.10±0.60	68.36±0.57	68.61±0.55	68.87±0.46	70.49±0.56	**71.82±0.53**

Table A9: Additional missing-ratio results at $\eta \in \{50\%, 90\%\}$. For compactness, this sensitivity table reports the matched MoRA and CODAR comparison; the full controlled baseline table is Table A7. MoRA and CODAR report mean (sample variance) over matched seeds $\{42, 43, 44\}$. Settings are trained separately; Average spans the three missing settings.

Dataset	η	Image	Text	MoRA	CODAR
MM-IMDb	50%	100%	50%	54.46 (0.139)	**55.60 (0.099)**
		50%	100%	57.24 (0.111)	**58.43 (0.082)**
		75%	75%	55.16 (0.106)	**56.24 (0.109)**
		Average		55.62 (0.015)	**56.76 (0.014)**
	90%	100%	10%	49.63 (0.102)	**50.77 (0.116)**
		10%	100%	54.49 (0.121)	**55.68 (0.135)**
		55%	55%	51.30 (0.133)	**52.51 (0.130)**
		Average		51.81 (0.013)	**52.99 (0.013)**
UPMC-Food101	50%	100%	50%	84.09 (0.082)	**85.21 (0.071)**
		50%	100%	89.79 (0.060)	**90.63 (0.051)**
		75%	75%	86.55 (0.103)	**87.70 (0.087)**
		Average		86.81 (0.081)	**87.85 (0.069)**
	90%	100%	10%	78.49 (0.109)	**79.58 (0.103)**
		10%	100%	86.98 (0.070)	**88.05 (0.068)**
		55%	55%	80.73 (0.110)	**81.78 (0.126)**
		Average		82.07 (0.095)	**83.13 (0.097)**
Hateful Memes	50%	100%	50%	71.83 (0.194)	**72.83 (0.153)**
		50%	100%	71.37 (0.173)	**72.30 (0.148)**
		75%	75%	68.76 (0.212)	**73.03 (0.194)**
		Average		70.65 (0.192)	**72.72 (0.164)**
	90%	100%	10%	66.90 (0.235)	**68.30 (0.221)**
		10%	100%	68.41 (0.198)	**69.71 (0.173)**
		55%	55%	65.07 (0.212)	**67.18 (0.207)**
		Average		66.80 (0.215)	**68.39 (0.200)**

training budget, missing masks, validation-selection rule, and seeds $\{42, 43, 44\}$. The text-missing, image-missing, and Both rows are trained independently, and the first three rows in each dataset block report those three checkpoints, respectively.

Across the three datasets, CODAR improves over the strongest controlled alternative, StateCondLoRA, by 0.93, 0.83, and 1.33 points on MM-IMDb, UPMC-Food101, and Hateful Memes, respectively. The gains over globally shared reconstruction methods are larger, supporting the benefit of combining state-conditioned geometry with reusable dual-path dictionaries.

E Additional Missing-Ratio Results

The text-missing, image-missing, and Both configurations are retrained independently at each missing ratio. Thus, every non-average CODAR entry in Table A9 corresponds to a separately selected checkpoint rather than evaluation of one checkpoint under multiple scenarios.

Table A10: Holm-adjusted paired-bootstrap p-values for CODAR against the controlled low-rank baselines.

Baseline	MM-IMDb	Food101	Hateful Memes
LoRA	< 0.0001	< 0.0001	< 0.0001
VeRA	< 0.0001	< 0.0001	< 0.0001
VB-LoRA	0.0003	0.0004	< 0.0001
Uni-LoRA	0.0008	0.0011	0.0002
MoRA	0.0062	0.0076	0.0007
StateCondLoRA	0.0147	0.0183	0.0116

In this descriptive sensitivity check, CODAR remains stronger than MoRA at both additional missing ratios across all three datasets. Relative to MoRA, its average gains at $\eta = 50\%/90\%$ are 1.14/1.18 points on MM-IMDb, 1.04/1.06 points on UPMC-Food101, and 2.07/1.59 points on Hateful Memes. The repeated-seed results show that the improvement observed at the default $\eta = 70\%$ persists under both milder and more severe missing-modality conditions, suggesting that the gain is not specific to a single missing ratio.

