Rosetta: Composable Native Multimodal Pretraining

Xiangyue Liu[1] **Zijian Zhang**[2] **Miles Yang**[2] **Zhao Zhong**[2]
Liefeng Bo[2] **Ping Tan**[1]*

[1]HKUST [2]Tencent Hunyuan

Abstract

Achieving true artificial general intelligence requires foundation models capable of integrating new modalities without forgetting prior knowledge. However, accommodating continuous generative objectives alongside discrete understanding tasks causes severe gradient conflicts. Existing architectures, including standard Mixture-of-Experts (MoE), are highly susceptible to representation overwriting. Even structurally partitioned paradigms like Mixture-of-Transformers (MoT) remain vulnerable to catastrophic forgetting, severely impeding multimodal scalability. In this work, we introduce Rosetta, a composable native multimodal pretraining framework designed for seamless and non-destructive modality expansion. Rosetta adopts a modular paradigm where core foundational knowledge is preserved within global shared experts, while modality-specific capabilities are distributed across plug-and-play experts. To guarantee non-destructive composition, we propose Momentum-Anchored Orthogonal Projection (MAOP). MAOP leverages the optimizer's momentum state as an implicit semantic anchor, selectively neutralizing conflicting gradient components from new modalities while preserving synergistic updates. To strictly isolate the architectural impact, we evaluate Rosetta against standard MoE and MoT baselines under strict active parameter parity. All models are trained from scratch within the Transfusion framework, using discrete next-token prediction for language and continuous visual diffusion. Extensive evaluations demonstrate that, while standard MoE and MoT architectures suffer catastrophic forgetting of previously acquired knowledge, Rosetta robustly preserves established language and visual understanding. Furthermore, it delivers superior image generation and unlocks cross-modal synergy, paving the way for truly composable and unified multimodal foundation models. To facilitate future research and reproducibility, our complete code, checkpoints, and project page are publicly available at https://rosetta-lmm.github.io/.

1 Introduction

The evolution toward general-purpose AI necessitates foundation models capable of natively integrating diverse modalities within a singular architecture [1, 58], spanning from discrete autoregressive language comprehension to continuous diffusion (or flow matching) visual synthesis [72, 66]. However, integrating these disparate training objectives intrinsically triggers severe gradient conflicts. Specifically, the high-variance gradients from generative tasks tend to overwrite the established representations of language modeling, creating a critical optimization bottleneck.

Scaling unified models effectively points toward Sparse Mixture-of-Experts (MoE) [22, 9]. Yet, standard MoE architectures typically deploy modality-agnostic routing mechanisms. When exposed to heterogeneous multimodal signals, this unconstrained routing leads to a catastrophic *routing collapse*: aggressive generative gradients monopolize and irreversibly overwrite the pre-established experts, severely degrading the model's foundational language (as shown in Fig. 1 Left) and visual

*Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

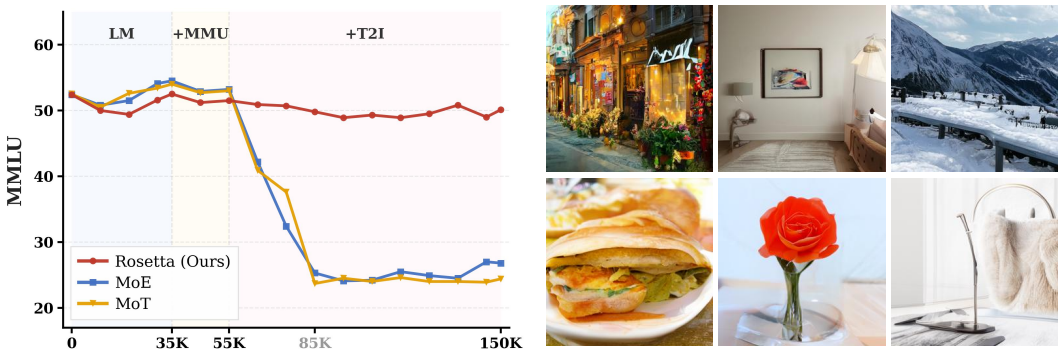


Figure 1: **Escaping the Forgetting-Synergy Dilemma. (Left)** Performance dynamics on MMLU benchmark across composable pretraining stages. While standard MoE and structurally isolated MoT suffer from catastrophic routing collapse and degradation upon the integration of continuous generative objectives (+T2I), our Rosetta architecture acts as a robust semantic anchor, maintaining a highly stable foundation. **(Right)** Qualitative results of Rosetta, demonstrating that the preservation of foundational knowledge seamlessly unlocks superior visual generation capabilities.

understanding capabilities. To mitigate this representation overwriting, structurally partitioned paradigms, such as Mixture-of-Transformers (MoT [31]) and Bagel [10], enforce physical isolation across both the Attention and Feed-Forward Network (FFN) layers within the Transformer [61] backbone. While effective at preserving prior knowledge, such rigid structural segregation can inadvertently restrict dense cross-modal interactions, making it challenging to fully leverage cross-modal synergy.

This exposes a critical *Forgetting-Synergy Dilemma*: how can a foundation model effectively expand its generative capabilities without compromising its foundational knowledge, while simultaneously fostering mutual enhancement across modalities?

In this work, we present Rosetta, a composable native multimodal pretraining framework designed to resolve this dilemma. Much like the historical Rosetta Stone that bridged disparate linguistic scripts, our framework serves as a universal semantic translator, seamlessly harmonizing discrete text, continuous visual perception, and generative latent spaces without mutual interference. Operating on a Lego-like modular paradigm, Rosetta retains a Unified Attention mechanism to preserve high-bandwidth cross-modal contextualization. Concurrently, it confines functional expansion to a composable FFN routing space. By structurally decoupling the FFN into plug-and-play task-specific experts (*e.g.*, dedicated to Text, ViT, or VAE tokens) and a Global Shared Expert, Rosetta isolates task-specific processing while maintaining a universal semantic bridge for cross-modal alignment.

However, the Global Shared Expert inherently remains vulnerable to gradient conflicts. Traditional gradient surgery techniques [69] require N separate backward passes and gradient buffers for pairwise orthogonalization, incurring an unacceptable $\mathcal{O}(N)$ memory overhead under large-scale distributed training frameworks like FSDP [71]. To overcome this limitation, we propose Momentum-Anchored Orthogonal Projection (MAOP), which innovatively repurposes the optimizer's running momentum state as an implicit semantic anchor. Destructive gradient components from incoming generative tasks are orthogonally projected against this anchor, surgically neutralizing cross-modal interference with virtually zero additional memory overhead.

Our main contributions are summarized as follows:

- We propose **Rosetta**, a composable native multimodal pretraining framework. By structurally decoupling global shared experts from plug-and-play task-specific experts, it seamlessly unifies discrete understanding and continuous generation within a single architecture.
- We introduce **Momentum-Anchored Orthogonal Projection (MAOP)**. To the best of our knowledge, we are the first to leverage optimizer momentum as an implicit semantic anchor to dynamically project gradients, eradicating representation overwriting with strictly zero additional memory overhead.
- Extensive experiments demonstrate that Rosetta eliminates catastrophic forgetting during functional expansion (as in Fig. 1). Furthermore, it accelerates convergence on new generative tasks and unlocks true cross-modal synergy against prevalent MoE and MoT baselines.

2 Related Work

Unified Multimodal Foundation Models. Building on the scaling success of foundation models [1, 58, 15], recent efforts have focused on integrating diverse modalities into a unified architecture [2, 21]. Methods such as Chameleon [57] and Janus [63] typically quantize images into discrete tokens to leverage autoregressive prediction [60, 48, 26, 41, 11, 68]. More recently, paradigms like Transfusion [72] and Show-o [66] have demonstrated the superiority of jointly modeling discrete text and continuous visual diffusion [19, 53, 46] within a single dense Transformer. However, forcing disparate generative and understanding objectives into a monolithic parameter space inevitably induces severe modality interference, bottlenecking scalability and cross-modal alignment [45].

Mixture-of-Experts in Multimodal Learning. Sparse Mixture-of-Experts (MoE) efficiently decouples computation from capacity [22, 9, 13, 27, 51], showing great promise in VLMs [32, 29]. Yet, standard modality-agnostic MoE suffers from catastrophic routing collapse when integrating continuous generation, irreversibly overwriting language experts. To prevent this, recent architectures enforce strict physical separation, such as splitting FFNs [42, 54] or entire Transformer blocks (MoT [31], Bagel [10]). While effectively preventing forgetting, this strict segregation completely severs dense cross-modal synergy. Although shared-expert routing [37] attempts to bridge modalities, it relies on heuristic constraints and lacks the mathematical rigor required for open-ended expansion.

Mitigating Forgetting and Gradient Conflicts. Expanding foundation models to new modalities frequently triggers catastrophic forgetting due to severe gradient conflicts between disparate objectives. Traditional Continual Learning techniques attempt to preserve prior knowledge via weight regularization [24, 30, 3] or experience replay [49, 52, 40, 62], but they struggle to scale to billion-parameter distributed pretraining. Alternatively, gradient surgery methods [69, 35, 7, 44] tackle this by orthogonally projecting conflicting task gradients. However, this requires materializing separate computational graphs for each task, incurring an unacceptable memory overhead that paralyzes large-scale frameworks [71, 47, 55, 50]. Breaking this limitation, our MAOP innovatively repurposes the optimizer's intrinsic momentum as an implicit semantic anchor representing foundational knowledge. By projecting incoming gradients against this anchor, MAOP neutralizes destructive interference with virtually zero additional memory overhead, enabling efficient massive-scale multimodal pretraining.

3 Method

In this section, we present Rosetta, a composable native multimodal framework for non-destructive modality expansion. It eliminates catastrophic forgetting and unlocks cross-modal synergy via two core components: (1) **Rosetta Architecture** (Sec. 3.2 and Fig. 2), which uses Unified Attention and plug-and-play FFN experts linked by a Global Shared Expert; and (2) **Conflict-Free Optimization** (Sec. 3.3), introducing Momentum-Anchored Orthogonal Projection (MAOP) to neutralize destructive gradients with zero memory overhead. These innovations are operationalized through our **Composable Pretraining Recipe** (Sec. 4.1 and App. C.1), seamlessly harmonizing visual understanding and generation within a native sparse foundation.

