FLAME: Adaptive Mixture-of-Experts for Continual Multimodal Multi-Task Learning

Xing Han[1]* **Shravan Chaudhari**[1]* **Tanvi Ranade**[1] **Rama Chellappa**[1] **Suchi Saria**[1,2]
[1]Johns Hopkins University [2]Bayesian Health
{xhan56, schaud35}@jh.edu {tranade1, rchella4, ssaria1}@jhu.edu

Abstract

Real-world model deployment across multiple domains requires multimodal models to operate under two complementary regimes: (1) multi-task pretraining, tasks are co-available at design time where related tasks could borrow representational strength from one another, (2) continual adaptation, in which new tasks emerge after deployment with previously unseen modality combinations. However, neither regime alone suffices: the pretraining task set is never exhaustive, while bypassing joint training forfeits the transfer gains and efficiency among co-trainable tasks. Sparse Mixture-of-Experts (MoE) is a natural fit for this dual requirement: sparse activation enables modular capacity expansion as new tasks arrive, while routing decouples modality-level computation from task-level composition. In this work, we propose a scalable MoE framework for multitask pretraining and continual learning across flexible modality combinations. The framework is designed to support training on multimodal tasks with diverse modality configurations by leveraging modality-specific routers that process tokens from each modality across tasks. Furthermore, it enables continual learning over sequential multimodal tasks within a fixed-capacity MoE by compressing accumulated expert knowledge into low-rank memory subspaces, while expanding only the lightweight routers. We validate the effectiveness of our method on multiple healthcare multimodal benchmarks. It demonstrates competitive multitask pretraining performance while alleviating catastrophic forgetting and improving parameter efficiency[2].

1 Introduction

The proliferation of large-scale multimodal foundation models [31, 64, 51] has demonstrated great potential of learning across diverse modality combinations. Recent multimodal foundation models [24, 80, 46, 22] have extended beyond conventional image-text pairs to support flexible modality combinations, accommodating arbitrary missingness and irregular temporal dynamics across a broader spectrum of modality types. This practical setting is referred as **Flexi-Modal Data** [24], demanding fusion frameworks that are simultaneously flexible, scalable, and temporally aware. However, as shown in Figure 1, real-world deployment requires models that simultaneously handle multiple predictive tasks [82, 74], each associated with a heterogeneous set of input modalities, a setting rarely addressed by existing frameworks. For example, in clinical situations, a mortality rate prediction task may rely on vital signs and clinical notes, whereas a related length-of-stay prediction task may additionally incorporate electrocardiogram (ECG) and imaging data [37, 36, 18]. Moreover, even for similar tasks, modality combinations can differ across populations [71, 65, 24]. Moreover, the set of tasks available during pretraining is rarely exhaustive: after deployment, new clinical

*Equal contribution. Corresponding authors.

[2]Project page: https://github.com/FLAME-NeuRIPS2026. Code: https://github.com/FLAME-NeuRIPS2026/FLAME.

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

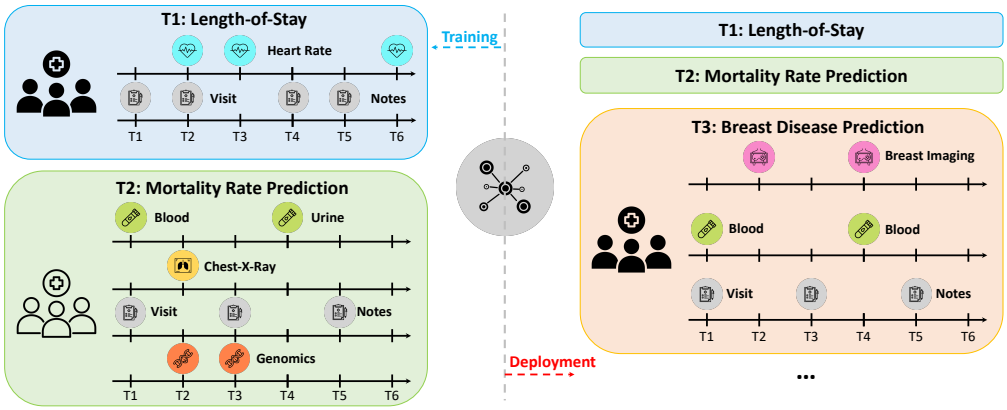


Figure 1: An illustration of the Flexi-Modal multi-task setting. Each task is associated with an arbitrary combination of modalities, where within each task, individual modality measurements may be recorded at irregular time points across channels. During pretraining, models are expected to simultaneously handle various tasks. In deployment, new tasks with unseen modality combinations frequently emerge. FLAME aims to address two core challenges: (1) how to jointly train a unified model across Flexi-Modal multitasks with heterogeneous modality combinations; and (2) how to handle arbitrary train-test modality combination shift and preserve continual learning capacity when new tasks with different modality compositions are introduced after deployment.

questions, sensor modalities, and institutional protocols continually emerge [42, 56], introducing tasks with previously unseen modality combinations that must be absorbed without retraining the model from scratch or sacrificing performance on existing tasks [40, 12, 70]. This variability is the norm rather than the exception. A straightforward solution is training separate models for each task, but this would incur substantial computational overhead, as each model requires extensive tuning, validation, and approval, and forfeits the transfer gains available when related tasks share modalities. Recent state-of-the-art multimodal models have demonstrated strong generalization across different tasks; however, these frameworks typically assume a unified and consistent input space across tasks [51, 43, 68, 15], failing to handle either the Flexi-Modal multi-task setting at training time or the arrival of new tasks with shifted modality configurations after deployment.

A principled entry point is to study modality sharing across tasks: a text encoder trained on clinical notes for mortality prediction should transfer useful representations to a Phenotyping task that also consumes clinical text, even if the remaining modalities differ. HighMMT [46] empirically supports this view, but relies on a pairwise Perceiver-based cross-attention fusion [33] that scales poorly as new tasks with varying modality configurations are introduced. M^{4}oE [74] models patient-level multimodal information via dedicated experts but assumes a fixed modality combination and ignores temporal dependencies in time-varying modalities. More broadly, sparse MoEs [63, 14, 35, 20] are a natural fit for joint pretraining and continual adaptation: the routing mechanism gives flexibility to allocate task-level composition, while the sparsely activated set of experts naturally supports expansion for continual adaptation. However, existing multimodal MoEs route over concatenated, flattened token sequences, coupling gate parameters to a specific modality subset and offering no mechanism to selectively share a single modality across tasks. Although MoEs are theoretically shown to mitigate forgetting [44] and have been explored in continual learning [9, 59], prior work is confined to small-scale unimodal settings with inefficient knowledge reuse.

To address these limitations, we propose **FLAME**, a **F**lexi-modal **L**earning **A**daptive **M**ixture-of-**E**xperts framework for multitask learning and continual adaptation across heterogeneous modality combinations, with a particular focus on healthcare applications. FLAME's contributions are as follows: (1) a per-modality routing mechanism that assigns each modality type a dedicated router dispatching its sequences (regardless of task origin) into a shared expert pool. It decouples the routing topology from any specific task's modality combination and naturally accommodates train/test modality shifts; (2) an efficient continual learning scheme that compresses each new task's contribution into a low-rank additive slice of a *fixed-size* expert pool while expanding only lightweight task-specific routers. This is motivated by our observation that the functional energy of trained experts concentrates

Table 1: Capability comparison between **FLAME** and representative baselines along different axes. Existing multimodal models rarely support both multitask pretraining and continual learning on new tasks simultaneously. Expert knowledge compression for efficient CL is another unique contribution. Capabilities are recorded as reported and implemented in the original papers; ✗ denotes not supported without modification.

Method	Multi-Modal Multi-Task Pretraining			Continual Learning and Shifts		Architecture & Efficiency	
	Flexi-Modal Inputs	Temporal Irregularity	Heterogeneous Task Modalities	New-Task Adaptation	Shift of Modality Composition	Sparse MoE Routing	Expert Knowledge Compression
HighMMT [46]	✓	✗	✓	✗	✓	✗	N.A.
FuseMoE [24]	✓	✓	✗	✗	✗	✓	✗
Flex-MoE [80]	✓	✗	✗	✗	✓	✓	✗
M^{4}oE [74]	✓	✗	✓	✗	✓	✓	✗
SEED [59]	✗	✗	N.A.	✓	N.A.	✓	✗
Lifelong-MoE [9]	✗	✗	N.A.	✓	N.A.	✓	✗
FLAME (Ours)	✓	✓	✓	✓	✓	✓	✓

in a sharply low-rank subspace despite near-full-rank weights. This yields a structural no-forgetting guarantee via cursor-based inference at $5 \sim 15\times$ fewer trained encoder parameters than fine-tuning and EWC; and (3) interpretable insights in which inter-task routing fingerprints reveal which tasks collapse onto overlapping experts, providing a data-driven recommender for which multimodal task combinations benefit from joint training.

Related Works. Mixture-of-Experts has emerged as a scalable paradigm for multimodal learning, capturing inter- and intra-modality interactions through learned routing rather than explicit pairwise attention [63, 53]. LIMoE [53] pioneered large-scale image-text MoE with entropy-based expert specialization; FuseMoE [24] introduced Laplace gating for irregularly sampled and missing modalities; Flex-MoE [80] added a missing-modality bank with dual routers for arbitrary modality availability; and MERGE [22] scaled to the massively multimodal regime via redundancy–uniqueness–synergy interactions. All assume a *fixed* modality structure across inputs, making them incompatible with flexi-modal multi-task settings. Orthogonally, sparse MoEs are effective multitask learners [27, 21], while continual learning (CL) addresses the forward-transfer/backward-interference trade-off [40, 58]. Recent MoE-based CL methods expand experts for new distributions (Lifelong-MoE [9]), use LoRA experts for lifelong LLM adaptation (MoRAL [77]), or selectively fine-tune the least-overlapping expert per task (SEED [59]). However, none investigate whether a multimodal MoE can continually adapt to new tasks with unseen modality combinations without retraining or degrading prior performance. We compare our work with representative baselines in Table 1.

2 Adaptive MoE for Continual Multimodal Multi-Task Learning

In this section, we present the core modules of the proposed FLAME framework, which consists of two components. Sec. 2.1 describes the training of multiple flexi-modal tasks, providing a foundation for multimodal models to acquire task-specific knowledge. We then introduce continual adaptation to new multimodal tasks in Sec.2.2. We first show that, during pretraining, the functional energy of expert weights tends to concentrate in lower-rank components, indicating that the representational capacity of experts is underutilized. This observation motivates our parameter-efficient continual learning approach, which compresses accumulated expert knowledge into low-rank subspaces.

2.1 Multi-Task Pretraining of FLAME

Preliminary: Flexi-Modal Formulation and Sparse MoE. Let $\mathcal{M} = \{m_1, \ldots, m_M\}$ denote all possible modality types, and let $\mathcal{T} = \{\mathcal{T}_1, \ldots, \mathcal{T}_T\}$ be a set of tasks. Each task $\mathcal{T}_k$ is associated with a modality subset $\mathcal{M}_k \subseteq \mathcal{M}$, a dataset $\mathcal{D}_k$, and a prediction objective $\mathcal{L}_k$. A sample of $\mathcal{T}_k$ is a collection of modality sequences $\mathbf{x} = \left\{ \mathbf{x}_m \in \mathbb{R}^{L_m \times d_m} \,\middle|\, m \in \mathcal{M}_k \right\}$, where L_m and d_m are the length and feature dimension of modality m. Both the modality subset $\mathcal{M}_k$ and the lengths $\{L_m\}_{m \in \mathcal{M}_k}$ vary arbitrarily across tasks, and within each task, measurements may be recorded at irregular time points. A standard sparse MoE layer with N experts $\{E_1, \ldots, E_N\}$ and a gating network G maps a token $\mathbf{z} \in \mathbb{R}^d$ to $\sum_{i=1}^{N} G(\mathbf{z})_i \cdot E_i(\mathbf{z})$, with sparsity enforced by activating only

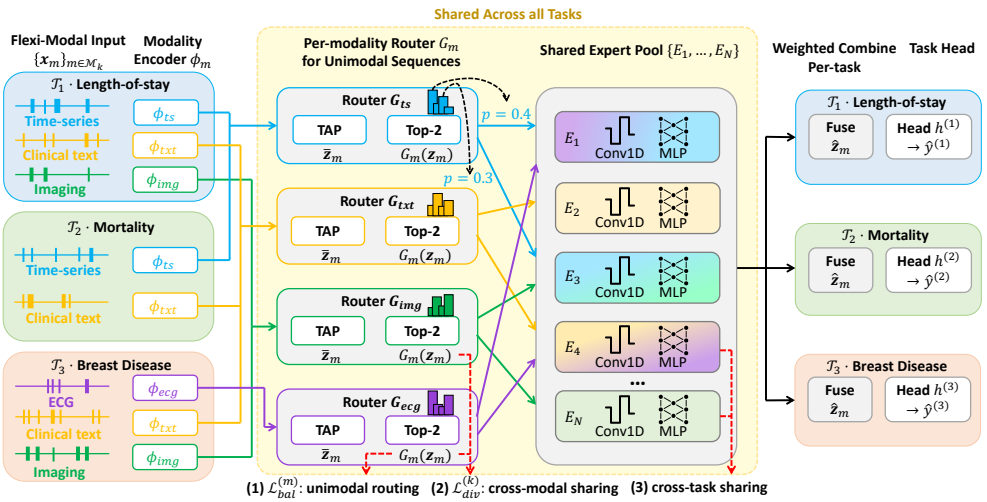


Figure 2: FLAME multi-task pretraining architecture. Flexi-Modal tasks with overlapping but distinct modality subsets are jointly trained. Each modality m is encoded by ϕ_m and routed by a dedicated router G_m (Temporal Attention Pooling + Top-K gating) into a shared expert pool. Each expert applies a 1D temporal convolution followed by a position-wise MLP. Same-modality inputs from different tasks (matching arrow/module color) traverse the same router and the same experts, providing structural inter-task sharing. The router balancing function $\mathcal{L}_{\text{bal}}^{(m)}$ regulates unimodal routing for m, while the routing distribution divergence $\mathcal{L}_{\text{div}}^{(k)}$ controls cross-modal sharing for $\mathcal{T}_k$.

the top-K experts. To enter the MoE, each modality $m \in \mathcal{M}_k$ is first projected by an encoder $\phi_m : \mathbb{R}^{L_m \times d_m} \to \mathbb{R}^{L_m \times d}$ into a sequence of d-dimensional embeddings $\mathbf{z}_m = \phi_m(\mathbf{x}_m)$.

Modality-Specific Routing. Existing multimodal MoEs [24, 80] concatenate all modalities into a single flattened sequence and route over its tokens. This couples the gate parameter shapes to a particular modality combination $\mathcal{M}_k$ and to the lengths $\{L_m\}$, preventing reuse across tasks with heterogeneous modality subsets. We instead instantiate one router G_m *per modality type*, dispatching modality-m embeddings from *any* task into a single shared expert pool $\{E_1, \ldots, E_N\}$. Routing is performed at the *sample* level rather than at the token level, so each decision sees an entire sequence's temporal context while the expert input shape remains $\mathbb{R}^{L_m \times d}$ regardless of how many modalities the current task contains. Sample-level routing requires a fixed-size summary of the variable-length sequence $\mathbf{z}_m \in \mathbb{R}^{L_m \times d}$. Mean pooling discards temporal structure, while flattening reintroduces L_m into the gate parameters. We instead use a learnable query $\mathbf{q}_m \in \mathbb{R}^d$ that attends over time:

$$\bar{\mathbf{z}}_m = \sum_{t=1}^{L_m} \alpha_{m,t}\, \mathbf{z}_{m,t}, \qquad \alpha_{m,t} = \frac{\exp(\langle \mathbf{z}_{m,t},\, \mathbf{q}_m \rangle / \sqrt{d})}{\sum_{t'} \exp(\langle \mathbf{z}_{m,t'},\, \mathbf{q}_m \rangle / \sqrt{d})}. \tag{1}$$

The summary $\bar{\mathbf{z}}_m \in \mathbb{R}^d$ has fixed size and emphasizes routing-relevant time steps. Router G_m then produces sparse top-K gates over the shared pool from $\bar{\mathbf{z}}_m$ via noisy top-K gating [63]; the gate parameters of G_m are shaped only by d and N and are independent of L_m and $|\mathcal{M}_k|$.

Temporal-Aware Expert. Each expert E_i processes a full single-modality sequence at its native temporal resolution. E_i first applies a length-preserving 1D convolution along the time axis with kernel size κ to capture local temporal dynamics, followed by a position-wise two-layer MLP applied identically at every step. All expert weights depend only on d and κ, so sequences of any length and any modality can traverse the same expert. The MoE output for modality m broadcasts the sample-level gate $G_m(\mathbf{z}_m) \in \mathbb{R}^N$ across the temporal axis, so the same K-expert subset processes every time step of a given sample, preserving temporal coherence while remaining sparse.

Cross-Task and Cross-Modal Interactions. Per-modality routing creates different interaction levels to be regulated jointly: (1) *intra-router* expert utilization within a single modality, (2) *inter-task* sharing across tasks that contain the same modality, and (3) *intra-task* interaction across the modalities of a single task. We address (1) and (3) with explicit objectives, while (2) is handled by parameter sharing. Specifically, (2) arises structurally: because the router G_m and the expert

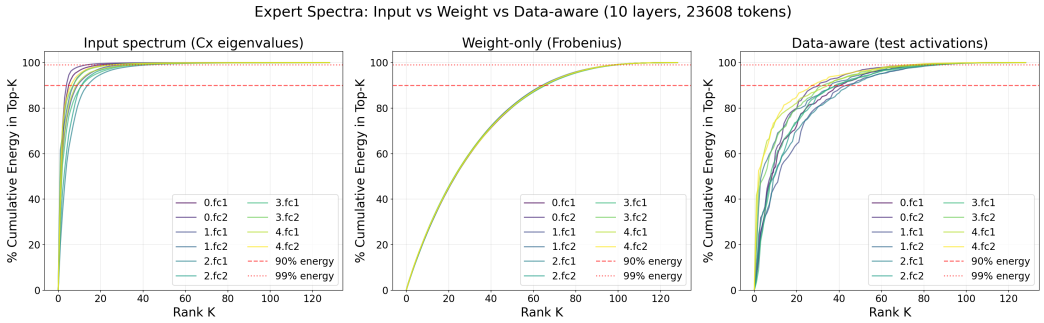


Figure 3: Cumulative top-K energy across all 10 expert sublayers (5 experts $\times$ {`fc1`, `fc2`}) of the first cross-modal MoE block, after FLAME multi-task pretraining on all 9 healthcare tasks. **Left:** input-covariance spectrum $\{\lambda_j\}$ of $C_i = \mathbb{E}_{\boldsymbol{z}}[\boldsymbol{z}\boldsymbol{z}^\top]$, computed over inputs the router actually dispatches to each expert. **Center:** weight-only Frobenius spectrum $\{\sigma_{i,k}^2\}$ of W_i. **Right:** data-aware functional energy $\mathcal{E}_{i,k} = \sigma_{i,k}^2 \boldsymbol{v}_{i,k}^\top C_i \boldsymbol{v}_{i,k}$ on test activations. The input spectrum and the data-aware energy both saturate at very small rank (90% energy reached well before K=20, 99% before K=40), while the weight-only spectrum is essentially full rank. This empirically confirms Proposition 1.

pool $\{E_i\}$ are shared, gradients from $\mathcal{L}_k$ update the same components that any other task $\mathcal{T}_{k'}$ with $m \in \mathcal{M}_k \cap \mathcal{M}_{k'}$ relies on. Similar patterns of modality m across tasks are thus routed to overlapping experts without an explicit alignment loss, while tasks with disjoint modality subsets interact only through experts that both routers happen to select, naturally bounding cross-task interference. Proposition 8 (App. B.3) makes this precise: the interaction scales with the probability that two tasks' routed expert supports overlap, and its sign follows their gradient inner product on the shared experts. For (1), we apply the coefficient-of-variation penalty [63] to each router separately, ensuring full utilization of the shared pool without forcing distinct modalities to use experts in identical proportions. For (3), the modalities in $\mathcal{M}_k$ may be either complementary or partially redundant [22]. Let $\bar{\mathbf{g}}_m^{(k)} = \mathbb{E}_{\mathbf{x}\sim\mathcal{D}_k}[G_m(\mathbf{z}_m)] \in \Delta^{N-1}$ denote the average routing distribution of modality m on task $\mathcal{T}_k$. We regulate cross-modal expert overlap with $\mathcal{L}_{\text{div}}^{(k)} = \beta \cdot \binom{|\mathcal{M}_k|}{2}^{-1} \sum_{\substack{m,m'\in\mathcal{M}_k \\ m\neq m'}} \cos\big(\bar{\mathbf{g}}_m^{(k)}, \bar{\mathbf{g}}_{m'}^{(k)}\big)$, where $\cos(\cdot,\cdot)$ denotes cosine similarity. Setting $\beta = +1$ (*spread*) penalizes overlap, so modalities specialize on disjoint experts and yield complementary representations; setting $\beta = -1$ (*concentrate*) rewards overlap, encouraging fused representations. The mode is selected per task family. Overall, FLAME minimizes the task loss together with the (1) router balancing loss and (3) interaction loss.