Statistical Testing Protocol

We use two-sided paired bootstrap tests with 10,000 resamples. Predictions are paired by test example, seed, availability setting, and the fixed missing mask. For a controlled low-rank comparison on one dataset, each bootstrap replicate resamples test indices with replacement independently within every seed–setting pair, while applying the same sampled indices to CODAR and the baseline. We recompute the task metric for each of the three availability settings and each of the three seeds, average the three seeds within a setting, and then take the arithmetic mean over text-missing, image-missing, and Both settings. This nine-run average metric difference is the dataset-level test statistic. The two-sided p-value is twice the smaller empirical tail probability around zero.

Holm correction is applied across the six controlled low-rank baselines within each dataset. For the capacity–transfer diagnostic, the same paired-bootstrap procedure is applied separately to the four fixed-checkpoint evaluation states, and Holm correction is applied across those four state-wise comparisons.

Under this paired test-example bootstrap, all controlled low-rank comparisons remain significant after Holm correction at the 0.05 level. The state-wise capacity–transfer comparisons are likewise significant at the 0.01 level. These tests quantify uncertainty over matched test examples across the three trained seeds; they should not be interpreted as a separate estimate of the full distribution of training-seed variability.

F Supplementary Results

Extended Ablations

Table A11 complements the core architectural ablations in Table 2 with generator and training-objective variants. All results use the same matched seeds, missing masks, optimization budgets, and checkpoint-selection protocol.

Replacing the hierarchical CP generator with an MLP reduces performance by 0.67/0.61/0.51 points on MM-IMDb, UPMC-Food101, and Hateful Memes, respectively, while removing the hierarchical generator causes a larger drop. Among auxiliary components, confidence-aware gating and full-to-missing distillation provide the largest gains, with routing consistency, load balancing, and sparse Top-K routing contributing further improvements. These results complement the main-text architectural ablations and show that the auxiliary mechanisms improve the core state-conditioned dual-path design.

Geometry diagnostics. For each update ΔW, we extract the leading r left and right singular vectors as orthonormal bases for its column and row subspaces, respectively. For two bases Q_1, Q_2, we compute principal angles as $\theta_i = \arccos \sigma_i(Q_1^\top Q_2)$ and report their mean. For three availability states, we further average this quantity over the three state pairs. To character-

Table A11: Extended ablations at $\eta = 70\%$, averaged over three missing-modality settings.

Variant	MM-IMDb	Food101	Hateful
CODAR	**55.19**	**85.40**	**71.82**
Generator architecture			
w/o Hierarchical Gen.	54.17	84.47	70.91
w/o Layer Residual	54.84	85.12	71.56
MLP Gen. (vs. CP)	54.52	84.79	71.31
Training objective			
w/o Confidence Gate	54.31	84.66	71.05
w/o Routing Consistency	54.47	84.78	71.18
w/o Load Balancing	54.79	85.04	71.46
w/o Distillation	54.39	84.73	71.11
Dense Routing ($K = N$)	54.68	84.91	71.34

Table A12: Geometry of CODAR conditioning variants under the Both setting (mean±std over 8 projections and 9 state–seed pairs).

Variant	Left angle	Right angle	Eff. union rank (L/R)	Tail ratio (L/R)
CODAR	$78.14^\circ \pm 0.86^\circ$	$79.63^\circ \pm 0.74^\circ$	10.86 ± 0.17 / 11.05 ± 0.14	0.53 ± 0.01 / 0.54 ± 0.01
State-invariant routing	$0.00^\circ \pm 0.00^\circ$	$0.00^\circ \pm 0.00^\circ$	4.00 ± 0.00 / 4.00 ± 0.00	0.00 ± 0.00 / 0.00 ± 0.00
Input-side cond. only	$0.00^\circ \pm 0.00^\circ$	$76.92^\circ \pm 1.06^\circ$	4.00 ± 0.00 / 10.61 ± 0.22	0.00 ± 0.00 / 0.50 ± 0.02
Output-side cond. only	$75.48^\circ \pm 1.14^\circ$	$0.00^\circ \pm 0.00^\circ$	10.39 ± 0.24 / 4.00 ± 0.00	0.49 ± 0.02 / 0.00 ± 0.00