3.1 Preliminaries

Standard Mixture-of-Experts (MoE). For an input token $\mathbf{x} \in \mathbb{R}^d$, a standard sparse MoE layer computes the output via a Top-K gating network $\mathcal{G}$:

$$\mathbf{h}' = \sum_{i=1}^{N} \mathcal{G}_i(\mathbf{x}) \mathcal{E}_i(\mathbf{x}), \tag{1}$$

where $\mathcal{E}_i(\cdot)$ represents the i-th expert out of N total experts. Inherent Flaw: Standard routers $\mathcal{G}(\cdot)$ are entirely modality-agnostic. Jointly optimizing heterogeneous signals (*e.g.*, discrete text and continuous vision) under such unconstrained routing triggers severe capacity collapse, irreversibly overwriting pre-established capabilities.

Gradient Conflicts in Multimodal Learning. Expanding a pretrained foundation model to new modalities essentially optimizes a joint objective: $\mathcal{L}_{total} = \mathcal{L}_{base} + \mathcal{L}_{new}$. Let $\mathbf{g}_{base} = \nabla_\theta \mathcal{L}_{base}$

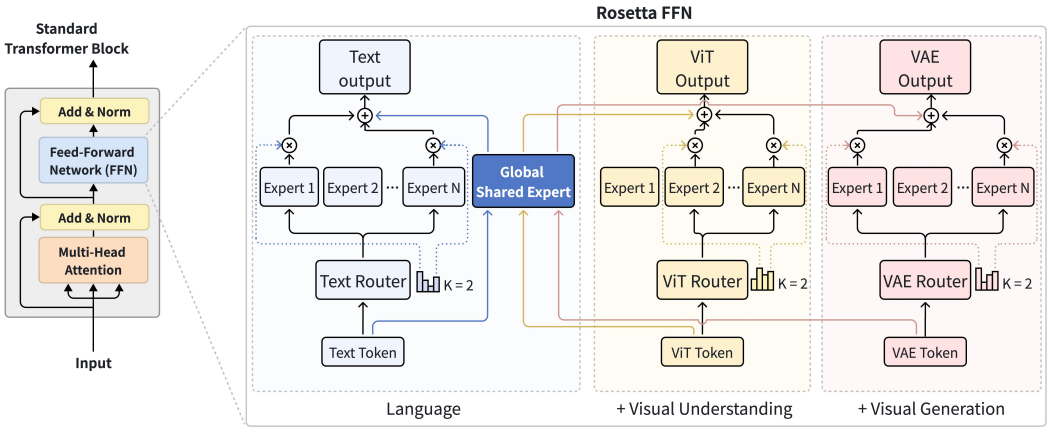


Figure 2: **Architectural Overview of Rosetta.** Our framework ensures non-destructive modality expansion via three mechanisms: (1) Unified Attention (left): Maintains globally shared QKV projections across all modalities to preserve dense cross-modal interactions. (2) Composable FFN (right): Selectively routes tokens to plug-and-play task-specific experts, bridged by a Global Shared Expert. (3) Conflict-Free Optimization: Momentum-Anchored Orthogonal Projection (MAOP) surgically neutralizes destructive gradients with zero memory overhead, converting modality interference into cross-modal synergy.

and $\mathbf{g}_{new} = \nabla_\theta \mathcal{L}_{new}$ represent their respective gradients for shared parameters θ. These objectives frequently diverge, resulting in conflicting gradients where the cosine similarity is negative ($\mathbf{g}_{new}^\top \mathbf{g}_{base} < 0$). Standard optimizers rashly aggregate them ($\mathbf{g}_{total} = \mathbf{g}_{base} + \mathbf{g}_{new}$), pulling shared parameters in opposing directions. This destructive interference is the fundamental optimization root of catastrophic forgetting, necessitating a robust gradient projection mechanism.

3.2 Rosetta Architecture

As illustrated in Fig. 2, Rosetta Transformer block adopts a hybrid paradigm to balance dense cross-modal alignment with interference-free expansion. Specifically, we maintain completely shared QKV projections for dense interactions, while utilizing modality-specific composable sparse FFN layer.

3.2.1 Unified Attention

Unlike approaches that enforce early structural isolation via modality-specific QKV projections (*e.g.*, Bagel [10], MoT [31]), Rosetta retains a strictly unified Multi-Head Attention (MHA). Applying modality-specific QKV forces tokens into disjoint representational subspaces prematurely, disrupting global contextualization. By unifying the entire attention operation, Rosetta ensures all tokens are projected into a cohesive semantic space, fostering dense cross-modal interactions prior to modality-aware FFN routing.

3.2.2 Composable FFN

While the MHA layer captures dense contextual dependencies, previous studies [14, 43] suggest that the Feed-Forward Network (FFN) acts as the primary knowledge repository of the Transformer. Consequently, it is within the FFN that heterogeneous multimodal objectives (*e.g.*, discrete token prediction and continuous diffusion) intrinsically conflict, causing representation overwriting. To fundamentally resolve this bottleneck, Rosetta strictly confines modality-aware expansion to the FFN layer through a highly composable dual-mechanism design.

Modality-Aware Routing with Plug-and-Play Experts. To solve the catastrophic routing collapse observed in standard MoE, Rosetta completely decouples the routing topology. Let $\mathbf{x}^{(t)}$ denote an input token belonging to a specific functional type $t \in \{\text{Text}, \text{ViT}, \text{VAE}\}$. Instead of a single, modality-agnostic router, Rosetta employs modality-specific routers $\mathcal{G}_t(\cdot)$ that restrict token assignment exclusively to a dedicated pool of plug-and-play experts $\{\mathcal{E}_{t,i}\}_{i=1}^{N}$. This explicit decoupling guarantees that high-frequency, noisy generative gradients cannot monopolize the capacity of language or visual understanding experts. Furthermore, this design is natively *composable*: integrating a

novel functionality only requires mounting a new router and its corresponding expert group, ensuring extensibility without compromising previously acquired foundational capabilities.

Global Shared Expert as a Cross-Modal Semantic Bridge. Complementing the modality-aware routing against destructive interference, the Global Shared Expert ($\mathcal{E}_{shared}$) serves as the central engine for cross-modal interactions. Rosetta mandates that every token, regardless of its functional type t, is deterministically processed by this shared expert. The final FFN output for token $\mathbf{x}^{(t)}$ is formulated as:

$$\mathbf{h}' = \mathcal{E}_{shared}(\mathbf{x}^{(t)}) + \sum_{i \in \text{TopK}(\mathcal{G}_t(\mathbf{x}^{(t)}))} g_{t,i}\, \mathcal{E}_{t,i}(\mathbf{x}^{(t)}), \tag{2}$$

where $g_{t,i}$ represents the routing probability. By absorbing gradients from all diverse tasks, the shared expert learns a universal, modality-agnostic semantic representation. It functions as a global semantic bridge, allowing fine-grained visual generative signals to implicitly enrich language representations, thereby unlocking mutually reinforcing cross-modal synergy.

3.3 Conflict-Free Optimization via MAOP

While the Rosetta architecture physically isolates modality-specific capabilities, the Global Shared Expert inevitably absorb gradients from all active tasks. When introducing continuous visual generation tasks alongside discrete understanding, the severe heterogeneity of the loss landscapes frequently results in gradient conflicts (*i.e.*, $\mathbf{g}_{new}^{\top}\mathbf{g}_{base} < 0$). Traditional gradient surgery methods (*e.g.*, PCGrad [69]) require instantiating and storing separate computational graphs for each task, introducing a prohibitive $\mathcal{O}(N)$ memory overhead that is fundamentally incompatible with large-scale distributed training frameworks like FSDP [71].

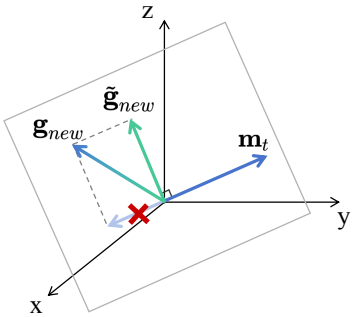


Figure 3: **Illustration of MAOP.**

To achieve non-destructive functional expansion, we introduce Momentum-Anchored Orthogonal Projection (MAOP). MAOP innovatively repurposes the optimizer's running momentum $\mathbf{m}_t$ as an *implicit semantic anchor* tracking foundational knowledge. If a conflict is detected ($\mathbf{g}_{new}^{\top}\mathbf{m}_t < 0$), MAOP surgically neutralizes the interfering component by projecting $\mathbf{g}_{new}$ onto the normal plane of $\mathbf{m}_t$:

$$\tilde{\mathbf{g}}_{new} = \mathbf{g}_{new} - \frac{\mathbf{g}_{new}^{\top}\mathbf{m}_t}{\|\mathbf{m}_t\|^2}\mathbf{m}_t, \quad \text{if} \quad \mathbf{g}_{new}^{\top}\mathbf{m}_t < 0. \tag{3}$$

For numerical stability, this projection is bypassed if $\|\mathbf{m}_t\|^2 < 10^{-12}$. Crucially, since $\mathbf{m}_t$ is inherently maintained by the optimizer, MAOP eradicates representation overwriting with virtually zero additional memory overhead, safeguarding synergistic cross-modal updates (details in App. A).

4 Experiments

In this section, we comprehensively evaluate Rosetta to answer four core research questions: (1) Can Rosetta natively integrate generative modalities without catastrophic forgetting? (2) Does it unlock true cross-modal synergy compared to physically isolated paradigms? (3) What are the underlying reasons that trigger catastrophic forgetting in standard architectures during expansion? and (4) How do our proposed architectural components (the Global Shared Expert and MAOP) guarantee non-destructive modality expansion?