2.2 Efficient Continual Learning on New Multimodal Tasks

Multitask pretraining requires all tasks to be available simultaneously, which is rarely the case in practice. In deployment, FLAME must continuously absorb a stream of new tasks $\mathcal{T}_{T+1}, \mathcal{T}_{T+2}, \ldots$ that arrive with previously unseen modality combinations, without revisiting earlier data. At stage t only $\mathcal{D}_t$ is available, and the model must retain performance on previously seen tasks. Sparse MoEs can absorb new tasks by adding experts [9], but the model then grows with every task, which is impractical for compute-heavy multimodal tasks. We instead keep a *fixed-size* expert pool $\{E_1, \ldots, E_N\}$ and combine spectral compression of expert knowledge with lightweight task-specific router expansion.

Motivation: Why is Functional Energy Low-Rank? After multi-task pretraining, we observed that the cumulative functional energy of expert weights saturates at much smaller ranks than the ranks of the weight matrices. Specifically, for each expert E_i with weight matrix W_i and Singular Value Decomposition (SVD) $W_i = U_i \Sigma_i V_i^\top$, let $C_i = \mathbb{E}_{\boldsymbol{z}}[\boldsymbol{z}\boldsymbol{z}^\top]$ denote the covariance of inputs that the router actually dispatches to E_i. The total functional energy admits two equivalent decompositions,

$$\mathcal{E}_i \;=\; \operatorname{tr}(W_i\, C_i\, W_i^\top) \;=\; \underbrace{\sum_k \sigma_{i,k}^2\, \boldsymbol{v}_{i,k}^\top C_i\, \boldsymbol{v}_{i,k}}_{\text{in } W_i\text{'s SVD basis}} \;=\; \underbrace{\sum_j \lambda_j \,\|W_i\, \boldsymbol{q}_j\|^2}_{\text{in } C_i\text{'s eigenbasis}}, \tag{2}$$

where $\{(\lambda_j, \boldsymbol{q}_j)\}$ are eigenpairs of C_i. The first sum yields the per-rank functional energy $\mathcal{E}_{i,k} = \sigma_{i,k}^2 \boldsymbol{v}_{i,k}^\top C_i \boldsymbol{v}_{i,k}$ plotted in Fig. 3 (all benchmarks in App. D.4); the second isolates its source.

Proposition 1. *If C_i has ϵ-effective rank r^*, i.e., $\sum_{j>r^*} \lambda_j \le \epsilon \operatorname{tr}(C_i)$, then $\sum_{j>r^*} \lambda_j \|W_i \boldsymbol{q}_j\|^2 \le \epsilon \|W_i\|_{\mathrm{op}}^2 \operatorname{tr}(C_i)$, so the output $W_i \boldsymbol{z}$ is concentrated in the rank-r^* subspace $W_i \operatorname{span}\{\boldsymbol{q}_1, \ldots, \boldsymbol{q}_{r^*}\}$,* regardless of $\operatorname{rank}(W_i)$.

Proposition 1 (proof in Appendix B) states that the functional rank of an expert is governed by the spectrum of its routed inputs, and not just by the spectrum of its weights as proposed in [39, 38]. Two companion results in Appendix B.2 bridge this input-side guarantee to the weight-side truncation in Eq. (3): gradient flow from the zero-initialized output layer of $\widetilde{W}_i^{(t)}$ aligns its right singular vectors with C_i's eigenvectors (Proposition 2; App. D.4.3), and the rank-r_t truncation error generalizes from $n \gtrsim d_{\mathrm{eff}}(C_i)$ stage-t samples (Proposition 3). We conjecture that modality-specific encoders ϕ_m project inputs onto manifolds with intrinsic dimensionality smaller than d, which drives C_i to be sharply low-rank. In addition, after sufficient training, W_i's top singular vectors $\boldsymbol{v}_{i,k}$ align with C_i's top eigenvectors $\boldsymbol{q}_j$, as shown in prior work [61, 19, 2], so $\mathcal{E}_{i,k}$ saturates as observed in Fig. 3: although W_i is nearly full rank, its singular directions beyond saturation are functionally idle, contributing negligibly to $\mathcal{E}_i$. These idle ranks are the capacity FLAME reallocates to incoming tasks.

Spectral Compression of Expert Knowledge. Let $W_i^{(0)}$ denote the expert weight obtained from the multitask pretraining of Sec. 2.1. At each continual stage $t \ge 1$, we attach to every expert a fresh, full-rank additive component $\widetilde{W}_i^{(t)}$ of the same shape as W_i whose output layer is zero-initialized, train it under unconstrained SGD on $\mathcal{L}_t$ with all components from prior stages frozen, and after convergence replace it by its rank-r_t truncation,

$$\widetilde{W}_i^{(t)} = U_i^{(t)} \Sigma_i^{(t)} V_i^{(t)\top}, \qquad W_i^{(t)} \leftarrow U_{i,\,1:r_t}^{(t)} \, \Sigma_{i,\,1:r_t}^{(t)} \, V_{i,\,1:r_t}^{(t)\top}, \tag{3}$$

which is appended to a frozen stack $\Pi_i = (W_i^{(1)}, W_i^{(2)}, \ldots)$ along the rank dimension, so after τ stages the cumulative delta atop $W_i^{(0)}$ has rank at most $\sum_{j=1}^{\tau} r_j$. Each task t carries a cursor $\tau = k(t)$ pointing to the stage at which it was first trained, and its effective expert weight is

$$W_i^{\mathrm{eff}}(\tau) = W_i^{(0)} + \sum_{j=1}^{\tau} W_i^{(j)}, \tag{4}$$

so the forward pass for task t excludes any component reserved at a later stage. The per-stage rank r_t is a hyperparameter, fixing the cumulative reserved rank $\sum_t r_t$ and turning capacity planning into an explicit choice under a fixed-size MoE (per-stage parameters and memory footprint in App. E).

Task-Specific Router and Head Expansion. Compressing experts preserves *what* each expert knows; the routing patterns that compose them into task-specific predictions must also be preserved. We extend the per-modality router G_m from Sec. 2.1 into a task-indexed family $\{G_m^{(t)}\}_t$. At stage t, for every $m \in \mathcal{M}_t$ we add a new lightweight router head $G_m^{(t)} \in \mathbb{R}^{d \times N}$ dispatching modality m on $\mathcal{T}_t$, while earlier heads $\{G_m^{(s)}\}_{s<t}$ are frozen. A new task head $h^{(t)}$ is added in parallel. Modality encoders ϕ_m are instantiated fresh for any genuinely new modality and reused for those already seen; in either case, the variable-length attention layers inside ϕ_m that standardize sequence lengths across modalities receive the same per-stage compress-and-stack treatment as the experts via Eq. (3). At inference, task identity selects the appropriate $\{G_m^{(t)}\}_{m \in \mathcal{M}_t}$, the cursor $\tau = k(t)$ on every stack Π_i, and the head $h^{(t)}$. The per-stage router overhead is $O(|\mathcal{M}_t|\, d\, N)$, negligible relative to the experts (Algorithm 1).

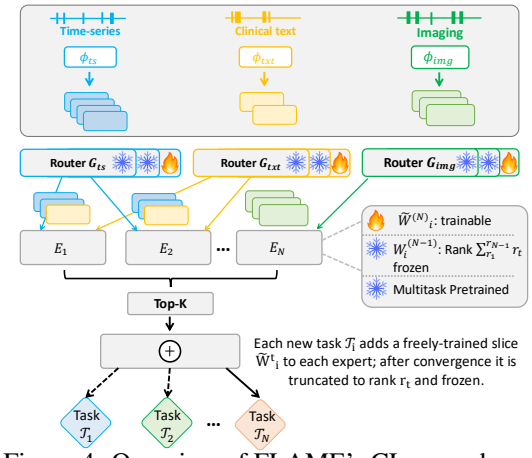


Figure 4: Overview of FLAME's CL procedure for MoE, with improved parameter efficiency.

Structural Forgetting Guarantee. The combination of frozen prior components and cursor-based inference delivers a no-forgetting property that is structural rather than statistical. *Parameter-level*

isolation: all $\{W_i^{(j)}\}_{j<t}$ are frozen during stage-t training, so the optimizer cannot modify any block reserved for a prior task. *Inference-level isolation*: under cursor $\tau = k(t)$, Eq. (4) sums only components reserved at or before stage $k(t)$, so any $W_i^{(j)}$ with $j > k(t)$ is excluded from task t's forward pass by construction. New tasks therefore cannot perturb the predictions of earlier tasks regardless of what they learn. This is strictly stronger than methods that prevent interference at the optimization level only, since those still combine all parameters at inference time. Appendix B.3 formalizes this and extends the single-expert analysis of Proposition 1 to the full stream: forgetting is exactly zero (Proposition 4), the truncation in Eq. (3) costs only the task it compresses, by a functional-energy tail paid once (Propositions 5–6), and the mean final-stage risk obeys a generalization bound with no forgetting term (Proposition 7).

Comparison to Subspace-Orthogonality CL. A complementary line of work enforces non-interference *during* training: GPM [60] projects gradients onto the orthogonal complement of past-task input subspaces at every step, InfLoRA [47] constrains the LoRA parameterization to a pre-designed orthogonal subspace, and O-LoRA [72] adds an orthogonality penalty between successive LoRA matrices. FLAME-CL instead trains $\widetilde{W}_i^{(t)}$ under unconstrained SGD and applies rank-r_t truncation post-hoc. This decouples the forgetting constraint from the optimizer: forgetting is prevented structurally by Eq. (4), regardless of how $\widetilde{W}_i^{(t)}$ is optimized. The validity of the truncation rests on Proposition 1 applied to $\widetilde{W}_i^{(t)}$, with C_i retaining its meaning as the routed-input covariance under stage-t data: the functional energy of $\widetilde{W}_i^{(t)}$ concentrates in a low-rank subspace, so discarding low-singular-value directions removes only directions with negligible contribution to stage-t behavior.

3 Experiments

Overview. We evaluate FLAME on multi-task pretraining and continual adaptation, organizing the analysis around five questions, each paired with the evidence that tests it: **(RQ1)** Is FLAME robust to heterogeneous modality combinations in joint multitask pretraining, matching task-specific multimodal baselines? (Tab. 2) **(RQ2)** Which task pairings benefit from joint training, and does the learned per-modality routing recover interpretable cross-task structure? (Figs. 7, 18) **(RQ3)** Can FLAME-CL absorb a stream of new tasks with shifting modality combinations without forgetting earlier ones, and which component is responsible? (Figs. 5, 6a) **(RQ4)** What is the parameter footprint of FLAME-CL as the task stream grows, and is the rank truncation justified in the continual phase? (Figs. 5, 6b) **(RQ5)** Do the routing and compression mechanisms carry over beyond healthcare and to a large pretrained MoE? (Tab. 3)

3.1 Experimental Setup

Dataset and Task Information. We evaluate FLAME on four publicly available datasets that together yield nine prediction tasks spanning critical care, breast imaging, and neurodegenerative disease. **MIMIC-IV** [37] provides ICU stays with three modalities: vital/laboratory time series, clinical notes, and chest X-rays via MIMIC-CXR [36], evaluated on three tasks (48-IHM, LOS, 25-PHENO). **eICU** [55] adds a multi-center ICU counterpart with time series and clinical notes for two tasks (MOR, RAD). **EMBED** [34] contributes mammography studies with four imaging views (FFDM and synthesized 2D C-View, in CC/MLO projections) and tabular metadata, evaluated on BIRADS, RISK, and DENSITY. **ADNI** [73] provides five per-subject modalities (MRI, FDG-PET, SNP profiles, biospecimen assays, tabular features) over modality-stratified subsets, evaluated on the three-way diagnosis task (DIAG). Further details in Appendix C.

Baselines. For multitask multimodal performance, we compare FLAME against single-task training of recent multimodal baselines: FLEXMOE [80], FUSEMOE [24], and HIGHMMT [46]. FUSEMOE and FLEXMOE assume a fixed modality combination and cannot be trained across tasks with heterogeneous modality subsets; HIGHMMT supports multitask training but its pairwise Perceiver cross-attention follows an offline modality- and interaction-heterogeneity sharing schedule fixed over a closed task-modality pool: any new modality combination forces re-estimation and joint retraining with all prior tasks, ruling out continual use. It also assumes regularly sampled inputs, with no mechanism for irregular clinical time series, restricting the comparison to multitask pretraining.

For continual learning, we benchmark FLAME-CL against simple fine-tuning (SIMPLE FT), EWC [40], LWF [45], Lifelong-PT [9], and LORA [30] with rank-32 adapters on the MoE ex-

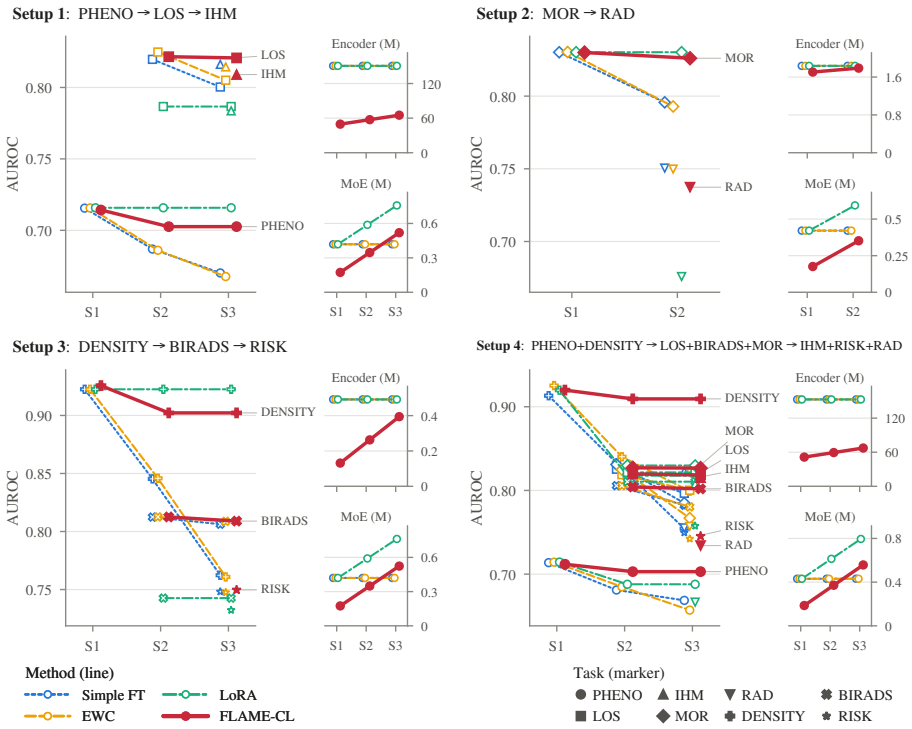


Figure 5: Per-stage continual-learning trajectories (seed 42): AUROC of every task seen so far (large panels) and parameters at inference (small panels); FLAME-CL's point at a task's own stage is before truncation. Numbers in Tab. 13, AUPRC in Fig. 12.

perts only (App. D.3.2 adds them to the encoders as well; Figs. 5, 11); LoRA is a reference point for adaptation quality rather than a practical CL solution, as its parameter count grows with the task stream. We also port two recent MoE-based CL methods, SAME [75] and TRGE [83], from LoRA experts on a pretrained vision-language backbone onto FLAME's MoE backbone, keeping their routing and update rules (App. D.3.3). We exclude InfLoRA [47], O-LoRA [72], and MoRAL [77], which attach per-task LoRA experts to a unimodal pretrained backbone rather than a multimodal MoE trained from scratch; SEED [59] replaces routing and heads with a per-task generative Gaussian ensemble, and PODNet [13] distills 2D convolutional feature maps, whereas our experts are MLPs.

Metrics. We report AUROC and, for CL, average accuracy (AA, mean final AUROC), forgetting (F) [8] and backward transfer (BWT) [49], each measured against the model deployed at the task's own stage (post-truncation for FLAME-CL), so F isolates interference from later stages. Task identity selects the output head at inference [67, 29]. Hyperparameters are in Tab. 4.

3.2 Results

A single FLAME backbone trained jointly across heterogeneous task and modality combinations matches task-specific multimodal baselines (Tables 2a, 2b). The aim is not to outperform single-task models per-task but to show that one architecture absorbs the modality gap typical of joint multimodal training: tasks with overlapping modality pools gain mildly under per-dataset multitask, while disjoint-pool tasks regress only modestly under the all-nine-task regime.

3.3 Discussions

(RQ1) FLAME matches task-specific multimodal baselines: Tables 2a, 2b place it within 0.005 AUROC of the best baseline on six of nine tasks and ahead on AUPRC for the imbalanced binaries (48-IHM, RISK, DIAG), with the all-nine-task regime trailing per-task models by under 0.02 AUROC—the invariance to modality-pool composition that makes a single shared backbone deployable. They are stable under timestamp-generated seeds and auxiliary-loss weights (Apps. D.1, D.2.5). **(RQ2)**

Table 2: Per-task performance across three runs. Multi-class/multi-label tasks use macro-averaged AUROC/AUPRC; all MoE methods use 5 experts. Cells in all FLAME rows are shaded green, darker for smaller gaps to the best value (darkest when FLAME *is* the best): **gap** $= 0$, gap ≤ 0.005, gap ≤ 0.015, gap > 0.015. See Table 6 for full statistics and Figure 8 for the experts ablation.