ize their union geometry, we concatenate the three rank-r bases as $Q_{\text{all}} = [Q_1, Q_2, Q_3]$. Let $p_i = \sigma_i^2(Q_{\text{all}}) / \sum_j \sigma_j^2(Q_{\text{all}})$. We define the effective union rank as $\exp(-\sum_i p_i \log p_i)$ and the tail ratio as $\sum_{i>r} \sigma_i^2(Q_{\text{all}}) / \sum_i \sigma_i^2(Q_{\text{all}})$. The same definitions are applied independently to the left and right subspaces.

One-Sided Conditioning Geometry

To isolate the contribution of conditioning each low-rank factor, we compare CODAR with state-invariant routing and variants that condition only the input or output factor. Results follow the same geometry protocol as Table 3.

State-invariant routing leaves both factor subspaces unchanged across availability states. Conditioning only the input factor expands the right subspace while leaving the left subspace fixed, whereas conditioning only the output factor produces the complementary pattern. Conditioning both factors enables CODAR to vary both subspaces jointly, yielding richer state-dependent update geometry.

Cross-Dataset Geometry Replication

Across MM-IMDb and UPMC-Food101, the cross-state geometry shows the same overall pattern as on Hateful Memes, although the magnitude of separation varies across datasets and factor sides. Different-state updates exhibit larger principal angles and higher union effective ranks than same-state updates across seeds, while remaining less separated than random rank-4 subspaces. The asymmetric gaps between the left and right subspaces further suggest that modality availability does not affect the two factor sides uniformly. Together, these results indicate that cross-state subspace separation is not specific to Hateful Memes, but recurs across all three benchmarks with dataset-dependent strength.

Energy-Weighted Fixed-Geometry Deficit. The subspace diagnostics above treat orthonormal basis directions uniformly. We therefore additionally evaluate the energy-weighted consequence of cross-state separation using the actual state-specific LoRA updates. For each adapted projection and seed, we form $\mathbf{S}_{\text{col}} = \sum_m \mathbf{D}_m \mathbf{D}_m^\top$ and $\mathbf{S}_{\text{row}} = \sum_m \mathbf{D}_m^\top \mathbf{D}_m$, and report the normalized spectral tail energy beyond rank r:

$$\rho_{\text{col}}^E = \frac{\sum_{j>r} \lambda_j(\mathbf{S}_{\text{col}})}{\text{tr}(\mathbf{S}_{\text{col}})}, \qquad \rho_{\text{row}}^E = \frac{\sum_{j>r} \lambda_j(\mathbf{S}_{\text{row}})}{\text{tr}(\mathbf{S}_{\text{row}})}. \tag{A59}$$

As shown in Table A14, the energy-weighted deficits remain substantially above zero on all three

Table A13: Cross-dataset replication of state-dependent update geometry. Results follow the same protocol as Table 3.

Dataset	Comparison	Left angle	Right angle	Eff. rank (L/R)	Tail ratio (L/R)
MM-IMDb	Same state, diff. seeds	$72.18^\circ \pm 3.74^\circ$	$69.86^\circ \pm 4.12^\circ$	10.81 ± 0.49 / 10.42 ± 0.61	0.53 ± 0.04 / 0.50 ± 0.05
	Diff. states, same seed	$78.46^\circ \pm 1.21^\circ$	$79.73^\circ \pm 0.88^\circ$	11.18 ± 0.11 / 11.26 ± 0.09	0.55 ± 0.01 / 0.56 ± 0.01
	Random rank-4 subspaces	$86.11^\circ \pm 0.29^\circ$	$86.27^\circ \pm 0.22^\circ$	11.91 ± 0.02 / 11.93 ± 0.01	0.61 ± 0.01 / 0.62 ± 0.00
UPMC-Food101	Same state, diff. seeds	$68.97^\circ \pm 4.46^\circ$	$72.64^\circ \pm 3.18^\circ$	10.29 ± 0.67 / 10.88 ± 0.43	0.49 ± 0.05 / 0.53 ± 0.03
	Diff. states, same seed	$82.06^\circ \pm 0.69^\circ$	$78.91^\circ \pm 1.34^\circ$	11.46 ± 0.06 / 11.17 ± 0.12	0.58 ± 0.01 / 0.55 ± 0.01
	Random rank-4 subspaces	$86.24^\circ \pm 0.21^\circ$	$86.15^\circ \pm 0.28^\circ$	11.93 ± 0.01 / 11.91 ± 0.02	0.62 ± 0.00 / 0.61 ± 0.01