4.1 Experimental Setup

Baseline Architectures & Active Parity. To isolate architectural advancements, all models are upcycled from Qwen3-0.6B Base [67] and trained from scratch. We guarantee strict *active parameter parity* ($\sim$0.97B) by mathematically constraining every framework to activate exactly 3 experts (2 routed + 1 shared) per token: (1) **Standard MoE** (denoted simply as MoE): A modality-agnostic baseline routing to 2 out of 12 experts plus 1 shared expert. (2) **MoT**: Representing structural isolation, we instantiate this following Bagel [10]. It employs modality-specific QKV projections

Table 1: **Comprehensive Performance Evaluations.** All methods are evaluated in their raw foundation state after the full multimodal pretraining phase (LM+MMU+T2I) under identical training constraints, strictly without any downstream instruction tuning. Rosetta fundamentally overcomes the catastrophic forgetting existing in MoE and MoT, achieving the best across all three capability domains (Language, Visual Understanding, and Visual Generation). The gray row Rosetta (Pre-T2I) represents the understanding performance of Rosetta prior integrating T2I. (**Bold**: Best)

Method	Total Params	Active Params	Training Iterations	Visual Generation						
				T2I-Comp↑	Color↑	Shape↑	Texture↑	FID↓	CLIPScore↑	HPSv2↑
MoE	3.77B	0.97B	400K	40.2	47.7	27.5	45.5	17.80	0.287	0.204
MoT (Bagel)	4.48B	0.97B	400K	43.5	50.7	29.8	50.0	15.58	0.288	0.211
Rosetta	3.77B	0.97B	400K	**45.5**	**52.9**	**31.7**	**51.9**	**14.05**	**0.290**	**0.219**

Method	Language				Visual Understanding					
	MMLU↑	BBH↑	ARC-c↑	MBPP↑	MMMU↑	MMB-EN↑	MMB-CN↑	POPE↑	AI2D↑	RealWorldQA↑
Rosetta (Pre-T2I)	51.6	48.8	67.3	36.2	34.1	51.3	45.2	78.1	54.2	47.4
MoE	26.3	0	22.8	0	26.8	42.1	31.6	77.0	41.6	39.5
MoT (Bagel)	27.1	0	26.5	0	27.1	46.2	40.7	74.0	47.6	45.1
Rosetta	**49.2**	**46.8**	**62.9**	**42.4**	**34.6**	**52.5**	**45.7**	**80.1**	**55.8**	**48.2**

and dual streams (understanding routes to 2 of 7 experts; generation to 2 of 6), each with 1 isolated shared expert. (3) **Rosetta (Ours)**: Maintains unified attention and strictly routes tokens to dedicated task-aware expert pools (3 Text, 3 ViT, 6 VAE), all bridged by a single Global Shared Expert. Notably, MoT's structural redundancy inflates its total parameter budget to 4.48B. In contrast, Rosetta preserves the exact 3.77B total parameter of standard MoE while unlocking superior cross-modal synergy (details in App. C.3).

Composable Pretraining Recipe. For strict fairness, all models process identical pretraining sequences via fixed seeds, without any subsequent scaling or tuning stages. The vision encoder Qwen3-VL ViT [5] and FLUX.2 VAE [6] remain permanently frozen in all settings. Pretraining spans three composable stages: (1) *Language Foundation (LM)*: Models are optimized on ∼300B text tokens for 35K steps. (2) *Visual Understanding (+MMU)*: A 3K-step LLaVA-style projector warmup [36] precedes 20K joint training steps (∼4M MMU samples, MMU:LM = 0.8:0.2). (3) *Visual Generation (+T2I)*: Joint optimization of all modalities using a data sampling ratio of T2I:MMU:LM = 0.6:0.25:0.15. Comprehensive recipes are in App. C.1 and hyperparameters in App. C.2.

Evaluation Metrics. To comprehensively evaluate models, we conduct experiments across three main domains. For *Language*, we measure general knowledge and reasoning via MMLU [16] and BBH [56], language understanding via ARC-challenge [8], and coding capabilities via MBPP [4]. For *Visual Understanding*, we utilize a diverse set of tests: MMMU [70] (expert-level reasoning), MMBench (English and Chinese) [39] (comprehensive perception), POPE [28] (hallucination robustness), AI2D [23] (diagram understanding), and RealWorldQA [65] (real-world comprehension). For *Visual Generation*, synthesis fidelity and semantic alignment are evaluated using FID [18], CLIPScore [17], and HPSv2 [64] on COCO-30K [33]. Furthermore, we employ T2I-CompBench [20] to explicitly quantify compositional prompt adherence across specific attributes (*i.e.*, Color, Shape, and Texture).

4.2 Main Results: Escaping the Forgetting-Synergy Dilemma

Language Ability. As evidenced by the MMLU dynamics in Fig. 1, integrating image generation triggers a severe routing collapse in baseline architectures. The underlying cause is revealed in Fig. 5 (bottom right): the aggressive injection of T2I gradients causes the LM Text Loss of MoE and MoT to catastrophically diverge. This optimization instability translates to massive performance degradation across all language metrics (as in Tab. 1). In stark contrast, Rosetta effectively absorbs these generative shocks, maintaining a strictly suppressed and stable loss trajectory. Benefiting from this architectural immunity, Rosetta preserves foundational language capabilities with minimal degradation compared to its pre-generative checkpoint Pre-T2I (trained exclusively on language and visual understanding). This stark divergence is most vividly illustrated in highly sensitive reasoning (BBH) and coding (MBPP) benchmarks. While generative interference completely destroys these capabilities in MoE and MoT—collapsing both their BBH and MBPP scores to 0, Rosetta robustly safeguards complex reasoning (BBH retaining 46.8 vs. 48.8) and even enhances coding proficiency (MBPP increasing from 36.2 to 42.4), demonstrating exceptional cross-modal resilience.

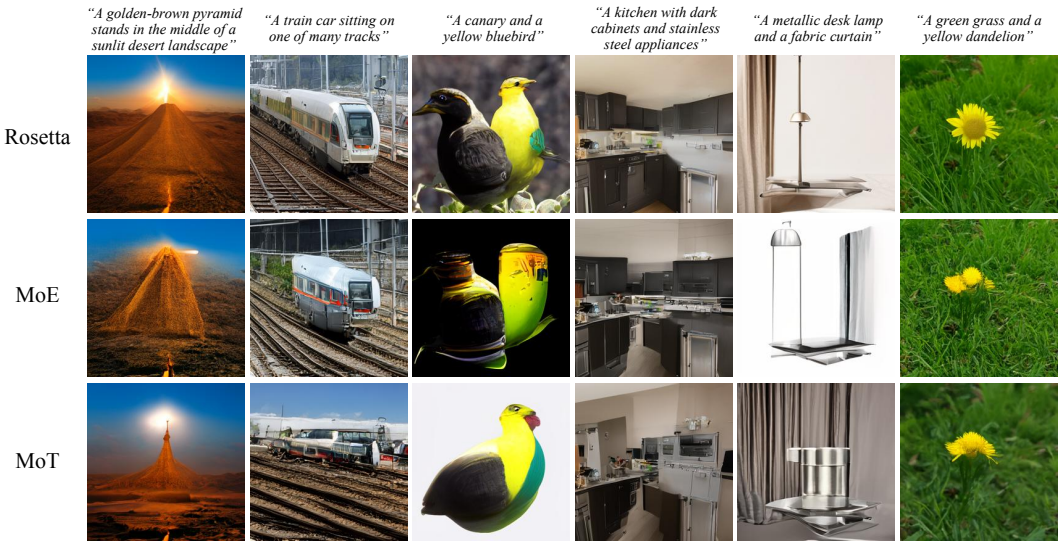


Figure 4: **Qualitative Comparisons.** Standard MoE suffers semantic drift (*e.g.*, bird to bottle) and MoT exhibits structural distortions (*e.g.*, broken lamp). In contrast, Rosetta leverages cross-modal synergy to synthesize high-fidelity images with precise spatial geometry and prompt adherence.

Visual Understanding. Building upon its stable language capabilities, Rosetta further improves visual understanding through cross-modal synergy. As detailed in Tab. 1, the integration of generative tasks causes a universal performance drop across all visual understanding metrics for both MoE and MoT. Although attempts like freezing the understanding branch in MoT (*e.g.*, Bagel) avoids interference, but strictly limits visual understanding performance to pre-trained levels and prevents further enhancement. In contrast, Rosetta achieves consistent improvements over its Pre-T2I baseline across all indicators. This superiority directly stems from the underlying optimization dynamics (Fig. 5, bottom center): while the MMU Text Loss of baselines severely diverges under generative interference, Rosetta maintains a highly stable trajectory. Consequently, as tracked by the MMBench dynamics (Fig. 5, top left), although all methods experience an initial performance drop, MoE and MoT suffer irreversible degradation. Rosetta, however, rapidly recovers and maintains a steady upward trend. This confirms that Rosetta successfully unlocks true cross-modal synergistic evolution.

Visual Generation. Crucially, Rosetta's preservation of foundational knowledge does not compromise generative plasticity. As detailed in Tab. 1, Rosetta consistently achieves the best performance across all visual generation metrics, delivering superior synthesis fidelity on COCO-30K and dominating compositional benchmarks like T2I-CompBench. This empirical success is directly supported by its optimization trajectory (Fig. 5, bottom left). Rosetta achieves the lowest T2I Image Loss in training, this optimization efficacy stems from robust cross-modal synergy, effectively resolving parameter competition and enabling the model to achieve superior generative performance. Qualitative comparisons (Fig. 4) further validate that unlike MoE and MoT which suffer semantic drift or structural distortions, Rosetta synthesizes high-fidelity images with precise prompt adherence, successfully dismantling the traditional stability-plasticity dilemma.

Unlocking Cross-Modal Synergy. In summary, Rosetta effectively overcomes the forgetting-synergy dilemma. Upon integrating continuous generative tasks, MoE and MoT suffer catastrophic degradation across all language and visual understanding metrics. Conversely, Rosetta robustly preserves foundational language priors with even improving coding proficiency, and universally enhances visual understanding performance. Coupled with significantly faster convergence in visual generation, Rosetta demonstrates that introducing new modalities can serve as a constructive regularizer rather than a disruptive force. This mutually beneficial synergy establishes a highly scalable and unified blueprint for extensible multimodal foundation models.