(a) MIMIC IV and eICU

Metric	Setting	Method	MIMIC IV			eICU	
			48-IHM	LOS	25-PHENO	MOR	RAD
AUROC	Single Task	HighMMT	0.809 ± 0.003	0.830 ± 0.004	0.724 ± 0.002	0.850 ± 0.001	0.767 ± 0.003
		FuseMoE	0.798 ± 0.008	0.819 ± 0.004	0.706 ± 0.002	0.846 ± 0.003	0.763 ± 0.001
		FlexMoE	0.789 ± 0.009	0.817 ± 0.004	0.722 ± 0.001	0.849 ± 0.003	0.768 ± 0.003
		FLAME	0.808 ± 0.001	0.816 ± 0.005	0.723 ± 0.001	0.846 ± 0.002	0.756 ± 0.006
	Multitask (per dataset)	FLAME	0.817 ± 0.005	0.821 ± 0.007	0.719 ± 0.012	0.843 ± 0.005	0.758 ± 0.001
	Multitask (all datasets)	FLAME	0.815 ± 0.002	0.821 ± 0.006	0.719 ± 0.006	0.835 ± 0.005	0.758 ± 0.001
AUPRC	Single Task	HighMMT	0.480 ± 0.019	0.748 ± 0.008	0.485 ± 0.004	0.300 ± 0.007	0.459 ± 0.003
		FuseMoE	0.444 ± 0.014	0.737 ± 0.003	0.455 ± 0.003	0.293 ± 0.007	0.449 ± 0.003
		FlexMoE	0.445 ± 0.020	0.726 ± 0.010	0.484 ± 0.002	0.300 ± 0.006	0.460 ± 0.003
		FLAME	0.492 ± 0.009	0.731 ± 0.007	0.481 ± 0.002	0.293 ± 0.006	0.449 ± 0.011
	Multitask (per dataset)	FLAME	0.450 ± 0.023	0.731 ± 0.012	0.471 ± 0.016	0.279 ± 0.004	0.451 ± 0.003
	Multitask (all datasets)	FLAME	0.470 ± 0.018	0.743 ± 0.008	0.471 ± 0.004	0.271 ± 0.008	0.451 ± 0.001

(b) EMBED and ADNI

Metric	Setting	Method	EMBED			ADNI
			BIRADS	RISK	DENSITY	DIAG
AUROC	Single Task	HighMMT	0.813 ± 0.002	0.733 ± 0.004	0.927 ± 0.001	0.743 ± 0.021
		FuseMoE	0.802 ± 0.001	0.738 ± 0.001	0.920 ± 0.003	0.747 ± 0.028
		FlexMoE	0.809 ± 0.003	0.735 ± 0.003	0.925 ± 0.000	0.702 ± 0.025
		FLAME	0.813 ± 0.002	0.736 ± 0.001	0.926 ± 0.001	0.788 ± 0.023
	Multitask (per dataset)	FLAME	0.811 ± 0.002	0.738 ± 0.005	0.927 ± 0.000	–
	Multitask (all datasets)	FLAME	0.791 ± 0.007	0.714 ± 0.011	0.915 ± 0.007	0.748 ± 0.006
AUPRC	Single Task	HighMMT	0.563 ± 0.001	0.142 ± 0.002	0.765 ± 0.003	0.592 ± 0.028
		FuseMoE	0.550 ± 0.003	0.151 ± 0.008	0.755 ± 0.008	0.601 ± 0.040
		FlexMoE	0.567 ± 0.002	0.142 ± 0.003	0.756 ± 0.004	0.549 ± 0.022
		FLAME	0.564 ± 0.003	0.153 ± 0.010	0.758 ± 0.002	0.660 ± 0.039
	Multitask (per dataset)	FLAME	0.560 ± 0.008	0.151 ± 0.005	0.764 ± 0.002	–
	Multitask (all datasets)	FLAME	0.542 ± 0.010	0.148 ± 0.012	0.743 ± 0.011	0.611 ± 0.018

Tasks pair well exactly when their per-modality routing fingerprints overlap, and routing recovers this structure without supervision: the pairwise heatmap (Fig. 7) and per-task routing panels (Fig. 18) both group EMBED's BIRADS/DENSITY/RISK, eICU's MOR/RAD, and MIMIC-IV's IHM/LOS onto overlapping expert subsets that reinforce, while cross-dataset pairs occupy disjoint experts and gain nothing. **(RQ3)** FLAME-CL absorbs the entire stream without forgetting, structurally rather than statistically: cursor-based inference plus rank reservation prevent later stages from perturbing prior-task forward passes by construction (Sec. 2.2), so after a task's one-time truncation its AUROC never changes (F=0 on all four sequences, Fig. 5), and the truncation costs at most 0.01 except on Setup 3's DENSITY (0.024). Simple FT and EWC drop 0.03–0.06 on PHENO and MOR and 0.13–0.16 on DENSITY, while LoRA, which freezes the base, still forgets DENSITY in Setup 4 (0.92→0.82) and trails FLAME-CL by 0.015–0.029 AA. On the hardest cross-dataset

Table 3: OLMoE-1B-7B after five stages (ROUGE-L; GSM8K exact match; MBPP pass@1). Rank-8 LoRA, truncated to rank 2 by FLAME-CL; IND: per-task adapters; SEQ: no cursor.

Method	SAMSum	GSM8K	MBPP	CNN/DM	CommonGen	Mean ↑	F ↓	Params ↓
Zero-shot	22.2	3.0	1.4	16.5	27.7	14.2	–	–
IND	39.7	16.2	20.4	26.1	38.0	28.1	0.00	388.0M
SEQ	21.4	2.6	16.0	22.2	36.9	19.8	6.71	77.6M
FLAME-CL	39.2	15.8	20.8	26.0	38.2	28.0	0.00	104.9M

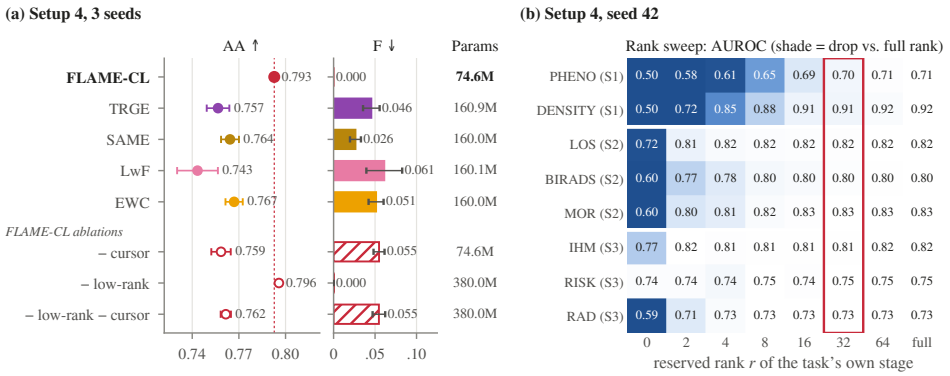


Figure 6: Setup 4. (a) AA, forgetting F and inference-time parameters over three timestamp seeds, with FLAME-CL's ablations (hollow/hatched). (b) CL-phase rank sweep: AUROC when a task's own stage is truncated to rank r (default r=32 outlined). Values in Tabs. 16, 19.

sequence (Fig. 6a), FLAME-CL is the only method with zero forgetting and has the highest AA (0.793 vs. 0.767); DENSITY, the earliest multi-class task, loses 0.06–0.27 AUROC under SAME, TRGE, LwF and EWC (App. D.3.3). Removing the cursor raises F to 0.055, on par with the baselines, with or without compression; removing compression keeps F at zero and AA within seed noise at 5.1× the parameters. The cursor buys retention and compression buys the budget. **(RQ4)** FLAME-CL trains 5–15× fewer encoder parameters while retaining more, and adds only about 0.17M MoE parameters per stage: data-aware spectra saturate by rank 20–40 on every task while weight-only spectra remain near full rank, so rank reservation reuses functionally idle capacity instead of duplicating weights (Proposition 1). Since forgetting is zero by construction, any residual drop in Fig. 5 is truncation cost, which Fig. 6b measures directly: at r=32 every task is within 0.01 AUROC of its untruncated component, though the reserved slices keep only 57–69% of their weight energy, and this cost is paid once (App. B.3). The rank needed falls with stage index (stage-3 tasks saturate by r=2–8), so r=32 is conservative, and the r=0 row shows forward transfer: with no new expert capacity, IHM reaches 0.77 (vs. 0.82 at full rank). Per-task tables, the new baselines on Setups 1–3, AUPRC trajectories and the Lifelong-PT comparison [9] are in App. D.3. **(RQ5)** Interleaving CMU-MOSEI [81] and UR-FUNNY [26] (video–audio–text) with MIMIC-IV tasks, every task stays within 0.01 AUROC of its first-learned value (App. D.7). On OLMoE-1B-7B [52] over SAMSum [17]→GSM8K [10]→MBPP [4]→CNN/DM [62, 28]→CommonGen [48] (Tab. 3), FLAME-CL matches independent adapters (mean 28.0 vs. 28.1) at 3.7× fewer parameters, while stacking adapters without the cursor (SEQ) loses 8.2 points; the LoRA updates keep 96–99% of their data-aware energy at rank 2 (App. D.7.2).

4 Conclusion and Future Works

FLAME unifies flexi-modal multitask pretraining and continual absorption of new tasks, two regimes existing frameworks treat in isolation, in a single sparse MoE: per-modality routers decouple gating from any task's modality subset, and a stacked, rank-truncated additive design with cursor-based inference makes no-forgetting structural rather than a tunable regularizer. FLAME matches task-specific multimodal baselines with one shared backbone, FLAME-CL retains every earlier task with zero forgetting at 5–15× fewer trained encoder parameters, and both mechanisms carry over to video–audio–text data and a 7B-parameter MoE language model. Future work includes large-scale vision–language continual learning and other domains such as robotics, decreasing rank schedules for very long task streams, and per-sample reliability gating across modalities (limitations in App. F).

Acknowledgments and Disclosure of Funding

X.H. and S.S. were funded by the Gordon and Betty Moore Foundation grant #12128, and National Science Foundation (NSF) grant #1840088. The authors declare no competing interests.

References

[1] Pedram Akbarian, Huy Nguyen, Xing Han, and Nhat Ho. Quadratic gating mixture of experts: Statistical insights into self-attention. *arXiv preprint arXiv:2410.11222*, 2024.

[2] Sanjeev Arora, Nadav Cohen, Wei Hu, and Yuping Luo. Implicit regularization in deep matrix factorization. *Advances in neural information processing systems*, 32, 2019.

[3] Alexander Atanasov, Blake Bordelon, and Cengiz Pehlevan. Neural networks as kernel learners: The silent alignment effect. In *International Conference on Learning Representations*, 2022.

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

[5] Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. Multimodal machine learning: A survey and taxonomy. *IEEE transactions on pattern analysis and machine intelligence*, 41(2):423–443, 2018.

[6] Peter L. Bartlett and Shahar Mendelson. Rademacher and Gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3:463–482, 2002.

[7] Rich Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997.

[8] Arslan Chaudhry, Puneet K. Dokania, Thalaiyasingam Ajanthan, and Philip H. S. Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *European Conference on Computer Vision (ECCV)*, pages 532–547, 2018.

[9] Wuyang Chen, Yanqi Zhou, Nan Du, Yanping Huang, James Laudon, Zhifeng Chen, and Claire Cui. Lifelong language pretraining with distribution-specialized experts. In *International Conference on Machine Learning*, pages 5383–5395. PMLR, 2023.

[10] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

[11] Chandler Davis and William M. Kahan. The rotation of eigenvectors by a perturbation. III. *SIAM Journal on Numerical Analysis*, 7(1):1–46, 1970.

[12] Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE transactions on pattern analysis and machine intelligence*, 44(7):3366–3385, 2021.

[13] Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. PODNet: Pooled outputs distillation for small-tasks incremental learning. In *European Conference on Computer Vision (ECCV)*, pages 86–102, 2020.

[14] 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.

[15] Chris Fifty, Ehsan Amid, Zhe Zhao, Tianhe Yu, Rohan Anil, and Chelsea Finn. Efficiently identifying task groupings for multi-task learning. *Advances in Neural Information Processing Systems*, 34:27503–27516, 2021.

[16] Gauthier Gidel, Francis Bach, and Simon Lacoste-Julien. Implicit regularization of discrete gradient dynamics in linear neural networks. In *Advances in Neural Information Processing Systems*, 2019.

[17] Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. SAMSum corpus: A human-annotated dialogue dataset for abstractive summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 70–79, 2019.

[18] Brian Gow, Tom Pollard, Larry A Nathanson, Alistair Johnson, Benjamin Moody, Chrystinne Fernandes, Nathaniel Greenbaum, Jonathan W Waks, Parastou Eslami, Tanner Carbonati, et al. Mimic-iv-ecg: Diagnostic electrocardiogram matched subset. *Type: dataset*, 2023.

[19] Suriya Gunasekar, Jason D Lee, Daniel Soudry, and Nati Srebro. Implicit bias of gradient descent on linear convolutional networks. *Advances in neural information processing systems*, 31, 2018.

[20] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[21] Shashank Gupta, Subhabrata Mukherjee, Krishan Subudhi, Eduardo Gonzalez, Damien Jose, Ahmed H Awadallah, and Jianfeng Gao. Sparsely activated mixture-of-experts are robust multi-task learners. *arXiv preprint arXiv:2204.07689*, 2022.

[22] Xing Han, Hsing-Huan Chung, Joydeep Ghosh, Paul Pu Liang, and Suchi Saria. Guiding mixture-of-experts with temporal multimodal interactions. *arXiv preprint arXiv:2509.25678*, 2025.

[23] Xing Han, Jing Hu, and Joydeep Ghosh. Dynamic combination of heterogeneous models for hierarchical time series. In *2022 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 1207–1216. IEEE, 2022.

[24] Xing Han, Huy Nguyen, Carl Harris, Nhat Ho, and Suchi Saria. Fusemoe: Mixture-of-experts transformers for fleximodal fusion. *Advances in Neural Information Processing Systems*, 37:67850–67900, 2024.

[25] Hrayr Harutyunyan, Hrant Khachatrian, David C Kale, Greg Ver Steeg, and Aram Galstyan. Multitask learning and benchmarking with clinical time series data. *Scientific data*, 6(1):96, 2019.

[26] Md Kamrul Hasan, Wasifur Rahman, AmirAli Bagher Zadeh, Jianyuan Zhong, Md Iftekhar Tanveer, Louis-Philippe Morency, and Mohammed Ehsan Hoque. UR-FUNNY: A multimodal language dataset for understanding humor. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2046–2056, 2019.

[27] Ahmed Hendawy, Jan Peters, and Carlo D'Eramo. Multi-task reinforcement learning with mixture of orthogonal experts. *arXiv preprint arXiv:2311.11385*, 2023.

[28] Karl Moritz Hermann, Tomáš Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, volume 28, 2015.

[29] Yen-Chang Hsu, Yen-Cheng Liu, Anita Ramasamy, and Zsolt Kira. Re-evaluating continual learning scenarios: A categorization and case for strong baselines. *arXiv preprint arXiv:1810.12488*, 2018.

[30] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[31] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[32] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems*, 2018.

[33] Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *International conference on machine learning*, pages 4651–4664. PMLR, 2021.

[34] Jiwoong J Jeong, Brianna L Vey, Ananth Bhimireddy, Thomas Kim, Thiago Santos, Ramon Correa, Raman Dutt, Marina Mosunjac, Gabriela Oprea-Ilies, Geoffrey Smith, Minjae Woo, Christopher R McAdams, Mary S Newell, Imon Banerjee, Judy Gichoya, and Hari Trivedi. The EMory BrEast imaging dataset (EMBED): A racially diverse, granular dataset of 3.4 million screening and diagnostic mammographic images. *Radiol. Artif. Intell.*, 5(1):e220047, January 2023.

[35] 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.

[36] Alistair Johnson, Tom Pollard, Roger Mark, Seth Berkowitz, and Steven Horng. Mimic-cxr database. *PhysioNet10*, 13026(C2JT1Q):5, 2024.

[37] Alistair EW Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, et al. Mimic-iv, a freely accessible electronic health record dataset. *Scientific data*, 10(1):1, 2023.

[38] Prakhar Kaushik, Shravan Chaudhari, Ankit Vaidya, Rama Chellappa, and Alan Yuille. The universal weight subspace hypothesis, 2025.

[39] Prakhar Kaushik, Ankit Vaidya, Shravan Chaudhari, and Alan Yuille. Eigenlorax: Recycling adapters to find principal subspaces for resource-efficient adaptation and inference. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 649–659, 2025.

[40] 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.

[41] Vladimir Koltchinskii and Karim Lounici. Concentration inequalities and moment bounds for sample covariance operators. *Bernoulli*, 23(1):110–133, 2017.

[42] Matthias Lenga, Heinrich Schulz, and Axel Saalbach. Continual learning for domain adaptation in chest x-ray classification. In *Medical Imaging with Deep Learning*, pages 413–423. PMLR, 2020.

[43] Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. Llava-med: Training a large language-and-vision assistant for biomedicine in one day. *Advances in Neural Information Processing Systems*, 36:28541–28564, 2023.

[44] Hongbo Li, Sen Lin, Lingjie Duan, Yingbin Liang, and Ness B Shroff. Theory on mixture-of-experts in continual learning. *arXiv preprint arXiv:2406.16437*, 2024.

[45] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2017.

[46] Paul Pu Liang, Yiwei Lyu, Xiang Fan, Jeffrey Tsaw, Yudong Liu, Shentong Mo, Dani Yogatama, Louis-Philippe Morency, and Ruslan Salakhutdinov. High-modality multimodal transformer: Quantifying modality & interaction heterogeneity for high-modality representation learning. *arXiv preprint arXiv:2203.01311*, 2022.

[47] Yan-Shuo Liang and Wu-Jun Li. Inflora: Interference-free low-rank adaptation for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[48] Bill Yuchen Lin, Wangchunshu Zhou, Ming Shen, Pei Zhou, Chandra Bhagavatula, Yejin Choi, and Xiang Ren. CommonGen: A constrained text generation challenge for generative commonsense reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1823–1840, 2020.

[49] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[50] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

[51] Michael Moor, Qian Huang, Shirley Wu, Michihiro Yasunaga, Cyril Zakka, Yash Dalmia, Eduardo Pontes Reis, Pranav Rajpurkar, and Jure Leskovec. Med-flamingo: a multimodal medical few-shot learner (2023). *URL: https://arxiv. org/abs/2307.15189*, 2023.

[52] Niklas Muennighoff, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Jacob Morrison, Sewon Min, Weijia Shi, Pete Walsh, Oyvind Tafjord, Nathan Lambert, et al. OLMoE: Open mixture-of-experts language models. *arXiv preprint arXiv:2409.02060*, 2024.