Table A14: Energy-weighted fixed-geometry deficits across datasets. Results are mean±std over adapted projections and random seeds. Larger values indicate a larger fraction of update energy that cannot be captured by a shared rank-r geometry.

Dataset	Column-side deficit	Row-side deficit
MM-IMDb	0.271 ± 0.034	0.304 ± 0.029
UPMC-Food101	0.342 ± 0.027	0.286 ± 0.036
Hateful Memes	0.309 ± 0.025	0.361 ± 0.031

datasets. The column-/row-side deficits are $0.271/0.304$ on MM-IMDb, $0.342/0.286$ on UPMC-Food101, and $0.309/0.361$ on Hateful Memes. Thus, the observed cross-state separation is not confined to low-energy basis directions: a non-negligible fraction of the actual update energy lies outside the best shared rank-4 geometry. This empirical result complements the basis-level principal-angle, effective-rank, and tail-ratio analyses and provides an empirical instantiation of the fixed-geometry reconstruction limitation derived in Appendix B.

Extension to Embedding Tasks

We evaluate semantic retrieval on the UPMC-Food101 test split. Every test image serves once as a query, while the gallery contains all images from the same complete test split. For a complete query, the reference image and its title are encoded separately and combined as

$$\mathbf{q}_{\text{full}} = \mathrm{Norm}\big(\mathbf{f}_v + \mathbf{f}_t\big), \tag{A60}$$

where $\mathrm{Norm}(\cdot)$ denotes ℓ_2 normalization. For text-missing retrieval, we use a modality-absent, image-only inference variant: the query is constructed solely from the reference image embedding, and CODAR is evaluated with state $m = \text{txt-miss}$. We do not construct an empty string or supply padding-only or empty-text tokens to the text encoder. This differs from the classification benchmark protocol, where both towers remain active and missing text is represented by an empty string processed by the standard CLIP tokenizer and passed to the text encoder. The difference reflects the retrieval query being defined directly in the shared embedding space rather than through the classification fusion head.

Gallery items use image embeddings only:

$$\mathbf{g}_j = \mathrm{Norm}\big(\mathbf{f}_{v,j}\big). \tag{A61}$$

Retrieval uses cosine similarity, implemented as the dot product between the normalized query and gallery embeddings. Because the query image also appears in the gallery, its self-similarity is set to $-\infty$ before ranking. All remaining gallery images with the same UPMC-Food101 class label as the query are treated as positives. Recall@K counts a query as correct when at least one non-self positive occurs among its top-K retrieved images.

Across three seeds, CODAR improves R@1/R@5/R@10 by 1.18/0.82/0.57 points for complete queries and 1.28/0.94/0.66 points for text-missing queries, indicating that state-conditioned composition also benefits the learned embedding space.

G Reproducibility

Unless otherwise specified, newly conducted experiments are averaged over three independent runs with random seeds $\{42, 43, 44\}$. Sourced prompt results are the explicit exception. The frozen backbone is CLIP ViT-B/16, and only CODAR, the classification head, and the corresponding baseline

Table A15: Semantic image retrieval on the UPMC-Food101 test split using multimodal complete queries or image-only text-missing queries. The gallery contains all test images, the identical query image is excluded, and results are Recall@K (mean±std over seeds $\{42, 43, 44\}$, %).

Condition	Method	R@1	R@5	R@10
Complete	MoRA	74.02±0.34	87.81±0.22	90.75±0.18
	CODAR	**75.20±0.31**	**88.63±0.20**	**91.32±0.17**
Text-missing	MoRA	66.27±0.41	81.77±0.29	85.82±0.24
	CODAR	**67.55±0.38**	**82.71±0.27**	**86.48±0.22**

Table A16: Software and hardware environment. Adapter counts include all tensors with `requires_grad=True` and exclude the dataset-specific classifier, which is reported separately in total counts.