4.3 In-depth Analysis and Ablation Studies

Deep Analysis: Unmasking the Collapse. To investigate the severe MMLU performance drop observed in Fig. 1, we visualize the routing distribution of Text tokens during inference. Fig. 6

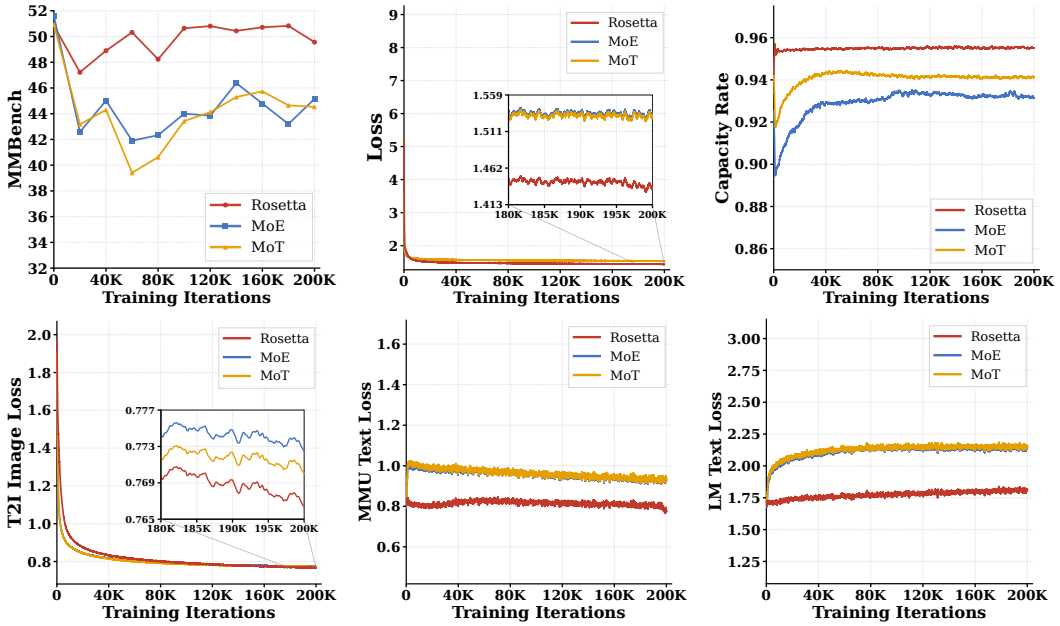


Figure 5: **Comprehensive Training Dynamics.** Evaluated over a 200K-step generative expansion. **(1) Overall Dynamics (Top Row):** Rosetta averts the irreversible MMBench degradation in MoE and MoT baselines, maintaining a synergistic upward trajectory (Left). It also achieves a deeper optimization bound (Center) and near-optimal capacity rate (*i.e.*, ratio of successfully routed, non-dropped tokens; ∼0.95, Right). **(2) Task-Specific Losses (Bottom Row):** Rosetta accelerates T2I convergence (Left) and neutralizes cross-modal gradient interference, guaranteeing strictly lower and stable trajectories for both visual (Center) and language understanding (Right).

compares the expert activation probabilities at two critical checkpoints from Fig. 1: before generative training (top row, iteration 55K) and after 30K steps of adding T2I training (bottom row, iteration 85K). For MoE (Left), the text routing distribution shifts significantly, indicating that continuous generative gradients severely interfere with the pre-trained modality-agnostic experts. Notably, MoT (Middle) also exhibits clear routing changes despite having physically separated experts. This reveals that structural isolation alone is insufficient, generative signals can still corrupt shared parameters during joint optimization. In contrast, Rosetta (Right) maintains nearly the same routing distribution before and after the generative expansion. By combining modality-aware routing with the MAOP mechanism, Rosetta effectively eliminates cross-modal interference and preserves pre-established capabilities.

Efficacy of the Global Shared Expert. Removing the Global Shared Expert reduces our FFN to a strictly isolated routing paradigm. Notably, unlike MoT, this variant retains unified QKV projections, allowing us to precisely isolate the impact of the FFN's shared representation space. As detailed in Tab. 2, while this strict isolation within the FFN successfully averts catastrophic forgetting, it incurs a severe penalty in compositional generation tasks (FID) and advanced multimodal reasoning (MMBench). This empirically confirms that rigid isolation inherently impedes cross-modal synergy, establishing our Global Shared Expert as a pivotal semantic bridge for harmonizing diverse modalities.

Table 2: **Ablation Results.** Impact of removing core components on multimodal synergy.

Variant	Iters	MMLU↑	MMBench↑	FID↓
w/o Shared Expert	100k	49.84	49.19	36.13
w/o MAOP	100k	45.48	46.07	34.29
Rosetta (Full)	100k	**50.35**	**50.64**	**34.21**

Necessity of MAOP. To verify our gradient projection mechanism, we train a variant without MAOP. As illustrated in Tab. 2, this variant experiences a clear degradation in language understanding metrics upon the introduction of visual generation. This indicates that while structural decoupling prevents routing collapse, the Shared Expert still suffers from representation overwriting due to gradient conflicts. MAOP surgically neutralizes this interference.

System-Level Efficiency and Zero Overhead. Finally, we highlight the architectural elegance of MAOP compared to conventional gradient surgery techniques. Mathematically, PCGrad requires N

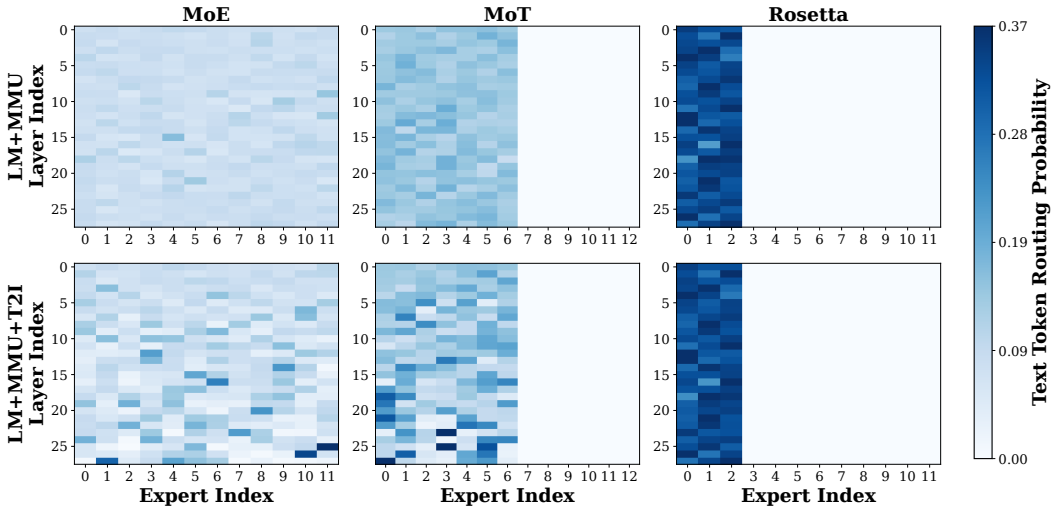


Figure 6: **Routing Distribution Heatmaps During Generative Expansion.** We visualize the routing probabilities of Text tokens across experts during MMLU inference. **Top Row:** Checkpoints under the LM+MMU configuration (iteration 55K in Fig. 1). **Bottom Row:** Checkpoints upon integrating 30K steps of T2I training (iteration 85K in Fig. 1). Both MoE and MoT exhibit significant distribution shifts, indicating severe cross-modal interference. In contrast, Rosetta maintains nearly identical routing distribution, successfully preserving pre-established language capabilities.

separate backward passes (one per task, *i.e.*, $N{=}3$ in our LM + MMU + T2I setting), reducing training throughput by approximately $3\times$, while materializing N distinct gradient tensors introduces the same $\mathcal{O}(N)$ memory overhead described above. Conversely, MAOP repurposes the optimizer's intrinsically pre-allocated momentum state, requiring no additional backward passes or gradient buffers. Empirical hardware profiling confirms that enabling MAOP introduces virtually zero additional peak memory (maintaining exactly 58.8 GB per GPU) and zero computational overhead (retaining an identical throughput of 8,602 tokens/s/GPU). This establishes MAOP as an exceptionally viable, zero-cost optimization solution for large-scale foundation models.

Expert Scalability. We furture analyze the scalability of Rosetta's plug-and-play experts. By varying the number of generation experts ($N_{VAE} \in \{2, 4, 6, 8, 10\}$) while maintaining the active parameter count ($\sim$0.97B) constant, we evaluate the generative fidelity at a 100K-step checkpoint to observe structural scalability. As illustrated in Fig. 7, expanding the expert pool monotonically improves synthesis fidelity (evidenced by a continuous decrease in FID) without degrading the established language understanding capabilities (MMLU remains rigorously stable). This demonstrates robust expert scaling behavior, confirming Rosetta's potential for highly efficient and non-destructive modality expansion.

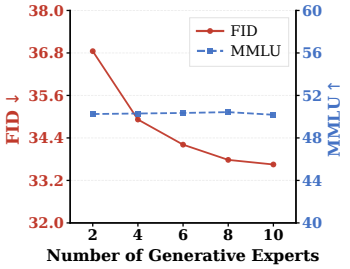


Figure 7: **Expert Scalability of Rosetta.**

5 Conclusion

In this work, we present Rosetta to resolve the Forgetting-Synergy Dilemma between discrete understanding and continuous generation. Structurally, it delegates modality-specific processing to plug-and-play experts while employing a Global Shared Expert as a semantic bridge. Mathematically, our Momentum-Anchored Orthogonal Projection (MAOP) neutralizes destructive gradient conflicts with virtually zero additional memory overhead. Ultimately, Rosetta goes beyond a mere architectural improvement; it introduces a scalable philosophy for next-generation foundation models. As the community accelerates toward Artificial General Intelligence, seamlessly integrating new modalities including audio, video, 3D perception, and embodied control becomes paramount. By mathematically and structurally guaranteeing this non-destructive expansion, Rosetta provides a blueprint for such a future: a truly unified and ever-expanding native multimodal foundation model.