[53] Basil Mustafa, Carlos Riquelme, Joan Puigcerver, Rodolphe Jenatton, and Neil Houlsby. Multimodal contrastive learning with limoe: the language-image mixture of experts. *Advances in Neural Information Processing Systems*, 35:9564–9576, 2022.

[54] Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, Andrew Y Ng, et al. Multimodal deep learning. In *Icml*, volume 11, pages 689–696, 2011.

[55] Tom Pollard, Alistair Johnson, Jesse Raffa, Leo Anthony Celi, Omar Badawi, and Roger Mark. eICU Collaborative Research Database. *PhysioNet*, April 2019. Version 2.0.

[56] Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletari, Holger R Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N Galtier, Bennett A Landman, Klaus Maier-Hein, et al. The future of digital health with federated learning. *NPJ digital medicine*, 3(1):119, 2020.

[57] Sebastian Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.

[58] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016.

[59] Grzegorz Rypeść, Sebastian Cygert, Valeriya Khan, Tomasz Trzciński, Bartosz Zieliński, and Bartłomiej Twardowski. Divide and not forget: Ensemble of selectively trained experts in continual learning. *arXiv preprint arXiv:2401.10191*, 2024.

[60] Gobinda Saha, Isha Garg, and Kaushik Roy. Gradient projection memory for continual learning. In *International Conference on Learning Representations*, 2021.

[61] Andrew M Saxe, James L McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv preprint arXiv:1312.6120*, 2013.

[62] Abigail See, Peter J. Liu, and Christopher D. Manning. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, 2017.

[63] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[64] 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.

[65] Luan Tran, Xiaoming Liu, Jiayu Zhou, and Rong Jin. Missing modalities imputation via cascaded residual autoencoder. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1405–1414, 2017.

[66] Joel A. Tropp. An introduction to matrix concentration inequalities. *Foundations and Trends in Machine Learning*, 8(1–2):1–230, 2015.

[67] Gido M. van de Ven and Andreas S. Tolias. Three scenarios for continual learning. *arXiv preprint arXiv:1904.07734*, 2019.

[68] Simon Vandenhende, Stamatios Georgoulis, Wouter Van Gansbeke, Marc Proesmans, Dengxin Dai, and Luc Van Gool. Multi-task learning for dense prediction tasks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(7):3614–3633, 2021.

[69] Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge University Press, 2018.

[70] Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *IEEE transactions on pattern analysis and machine intelligence*, 46(8):5362–5383, 2024.

[71] Qi Wang, Liang Zhan, Paul Thompson, and Jiayu Zhou. Multimodal learning with incomplete modalities by knowledge distillation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1828–1838, 2020.

[72] Xiao Wang, Tianze Chen, Qiming Ge, Han Xia, Rong Bao, Rui Zheng, Qi Zhang, Tao Gui, and Xuanjing Huang. Orthogonal subspace learning for language model continual learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023.

[73] Michael W Weiner, Dallas P Veitch, Paul S Aisen, Laurel A Beckett, Nigel J Cairns, Robert C Green, Danielle Harvey, Clifford R Jack, Jr, William Jagust, John C Morris, Ronald C Petersen, Jennifer Salazar, Andrew J Saykin, Leslie M Shaw, Arthur W Toga, John Q Trojanowski, and Alzheimer's Disease Neuroimaging Initiative. The alzheimer's disease neuroimaging initiative 3: Continued innovation for clinical trial improvement. *Alzheimers Dement*, 13(5):561–571, December 2016.

[74] Chenwei Wu, Zitao Shuai, Zhengxu Tang, Luning Wang, and Liyue Shen. Dynamic modeling of patients, modalities and tasks via multi-modal multi-task mixture of experts. In *The thirteenth international conference on learning representations*, 2025.

[75] Zhen-Hao Xie, Jun-Tao Tang, Yu-Cheng Shi, Han-Jia Ye, De-Chuan Zhan, and Da-Wei Zhou. SAME: Stabilized mixture-of-experts for multimodal continual instruction tuning. *arXiv preprint arXiv:2602.01990*, 2026.

[76] Peng Xu, Xiatian Zhu, and David A Clifton. Multimodal learning with transformers: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(10):12113–12132, 2023.

[77] Shu Yang, Muhammad Asif Ali, Cheng-Long Wang, Lijie Hu, and Di Wang. Moral: Moe augmented lora for llms' lifelong learning. *arXiv preprint arXiv:2402.11260*, 2024.

[78] Jun Yu, Yutong Dai, Xiaokang Liu, Jin Huang, Yishan Shen, Ke Zhang, Rong Zhou, Eashan Adhikarla, Wenxuan Ye, Yixin Liu, et al. Unleashing the power of multi-task learning: A comprehensive survey spanning traditional, deep, and pretrained foundation model eras. *arXiv preprint arXiv:2404.18961*, 2024.

[79] Yi Yu, Tengyao Wang, and Richard J. Samworth. A useful variant of the Davis–Kahan theorem for statisticians. *Biometrika*, 102(2):315–323, 2015.

[80] Sukwon Yun, Inyoung Choi, Jie Peng, Yangfan Wu, Jingxuan Bao, Qiyiwen Zhang, Jiayi Xin, Qi Long, and Tianlong Chen. Flex-moe: Modeling arbitrary modality combination via the flexible mixture-of-experts. *Advances in Neural Information Processing Systems*, 37:98782–98805, 2024.

[81] AmirAli Bagher Zadeh, Paul Pu Liang, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. Multimodal language analysis in the wild: CMU-MOSEI dataset and interpretable dynamic fusion graph. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2236–2246, 2018.

[82] Yu Zhang and Qiang Yang. A survey on multi-task learning. *IEEE transactions on knowledge and data engineering*, 34(12):5586–5609, 2021.

[83] Jialu Zhou, Dianxi Shi, Shaowu Yang, Xinyu Wei, Mingyue Yang, Leqian Li, Mengzhu Wang, and Chunping Qiu. Separation and collaboration: Two-level routing grouped mixture-of-experts for multi-domain continual learning. *arXiv preprint arXiv:2508.07738*, 2025.

Supplementary Material for "FLAME: Adaptive Mixture-of-Experts for Continual Multimodal Multi-Task Learning"

Appendix Contents

- **A.** Extended Related Works
- **B.** Theoretical Results and Proofs — Proposition 1, alignment and sample complexity of the truncation, and continual-learning guarantees (no forgetting, truncation cost, stream-level generalization, transfer and interference)
- **C.** Dataset Details
- **D.** Additional Experimental Results
  - D.1 Hardware, Implementation, and Hyperparameters — hyperparameter table, timestamp-seed re-runs
  - D.2 Detailed Multitask Pretraining Results — full per-task table, pairwise interactions, number-of-experts ablation, gating-function ablation, auxiliary-loss weight sensitivity
  - D.3 Detailed Continual Learning Results — per-stage numbers against Simple FT/EWC/LoRA, MoE-based and distillation baselines (SAME, TRGE, LwF) on all four setups with AA/F/BWT, retention vs. plasticity at a fixed budget, CL-phase rank sweep, component ablation, training cost, comparison against Lifelong-PT, per-stage AUPRC trajectories
  - D.4 Spectral Validation of Proposition 1 — single-task, multi-task, and continual-stage expert spectra, and singular-vector alignment on all four continual sequences
  - D.5 Per-Modality Routing Distributions — single-task, multi-task, and continual-stage routing fingerprints
  - D.6 Robustness to Modality Corruption, Disagreement, and Missingness
  - D.7 Beyond Healthcare — CMU-MOSEI and UR-FUNNY, and scaling FLAME-CL to OLMoE-1B-7B
- **E.** Additional Methodological Details
- **F.** Limitations
- **G.** Broader Impact
- **H.** Asset Licenses

A Extended Related Works

Multimodal Learning and MoEs. Multimodal learning spans early fusion, late fusion, and cross-modal attention architectures [54, 5, 76], with recent Transformer-based approaches enabling scalable joint modeling of heterogeneous inputs. MoE [63, 14, 23, 35, 1, 20] has emerged as a powerful paradigm for scaling model capacity without proportional increases in computation, by dynamically routing inputs through a sparse subset of specialized experts. It has achieved superior results in multimodal learning, where the multimodal embeddings are routed to experts based on the learned routing mechanisms. The inter and intra-modality relationships are captured during the routing process without the need to explicitly specify pairwise modality as in cross-attention. LIMoE [53] pioneered large-scale multimodal MoE by processing images and text through a shared sparse transformer with contrastive learning, demonstrating natural expert specialization via entropy-based regularization. FuseMoE [24] addressed the flexi-modal setting with irregularly sampled and missing modalities, introducing a Laplace gating function with theoretical convergence guarantees superior to softmax routing. Building on this, Flex-MoE [80] proposed a missing modality bank and dual-router design to handle arbitrary modality availability at inference. Most recently, MERGE [22] extended multimodal MoE to the massively multimodal regime by using temporal redundancy, uniqueness, and synergy interactions to guide routing. Despite these advances, existing multimodal MoE methods operate under a fixed and consistent modality structure across inputs, making them fundamentally incompatible with the Flexi-Modal multi-task scenario.

Multi-Task and Continual Learning. Multi-task learning (MTL) [7, 82, 78] is a principled approach to improve generalization by jointly optimizing across related tasks, leveraging shared representations to reduce sample complexity and improve model efficiency. MTL has been extended to complex settings involving gradient conflict resolution [15], task grouping, and parameter sharing strategies [68, 57]. Recent works have shown that sparse MoEs are effective at generalizing across multiple tasks and serve as robust multitask learners [27, 21, 27]. Complementary to MTL, continual learning (CL) [70, 12] addresses the sequential task setting, where a model must acquire new knowledge over time without catastrophically forgetting previously learned tasks [40]. Foundational approaches such as Elastic Weight Consolidation (EWC) [40] and Progressive Neural Networks [58] have established the canonical trade-off between forward transfer and backward interference. A growing line of work has specifically examined the continual learning capacity of MoE. Lifelong-MoE [9] progressively expands experts and gating dimensions to accommodate new pretraining data distributions. MoRAL [77] couples MoE with LoRA, using low-rank adapters as experts so that lifelong adaptation of LLMs can be achieved with minimal trainable parameters. SEED [59] tackles class-incremental learning by maintaining a fixed-size expert ensemble and selectively fine-tuning only the single expert whose distributions overlap least with the new task. Despite these advances, the capacity for multitask and continual learning in multimodal models remains substantially understudied. Many multimodal multitask frameworks typically assume a fixed and unified set of modalities across all tasks, and none have explored the CL capabilities of multimodal MoE. A key question is whether a flexi-modal MoE model trained on an initial set of tasks can rapidly adapt to new tasks with previously unseen modality combinations after deployment, without retraining or degrading performance on existing tasks.

B Theoretical Results and Proofs

B.1 Proof of Proposition 1

Proposition 1. *Let $W_i \in \mathbb{R}^{p\times d}$ be the weight matrix of expert E_i and let $C_i = \mathbb{E}_{\boldsymbol{z}}[\boldsymbol{z}\boldsymbol{z}^\top] \in \mathbb{R}^{d\times d}$ be the covariance of the inputs routed to E_i, with eigendecomposition $C_i = \sum_{j=1}^{d} \lambda_j \boldsymbol{q}_j \boldsymbol{q}_j^\top$ where $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d \geq 0$. Suppose C_i has ϵ-effective rank r^*, i.e.,*

$$\sum_{j>r^*} \lambda_j \;\leq\; \epsilon \,\mathrm{tr}(C_i). \tag{5}$$

Then the orthogonal-tail contribution to the functional energy satisfies

$$\sum_{j>r^*} \lambda_j \,\|W_i\,\boldsymbol{q}_j\|^2 \;\leq\; \epsilon \,\|W_i\|_{\mathrm{op}}^2 \,\mathrm{tr}(C_i), \tag{6}$$

where $\|W_i\|_{\mathrm{op}} = \sigma_{i,1}$ denotes the spectral norm of W_i. Consequently, for any input $\boldsymbol{z}$ drawn from the routed distribution, the output $W_i\boldsymbol{z}$ is concentrated, in expected squared norm, in the rank-r^* subspace $W_i\,\mathrm{span}\{\boldsymbol{q}_1, \ldots, \boldsymbol{q}_{r^*}\}$, regardless of $\mathrm{rank}(W_i)$.

Proof. We proceed in three steps: (i) decompose the total functional energy in C_i's eigenbasis, (ii) bound the orthogonal tail using the spectral norm of W_i, and (iii) translate the bound into a statement about the output $W_i\boldsymbol{z}$.

Step 1: Decomposition. Since $\{\boldsymbol{q}_j\}_{j=1}^d$ is an orthonormal eigenbasis of C_i, we may write $C_i = \sum_j \lambda_j \,\boldsymbol{q}_j\boldsymbol{q}_j^\top$. The total functional energy of W_i on the routed input distribution is

$$\begin{aligned}
\mathcal{E}_i \;&=\; \mathbb{E}_{\boldsymbol{z}}\big[\|W_i\boldsymbol{z}\|^2\big] \;=\; \mathbb{E}_{\boldsymbol{z}}\big[\mathrm{tr}(W_i\boldsymbol{z}\boldsymbol{z}^\top W_i^\top)\big] \;=\; \mathrm{tr}(W_i\,C_i\,W_i^\top) \\
&=\; \mathrm{tr}\Big(W_i\Big(\textstyle\sum_j \lambda_j\,\boldsymbol{q}_j\boldsymbol{q}_j^\top\Big)W_i^\top\Big) \;=\; \displaystyle\sum_{j=1}^{d} \lambda_j\,\mathrm{tr}(W_i\boldsymbol{q}_j\boldsymbol{q}_j^\top W_i^\top) \\
&=\; \sum_{j=1}^{d} \lambda_j\,\|W_i\boldsymbol{q}_j\|^2,
\end{aligned} \tag{7}$$

where the last equality uses $\mathrm{tr}(\boldsymbol{a}\boldsymbol{a}^\top) = \|\boldsymbol{a}\|^2$ with $\boldsymbol{a} = W_i\boldsymbol{q}_j$. We split this sum at the cutoff r^*:

$$\mathcal{E}_i \;=\; \underbrace{\sum_{j=1}^{r^*} \lambda_j \,\|W_i\boldsymbol{q}_j\|^2}_{\mathcal{E}_i^{\le r^*}} \;+\; \underbrace{\sum_{j=r^*+1}^{d} \lambda_j \,\|W_i\boldsymbol{q}_j\|^2}_{\mathcal{E}_i^{>r^*}}. \tag{8}$$

Step 2: Tail bound. For each j, the spectral norm of W_i implies the operator inequality $\|W_i\boldsymbol{q}_j\|^2 \le \|W_i\|_{\mathrm{op}}^2\|\boldsymbol{q}_j\|^2 = \|W_i\|_{\mathrm{op}}^2$, since $\boldsymbol{q}_j$ is a unit vector. Substituting into $\mathcal{E}_i^{>r^*}$,

$$\mathcal{E}_i^{>r^*} \;=\; \sum_{j>r^*} \lambda_j \,\|W_i\boldsymbol{q}_j\|^2 \;\le\; \|W_i\|_{\mathrm{op}}^2 \sum_{j>r^*} \lambda_j \;\le\; \epsilon\,\|W_i\|_{\mathrm{op}}^2\,\mathrm{tr}(C_i), \tag{9}$$

where the last inequality applies the effective-rank assumption (5). This establishes (6).

Step 3: Output concentration. Let $\Pi_{r^*} = \sum_{j=1}^{r^*} \boldsymbol{q}_j\boldsymbol{q}_j^\top$ project onto the top-r^* eigensubspace of C_i, and $\Pi_{r^*}^\perp = I - \Pi_{r^*}$. Decompose $\boldsymbol{z} = \Pi_{r^*}\boldsymbol{z} + \Pi_{r^*}^\perp\boldsymbol{z}$ and note that the cross term vanishes in expectation because the eigenbasis diagonalizes C_i:

$$\mathbb{E}\big[\|W_i\boldsymbol{z}\|^2\big] \;=\; \mathbb{E}\big[\|W_i\Pi_{r^*}\boldsymbol{z}\|^2\big] \;+\; \mathbb{E}\big[\|W_i\Pi_{r^*}^\perp\boldsymbol{z}\|^2\big], \tag{10}$$

since $\mathbb{E}[(\Pi_{r^*}\boldsymbol{z})^\top W_i^\top W_i(\Pi_{r^*}^\perp\boldsymbol{z})] = \mathrm{tr}(W_i^\top W_i\Pi_{r^*}^\perp C_i\Pi_{r^*}) = 0$. The same Step-1 calculation applied to each piece yields $\mathbb{E}[\|W_i\Pi_{r^*}^\perp\boldsymbol{z}\|^2] = \mathcal{E}_i^{>r^*}$. Combining (10) with (9),

$$\mathbb{E}\big[\|W_i\Pi_{r^*}^\perp\boldsymbol{z}\|^2\big] \;\le\; \epsilon\,\|W_i\|_{\mathrm{op}}^2\,\mathrm{tr}(C_i), \tag{11}$$

which is independent of $\mathrm{rank}(W_i)$. Hence the output $W_i\boldsymbol{z}$ is concentrated, in expected squared norm, in $W_i\,\mathrm{span}\{\boldsymbol{q}_1,\ldots,\boldsymbol{q}_{r^*}\}$. □

Remark 1. *Dividing both sides of* (6) *by $\mathcal{E}_i$ gives the relative tail energy $\mathcal{E}_i^{>r^*}/\mathcal{E}_i \le \epsilon\,\kappa_i$, where $\kappa_i = \|W_i\|_{\mathrm{op}}^2\,\mathrm{tr}(C_i)/\mathcal{E}_i$ measures the misalignment between W_i's top singular directions and C_i's top eigendirections; $\kappa_i = 1$ in the perfectly aligned case and grows when W_i wastes capacity on low-energy directions. Empirically, gradient training drives κ_i close to* 1*, giving the sharp saturation observed in Fig. 3.*

Remark 2. *Proposition 1 underwrites the spectral truncation in Eq.* (3)*. The truncation in* FLAME-CL *is applied not to the full expert weight W_i but to the stage-t additive component $\widetilde{W}_i^{(t)}$, whose role is to encode the new task on top of the already-frozen base $W_i^{(0)} + \sum_{j<t} W_i^{(j)}$. Applying Proposition 1 to $\widetilde{W}_i^{(t)}$, with C_i retaining its meaning as the routed-input covariance under stage-t data, shows that the functional energy of $\widetilde{W}_i^{(t)}$ concentrates in a low-rank subspace of its routed-input distribution. The post-training alignment of Remark 1 further implies that low-singular-value directions of $\widetilde{W}_i^{(t)}$ correspond to low-functional-energy directions, so the rank-r_t truncation defining $W_i^{(t)}$ is approximately functionally lossless on the stage-t distribution while reducing the per-stage memory footprint to $r_t(p+d+1)$ scalars.*

B.2 Alignment and Sample Complexity for the Spectral Truncation

Proposition 1 establishes input-side concentration: the output $W_i\boldsymbol{z}$ lives in the rank-r^* eigensubspace of C_i. The truncation in Eq. (3) is applied weight-side, however, to the singular vectors of $\widetilde{W}_i^{(t)}$. We close that gap with two complementary results: an alignment argument under gradient flow, showing that with the zero initialization used for the output layer of $\widetilde{W}_i^{(t)}$ in Eq. (3) its right singular vectors coincide with the eigenvectors of C_i (Proposition 2), and a sample-complexity bound, showing that the resulting truncation error is faithfully estimated from finite stage-t data (Proposition 3). Both results study a linear surrogate of the expert nonlinearity; their conclusions are corroborated empirically across all benchmarks in Sec. D.4.