Component	Version / setting
Operating system	Ubuntu 22.04.4 LTS
Python	3.10.13
PyTorch	2.2.2
CUDA toolkit	12.1
cuDNN	8.9.7
torchvision	0.17.2
transformers	4.40.2
open_clip_torch	2.24.0
NumPy	1.26.4
scikit-learn	1.4.2
AMP	FP16 autocast + GradScaler
GPU	RTX 4090, 24 GB

adaptation modules are optimized. Missing masks are shared between CODAR and reproduced baselines under the same seed. All experiments were conducted on a single NVIDIA GeForce RTX 4090 GPU with 24 GB memory. The software environment is summarized in Table A16.

H Extended Related Work

Missing-Modality Learning

Completion, alignment, and distillation. Reconstruction methods infer missing signals or representations through cross-modal generation, translation, or decoding (Zhao et al., 2021; Karimijafarbigloo et al., 2024); alignment and distillation instead transfer complete-input supervision or preserve cross-modal correlations (Li et al., 2024a; Zhuang et al., 2025). These approaches improve robustness without directly asking whether the PEFT update itself should vary with modality availability.

Prompt-space adaptation. Prompt methods modify the context of a frozen backbone. MMP conditions prompts on missing patterns (Lee et al., 2023), while later methods incorporate input semantics or combine static and dynamic prompts (Pipoli et al., 2025; Zhang et al., 2025). They establish the value of availability awareness, but remain in prompt space.

Parameter-space adaptation. MoRA (Zhao et al., 2026) combines modality-common low-rank parameters with modality-specific components. CODAR instead conditions both low-rank factors on per-sample availability while sharing learned dictionaries across states.

PEFT and Conditional Low-Rank Adaptation

LoRA (Hu et al., 2022) learns a low-rank update to a frozen backbone. Parameter-sharing variants reuse matrices, vectors, or compact reconstruction spaces across layers (Kopiczko et al., 2024; Li et al., 2024b; 2025), but their deployed composition rules generally do not depend explicitly on the sample's availability state.

A complementary line of work studies conditional or routed low-rank adaptation, where multiple low-rank components or experts are selected or combined according to task, input, or token context (Gou et al., 2026; Buehler & Buehler, 2024). Related approaches further employ mixture-of-experts-style routing or dynamic fusion to construct context-dependent adapter compositions (Wu et al., 2024; Wang et al., 2024a). These methods establish the value of conditional low-rank composition for handling heterogeneous tasks or inputs, but do not directly study how modality availability affects the row and column subspaces of the adaptation update.

CODAR focuses on this availability-dependent geometry. Rather than only selecting or mixing adaptation branches, CODAR reconstructs both low-rank factors from reusable modality-specific and transferable dictionaries conditioned on the availability state. This allows the realized row and column subspaces to vary across availability states while retaining explicit cross-state parameter sharing.

I Limitations

Our primary evaluation focuses on three image–text classification benchmarks with CLIP ViT-B/16; broader domains, tasks, and backbones remain to be tested.CODAR also assumes known, discrete availability states, whereas real inputs may be partial, corrupted, or silently unreliable. Continuous quality indicators, uncertainty estimation, and an availability encoder would address these cases and scale better than separate embeddings for many modality subsets.

The theoretical analysis establishes a representational separation between CODAR and fixed shared rank-r geometry for a nonempty family of state-dependent target updates, and provides a perturbation-based sufficient condition beyond the exact construction. These results concern target-update approximation rather than downstream task risk. They do not establish a universal approximation rate in dictionary size or routing sparsity, nor guarantee convergence of the non-convex STE optimization.

J Ethical Considerations

Missing-modality robustness can improve accessibility and reliability under sensor failure, user withholding, or transmission loss, but may also enable unwanted inference when users expect absence to preserve privacy. Practitioners should audit correlations between availability and protected attributes and report complete and incomplete performance separately. Because incomplete-input predictions are intrinsically less informed, safety-critical use requires calibrated uncertainty, conservative deferral, and human oversight.