Acknowledgements and Disclosures

The authors declare no competing interests beyond their stated institutional affiliations, and this work received no external funding.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022.

[3] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the European conference on computer vision (ECCV)*, pages 139–154, 2018.

[4] Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

[5] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

[6] Black Forest Labs. FLUX.2: Frontier visual intelligence. `https://bfl.ai/blog/flux-2`, 2025.

[7] Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *International conference on machine learning*, pages 794–803. PMLR, 2018.

[8] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

[9] Damai Dai, Chengqi Deng, Chenggang Zhao, RX Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Yu Wu, et al. Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1280–1297, 2024.

[10] Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, et al. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025.

[11] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021.

[12] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

[13] 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.

[14] Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495, 2021.

[15] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[16] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[17] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pages 7514–7528, 2021.

[18] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[19] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[20] Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. *Advances in Neural Information Processing Systems*, 36:78723–78747, 2023.

[21] Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Barun Patra, et al. Language is not all you need: Aligning perception with language models. *Advances in Neural Information Processing Systems*, 36:72096–72109, 2023.

[22] Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

[23] Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *European conference on computer vision*, pages 235–251. Springer, 2016.

[24] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

[25] Aran Komatsuzaki, Joan Puigcerver, James Lee-Thorp, Carlos Riquelme Ruiz, Basil Mustafa, Joshua Ainslie, Yi Tay, Mostafa Dehghani, and Neil Houlsby. Sparse upcycling: Training mixture-of-experts from dense checkpoints. *arXiv preprint arXiv:2212.05055*, 2022.

[26] Doyup Lee, Chiheon Kim, Saehoon Kim, Minsu Cho, and Wook-Shin Han. Autoregressive image generation using residual quantization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11523–11532, 2022.

[27] Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. Gshard: Scaling giant models with conditional computation and automatic sharding. *arXiv preprint arXiv:2006.16668*, 2020.

[28] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 292–305, 2023.

[29] Yunxin Li, Shenyuan Jiang, Baotian Hu, Longyue Wang, Wanqi Zhong, Wenhan Luo, Lin Ma, and Min Zhang. Uni-moe: Scaling unified multimodal llms with mixture of experts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[30] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017.

[31] Weixin Liang, LILI YU, Liang Luo, Srini Iyer, Ning Dong, Chunting Zhou, Gargi Ghosh, Mike Lewis, Wen tau Yih, Luke Zettlemoyer, and Xi Victoria Lin. Mixture-of-transformers: A sparse and scalable architecture for multi-modal foundation models. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL `https://openreview.net/forum?id=Nu6N69i8SB`.

[32] Bin Lin, Zhenyu Tang, Yang Ye, Jinfa Huang, Junwu Zhang, Yatian Pang, Peng Jin, Munan Ning, Jiebo Luo, and Li Yuan. Moe-llava: Mixture of experts for large vision-language models. *IEEE Transactions on Multimedia*, 2026.

[33] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[34] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

[35] Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. *Advances in neural information processing systems*, 34:18878–18890, 2021.

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

[37] Xiangyue Liu, Zijian Zhang, Miles Yang, Zhao Zhong, Liefeng Bo, and Ping Tan. Symbiotic-moe: Unlocking the synergy between generation and understanding. *arXiv preprint arXiv:2604.07753*, 2026.

[38] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

[39] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pages 216–233. Springer, 2024.

[40] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in neural information processing systems*, 30, 2017.

[41] Jiasen Lu, Christopher Clark, Sangho Lee, Zichen Zhang, Savya Khosla, Ryan Marten, Derek Hoiem, and Aniruddha Kembhavi. Unified-io 2: Scaling autoregressive multimodal models with vision language audio and action. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26439–26455, 2024.

[42] Brandon McKinzie, Zhe Gan, Jean-Philippe Fauconnier, Sam Dodge, Bowen Zhang, Philipp Dufter, Dhruti Shah, Xianzhi Du, Futang Peng, Anton Belyi, et al. Mm1: methods, analysis and insights from multimodal llm pre-training. In *European Conference on Computer Vision*, pages 304–323. Springer, 2024.

[43] Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372, 2022.

[44] Aviv Navon, Aviv Shamsian, Idan Achituve, Haggai Maron, Kenji Kawaguchi, Gal Chechik, and Ethan Fetaya. Multi-task learning as a bargaining game. *arXiv preprint arXiv:2202.01017*, 2022.

[45] Yuwei Niu, Munan Ning, Mengren Zheng, Weiyang Jin, Bin Lin, Peng Jin, Jiaqi Liao, Chaoran Feng, Kunpeng Ning, Bin Zhu, et al. Wise: A world knowledge-informed semantic evaluation for text-to-image generation. *arXiv preprint arXiv:2503.07265*, 2025.

[46] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

[47] Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. Zero: Memory optimizations toward training trillion parameter models. In *SC20: international conference for high performance computing, networking, storage and analysis*, pages 1–16. IEEE, 2020.

[48] Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

[49] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017.

[50] Jie Ren, Samyam Rajbhandari, Reza Yazdani Aminabadi, Olatunji Ruwase, Shuangyan Yang, Minjia Zhang, Dong Li, and Yuxiong He. {Zero-offload}: Democratizing {billion-scale} model training. In *2021 USENIX Annual Technical Conference (USENIX ATC 21)*, pages 551–564, 2021.

[51] Carlos Riquelme, Joan Puigcerver, Basil Mustafa, Maxim Neumann, Rodolphe Jenatton, André Susano Pinto, Daniel Keysers, and Neil Houlsby. Scaling vision with sparse mixture of experts. *Advances in Neural Information Processing Systems*, 34:8583–8595, 2021.

[52] David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. Experience replay for continual learning. *Advances in neural information processing systems*, 32, 2019.

[53] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[54] Sheng Shen, Zhewei Yao, Chunyuan Li, Trevor Darrell, Kurt Keutzer, and Yuxiong He. Scaling vision-language models with sparse mixture of experts. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11329–11344, 2023.

[55] Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*, 2019.

[56] Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, et al. Challenging big-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13003–13051, 2023.

[57] Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024.

[58] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[59] Shengbang Tong, David Fan, John Nguyen, Ellis Brown, Gaoyue Zhou, Shengyi Qian, Boyang Zheng, Théophane Vallaeys, Junlin Han, Rob Fergus, et al. Beyond language modeling: An exploration of multimodal pretraining. *arXiv preprint arXiv:2603.03276*, 2026.

[60] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[61] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[62] Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 139–149, 2022.

[63] Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, Chong Ruan, et al. Janus: Decoupling visual encoding for unified multimodal understanding and generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 12966–12977, 2025.

[64] Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023.

[65] X.AI Corp. Grok-1.5 Vision Preview: Connecting the digital and physical worlds with our first multimodal model. `https://x.ai/news/grok-1.5v`, 2024.

[66] Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. *arXiv preprint arXiv:2408.12528*, 2024.

[67] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[68] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2(3):5, 2022.

[69] Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in neural information processing systems*, 33:5824–5836, 2020.

[70] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9556–9567, 2024.

[71] Yanli Zhao, Andrew Gu, Rohan Varma, Liang Luo, Chien-Chin Huang, Min Xu, Less Wright, Hamid Shojanazeri, Myle Ott, Sam Shleifer, et al. Pytorch fsdp: experiences on scaling fully sharded data parallel. *arXiv preprint arXiv:2304.11277*, 2023.

[72] Chunting Zhou, Lili Yu, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. Transfusion: Predict the next token and diffuse images with one multi-modal model. *arXiv preprint arXiv:2408.11039*, 2024.

Supplementary Materials for Rosetta: Composable Native Multimodal Pretraining

This supplementary document provides comprehensive theoretical foundations, system-level implementation specifics, extended empirical configurations, and additional visualizations to guarantee the absolute reproducibility of our framework. The contents are systematically organized as follows:

- **Appendix A: Algorithmic Foundations and Mathematical Proofs.** Provides the formal formulation of Momentum-Anchored Orthogonal Projection (MAOP), proves its strict mathematical correctness and zero-overhead scalability under FSDP, and details the magnitude-preserving sparse upcycling strategy.
- **Appendix B: System-Level Implementation Details.** Elaborates on the two-stage synergistic protection mechanism and the Differential Learning Rate (DiffLR) strategy, providing explicit pseudo-code for its native optimization overriding within distributed frameworks.
- **Appendix C: Comprehensive Pretraining Curriculum and Configurations.** Presents the detailed modality expansion pipeline, comprehensive hyperparameter settings, data sampling ratios, and the exact architectural parameter breakdown that rigorously ensures active computational parity.
- **Appendix D: Additional Qualitative Results.** More image generation results that corroborate Rosetta's superior cross-modal alignment.

A Algorithmic Foundations and Mathematical Proofs

A.1 Momentum-Anchored Orthogonal Projection (MAOP)

Let $\mathbf{g} \in \mathbb{R}^D$ denote the current mixed gradient (aggregated from language and visual generative tokens) for the Shared Expert parameters. Let $\mathbf{m} \in \mathbb{R}^D$ denote the optimizer's first-moment estimate (*e.g.*, `exp_avg` in AdamW), representing the exponentially moving average of historical gradients. This $\mathbf{m}$ serves as an implicit semantic anchor for the established learning trajectory.

At each backward pass, MAOP evaluates the cosine similarity between the current gradient and the historical momentum. A destructive conflict is defined as their inner product being negative ($\mathbf{g}^\top \mathbf{m} < 0$). To neutralize this interference, MAOP orthogonally projects $\mathbf{g}$ onto the normal plane of $\mathbf{m}$:

$$\mathbf{g}_{\text{orth}} = \begin{cases} \mathbf{g} - \frac{\mathbf{g}^\top \mathbf{m}}{\|\mathbf{m}\|^2}\mathbf{m}, & \text{if } \mathbf{g}^\top \mathbf{m} < 0 \\ \mathbf{g}, & \text{otherwise.} \end{cases} \tag{4}$$

For numerical stability, the projection is bypassed if $\|\mathbf{m}\|^2 < 10^{-12}$.