Linear quadratic surrogate. For the alignment argument we replace the expert by its linear part $\boldsymbol{z} \mapsto W\boldsymbol{z}$ and consider continuous-time gradient flow on the population least-squares objective

$$\mathcal{L}(W) \;=\; \tfrac{1}{2}\,\mathbb{E}_{\boldsymbol{z}\sim p_i}\big\|W^\star \boldsymbol{z} - W\boldsymbol{z}\big\|^2 \;=\; \tfrac{1}{2}\,\mathrm{tr}\big((W-W^\star)\,C_i\,(W-W^\star)^\top\big), \tag{12}$$

with stage-t optimum $W^\star \in \mathbb{R}^{p\times d}$ and routed-input distribution p_i of covariance C_i. The flow $\dot W_t = -\nabla_W \mathcal{L}(W_t) = (W^\star - W_t)\,C_i$ models the training dynamics of $\widetilde{W}_i^{(t)}$ on stage-t data, with $W^\star$ the optimal linearization at convergence; the linear-surrogate assumption is standard in implicit-bias analyses [61, 19, 2].

Proposition 2 (Alignment under zero-initialized gradient flow)**.** *Let W_t evolve under the gradient flow on* (12) *with $W_0 = 0$, matching the zero-initialized output layer of $\widetilde{W}_i^{(t)}$ in Eq.* (3)*. Then for every $t \ge 0$,*

$$W_t \;=\; W^\star\,\big(I - e^{-C_i t}\big), \qquad W_t\boldsymbol{q}_j \;=\; \big(1-e^{-\lambda_j t}\big)\,W^\star\boldsymbol{q}_j. \tag{13}$$

(i) Convergence on the effective subspace. *For any index set $J \subseteq [d]$ with $\lambda_J := \min_{j\in J}\lambda_j > 0$, writing $Q_J = [\boldsymbol{q}_j]_{j\in J}$,*

$$\big\|(W_t - W^\star)\,Q_J\big\|_F^2 \;=\; \sum_{j\in J}\|W^\star\boldsymbol{q}_j\|^2\,e^{-2\lambda_j t} \;\le\; \|W^\star\|_F^2\,e^{-2\lambda_J t}. \tag{14}$$

(ii) Exact SVD alignment under commutativity. *If $W^{\star\top}W^\star$ commutes with C_i, then for every $t \ge 0$,*

$$W_t^\top W_t \;=\; \sum_{j=1}^{d}\big(1-e^{-\lambda_j t}\big)^2\,\sigma_j^{\star 2}\,\boldsymbol{q}_j\boldsymbol{q}_j^\top, \qquad \sigma_j^{\star 2} := \boldsymbol{q}_j^\top W^{\star\top}W^\star\boldsymbol{q}_j, \tag{15}$$

so the right singular vectors of W_t coincide with the eigenvectors $\{\boldsymbol{q}_j\}$ of C_i, with squared singular values $(1-e^{-\lambda_j t})^2\sigma_j^{\star 2}$.

Proof. Closed form. Let $X_t := W_t - W^\star$, so $\dot X_t = -X_t C_i$ with $X_0 = -W^\star$. The matrix-valued linear ODE has solution $X_t = X_0\,e^{-C_i t} = -W^\star\,e^{-C_i t}$, hence $W_t = W^\star(I - e^{-C_i t})$. Right-multiplying by $\boldsymbol{q}_j$ and using $C_i\boldsymbol{q}_j = \lambda_j\boldsymbol{q}_j$ gives (13).

(i) Convergence. From (13), $(W_t - W^\star)\boldsymbol{q}_j = -e^{-\lambda_j t}\,W^\star\boldsymbol{q}_j$. Squaring and summing over $j \in J$,

$$\|(W_t - W^\star)Q_J\|_F^2 \;=\; \sum_{j\in J}\|W^\star\boldsymbol{q}_j\|^2\,e^{-2\lambda_j t} \;\le\; \Big(\sum_{j\in J}\|W^\star\boldsymbol{q}_j\|^2\Big)\,e^{-2\lambda_J t} \;\le\; \|W^\star\|_F^2\,e^{-2\lambda_J t},$$

using $e^{-2\lambda_j t} \le e^{-2\lambda_J t}$ for $j \in J$.

(ii) Alignment. From the closed form, $W_t^\top W_t = (I - e^{-C_i t})\,W^{\star\top}W^\star\,(I - e^{-C_i t})$. The factor $I - e^{-C_i t}$ is a function of C_i, hence diagonal in $\{\boldsymbol{q}_j\}$ with entries $(1-e^{-\lambda_j t})$. Under the commutativity assumption $[W^{\star\top}W^\star, C_i] = 0$, both matrices are simultaneously diagonalizable in $\{\boldsymbol{q}_j\}$, so $W^{\star\top}W^\star = \sum_j \sigma_j^{\star 2}\boldsymbol{q}_j\boldsymbol{q}_j^\top$, giving (15). □

Remark 3. *The closed form* (13) *is the $\eta \to 0$ limit of SGD with step size η. Discrete-step convergence at rate $(1-\eta\lambda_K)^{2t}$ for $\eta < 1/\lambda_1$ follows by standard arguments on linear gradient dynamics [16]. Relaxing the commutativity assumption in (ii) incurs a Davis-Kahan correction [11, 79] of order $\|W_t^\top W_t - W^{\star\top}W^\star\|_F/\delta_K^\star$, where $\delta_K^\star$ is the squared-singular-value gap of $W^\star$; this perturbation itself decays exponentially at rate λ_{r^*} on the effective subspace, so the perfectly aligned conclusion of (ii) is the limit of approximate alignment statements that hold whenever the commutator $[W^{\star\top}W^\star, C_i]$ is small.*

Corollary 1 (Functional truncation error under alignment)**.** *Suppose Propositions 1 (with ϵ-effective rank r^*) and 2 (ii) hold, and let $W_t^{(K)}$ denote the rank-K SVD truncation of W_t for some $K \ge r^*$ chosen so that the top-K right singular subspace of W_t contains* $\mathrm{span}\{\boldsymbol{q}_1,\dots,\boldsymbol{q}_{r^*}\}$*. Then*

$$\mathbb{E}_{\boldsymbol{z}\sim p_i}\big\|W_t\boldsymbol{z} - W_t^{(K)}\boldsymbol{z}\big\|^2 \;\le\; \epsilon\,\|W_t\|_{\mathrm{op}}^2\,\mathrm{tr}(C_i). \tag{16}$$

Proof. By Proposition 2 (ii), the right singular vectors of W_t are exactly $\{\boldsymbol{q}_j\}_{j=1}^d$. Let $\mathcal{J}_K \subseteq [d]$ index the top-K right singular directions of W_t, so that $W_t^{(K)}\boldsymbol{z} = \sum_{j\in\mathcal{J}_K}(1-e^{-\lambda_j t})\,W^\star\boldsymbol{q}_j\,(\boldsymbol{q}_j^\top\boldsymbol{z})$.

By assumption $[r^*] \subseteq \mathcal{J}_K$, so the discarded directions all lie outside the top-r^* eigensubspace of C_i. Applying the Step-1 decomposition of Proposition 1's proof to the residual $W_t - W_t^{(K)}$,

$$\mathbb{E}\|W_t \boldsymbol{z} - W_t^{(K)} \boldsymbol{z}\|^2 \;=\; \sum_{j \notin \mathcal{J}_K} \lambda_j \, \|W_t \boldsymbol{q}_j\|^2 \;\le\; \sum_{j > r^*} \lambda_j \, \|W_t \boldsymbol{q}_j\|^2 \;\le\; \epsilon \, \|W_t\|_{\mathrm{op}}^2 \, \mathrm{tr}(C_i),$$

where the last inequality is the bound (6) of Proposition 1. □

Proposition 3 (Sample complexity for the empirical truncation error)**.** *Fix $W \in \mathbb{R}^{p \times d}$ and its rank-K SVD truncation $W^{(K)}$, and write $W^{(>K)} := W - W^{(K)}$. Suppose $\boldsymbol{z} \in \mathbb{R}^d$ is mean-zero sub-Gaussian with parameter ν, i.e. $\mathbb{E}\big[\exp(\langle \boldsymbol{u}, \boldsymbol{z} \rangle)\big] \le \exp(\nu^2 \|\boldsymbol{u}\|^2/2)$ for all $\boldsymbol{u} \in \mathbb{R}^d$. Let $\{\boldsymbol{z}_i\}_{i=1}^n$ be iid samples from p_i and define the population and empirical truncation errors*

$$\mathcal{T} \;:=\; \mathbb{E}_{\boldsymbol{z} \sim p_i} \big\| W^{(>K)} \boldsymbol{z} \big\|^2, \qquad \widehat{\mathcal{T}}_n \;:=\; \tfrac{1}{n} \sum_{i=1}^n \big\| W^{(>K)} \boldsymbol{z}_i \big\|^2,$$

together with the effective rank $d_{\mathrm{eff}}(C_i) := \mathrm{tr}(C_i)/\|C_i\|_{\mathrm{op}}$. There exists an absolute constant $c > 0$ such that for any $\delta \in (0,1)$ and any $n \ge d_{\mathrm{eff}}(C_i) + \log(2/\delta)$, with probability at least $1 - \delta$,

$$\big|\widehat{\mathcal{T}}_n - \mathcal{T}\big| \;\le\; c\,\nu^2 \, \big\|W^{(>K)}\big\|_F^2 \sqrt{\frac{d_{\mathrm{eff}}(C_i) + \log(2/\delta)}{n}}. \tag{17}$$

Proof. Write $\mathcal{T} = \mathrm{tr}\big(W^{(>K)} C_i W^{(>K)\top}\big)$ and $\widehat{\mathcal{T}}_n = \mathrm{tr}\big(W^{(>K)} \widehat{C}_n W^{(>K)\top}\big)$, where $\widehat{C}_n = \frac{1}{n}\sum_i \boldsymbol{z}_i \boldsymbol{z}_i^\top$. Using $|\mathrm{tr}(MN)| \le \|M\|_{\mathrm{op}} \|N\|_*$ with $N = W^{(>K)\top} W^{(>K)} \succeq 0$ and $\|N\|_* = \mathrm{tr}(N) = \|W^{(>K)}\|_F^2$,

$$\big|\widehat{\mathcal{T}}_n - \mathcal{T}\big| \;=\; \big|\mathrm{tr}\big((\widehat{C}_n - C_i)\, W^{(>K)\top} W^{(>K)}\big)\big| \;\le\; \big\|\widehat{C}_n - C_i\big\|_{\mathrm{op}} \big\|W^{(>K)}\big\|_F^2. \tag{18}$$

Sub-Gaussianity gives matrix-Bernstein concentration of $\widehat{C}_n$ around C_i [66, 69, 41]: there exists $c' > 0$ such that, with probability at least $1 - \delta$,

$$\big\|\widehat{C}_n - C_i\big\|_{\mathrm{op}} \;\le\; c'\,\nu^2 \left(\sqrt{\tfrac{d_{\mathrm{eff}}(C_i) + \log(2/\delta)}{n}} + \tfrac{d_{\mathrm{eff}}(C_i) + \log(2/\delta)}{n} \right).$$

For $n \ge d_{\mathrm{eff}}(C_i) + \log(2/\delta)$ the first term dominates, so $\|\widehat{C}_n - C_i\|_{\mathrm{op}} \le 2c'\,\nu^2 \sqrt{(d_{\mathrm{eff}}(C_i) + \log(2/\delta))/n}$. Substituting into (18) and absorbing the constant yields (17). □

Remark 4. *The bound* (17) *scales with $d_{\mathrm{eff}}(C_i)$, not the ambient dimension d. When C_i is sharply low-rank—empirically the case for FLAME's experts on every benchmark in Sec. D.4—we have $d_{\mathrm{eff}}(C_i) \approx r^*$, so $n = O(r^*/\varepsilon^2)$ samples suffice to estimate $\mathcal{T}$ to additive accuracy $\varepsilon \cdot \|W^{(>K)}\|_F^2$. The same low-rank structure that motivates the truncation in Proposition 1 also controls the sample complexity of validating it from finite data.*

Remark 5. *Propositions 2 and 3 are stated for the linear surrogate* (12)*; FLAME's experts add a length-preserving 1-D convolution and a position-wise MLP. NTK-regime analysis [32, 3] extends Proposition 2 to wide nonlinear networks: in the lazy regime the network behaves as a linear predictor in NTK feature space, and the alignment argument applies to the NTK feature map. We invoke both propositions as motivation for the truncation in Eq.* (3)*; the spectral plots in Fig. 3 and Sec. D.4 provide direct empirical confirmation that the conclusions hold for the nonlinear experts on every benchmark in our suite.*

B.3 Guarantees for Continual Learning

The results above concern a single expert matrix. We now state what FLAME-CL guarantees over a stream of stages: forgetting is exactly zero (Proposition 4); the rank-r_t truncation costs the task it compresses an amount equal to a functional-energy tail, paid once (Propositions 5, 6 and Remark 8); the mean final-stage risk admits a generalization bound with no forgetting term (Proposition 7); and shared-expert training transfers or interferes according to the gradient inner product on the experts two tasks share (Proposition 8). None of these results depends on the number of experts, the model width, or the backbone size.

Notation. At stage j, $W_i^{(j)}$ is the rank-r_j component reserved for expert i and Π_i the frozen stack of reserved components. Task t carries cursor $\tau = k(t)$, and its effective expert weight is $W_i^{\text{eff}}(\tau)$ (Eq. (4)). We write $f_t^{(s)}$ for the predictor of task t after stage s has completed, R_t and $\widehat{R}_t$ for its population and empirical risks under $\mathcal{D}_t$, and $S_t(\boldsymbol{x}) \subseteq [N]$ for the top-K expert support selected for sample $\boldsymbol{x}$. For readability we index each task by the stage at which it is learned, so $k(t) = t$ below.

Proposition 4 (Structural no forgetting)**.** *Suppose that (a) after stage $\tau = k(t)$, every parameter and state variable used by the forward pass of task t is frozen: all expert and encoder components with index $j \le \tau$, the router heads $\{G_m^{(\tau)}\}_{m \in \mathcal{M}_t}$, the task head $h^{(t)}$, and any normalization or preprocessing statistics; and (b) at inference, task identity determines τ, the router heads and the task head, and every compressed stack is evaluated only through components with index $j \le \tau$. Then for every stage $s \ge \tau$ and every input $\boldsymbol{x}$, $f_t^{(s)}(\boldsymbol{x}) = f_t^{(\tau)}(\boldsymbol{x})$. Hence $R_t(f_t^{(s)}) = R_t(f_t^{(\tau)})$, and any forgetting measure defined as deterioration relative to task t's post-stage-τ performance is exactly zero.*

Proof. Fix $s > \tau$; the case $s = \tau$ is trivial. We check the four objects on which the forward pass of task t depends. *Expert weights.* Stage s trains $\widetilde{W}_i^{(s)}$ and appends its rank-r_s truncation $W_i^{(s)}$ to Π_i; by (a) it modifies no $W_i^{(j)}$ with $j < s$. Task t evaluates $W_i^{\text{eff}}(\tau)$, which sums over $j \le \tau$ only, so $W_i^{(s)}$, and every block appended between stages τ and s, is absent from that sum. *Encoders.* The variable-length attention layers inside ϕ_m are compressed and stacked like the experts (Sec. 2.2), so the same cursor argument applies to each encoder stack. *Routers.* Stage s instantiates fresh heads $\{G_m^{(s)}\}$. The heads $\{G_m^{(\tau)}\}$ used by task t were frozen at the end of stage τ and are selected by task identity, not by recency. *Task head.* $h^{(t)}$ was frozen at the end of stage τ and is likewise selected by task identity. Every operation in the computation graph of $f_t^{(s)}$ is therefore identical to its counterpart in $f_t^{(\tau)}$, on every input. Pointwise equality of the two functions gives equality of risks by taking expectations under $\mathcal{D}_t$. □

Remark 6. *The argument never refers to the optimizer, the loss, the task ordering, or the data distribution of any stage. Freezing alone trivially preserves outputs; the point is that* FLAME-CL *does so while every new stage remains fully trainable. Condition (a) must hold for* every *state used by the forward pass, including buffers such as normalization running statistics that are not parameters: if such a buffer is left live, the identity fails and a small residual drift appears. Our implementation freezes each encoder's state at the stage where its modality first appears.*

Proposition 5 (The truncation error is a functional-energy tail)**.** *Let $\widetilde{W}_i^{(t)} = \sum_k \sigma_{i,k}\, \boldsymbol{u}_{i,k} \boldsymbol{v}_{i,k}^\top$ be the trained stage-t update, $W_i^{(t)}$ its rank-r_t truncation, and $\Delta_i = \widetilde{W}_i^{(t)} - W_i^{(t)} = \sum_{k > r_t} \sigma_{i,k}\, \boldsymbol{u}_{i,k} \boldsymbol{v}_{i,k}^\top$ the discarded part. Let $C_i^{(t)} = \mathbb{E}[\boldsymbol{z}\boldsymbol{z}^\top \mid i \in S_t(\boldsymbol{x})]$ be the second moment of the inputs routed to expert i at stage t. Then*

$$\varepsilon_{i,t}(r_t) \;:=\; \mathbb{E}\big[\|\Delta_i \boldsymbol{z}\|_2^2 \,\big|\, i \in S_t(\boldsymbol{x})\big] \;=\; \sum_{k > r_t} \sigma_{i,k}^2\, \boldsymbol{v}_{i,k}^\top C_i^{(t)} \boldsymbol{v}_{i,k}. \tag{19}$$

Proof. Using $\|\boldsymbol{a}\|_2^2 = \operatorname{tr}(\boldsymbol{a}\boldsymbol{a}^\top)$ and linearity of trace and expectation, $\mathbb{E}\|\Delta_i \boldsymbol{z}\|_2^2 = \mathbb{E}\operatorname{tr}(\Delta_i \boldsymbol{z}\boldsymbol{z}^\top \Delta_i^\top) = \operatorname{tr}(\Delta_i^\top \Delta_i\, C_i^{(t)})$. Orthonormality of the left singular vectors, $\boldsymbol{u}_{i,k}^\top \boldsymbol{u}_{i,l} = \delta_{kl}$, gives $\Delta_i^\top \Delta_i = \sum_{k > r_t} \sigma_{i,k}^2\, \boldsymbol{v}_{i,k} \boldsymbol{v}_{i,k}^\top$, and $\operatorname{tr}(\boldsymbol{v}\boldsymbol{v}^\top C) = \boldsymbol{v}^\top C \boldsymbol{v}$ completes the computation. □

Eq. (19) is an identity, not a bound: its summands are exactly the per-rank functional energies $\mathcal{E}_{i,k}$ of Eq. (2) evaluated for the stage-t update, so the cumulative-energy curves of Figs. 3, 15 and 23 read the truncation error off directly rather than through a proxy. $C_i^{(t)}$ must be the second moment rather than the centred covariance, since centring would leave an uncancelled mean term.