Proof of Non-Interference: To mathematically verify that the projected gradient $\mathbf{g}_{\text{orth}}$ exerts exactly zero interference on the momentum trajectory $\mathbf{m}$, we compute their inner product in the conflict scenario ($\mathbf{g}^\top \mathbf{m} < 0$):

$$\mathbf{g}_{\text{orth}}^\top \mathbf{m} = \left(\mathbf{g} - \frac{\mathbf{g}^\top \mathbf{m}}{\|\mathbf{m}\|^2}\mathbf{m}\right)^\top \mathbf{m} = \mathbf{g}^\top \mathbf{m} - \frac{\mathbf{g}^\top \mathbf{m}}{\|\mathbf{m}\|^2}(\mathbf{m}^\top \mathbf{m}) = 0. \tag{5}$$

This confirms that MAOP removes the antagonistic component while preserving synergistic updates ($\mathbf{g}_{\text{orth}}^\top \mathbf{m} \geq 0$), providing empirical regularization against representation overwriting.

A.2 Scalable Distributed Implementation in FSDP

A critical engineering challenge in modern massive-scale pretraining is that model parameters and their corresponding gradients are heavily sharded across multiple GPUs. Computing the global inner product $\mathbf{g}^\top \mathbf{m}$ directly on local shards would mathematically violate the projection geometry.

To maintain rigorous mathematical equivalence with strictly zero memory overhead, MAOP is implemented using a synchronized distributed reduction. Specifically, we first compute the local

inner product on each rank's shard: $S_{\text{local}} = \mathbf{g}^{(rank)} \cdot \mathbf{m}^{(rank)}$ and $N_{\text{local}} = \|\mathbf{m}^{(rank)}\|^2$. We then execute a highly efficient `All-Reduce (SUM)` operation across the distributed communication group to obtain the exact global scalars $S_{\text{global}} = \mathbf{g}^\top \mathbf{m}$ and $N_{\text{global}} = \|\mathbf{m}\|^2$.

These globally synchronized coefficients are subsequently utilized to orthogonally project the local gradient shards independently. This elegant implementation ensures that the projected distributed gradients are mathematically identical to projecting the full, unsharded parameters, demonstrating MAOP's pristine compatibility with extreme-scale FSDP infrastructures.

A.3 Pseudocode for Distributed MAOP

To ensure communication overhead remains negligible under FSDP, we adopt a vectorized batching strategy. Specifically, each distributed worker computes the local dot products ($\mathbf{g}_p \cdot \mathbf{m}_p$ and $\|\mathbf{m}_p\|^2$) for all shared parameter shards and stacks these scalar pairs into a single compact tensor of shape $[N_{\text{shared_params}}, 2]$, requiring **exactly one** vectorized `all_reduce` communication per optimization step. As demonstrated in Listing 1, MAOP applies this projection directly to the gradient tensors to ensure nonnegative alignment with historical momentum *before* the AdamW preconditioning step, requiring no task-specific gradient buffers and incurring negligible overhead.

Listing 1: PyTorch implementation of Distributed MAOP (`maop.py`).

```
local_dots = []
# 1. Compute local dot products for all shared parameter shards (cast to FP32 for numerical stability)
for p in shared_params:
    g_local = p.grad.to_local().float().flatten()
    m_local = optim.state[p]["exp_avg"].to_local().float().flatten()
    local_dots.append( torch.stack([torch.dot(g_local, m_local), torch.dot(m_local, m_local)]) )

# 2. Stack into a tensor of shape [N_shared_params, 2]
dots_tensor = torch.stack(local_dots)

# 3. EXACTLY ONE vectorized all_reduce communication per optimization step
dist.all_reduce(dots_tensor, op=dist.ReduceOp.SUM, group=fsdp_pg)

# 4. Apply the projection per parameter using globally synced statistics
for i, p in enumerate(shared_params):
    gm_dot = dots_tensor[i, 0].item()
    m_norm_sq = dots_tensor[i, 1].item()

    if gm_dot < 0.0 and m_norm_sq >= 1e-12:
        g_fp32 = p.grad.to_local().float()
        m_fp32 = optim.state[p]["exp_avg"].to_local().float()

        g_orth = g_fp32 - (gm_dot / m_norm_sq) * m_fp32
        p.grad.to_local().copy_(g_orth.to(p.grad.dtype))
```

A.4 Magnitude-Preserving Sparse Upcycling

As introduced in Sec. C.1.1, we first upcycle a dense LLM into a sparse MoE architecture. A naive weight duplication, however, would precipitate optimization instability. If all experts were exact identical copies of the dense FFN (denoted as $\hat{\mathcal{E}}$), the standard routing mechanism would double the initial activation magnitude, since the Top-2 routing weights sum to 1:

$$\hat{\mathcal{E}}_{shared}(\mathbf{x}) + \sum_{i \in \text{Top-2}} g_{text,i}\, \hat{\mathcal{E}}_{text,i}(\mathbf{x}) \approx \text{FFN}_{dense}(\mathbf{x}) + \text{FFN}_{dense}(\mathbf{x}) = 2\text{FFN}_{dense}(\mathbf{x}). \quad (6)$$

To strictly preserve the pre-trained dense capabilities and ensure a seamless zero-shot transition, we perform a surgical intervention *exclusively during weight initialization*. We apply a deterministic scaling factor of 0.5 to the `down_proj` weight matrix of all upcycled experts (*i.e.*, $\mathbf{W}_{down} \leftarrow 0.5\mathbf{W}_{down}$). Since `down_proj` constitutes the final linear projection, this initializes our actual experts to $\mathcal{E}(\mathbf{x}) \approx 0.5\text{FFN}_{dense}(\mathbf{x})$, elegantly avoiding the cubic decay effect that would arise from scaling all internal expert matrices. Consequently, the native forward computation inherently recovers the exact original output magnitude without requiring any runtime architectural modifications:

$$\mathbf{h}' = \mathcal{E}_{shared}(\mathbf{x}) + \sum_{i \in \text{Top-2}} g_{text,i}\, \mathcal{E}_{text,i}(\mathbf{x}) \approx 0.5\text{FFN}_{dense}(\mathbf{x}) + 0.5\text{FFN}_{dense}(\mathbf{x}) = \text{FFN}_{dense}(\mathbf{x}). \quad (7)$$

As depicted at iteration 0 in Fig. 1, all evaluated sparse frameworks (Standard MoE, MoT, and Rosetta) inherit an exceptional zero-shot MMLU [16] score of 52.40 without any training. This closely mirrors the officially reported 52.81 MMLU score of the dense Qwen3-0.6B Base model [67]. Such near-lossless capability inheritance confirms that our magnitude-preserving upcycling flawlessly transfers pre-established knowledge into the sparse topology. This rigorous weight scaling mathematically and empirically guarantees early-stage convergence stability, allowing the subsequent 300B-token text pretraining to proceed optimally.

B System-Level Implementation Details

B.1 Two-Stage Synergistic Protection Mechanism

While MAOP provides a mathematically rigorous guarantee against gradient conflicts, we embed it within a holistic two-stage protection mechanism to optimize the Global Shared Expert throughout the pretraining lifecycle:

- **Stage 1: Warmup Gradient Shielding.** During the initial warmup phase (*e.g.*, the first 5,000 steps), we explicitly detach the backward gradient paths for non-text tokens (*e.g.*, ViT and VAE tokens) routed to the Shared Expert. This absolute physical isolation allows the Global Shared Expert to establish a stable, text-centric semantic foundation without early-stage high-frequency noise.
- **Stage 2: Permanent MAOP Intervention.** Once the semantic foundation is stabilized, the warmup shielding is lifted, and MAOP is activated permanently. It allows gradients from all modalities to update the Global Shared Expert.

B.2 Differential Learning Rate (DiffLR) Strategy

To further stabilize the pretraining process, we adapt the differential learning rate (DiffLR) practice from Symbiotic-MoE [37]. We implement this natively within the FSDP framework by overriding the optimizer's parameter groups, as demonstrated in Algorithm 1.

Algorithm 1 Differential Learning Rate (DiffLR) Initialization under FSDP

Input: FSDP wrapped model f_θ, Base LR η_{base}, Generation LR η_{new}
Output: Dual-group Optimizer $\mathcal{O}_{dual}$

$\theta_{base} \leftarrow \emptyset$, $\theta_{gen} \leftarrow \emptyset$
for each name, param **in** f_θ.named_parameters() **do**
    clean_name $\leftarrow$ Strip FSDP wrapper prefixes from name
    if clean_name matches VAE router **or** VAE experts **then**
        $\theta_{gen} \leftarrow \theta_{gen} \cup \{\text{param}\}$
    else
        $\theta_{base} \leftarrow \theta_{base} \cup \{\text{param}\}$ ▷ Includes Global Shared Experts & Backbone
    end if
end for
ParamGroups $\leftarrow$[{'params': θ_{gen}, 'lr': η_{new}}, {'params': θ_{base}, 'lr': η_{base}}]
$\mathcal{O}_{dual} \leftarrow$ AdamW(ParamGroups) ▷ Ensures correct global LR inheritance
return $\mathcal{O}_{dual}$

C Comprehensive Pretraining Curriculum and Configurations

C.1 Detailed Composable Pretraining Recipe

Rosetta is fundamentally designed as a dynamic, Lego-like framework. Rather than enforcing a rigid training curriculum, we provide a composable training recipe to empirically demonstrate how modality expansion can be achieved seamlessly. To rigorously isolate the architectural impact, all empirical evaluations in this work are confined exclusively to the pretraining phase—omitting any downstream instruction tuning (SFT) or continual training (CT)—ensuring a strictly fair structural comparison.