Proposition 6 (From truncation error to risk)**.** *Consider a single MoE layer whose experts are compressed as in Proposition 5, with each expert treated as its compressed linear map as in Sec. B.2. Suppose that (i) the loss $\mathcal{L}_t$ is L_t-Lipschitz in its first argument; (ii) conditional on the trained routers and all other parameters, the part of the task-t network downstream of the MoE output is Γ_t-Lipschitz; and (iii) the active gate weights are nonnegative and sum to one. Let $f_{t,\text{full}}$ be task*

t's predictor at its own stage before truncation and $f_t^{(t)}$ the stored predictor after it. Then, with $\rho_{i,t} = \mathbb{P}_{\boldsymbol{x}\sim\mathcal{D}_t}\big(i \in S_t(\boldsymbol{x})\big)$,

$$R_t\big(f_t^{(t)}\big) - R_t\big(f_{t,\mathrm{full}}\big) \;\le\; L_t\,\Gamma_t\Big[\sum_{i=1}^{N} \rho_{i,t}\,\varepsilon_{i,t}(r_t)\Big]^{1/2}. \tag{20}$$

Proof. Fix a stage-t sample $\boldsymbol{x}$ with support $S_t(\boldsymbol{x})$ and gate weights $G_i(\boldsymbol{x})$, and let $\boldsymbol{e}_i(\boldsymbol{x}) = \Delta_i \boldsymbol{z}_i(\boldsymbol{x})$ be the per-expert perturbation. The support is computed from the router summary $\bar{\boldsymbol{z}}_m$, which does not depend on the expert matrices, so it is the same in the full and compressed models, and their MoE outputs differ only by $\boldsymbol{\delta}_t(\boldsymbol{x}) = \sum_{i\in S_t(\boldsymbol{x})} G_i(\boldsymbol{x})\,\boldsymbol{e}_i(\boldsymbol{x})$. By (iii) the gate weights form a convex combination, so Jensen's inequality for $\boldsymbol{u} \mapsto \|\boldsymbol{u}\|_2^2$, followed by $G_i(\boldsymbol{x}) \le 1$, gives

$$\|\boldsymbol{\delta}_t(\boldsymbol{x})\|_2^2 \;\le\; \sum_{i\in S_t(\boldsymbol{x})} G_i(\boldsymbol{x})\,\|\boldsymbol{e}_i(\boldsymbol{x})\|_2^2 \;\le\; \sum_{i=1}^{N} \mathbf{1}\{i \in S_t(\boldsymbol{x})\}\,\|\boldsymbol{e}_i(\boldsymbol{x})\|_2^2.$$

Taking expectations and conditioning on the routing event, $\mathbb{E}\|\boldsymbol{\delta}_t(\boldsymbol{x})\|_2^2 \le \sum_i \rho_{i,t}\,\mathbb{E}\big[\|\boldsymbol{e}_i(\boldsymbol{x})\|_2^2 \mid i \in S_t(\boldsymbol{x})\big] = \sum_i \rho_{i,t}\,\varepsilon_{i,t}(r_t)$ by Proposition 5. Conditions (i) and (ii) give, pointwise, $\big|\mathcal{L}_t(f_t^{(t)}(\boldsymbol{x}), y) - \mathcal{L}_t(f_{t,\mathrm{full}}(\boldsymbol{x}), y)\big| \le L_t\Gamma_t\|\boldsymbol{\delta}_t(\boldsymbol{x})\|_2$. Taking expectations and applying $\mathbb{E}\|\boldsymbol{\delta}_t\|_2 \le (\mathbb{E}\|\boldsymbol{\delta}_t\|_2^2)^{1/2}$ yields (20). □

Remark 7. *For nonlinear experts, the Lipschitz constant of the expert computation that follows the compressed sublayer multiplies the right-hand side of* (20)*. The proof uses that truncating* expert *matrices leaves the routing support unchanged, so it does not extend as stated to compressed encoder matrices: perturbing an encoder changes $\bar{\boldsymbol{z}}_m$, the router's input, and top-K selection is discontinuous in that input, so $S_t(\boldsymbol{x})$ may change between the full and compressed models. Extending the bound would require a margin condition on the gate scores together with a term for the probability of a support change.*

Remark 8 (The cost is paid once; the stream consumes capacity)**.** *By Proposition 4, $R_t(f_t^{(s)}) = R_t(f_t^{(t)})$ for every $s \ge t$, so the compression error bounded in* (20) *is incurred at task t's own stage and does not grow as later stages arrive. What the stream consumes is capacity. Reserved components accumulate along the rank dimension and the cumulative delta has rank at most $\min\{\sum_{j\le T} r_j,\, d\}$, so later stages stop contributing fresh input directions once $\sum_j r_j$ exceeds d; a constant per-stage budget r thus admits $\lfloor d/r \rfloor$ stages of fresh directions (four at $d = 128$, $r = 32$, consistent with the stream lengths of Setups 1–4). Eq.* (20) *quantifies what is given up by lowering r to extend the stream, and the rank sweep in Tab. 19 shows that the rank actually required falls with stage index.*

Proposition 7 (Stream-level generalization)**.** *Let $\mathcal{F}_t$ be the post-compression predictor class at stage t: the stage-t compressed expert and encoder components, the routers $\{G_m^{(t)}\}$, and the head $h^{(t)}$, with all earlier components held fixed. Assume $0 \le \mathcal{L}_t \le 1$ and that the n_t stage-t samples are drawn i.i.d. from $\mathcal{D}_t$, independently of the data that determined the components with index $j < t$. Then with probability at least $1-\delta$, simultaneously for all $t \in [T]$ and all $f \in \mathcal{F}_t$,*

$$R_t(f) \;\le\; \widehat{R}_t(f) + 2\,\mathfrak{R}_{n_t}(\mathcal{L}_t \circ \mathcal{F}_t) + \sqrt{\frac{\log(T/\delta)}{2n_t}}, \tag{21}$$

and consequently

$$\frac{1}{T}\sum_{t=1}^{T} R_t\big(f_t^{(T)}\big) \;\le\; \frac{1}{T}\sum_{t=1}^{T}\Big[\widehat{R}_t\big(f_t^{(t)}\big) + 2\,\mathfrak{R}_{n_t}(\mathcal{L}_t \circ \mathcal{F}_t) + \sqrt{\tfrac{\log(T/\delta)}{2n_t}}\Big]. \tag{22}$$

Proof. Conditional on the components with index $j < t$, $\mathcal{F}_t$ is a fixed class and the stage-t sample is i.i.d. from $\mathcal{D}_t$, so the standard Rademacher uniform-deviation bound [6] gives (21) for stage t with probability at least $1 - \delta/T$. A union bound over $t \in [T]$ makes it hold for all stages simultaneously with probability at least $1-\delta$. Applying it to $f_t^{(t)} \in \mathcal{F}_t$, averaging over t, and using $R_t(f_t^{(T)}) = R_t(f_t^{(t)})$ from Proposition 4 gives (22). □

The bound on the mean final-stage risk contains no forgetting term. Compression enters only through the empirical risk of the stored predictor, which the empirical analogue of Proposition 6 controls. We do not estimate $\mathfrak{R}_{n_t}$ or claim that it is tight; its complexity cannot be reduced to a union over the $\binom{N}{K}$ possible supports, because the support is a learned, input-dependent function and the gate values are continuous. The role of (22) is structural: it shows where compression enters the risk and that nothing accumulates across stages.

Proposition 8 (Transfer and interference through shared experts)**.** *This result concerns shared-expert training rather than the cursor-isolated inference of a stored task. Hold the routers and encoders fixed, let $E = (E_1, \dots, E_N)$ denote the shared expert parameters, and let $g_k = \nabla_E R_k(E)$ and $g_{k'} = \nabla_E R_{k'}(E)$ for two tasks trained jointly, with per-sample losses ℓ_k and $\ell_{k'}$.*

(i) If there exist disjoint $S_k, S_{k'} \subseteq [N]$ that contain the routed supports of k and k' almost surely, then $\langle g_k, g_{k'} \rangle = 0$.

(ii) If $\|\nabla_E \ell_k(\boldsymbol{x})\|_2 \le G_k$ and $\|\nabla_E \ell_{k'}(\boldsymbol{x}')\|_2 \le G_{k'}$ almost surely, then $|\langle g_k, g_{k'} \rangle| \le \rho_{kk'}\, G_k\, G_{k'}$, where $\rho_{kk'} = \mathbb{P}\big(S_k(\boldsymbol{x}) \cap S_{k'}(\boldsymbol{x}') \neq \varnothing\big)$ for independent $\boldsymbol{x} \sim \mathcal{D}_k$, $\boldsymbol{x}' \sim \mathcal{D}_{k'}$.

(iii) If R_k has a β_k-Lipschitz gradient in E on the segment between E and $E^+ = E - \eta\, g_{k'}$, then $R_k(E^+) \le R_k(E) - \eta\, \langle g_k, g_{k'} \rangle + \frac{\beta_k \eta^2}{2}\, \|g_{k'}\|_2^2$.

Proof. Write gradients blockwise, $g_k = (\nabla_{E_1} R_k, \dots, \nabla_{E_N} R_k)$. Under sparse routing, a sample's loss has no functional dependence on an expert outside its support, so $\nabla_{E_i} \ell_k(\boldsymbol{x}) = 0$ for $i \notin S_k(\boldsymbol{x})$. (i) It follows that $\nabla_{E_i} R_k = 0$ for $i \notin S_k$ and $\nabla_{E_i} R_{k'} = 0$ for $i \notin S_{k'}$. Since $S_k \cap S_{k'} = \varnothing$, at least one of the two blocks vanishes for every i, so $\langle g_k, g_{k'} \rangle = \sum_i \langle \nabla_{E_i} R_k, \nabla_{E_i} R_{k'} \rangle = 0$. Equivalently, $\frac{d}{d\eta} R_k(E - \eta g_{k'})\big|_{\eta=0} = -\langle g_k, g_{k'} \rangle = 0$: an infinitesimal step on task k' has no first-order effect on R_k through the experts. (ii) For independent $\boldsymbol{x}, \boldsymbol{x}'$, $\langle g_k, g_{k'} \rangle = \mathbb{E}\big[\langle \nabla_E \ell_k(\boldsymbol{x}), \nabla_E \ell_{k'}(\boldsymbol{x}') \rangle\big]$, and by the blockwise argument of (i) the integrand vanishes unless $S_k(\boldsymbol{x}) \cap S_{k'}(\boldsymbol{x}') \neq \varnothing$. Cauchy–Schwarz on the remaining event gives $|\langle g_k, g_{k'} \rangle| \le \mathbb{E}\big[\mathbf{1}\{S_k(\boldsymbol{x}) \cap S_{k'}(\boldsymbol{x}') \neq \varnothing\}\big]\, G_k G_{k'} = \rho_{kk'}\, G_k G_{k'}$, which recovers (i) at $\rho_{kk'} = 0$. (iii) β_k-smoothness gives $R_k(E + \Delta) \le R_k(E) + \langle g_k, \Delta \rangle + \frac{\beta_k}{2} \|\Delta\|_2^2$ for any increment Δ; substitute $\Delta = -\eta\, g_{k'}$. □

Remark 9. *By (iii), if $\langle g_k, g_{k'} \rangle > 0$, a step on task k' with $0 < \eta < 2\langle g_k, g_{k'} \rangle / (\beta_k \|g_{k'}\|_2^2)$ strictly decreases R_k (transfer); if the inner product is negative, the step is first-order interfering. Routing overlap therefore determines whether a pathway for direct expert-level interaction exists, but not its sign, which is set locally by the gradient inner product on the shared blocks. The cosine similarity between average routing fingerprints, which $\mathcal{L}_{\text{div}}^{(k)}$ regulates and Fig. 7 reflects, indicates an* opportunity *for sharing rather than a guarantee of positive transfer. The argument covers direct interaction through the shared experts only; disjoint expert supports do not preclude interaction through auxiliary balancing losses or jointly updated statistics.*

C Dataset Details

We provide additional details on the four healthcare datasets used in our experiments, including cohort construction, label distributions, modality preprocessing, and train/validation/test splits. All datasets are publicly available under their respective data use agreements.

MIMIC-IV. MIMIC-IV [37] is a de-identified critical care database collected at the Beth Israel Deaconess Medical Center, a single tertiary academic medical center. We follow the cohort construction and preprocessing pipeline of FuseMoE [24], which adapts the multi-task ICU benchmark of [25] to MIMIC-IV. ICU stays are restricted to adult patients and the first stay per admission. Vital signs and laboratory measurements are assembled into an irregularly sampled multivariate time series and discretized for the multi-time attention encoder. Clinical notes within the prediction window are concatenated and tokenized for a clinical text encoder. Chest X-rays linked through MIMIC-CXR [36] are matched at the stay level; stays missing imaging are retained with the modality marked absent. *48-IHM* predicts in-hospital mortality from the first 48 hours of the stay, *LOS* predicts whether the remaining stay exceeds seven days, and *25-PHENO* predicts a 25-way multi-label phenotype indicator over acute care conditions. We adopt the standard 70/10/20 train/validation/test split with patient-level disjointness.

eICU. The eICU Collaborative Research Database (v2.0) [55] comprises de-identified records from 208 U.S. hospitals and $\sim$139,367 unique patients, providing inter-institutional variation in coding practices and care patterns that complements MIMIC-IV's single-site cohort and exercises the routing under realistic distributional shift. We mirror the MIMIC-IV preprocessing pipeline for vital and laboratory time series and clinical notes. *MOR* predicts in-ICU mortality and *RAD* predicts 30-day all-cause readmission; both tasks use the standard cohort filters (adult patients, length of stay ≥ 24 hours) and a 70/10/20 patient-level split.

EMBED. The Emory Breast Imaging Dataset [34] contains 3.4M screening and diagnostic mammographic images across a racially diverse population. Each breast study yields four image streams—FFDM CC, FFDM MLO, synthesized 2D C-View CC, and synthesized 2D C-View MLO—which are resized and normalized for a shared image encoder. Tabular metadata (demographics, prior history) is one-hot/standardized. *BIRADS* is a multi-class classification over BI-RADS final assessment categories, *RISK* is a binary short-term cancer risk label, and *DENSITY* is the multi-class ACR breast density category. Splits are constructed at the patient level to prevent leakage across breast laterality.

ADNI. The Alzheimer's Disease Neuroimaging Initiative cohort [73] provides up to five modalities per subject: structural MRI, FDG-PET, genomic SNP profiles, biospecimen assays (CSF/blood biomarkers), and tabular clinical and cognitive assessments. Following Flex-MoE [80], we form modality-stratified subsets ranging from 757 subjects with all five modalities to 5,073 subjects with the most populous subset, and use the same five-modality combination at training time with the rest of the modalities marked absent. *DIAG* is a three-way classification across Cognitively Normal (CN), Mild Cognitive Impairment (MCI), and Alzheimer's Dementia (AD), with a 70/10/20 subject-level split.

D Additional Experimental Results

This section supplements the main-text experiments with full per-task tables, ablations, an additional CL baseline, spectral diagnostics, and routing visualizations. Subsections mirror the four research questions of Sec. 3: Sec. D.2 expands on multitask pretraining (RQ1, RQ2), Sec. D.3 on continual learning (RQ3, RQ4), and Secs. D.4, D.5 provide the spectral and routing evidence that explains the multitask gains and the parameter savings (RQ2, RQ4).

D.1 Hardware, Implementation, and Hyperparameters

All experiments are run on $8\times$ NVIDIA A5500 GPUs with 24 GB memory per GPU. Tab. 4 lists the hyperparameters shared by the multitask and continual experiments; the full configurations are in the released code. We use a single learning rate for every trainable parameter (encoders, MoE, heads) and do not tune it per stage, per task, or per module. The weight decay is large in absolute terms but standard in effect: AdamW [50] applies decoupled decay, so the per-step shrinkage is $\eta\lambda = 10^{-4}$, the same as, e.g., $\eta = 10^{-3}$ with $\lambda = 0.1$. Each optimizer step sums the per-task losses of the stage with equal weight.

Table 4: Hyperparameters shared by all experiments.

Setting	Value
Optimizer	AdamW [50]
Learning rate η	10^{-4} (all trainable parameters: encoders, MoE, heads)
Weight decay λ	1.0 (decoupled; per-step shrinkage $\eta\lambda = 10^{-4}$)
Max epochs	20 per stage, early stopping with patience 3
Batch size (train)	MIMIC-IV 8, eICU 128, EMBED 512, ADNI 64
Experts N / top-K	5 / 2
Gating	Laplace
Reserved rank r_t	32
Router-balancing loss weight λ_{bal}	0.01
Interaction loss weight λ_{int}	0.1

Timestamp-generated seeds. To rule out favorable seed selection, we re-ran joint training on all nine tasks and continual Setup 4 with three seeds generated from timestamps (Tab. 5); all continual comparisons in Fig. 6a and Tabs. 15, 16 use these re-runs. The seed standard deviation is at most

0.01 on eight of nine tasks (0.03 on DIAG, the smallest cohort), and the joint numbers are close to the three-seed results of Tab. 2, with the new seeds higher on RISK and DIAG. The continual model matches or exceeds the jointly trained model on four of the eight shared tasks despite never seeing them simultaneously.

Table 5: AUROC over three timestamp-generated seeds (mean ± std). Joint: all nine tasks trained together. CL: Setup 4 ({PHENO, DENSITY}→{LOS, BIRADS, MOR}→{IHM, RISK, RAD}) evaluated after the final stage; DIAG is not part of Setup 4.

Model	48-IHM	LOS	25-PHENO	MOR	RAD	BIRADS	RISK	DENSITY	DIAG
FLAME (joint)	0.81±0.00	0.81±0.01	0.72±0.00	0.84±0.00	0.76±0.00	0.80±0.01	0.75±0.01	0.91±0.01	0.77±0.03
FLAME-CL (final)	0.81±0.01	0.82±0.00	0.71±0.00	0.82±0.00	0.73±0.00	0.81±0.00	0.74±0.00	0.91±0.01	N/A

D.2 Detailed Multitask Pretraining Results

D.2.1 Full Per-Task Performance

Tab. 6 extends the main-text per-task tables (Tabs. 2a, 2b) along three axes: (i) it adds F1 and accuracy alongside AUROC and AUPRC, (ii) it adds FuseMoE [24] as a fourth single-task baseline, and (iii) it reports both the per-dataset multitask FLAME run and the all-nine-task multitask FLAME run alongside the single-task numbers. Across all four metrics, FLAME's single-task numbers track HighMMT within 0.005 AUROC on six of nine tasks and exceed it on AUPRC for the imbalanced binary tasks, confirming that the AUROC story in the main text is not a metric-specific artifact.