C.1.1 Base Modality: Language Foundation via Sparse Upcycling

To establish a robust semantic prior for subsequent multimodal expansion, we construct sparse language foundation via sparse upcycling [25] from the dense Qwen3-0.6B Base model. Specifically, the original dense FFN weights are duplicated to initialize the experts. While standard MoE uses 13 copies (12 routed + 1 shared) and MoT uses 8 copies for its understanding stream (7 routed + 1 shared), Rosetta allocates only 4 copies (3 text-specific experts + 1 global shared expert). All models enforce a Top-2 ($K = 2$) routing strategy alongside the shared expert.

To ensure optimization stability during this dense-to-sparse transition, we apply a surgical *Magnitude-Preserving Initialization* by scaling the `down_proj` weights (detailed mathematical proofs are provided in App. A.4). This design allows all architectures to achieve a near-lossless capability transfer. As shown at iteration 0 in Fig. 1, all sparse models start with an identical MMLU score of 52.4, which closely matches the original dense Qwen3-0.6B model score of 52.8.

Following initialization, all architectures undergo rigorous pretraining on approximately 300B text tokens for 35K steps. The models are optimized using standard autoregressive cross-entropy loss alongside an expert load balancing loss [9], formulated as $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{CE}} + \lambda_{\text{aux}}\mathcal{L}_{\text{aux}}$, where $\lambda_{\text{aux}} = 0.01$.

As illustrated in Fig. 1, after 35K steps of text-only training (LM), standard MoE and MoT reach slightly higher MMLU scores (54.5 and 54.0, respectively) compared to Rosetta (52.5). This performance gap fundamentally aligns with established MoE scaling principles [59]: expanding the total expert count while holding the active parameter budget constant intrinsically increases the model's representational capacity. Consequently, since MoE and MoT allocate significantly more total experts strictly to language modeling, they naturally memorize more text data. Crucially, the objective of this stage is not to train the language models to full saturation, but rather to safely recover the dense capabilities after upcycling. This provides an absolutely fair and stabilized foundation for our primary focus: evaluating continuous multimodal expansion.

C.1.2 Plug-in Extension: Visual Understanding

To equip the model with visual semantics, we "plug in" the Visual Understanding module (Fig. 2 yellow block). For Rosetta, we instantiate dedicated visual experts ($\{\mathcal{E}_{vit,i}\}_{i=1}^{3}$) and a Top-2 router ($\mathcal{G}_{vit}$) by directly duplicating their textual counterparts, ensuring a mature warm-start initialization. For standard MoE, it continually updates its 12 modality-agnostic routed experts and 1 shared expert. For MoT, it continually updates its pre-established understanding stream (7 routed experts + 1 shared expert).

For visual feature extraction, all architectures adopt the vision encoder from Qwen3-VL-30B-Instruct [5]. The ViT backbone is kept frozen, while a trainable linear projector is introduced to align the visual dimension with the LLM semantic space. Following LLaVA [36], we execute a progressive two-stage training across all models. First, we freeze the entire Transformer backbone and exclusively warm up the projector for 3K steps. Second, we unfreeze the backbone and jointly optimize the projector alongside the active expert routing spaces for 20K steps, consuming $\sim$4M multimodal understanding (MMU) samples sampled at a strict ratio of MMU:LM = 0.8:0.2. During this stage, the objective remains $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{CE}} + \lambda_{\text{aux}}\mathcal{L}_{\text{aux}}$, but crucially, the cross-entropy loss mask is applied exclusively to text response tokens, treating image tokens strictly as visual conditions.

C.1.3 Plug-in Extension: Visual Generation

To enable image generation, we "plug in" the Visual Generation module (Fig. 2 red block). For Rosetta, we instantiate six generative experts ($\{\mathcal{E}_{vae,i}\}_{i=1}^{6}$) and a Top-2 router ($\mathcal{G}_{vae}$) by duplicating their pre-trained textual counterparts twice, ensuring a mature warm-start initialization. To guarantee a strictly fair capacity comparison across baselines, standard MoE simply continues optimizing its pre-existing 12 modality-agnostic experts. Conversely, MoT instantiates a physically partitioned generation stream by duplicating its understanding QKV projections and copying the first 6 routed experts (with their corresponding gating slices) plus the shared expert from its understanding FFN. This rigorously enforces an identical generative expert allocation (6 routed + 1 shared) between MoT and Rosetta.

For continuous visual tokenization, all architectures adopt the FLUX.2 VAE [6]. Applying a 2×2 patchify operation yields an effective 16×16 downsampling ratio with 128-dimensional latent chan-

nels. The VAE encoder is always kept frozen. In this stage, all models are jointly trained on a massive mixture of text-to-image (T2I), MMU, and LM samples for 400K steps. The overall objective seamlessly expands to $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{CE}} + \lambda_{\text{aux}}\mathcal{L}_{\text{aux}} + \lambda_{\text{img}}\mathcal{L}_{\text{flow}}$. Specifically, we employ a Flow Matching loss (velocity prediction) [34, 38] for image generation: $\mathcal{L}_{\text{flow}} = \mathbb{E}_{t,\mathbf{x}_0,\mathbf{x}_1}\left[\|v_\theta(\mathbf{x}_t, t) - (\mathbf{x}_0 - \mathbf{x}_1)\|^2\right]$, utilizing a linear flow path with log-normal timestep sampling [12], setting $\lambda_{\text{img}} = 1.0$. Crucially, for Rosetta, MAOP is activated within the Global Shared Expert during this phase. By surgically neutralizing the conflicting high-frequency diffusion gradients, Rosetta successfully prevents representation overwriting, thereby preserving foundational understanding while unlocking cross-modal synergy. Detailed hyperparameters are provided in Table 3.

C.2 Comprehensive Hyperparameters

Table 3: **Pretraining Configurations across Modality Expansions.** All baselines and Rosetta are trained under identical data sequences and hyperparameter settings.

Hyperparameters	LM Foundation	+ Visual Understanding	+ Visual Generation
Total Training Steps	35,000	23,000 (3K Warmup + 20K Joint)	400,000
Global Batch Size	256	64	64
Peak Learning Rate	1e-4	Warmup (1e-4) $\rightarrow$ Joint (2e-5)	1e-4 (Generative) / 1e-5 (Base)
Optimizer	AdamW	AdamW	AdamW with DiffLR (Ours)
Total Tokens / Samples	300B Text Tokens	4M MMU Samples	100M T2I Samples
Data Sampling Ratio	LM	MMU:LM = 0.8 : 0.2	T2I:MMU:LM = 0.6 : 0.25 : 0.15
Trainable Parameters	Full Backbone	Projector (3K steps) $\rightarrow$ Joint (20K steps)	Full Backbone
Frozen Modules	N/A	ViT	ViT, VAE

All experiments are conducted on an industrial-scale NVIDIA H20 GPU cluster. To guarantee a rigorously fair comparison, Rosetta and all baselines are subjected to the exact same hardware allocation and batch size configurations in training. Specifically, the initial language foundation pretraining is scaled across 256 GPUs with a global batch size of 256. The subsequent multimodal expansion phases (for visual understanding and visual generation) are executed on 64 GPUs with a global batch size of 64. To ensure absolute parity, all random seeds are fixed across all baselines. For visual encoding and generation, we utilize the pre-trained Qwen3-VL ViT and FLUX.2 VAE respectively, both of which remain completely frozen during our training. For the final generative expansion, we utilize a dynamic bucketing strategy centered $\approx 256 \times 256$ resolution. More hyperparameters of the pretraining regimen are provided in Table 3.

C.3 Detailed Architectures

To ensure a rigorously fair evaluation, all evaluated architectures (Standard MoE, MoT, and Rosetta) are upcycled from the identical dense foundation (Qwen3-0.6B Base) and guarantee strict active parameter parity during forward and backward passes. As detailed in Table 4, all frameworks activate exactly 1 shared expert and 2 routed experts per token ($K = 2$), resulting in a uniform active parameter count of $\sim$0.97B per token.

However, structurally partitioned paradigms like MoT require dual attention streams and redundant expert allocations, leading to an inflated total parameter footprint (+0.71B). In contrast, Rosetta maintains the identical parameter efficiency as Standard MoE while eliminating routing collapse.

D Additional Qualitative Results

To visually corroborate the catastrophic forgetting analyzed in the main text, we provide extensive qualitative comparisons in Fig. 8. All models are evaluated using identical complex prompts after completing the full multimodal pretraining.

The generated samples reveal specific challenges faced by the baseline architectures during modality expansion:

- **Semantic Drift and Visual Artifacts (MoE):** Without modality-aware constraints, continuous generative gradients can inadvertently interfere with pre-established experts. As a result, standard MoE occasionally exhibits visual artifacts (*e.g.*, unnatural textures in the sky for the "bridge"