Single-task baselines. FLAME trails HighMMT by 0.011 to 0.014 AUROC on LOS and RAD but exceeds it on ADNI DIAG by 0.045 (0.788 vs 0.743). On AUPRC FLAME wins on the imbalanced binary tasks where positives are rare: 48-IHM (0.492 vs 0.480), RISK (0.153 vs 0.142), and DIAG (0.660 vs 0.592); the win on the imbalanced tail is the architectural signature of modality-conditioned dispatch sharpening the boundary on rare positives.

Joint training is task-dependent. Per-dataset MIMIC-IV multitask raises 48-IHM AUROC from 0.808 to 0.817 and LOS to 0.821 but drops 48-IHM AUPRC by 0.042, indicating that the AUROC gain comes from re-ranking near the operating threshold rather than from improved positive recall. Training all nine tasks at once preserves the binary-task AUROC gains and gives back 0.040 AUROC on DIAG, the only task whose modality pool (MRI, FDG-PET, SNP, biospecimen, clinical) shares no encoder with any other. Tasks that share modalities pay no measurable cost for joint training; tasks with disjoint modality combinations regress modestly when squeezed into a single shared expert pool.

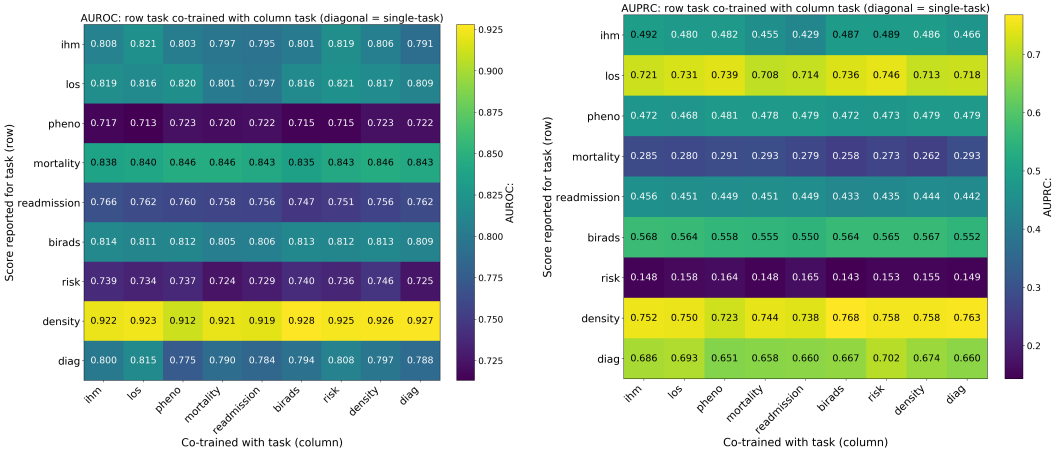
Table 6: Per-task performance (mean ± std across 3 seeds, population std). Multi-class/multi-label tasks use macro-averaged AUROC/AUPRC/F1; PHENO accuracy is Hamming. We fix number of experts to 5 for all the MoE methods. Ablation on number of experts is provided in Figure 8.

Metric	Setting	Method	MIMIC IV			eICU		EMBED			ADNI
			48-IHM	LOS	25-PHENO	MOR	RAD	BIRADS	RISK	DENSITY	DIAG
AUROC	Single Task	HighMMT	0.809 ± 0.003	0.830 ± 0.004	0.724 ± 0.002	0.850 ± 0.001	0.767 ± 0.003	0.813 ± 0.002	0.733 ± 0.004	0.927 ± 0.001	0.743 ± 0.021
		FuseMoE	0.798 ± 0.008	0.819 ± 0.004	0.706 ± 0.002	0.846 ± 0.003	0.763 ± 0.001	0.802 ± 0.001	0.738 ± 0.001	0.920 ± 0.003	0.747 ± 0.028
		FlexMoE	0.789 ± 0.009	0.817 ± 0.004	0.722 ± 0.001	0.849 ± 0.003	0.768 ± 0.003	0.809 ± 0.003	0.735 ± 0.003	0.925 ± 0.000	0.702 ± 0.025
		FLAME	0.808 ± 0.001	0.816 ± 0.005	0.723 ± 0.001	0.846 ± 0.002	0.756 ± 0.006	0.813 ± 0.002	0.736 ± 0.001	0.926 ± 0.001	0.788 ± 0.023
	Multitask (per dataset)	FLAME	0.817 ± 0.005	0.821 ± 0.007	0.719 ± 0.012	0.843 ± 0.005	0.758 ± 0.001	0.811 ± 0.002	0.738 ± 0.005	0.927 ± 0.000	–
	Multitask (all datasets)	FLAME	0.815 ± 0.002	0.821 ± 0.006	0.719 ± 0.006	0.835 ± 0.005	0.758 ± 0.001	0.791 ± 0.007	0.714 ± 0.011	0.915 ± 0.007	0.748 ± 0.006
AUPRC	Single Task	HighMMT	0.480 ± 0.019	0.748 ± 0.008	0.485 ± 0.004	0.300 ± 0.007	0.459 ± 0.003	0.563 ± 0.001	0.142 ± 0.002	0.765 ± 0.003	0.592 ± 0.028
		FuseMoE	0.444 ± 0.014	0.737 ± 0.003	0.455 ± 0.003	0.293 ± 0.007	0.449 ± 0.003	0.550 ± 0.003	0.151 ± 0.008	0.755 ± 0.008	0.601 ± 0.040
		FlexMoE	0.445 ± 0.020	0.726 ± 0.010	0.484 ± 0.002	0.300 ± 0.006	0.460 ± 0.003	0.567 ± 0.002	0.142 ± 0.003	0.756 ± 0.004	0.549 ± 0.022
		FLAME	0.492 ± 0.009	0.731 ± 0.007	0.481 ± 0.002	0.293 ± 0.006	0.449 ± 0.011	0.564 ± 0.003	0.153 ± 0.010	0.758 ± 0.002	0.660 ± 0.039
	Multitask (per dataset)	FLAME	0.450 ± 0.023	0.731 ± 0.012	0.471 ± 0.016	0.279 ± 0.004	0.451 ± 0.003	0.560 ± 0.008	0.151 ± 0.005	0.764 ± 0.002	–
	Multitask (all datasets)	FLAME	0.470 ± 0.018	0.743 ± 0.008	0.471 ± 0.004	0.271 ± 0.008	0.451 ± 0.001	0.542 ± 0.010	0.148 ± 0.012	0.743 ± 0.011	0.611 ± 0.018
F1	Single Task	HighMMT	0.386 ± 0.078	0.722 ± 0.005	0.354 ± 0.013	0.310 ± 0.036	0.432 ± 0.043	0.501 ± 0.006	0.150 ± 0.039	0.700 ± 0.008	0.549 ± 0.014
		FuseMoE	0.323 ± 0.053	0.733 ± 0.010	0.326 ± 0.002	0.304 ± 0.015	0.468 ± 0.006	0.518 ± 0.013	0.187 ± 0.035	0.694 ± 0.008	0.520 ± 0.031
		FlexMoE	0.324 ± 0.103	0.702 ± 0.012	0.370 ± 0.013	0.188 ± 0.041	0.443 ± 0.027	0.510 ± 0.017	0.120 ± 0.015	0.702 ± 0.004	0.370 ± 0.123
		FLAME	0.405 ± 0.028	0.710 ± 0.023	0.316 ± 0.017	0.229 ± 0.029	0.388 ± 0.035	0.506 ± 0.018	0.171 ± 0.009	0.704 ± 0.002	0.576 ± 0.026
	Multitask (per dataset)	FLAME	0.321 ± 0.106	0.717 ± 0.055	0.306 ± 0.008	0.147 ± 0.053	0.398 ± 0.028	0.533 ± 0.011	0.145 ± 0.009	0.701 ± 0.003	–
	Multitask (all datasets)	FLAME	0.209 ± 0.050	0.704 ± 0.030	0.292 ± 0.001	0.164 ± 0.050	0.431 ± 0.012	0.489 ± 0.019	0.096 ± 0.014	0.677 ± 0.012	0.478 ± 0.065
Accuracy	Single Task	HighMMT	0.853 ± 0.008	0.753 ± 0.004	0.768 ± 0.008	0.954 ± 0.004	0.865 ± 0.001	0.729 ± 0.018	0.743 ± 0.145	0.760 ± 0.001	0.556 ± 0.015
		FuseMoE	0.845 ± 0.003	0.738 ± 0.006	0.765 ± 0.004	0.954 ± 0.003	0.860 ± 0.002	0.702 ± 0.021	0.842 ± 0.075	0.734 ± 0.010	0.538 ± 0.037
		FlexMoE	0.846 ± 0.002	0.739 ± 0.007	0.770 ± 0.001	0.959 ± 0.000	0.866 ± 0.002	0.737 ± 0.010	0.662 ± 0.071	0.756 ± 0.001	0.474 ± 0.048
		FLAME	0.857 ± 0.001	0.723 ± 0.016	0.774 ± 0.000	0.958 ± 0.002	0.863 ± 0.001	0.736 ± 0.026	0.839 ± 0.013	0.757 ± 0.000	0.601 ± 0.022
	Multitask (per dataset)	FLAME	0.855 ± 0.011	0.749 ± 0.016	0.770 ± 0.004	0.958 ± 0.000	0.865 ± 0.002	0.734 ± 0.037	0.772 ± 0.027	0.758 ± 0.000	–
	Multitask (all datasets)	FLAME	0.852 ± 0.005	0.743 ± 0.011	0.770 ± 0.006	0.959 ± 0.000	0.867 ± 0.000	0.666 ± 0.083	0.516 ± 0.153	0.729 ± 0.022	0.516 ± 0.049

D.2.2 Pairwise Task Interactions

Fig. 7 reports the pairwise multitask-gain matrix that grounds the main-text RQ2 discussion. Each row corresponds to a focal task; each column gives a co-trained partner; cell (t, t') is the AUROC (left) or AUPRC (right) of task t when FLAME is jointly trained on the pair $\{t, t'\}$. Two structural

patterns are visible. *Within-dataset stability*: the EMBED block (BIRADS, RISK, DENSITY) is uniformly bright, with DENSITY above 0.92 AUROC for every column it appears in, indicating that mammography representations transfer readily within the dataset and resist interference from unrelated co-trained tasks. *Cross-dataset asymmetry*: the IHM row shifts by 0.02–0.03 depending on the partner, with the largest gain coming from co-training with LOS, while ADNI's DIAG (whose modality pool is disjoint from every other task) regresses modestly under most pairings. Tasks pair well precisely when their per-modality routers (Sec. D.5.2) overlap; the heatmap therefore serves as a data-driven recommender for which task pairs should share a model.



(a) AUROC (b) AUPRC

Figure 7: Pairwise task performance. Left: AUROC, Right: AUPRC. Each row represents the focal task's performance when trained jointly with the partner task on the column.

D.2.3 Number-of-Experts Ablation

We sweep the shared expert pool size $N \in \{2, 4, 8, 16, 32\}$ with FLAME jointly trained on all nine tasks at seed 42, and add $N \in \{5, 6, 7\}$ to the AUROC sweep to resolve the range around the default. Fig. 8 shows the AUROC and AUPRC trajectories; Tabs. 7, 8 give the numbers per task. Within $N \in [4, 8]$ the per-task best is spread across pool sizes with no systematic trend, so we fixed $N{=}5$ for all main experiments on this basis rather than by selecting the per-task maximum. Performance is largely insensitive to N for $N \geq 4$: per-task AUROC moves by less than 0.02 across the entire sweep on every task except 48-IHM (where $N{=}16$ regresses but $N{=}32$ recovers). The sweet spot is $N \in \{4, 8\}$, balancing capacity against per-modality router collapse risk; all main-text experiments use $N{=}5$. The flatness of the curve indicates that FLAME's gains do not rest on a particular pool size: the per-modality dispatch already concentrates each modality on a small expert subset, so adding more experts saturates rather than shifts the routing.

Table 7: Ablation over number of experts (AUROC; seed=42; FLAME jointly trained on all 9 tasks). Multi-class/multi-label tasks use macro-averaged AUROC.

# Experts	MIMIC IV			eICU		EMBED			ADNI
	48-IHM	LOS	25-PHENO	MOR	RAD	BIRADS	RISK	DENSITY	DIAG
2	0.787	0.814	0.723	0.843	0.759	0.785	0.690	0.917	0.751
4	0.809	0.821	0.718	0.835	0.758	0.802	0.727	0.917	0.788
5	0.815	0.821	0.719	0.835	0.759	0.810	0.714	0.920	0.754
6	0.795	0.810	0.717	0.839	0.756	0.773	0.749	0.919	0.770
7	0.804	0.826	0.713	0.836	0.757	0.804	0.756	0.913	0.739
8	0.811	0.813	0.716	0.831	0.760	0.770	0.680	0.909	0.774
16	0.760	0.791	0.710	0.834	0.754	0.783	0.672	0.904	0.782
32	0.812	0.814	0.710	0.842	0.759	0.798	0.714	0.900	0.725

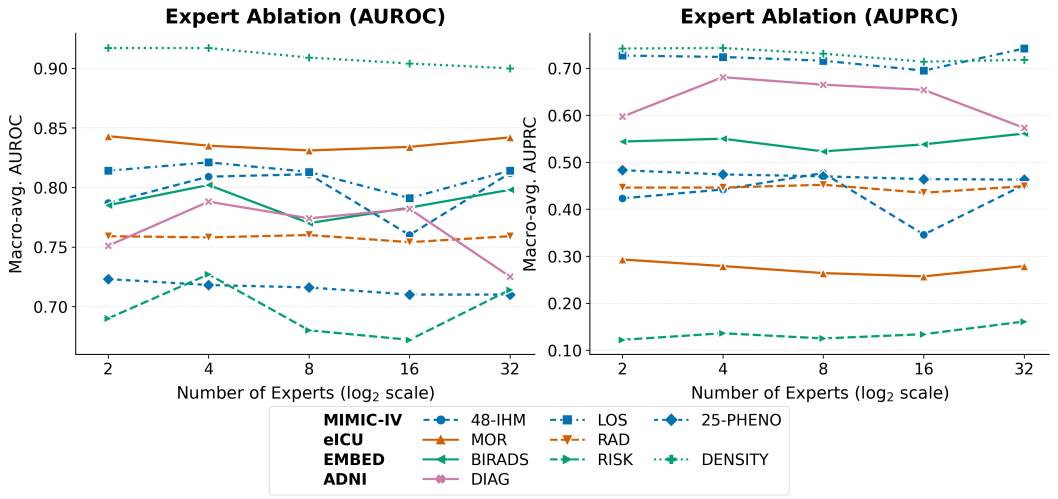


Figure 8: Ablation over number of experts; Left: AUROC, Right: AUPRC. FLAME jointly trained on all 9 tasks. Multi-class/multi-label tasks use macro-averaged AUROC and AUPRC.

Table 8: Ablation over number of experts (AUPRC; seed=42; FLAME jointly trained on all 9 tasks). Multi-class/multi-label tasks use macro-averaged AUPRC.

# Experts	MIMIC IV			eICU		EMBED			ADNI
	48-IHM	LOS	25-PHENO	MOR	RAD	BIRADS	RISK	DENSITY	DIAG
2	0.423	0.727	0.483	0.293	0.446	0.544	0.122	0.742	0.597
4	0.442	0.724	0.474	0.279	0.446	0.550	0.136	0.743	0.681
8	0.477	0.716	0.470	0.264	0.452	0.523	0.125	0.731	0.665
16	0.346	0.695	0.464	0.257	0.435	0.538	0.134	0.714	0.654
32	0.452	0.742	0.463	0.279	0.449	0.561	0.161	0.718	0.573

D.2.4 Gating-Function Ablation

Tab. 9 compares Laplace [24], Gaussian, and Softmax [63] gating under three regimes (single-task, per-dataset multitask, all-nine-task multitask) at $N{=}8$ and seed 42. The three gates are within 0.01 AUROC and 0.02 AUPRC of one another on every task and every regime, and no gate wins uniformly. The methodological implication is sharp: FLAME's parameter-efficiency and continual-retention gains do not come from the gating non-linearity but from the structural choices around it—per-modality routing, sample-level dispatch via TAP, and rank-reserved spectral compression. Replacing the gate is essentially a no-op; replacing any of the structural pieces is not.

D.2.5 Auxiliary-Loss Weight Sensitivity

We sweep the two auxiliary weights in $\mathcal{L} = \mathcal{L}_{\text{task}} + \lambda_{\text{bal}}\,\mathcal{L}_{\text{bal}} + \lambda_{\text{int}}\,\mathcal{L}_{\text{div}}$ on joint nine-task training with three seeds (42, 20492, 28411), varying one weight with the other at its default (Tab. 10). Mean AUROC stays within 0.004 as either weight varies from 0 to 1, and this holds per task: no task varies by more than 0.014 across configurations except DIAG (0.036; per-task values in Tabs. 11, 12). FLAME's gains therefore come from its structural choices (per-modality routing, sample-level dispatch, rank reservation) rather than from tuned auxiliary weights, consistent with the gating (Tab. 9) and expert-count (Fig. 8) ablations. The interaction loss mainly controls *which* experts modalities share (the sign of β in Sec. 2.1), shaping the routing structure behind RQ2 rather than per-task accuracy.

D.3 Detailed Continual Learning Results

D.3.1 Walkthrough Against Standard CL Baselines

This subsubsection unpacks the continual-learning headline of Sec. 3.2 into per-method and per-stage detail.

Table 9: Gating-style ablation (FLAME, 8 experts, seed=42). Rows are tasks grouped by training setup: Single Task trains IHM alone; Per-dataset (MIMIC IV) trains IHM, LOS, PHENO jointly; All Datasets trains all 9 tasks jointly. Multi-class/multi-label tasks (PHENO, BIRADS, DENSITY, DIAG) use macro-averaged AUROC/AUPRC.

Setup	Task	AUROC			AUPRC		
		Laplace	Gaussian	Softmax	Laplace	Gaussian	Softmax
Single Task	48-IHM	0.794	0.797	0.802	0.442	0.461	0.433
Per-dataset (MIMIC IV)	48-IHM	0.805	0.804	0.810	0.419	0.426	0.443
	LOS	0.819	0.806	0.809	0.719	0.709	0.714
	25-PHENO	0.710	0.711	0.714	0.467	0.468	0.472
All Datasets	48-IHM	0.811	0.816	0.817	0.477	0.487	0.487
	LOS	0.813	0.817	0.815	0.716	0.721	0.705
	25-PHENO	0.716	0.717	0.719	0.470	0.472	0.475
	RAD	0.760	0.759	0.758	0.452	0.453	0.455
	MOR	0.831	0.841	0.843	0.264	0.274	0.271
	BIRADS	0.770	0.779	0.792	0.523	0.519	0.551
	RISK	0.680	0.732	0.729	0.125	0.126	0.143
	DENSITY	0.909	0.914	0.900	0.731	0.739	0.711
	DIAG	0.774	0.803	0.755	0.665	0.691	0.603

Table 10: Auxiliary-loss weight sweep: mean AUROC over the nine tasks (joint training, 3 seeds, mean $\pm$ std). † marks the default used throughout.