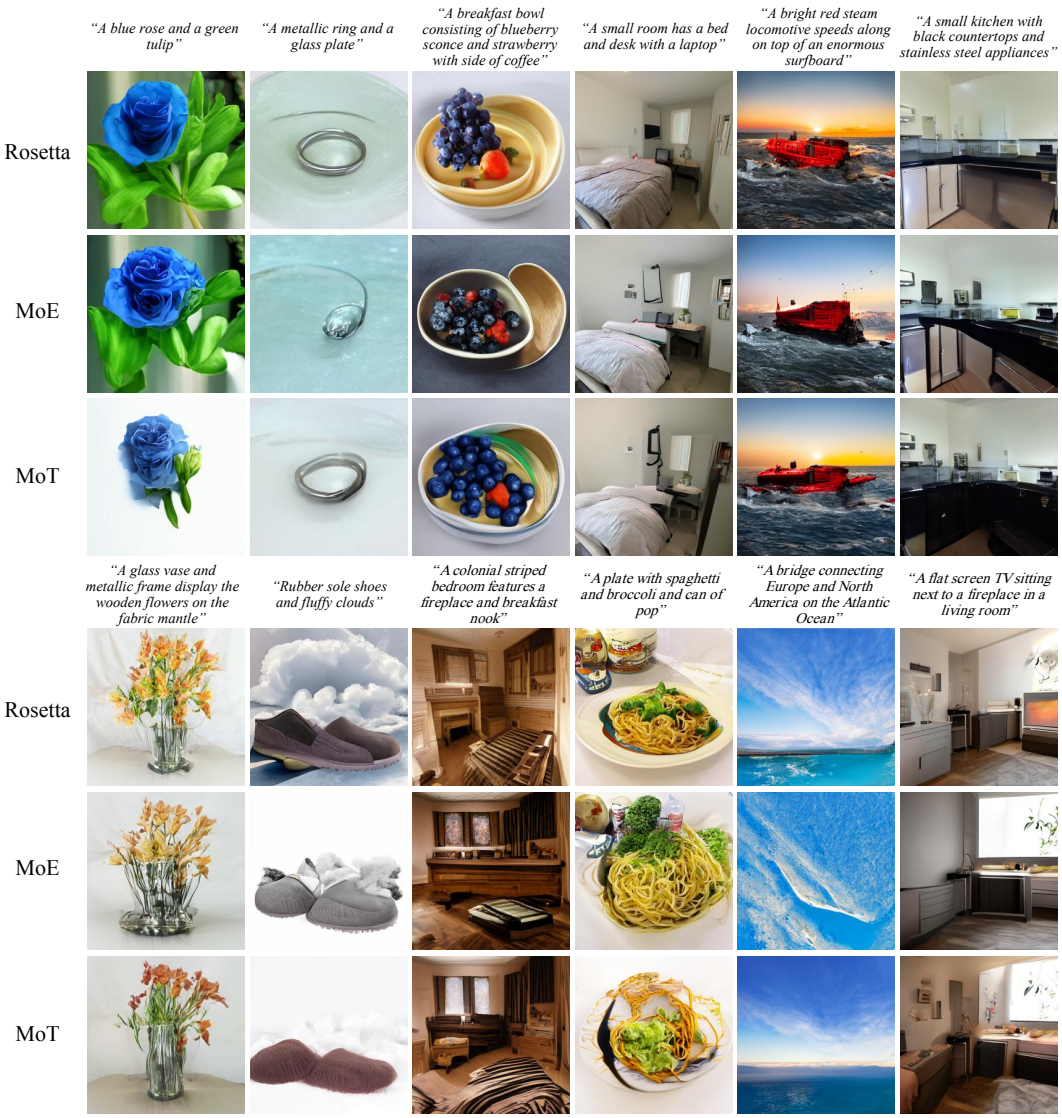


Figure 8: **Qualitative Comparisons of Image Generation.** Images generated under identical complex text prompts. **Middle Rows (MoE):** Exhibits semantic drift and visual artifacts (*e.g.*, corrupted sky textures and mutated food geometries) due to representation overwriting. **Bottom Rows (MoT):** Suffers from structural collapse in indoor scenes and fails at compositional adherence (*e.g.*, entirely omitting the bridge). **Top Rows (Rosetta):** Natively leverages cross-modal synergy to synthesize high-fidelity images, demonstrating precise spatial geometry, rich material textures, and strict compositional prompt adherence.

prompt) and semantic blending (*e.g.*, losing distinct object boundaries between "spaghetti and broccoli", or struggling with the precise geometry of the "metallic ring").

- **Compositional and Structural Inconsistencies (MoT):** While physical isolation protects foundational knowledge, the lack of a shared semantic bridge can limit dense cross-modal alignment. Consequently, MoT sometimes struggles with compositional completeness (*e.g.*, omitting specific subjects like the "bridge") and may present structural distortions in complex indoor environments (*e.g.*, inaccurate perspective or object placements in the "kitchen" and "bedroom" prompts).

In stark contrast, **Rosetta** neutralizes cross-modal interference. Whether rendering fine-grained material textures (*e.g.*, "metallic ring and a glass plate"), or accurate spatial geometries, Rosetta

Table 4: **Detailed Architecture Parameters.** Calculations are based on Qwen3-0.6B Base hyper-parameters ($L = 28, d_{model} = 1024, d_{ffn} = 3072$, vocab=157,420). All frameworks rigorously enforce strict *active parameter parity* by activating exactly 3 experts (2 routed + 1 shared) and 1 attention block per token.

Component	**Standard MoE**	**Rosetta (Ours)**	**MoT (*e.g.*, Bagel)**
Structural Configuration			
Attention Mechanism	Unified (1 Stream)	Unified (1 Stream)	Isolated (2 Streams)
Routed Experts Pool	12 (Agnostic)	3 Text + 3 ViT + 6 VAE	7 Und. + 6 Gen.
Shared Experts Pool	1 (Agnostic)	1 (Global)	1 Und. + 1 Gen.
Active Experts / Token	2 Routed + 1 Shared	2 Routed + 1 Shared	2 Routed + 1 Shared
Per-Layer Parameters			
Attention Block	6.29M (1 stream)	6.29M (1 stream)	12.58M (2 streams)
FFN Experts	122.7M (13 × 9.44M)	122.7M (13 × 9.44M)	141.6M (15 × 9.44M)
Layer Norms	0.002M	0.002M	0.004M
Per-Layer Total	**≈ 129.0M**	**≈ 129.0M**	**≈ 154.2M**
Macro Parameter Budget			
28-Layer Backbone	3,612M	3,612M	4,318M
Embedding Layer	161M	161M	161M
Total Parameters	**3.77B**	**3.77B**	**4.48B**
Active Parameters (Per Token)			
Attention Activation	6.29M	6.29M	6.29M
FFN Activation	28.3M (3 × 9.44M)	28.3M (3 × 9.44M)	28.3M (3 × 9.44M)
Total Active / Layer	34.6M	34.6M	34.6M
Total Active Params	**∼0.97B**	**∼0.97B**	**∼0.97B**

consistently synthesizes high-fidelity images with superior prompt adherence. This visually confirms the efficacy of our mechanisms in unlocking true cross-modal synergy.

E Limitations

While Rosetta successfully overcomes the catastrophic forgetting and gradient conflicts inherent in multimodal expansion, we acknowledge certain limitations that present exciting avenues for future research:

Parameter Scale Constraints. Our empirical evaluations rigorously validate the Rosetta architecture at a highly efficient 3.77B parameter scale (with strict ∼0.97B active parameter parity). Due to computational resource constraints, scaling the framework to massive-scale regimes (*e.g.*, 70B+ parameters) falls outside the scope of the current study. Nevertheless, because the Momentum-Anchored Orthogonal Projection (MAOP) relies on intrinsic optimizer states with zero memory overhead, its mathematical formulation is inherently scale-agnostic, suggesting strong viability for future extreme-scale foundation models.

Scope of Modality Expansion. This work primarily focuses on harmonizing discrete language with continuous visual synthesis and understanding, as they represent the most fundamentally disparate optimization dynamics. Although the composable, Lego-like nature of our plug-and-play experts theoretically supports infinite modality expansion, empirical extension to temporal or spatial modalities (*e.g.*, video generation, 3D assets, and audio) remains unexplored. Validating Rosetta across these broader modalities is a promising next step.

Dynamic Data Curricula. Our current composable pretraining recipe employs a fixed, predefined data sampling ratio (T2I:MMU:LM) to ensure rigorous and fair baseline comparisons. In practice, the loss landscapes of different modalities converge at varying velocities. Developing a dynamic curriculum learning strategy that adaptively adjusts data sampling based on real-time expert capacity could potentially unlock even deeper cross-modal synergies.

F Broader Impacts

Our work on Rosetta introduces a highly scalable and composable framework for unifying discrete language understanding and continuous visual generation. We discuss the broader societal impacts of this research from both positive and potential negative perspectives.

Potential Positive Impacts. From a technological and environmental standpoint, Rosetta significantly reduces the computational overhead typically required for multimodal expansion. By leveraging the zero-memory-overhead MAOP mechanism and efficient sparse routing, our framework enables the training of highly capable multimodal foundation models with a reduced carbon footprint. This improved efficiency can help democratize access to advanced AI research, lowering the barrier for academic laboratories with limited computational resources.

Potential Negative Impacts and Mitigations. While Rosetta advances the architectural design of multimodal foundation models, it shares common challenges with existing systems. Since the model relies on web-scale pretraining corpora, it may inadvertently inherit or reflect historical biases present in the data. Furthermore, although our routing paradigm significantly improves cross-modal alignment, the model may still occasionally generate hallucinatory or contextually imprecise visual content. To mitigate these concerns, Rosetta is explicitly released as a research-stage prototype intended solely for academic exploration. Any future practical deployment adopting this architecture should incorporate standard safety alignment, rigorous data curation, and output moderation to ensure responsible usage.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We elucidate our contributions in the Abstract and Intrudctions.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations in App. E

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

Justification: Our study is mainly empirical exploration.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We report our detailed training settings in Sec. 4.1 and App. C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: Due to institutional policies and internal code review processes, the codebase and training datasets are not included in the initial submission. However, we are fully committed to the open-source community and plan to publicly release the core training code, evaluation scripts, and partial datasets upon publication.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We report our detailed training settings including hyperparameters in Sec. 4.2 and App. C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: MLLMs pre-training consumes a significant amount of computational resources, making it impractical to conduct multiple experiments to obtain error bars.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We discussion computation cost in App. C.2.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have read this code.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the broader impacts in App. F.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [No]

    Justification: Our study is an empirical exploration. The codes and models we plan to release are intended solely for further research and are not meant for direct industrial application.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We explicitly cite the original papers and official repositories for all pre-trained base models and evaluation datasets used in our experiments. We strictly respect and adhere to their respective open-source licenses and terms of use for academic research purposes.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [No]

    Justification: Our primary contribution is the architectural paradigm of Rosetta and the corresponding optimization mechanism. Due to corporate proprietary policies and the strict double-blind review constraints, the full model checkpoints and the complete pretraining codebase are not included in this initial submission. However, we are committed to open science and will publicly release the codebase, model weights, and comprehensive documentation in the camera-ready version upon acceptance.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This work does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This work does not involve crowdsourcing nor research with human subjects.

    Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [No]

    Justification: LLM is used only for writing.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.