λ_{bal} (with $\lambda_{\text{int}} = 0.1$)	0	0.001	$0.01^\dagger$	0.1	1
Mean AUROC	0.795 ± 0.001	0.799 ± 0.002	0.798 ± 0.005	0.795 ± 0.003	0.795 ± 0.003
λ_{int} (with $\lambda_{\text{bal}} = 0.01$)	0	0.01	$0.1^\dagger$	1	
Mean AUROC	0.796 ± 0.004	0.795 ± 0.002	0.798 ± 0.005	0.798 ± 0.002	

Table 11: Per-task AUROC under the router-balancing loss weight sweep (joint nine-task training, $\lambda_{\text{int}} = 0.1$; mean $\pm$ std over seeds 42, 20492, 28411). † default. Bold: best weight per task. Range: max $-$ min of the task mean across this sweep.

		λ_{bal}					
Dataset	Task	0	0.001	$0.01^\dagger$	0.1	1	Range
MIMIC-IV	48-IHM	0.801 ± 0.013	0.806 ± 0.009	0.809 ± 0.004	0.806 ± 0.007	$\mathbf{0.815 \pm 0.005}$	0.013
	LOS	0.812 ± 0.007	0.808 ± 0.003	$\mathbf{0.817 \pm 0.002}$	0.812 ± 0.005	0.811 ± 0.007	0.009
	25-PHENO	0.715 ± 0.003	$\mathbf{0.718 \pm 0.001}$	0.718 ± 0.001	0.716 ± 0.002	0.712 ± 0.001	0.006
eICU	MOR	0.838 ± 0.007	0.839 ± 0.007	0.839 ± 0.005	0.837 ± 0.002	$\mathbf{0.839 \pm 0.006}$	0.003
	RAD	0.759 ± 0.003	$\mathbf{0.760 \pm 0.002}$	0.759 ± 0.001	0.754 ± 0.006	0.758 ± 0.005	0.006
EMBED	BIRADS	$\mathbf{0.809 \pm 0.004}$	0.806 ± 0.001	0.804 ± 0.003	0.804 ± 0.003	0.803 ± 0.003	0.006
	RISK	0.747 ± 0.004	0.752 ± 0.004	0.752 ± 0.006	$\mathbf{0.753 \pm 0.006}$	0.750 ± 0.005	0.006
	DENSITY	0.914 ± 0.012	0.914 ± 0.003	0.915 ± 0.003	0.910 ± 0.006	$\mathbf{0.916 \pm 0.001}$	0.006
ADNI	DIAG	0.755 ± 0.033	$\mathbf{0.788 \pm 0.020}$	0.771 ± 0.044	0.765 ± 0.026	0.751 ± 0.029	0.036
Mean (9 tasks)		0.795 ± 0.001	$\mathbf{0.799 \pm 0.002}$	0.798 ± 0.005	0.795 ± 0.003	0.795 ± 0.003	0.004

Setup. We compare FLAME-CL against simple fine-tuning (SIMPLE FT), Elastic Weight Consolidation (EWC), and Low-Rank Adaptation (LORA) on four task sequences that span single-task and multitask stages over MIMIC-IV, eICU, and EMBED (Setups 1–4 in Fig. 5). SIMPLE FT updates the full encoder in place at every stage; EWC adds a Fisher-weighted regularizer; LORA freezes the base and stacks a fresh rank-32 adapter on the MoE experts per stage; FLAME-CL trains a fresh additive component per stage and freezes a rank-r truncation of it on completion (Sec. 2.2).

Retention. Tab. 13 gives the numbers behind Fig. 5. Among the methods of Fig. 5, FLAME-CL has the highest AA on every sequence, exceeding LORA by 0.015–0.029 and SIMPLE FT and EWC by 0.01–0.05, with zero forgetting after each task's one-time truncation. SIMPLE FT and EWC forget 0.03–0.06 on PHENO and MOR and 0.13–0.16 on DENSITY, where FLAME-CL beats them by 0.11–0.16. LORA retains Setups 1–3 because its frozen base shields prior-task weights, but in Setup 4 it loses DENSITY (0.92→0.82) and PHENO (0.71→0.69; F=0.026), and it is less plastic on new tasks

Table 12: Per-task AUROC under the interaction loss weight sweep (joint nine-task training, $\lambda_{\text{bal}} = 0.01$; mean $\pm$ std over seeds 42, 20492, 28411). † default. Bold: best weight per task. Range: max $-$ min of the task mean across this sweep.

		λ_{int}				
Dataset	Task	0	0.01	0.1†	1	Range
MIMIC-IV	48-IHM	0.806±0.006	0.808±0.004	0.809±0.004	**0.816±0.006**	0.010
	LOS	0.814±0.008	0.808±0.003	0.817±0.002	**0.817±0.006**	0.009
	25-PHENO	0.713±0.002	**0.718±0.001**	0.718±0.001	0.712±0.003	0.006
eICU	MOR	0.835±0.005	0.839±0.005	0.839±0.005	**0.840±0.006**	0.005
	RAD	0.757±0.002	0.757±0.006	0.759±0.001	**0.762±0.004**	0.006
EMBED	BIRADS	0.804±0.005	0.801±0.004	0.804±0.003	**0.806±0.002**	0.006
	RISK	0.747±0.004	0.749±0.009	**0.752±0.006**	0.748±0.001	0.005
	DENSITY	**0.917±0.003**	0.910±0.009	0.915±0.003	0.915±0.007	0.007
ADNI	DIAG	0.770±0.027	0.762±0.028	**0.771±0.044**	0.762±0.031	0.009
Mean (9 tasks)		0.796±0.004	0.795±0.002	**0.798±0.005**	0.798±0.002	0.003

(RAD 0.68 vs. 0.74 in Setup 2, BIRADS 0.74 vs. 0.81 in Setup 3, LOS 0.79 vs. 0.82 in Setup 1). FLAME-CL's truncation cost is at most 0.01 except on Setup 3's DENSITY (0.926→0.902).

Per-stage parameter footprint. The small panels of Fig. 5 report parameters at inference. In Setup 4, SIMPLE FT, EWC and LORA all keep the full 153.0M encoder, whereas FLAME-CL grows from 51.4M to 67.4M; Setup 1 is similar (49.6M→65.1M vs. 150.6M), and the small eICU encoder of Setup 2 leaves almost no gap (1.71M→1.79M vs. 1.84M). The MoE budget grows for both adapting methods: FLAME-CL adds about 0.17M MoE parameters per stage (0.19M to 0.56M in Setup 4), versus 0.43M to 0.80M for LORA. Counting only the parameters each method trains (the 5–15× ratio quoted in the main text), FLAME-CL's encoder components grow from 7.99M to 23.96M in Setup 4, 4.6–14× fewer than the 109M that SIMPLE FT and EWC update in place, while LORA trains none in the encoder.

Spectral hypothesis link. Proposition 1 predicts this efficiency: a trained expert's functional energy concentrates in a low-rank subspace because the routed-input distribution is itself low-rank. Sec. D.4 confirms it: across every task and every joint configuration the data-aware spectrum saturates well before rank 40 of an ambient 128 while the weight-only spectrum stays close to full rank. The compression in Eq. (3) therefore removes only directions that contribute negligibly to the expert's output on the task it was trained for.

D.3.2 LoRA on All Layers

The LORA baseline of Fig. 5 places rank-32 adapters on the MoE experts only. As an ablation, we also apply them to the modality encoders (Fig. 9, Tab. 14; seed 42). Freezing the base and stacking a fresh adapter on every layer at each stage prevents forgetting by design, so this variant serves as an upper reference for retention rather than as a competing CL method. FLAME-CL matches it: both have F=0 on every sequence, and AA agrees within 0.01 (0.777 vs. 0.785, 0.782 vs. 0.783, 0.820 vs. 0.819 and 0.794 vs. 0.800 on Setups 1–4).

Why not use LoRA on every layer, then? Its footprint grows with the task stream at a rank fixed in advance on every layer. In Setups 1 and 4, 124–125M encoder parameters are attributable to it, against 23–24M for FLAME-CL (5.2–5.3× fewer), and it holds 166–169M encoder parameters at inference against 65–67M. FLAME-CL instead keeps only the directions that each stage's routed inputs use (Proposition 1), so its per-stage budget is set by the data: the rank needed falls with stage index, halving r from 32 to 16 costs at most 0.01 AUROC (Tab. 19), and with no new expert capacity a later task still reuses earlier components (IHM reaches 0.77 at r=0).

D.3.3 MoE-Based and Distillation Baselines

Adapting SAME and TRGE. Both methods were proposed for LoRA experts on a frozen pretrained vision-language backbone, which cannot ingest our modality combinations. To isolate the CL mechanism from these architectural differences, we port each onto FLAME's MoE backbone. For SAME [75] we keep its prescribed routing (dense during training, sparse at test, no load-balancing

Table 13: Per-stage numbers behind Fig. 5: AUROC when first learned → after the final stage (seed 42). Stage-K tasks report one value; drops of at least 0.02 in bold. For FLAME-CL the first value is before rank-32 truncation, so its deltas are the one-time truncation cost; F is measured against the deployed (post-truncation) model. Parameter rows give inference-time totals after the first → last stage; LoRA adapts only the MoE and keeps the full encoder.

Task (stage)	Simple FT	EWC	LoRA	FLAME-CL
Setup 1: Pheno → Los → Ihm				
PHENO (s1)	0.716 → **0.670**	0.716 → **0.668**	0.716 → 0.716	0.714 → 0.703
LOS (s2)	0.820 → 0.800	0.825 → 0.805	0.787 → 0.787	0.822 → 0.821
IHM (s3)	0.816	0.814	0.784	0.809
AA ↑	0.762	0.762	0.762	**0.777**
F ↓	0.032	0.034	0.000	0.000
Enc. params, at inference (M)	150.6	150.6	150.6	49.6→65.1
MoE params, at inference (M)	0.42	0.42	0.42→0.76	0.17→0.52
Setup 2: Mor → Rad				
MOR (s1)	0.830 → **0.796**	0.830 → **0.793**	0.830 → 0.830	0.830 → 0.826
RAD (s2)	0.751	0.750	0.676	0.737
AA ↑	0.773	0.771	0.753	**0.782**
F ↓	0.034	0.037	0.000	0.000
Enc. params, at inference (M)	1.84	1.84	1.84	1.71→1.79
MoE params, at inference (M)	0.42	0.42	0.42→0.59	0.18→0.35
Setup 3: Density → Birads → Risk				
DENSITY (s1)	0.923 → **0.762**	0.923 → **0.761**	0.923 → 0.923	0.926 → **0.902**
BIRADS (s2)	0.812 → 0.806	0.813 → 0.809	0.743 → 0.743	0.813 → 0.809
RISK (s3)	0.748	0.748	0.732	0.750
AA ↑	0.772	0.772	0.799	**0.820**
F ↓	0.083	0.083	0.000	0.000
Enc. params, at inference (M)	0.49	0.49	0.49	0.13→0.39
MoE params, at inference (M)	0.42	0.42	0.42→0.76	0.17→0.52
Setup 4: {Pheno, Density} → {Los, Birads, Mor} → {Ihm, Risk, Rad}				
PHENO (s1)	0.714 → **0.669**	0.714 → **0.657**	0.714 → **0.688**	0.712 → 0.703
DENSITY (s1)	0.913 → **0.754**	0.925 → **0.799**	0.920 → **0.818**	0.920 → 0.909
LOS (s2)	0.825 → **0.797**	0.819 → 0.800	0.822 → 0.822	0.820 → 0.819
BIRADS (s2)	0.806 → **0.782**	0.806 → **0.780**	0.810 → 0.810	0.804 → 0.802
MOR (s2)	0.831 → **0.785**	0.825 → **0.767**	0.830 → 0.830	0.827 → 0.827
IHM (s3)	0.823	0.824	0.808	0.815
RISK (s3)	0.750	0.742	0.758	0.745
RAD (s3)	0.755	0.757	0.667	0.734
AA ↑	0.764	0.766	0.775	**0.794**
F ↓	0.060	0.057	0.026	0.000
Enc. params, at inference (M)	153.0	153.0	153.0	51.4→67.4
MoE params, at inference (M)	0.43	0.43	0.43→0.80	0.19→0.56

Table 14: FLAME-CL against LoRA with rank-32 adapters on the MoE experts only (MoE; the baseline of Fig. 5) and on the encoders and the MoE (enc.+MoE), seed 42. AA and F over all tasks after the final stage; encoder parameters after the final stage, as the total resident at inference and as the part attributable to the CL method (FLAME-CL: its frozen low-rank stack; LoRA: base encoder plus adapters it trains, 0 for MoE-only adapters).

	AA ↑			F ↓			Enc. params, inference (M)			Enc. params, method-attr. (M)		
Setup	FLAME	MoE	enc.+MoE	FLAME	MoE	enc.+MoE	FLAME	MoE	enc.+MoE	FLAME	MoE	enc.+MoE
1	0.777	0.762	**0.785**	0.000	0.000	0.000	65.1	150.6	165.9	23.3	0.00	124.2
2	0.782	0.753	**0.783**	0.000	0.000	0.000	1.79	1.84	1.92	0.17	0.00	0.30
3	**0.820**	0.799	0.819	0.000	0.000	0.000	0.39	0.49	0.75	0.39	0.00	0.75
4	0.794	0.775	**0.800**	0.000	0.026	0.000	67.4	153.0	168.7	24.0	0.00	125.3

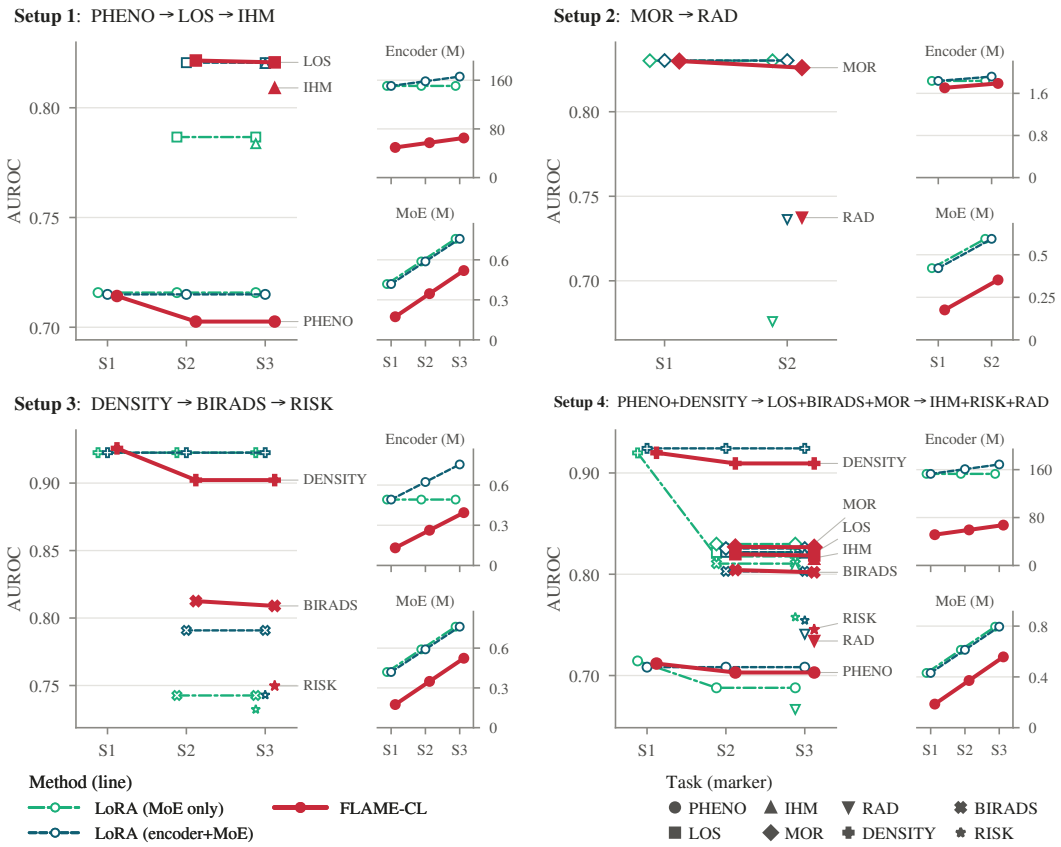


Figure 9: FLAME-CL against LoRA with rank-32 adapters on the MoE experts only (the baseline of Fig. 5) and on the encoders and the MoE (seed 42), in the format of Fig. 5. Parameter panels are inference-time totals.

loss) and its gradient hooks, applied to our fully trainable experts rather than to MoE-LoRA adapters. For TRGE [83] we keep its two-level grouped routing, over full-rank experts grown per stage rather than per-task LoRA experts on a frozen CLIP. LwF [45] regularizes the outputs of earlier task heads toward those of the previous-stage model, and EWC is as in Sec. D.3.1. All methods are evaluated on Setup 4 with the same three timestamp-generated seeds.

Per-task retention. Tab. 15 reports each task's AUROC when first learned and after the final stage, and Tab. 16 the per-task forgetting with AA and BWT. AA averages final AUROC over all eight tasks; F and BWT average over the five tasks learned before the final stage, since IHM, RISK and RAD enter at stage 3 and have zero forgetting by definition. Each task's reference value is the model deployed at the stage where it was learned; for FLAME-CL this is after rank-32 truncation, the configuration served from then on, so F measures interference from later stages only. The one-time cost of truncation itself is measured separately by the rank sweep (Tab. 19).

FLAME-CL is the only method with no drop on any task: measured from the deployed post-truncation model, its values do not change after the stage at which each task is learned (Proposition 4), whereas the baselines' deltas are interference from later stages. The largest gap is on DENSITY: it is introduced at the first stage and is therefore exposed to the most subsequent updates, and as a multi-class task its decision boundaries are more sensitive to drift than those of the binary tasks. The stage-3 tasks measure plasticity rather than retention: FLAME-CL reaches 0.73 on RAD against 0.66 (TRGE), 0.68 (SAME) and 0.67 (LwF), and EWC exceeds it (0.75) but loses 0.14 on DENSITY, which is why it is the strongest baseline on AA yet only third among baselines on F: the two metrics are complementary.

Setups 1–3. Fig. 10 and Tab. 17 repeat the comparison on the three shorter sequences (seed 42). FLAME-CL has the highest AA on all three (0.777, 0.782, 0.820). On these sequences the MoE-

Table 15: Setup 4 per-task AUROC when first learned (post-truncation for FLAME-CL) → after the final stage (3 timestamp-generated seeds, mean ± std). Stage-3 tasks are learned last and report one value. Drops of at least 0.02 in bold.

Task (stage)	FLAME-CL	TRGE	SAME	LwF	EWC
PHENO (s1)	0.71±0.00 → 0.71±0.00	0.71±0.00 → **0.67±0.01**	0.71±0.00 → **0.65±0.01**	0.71±0.01 → **0.69±0.01**	0.71±0.00 → **0.67±0.00**
DENSITY (s1)	0.91±0.01 → 0.91±0.01	0.92±0.00 → **0.75±0.06**	0.91±0.01 → **0.85±0.01**	0.92±0.01 → **0.65±0.10**	0.92±0.01 → **0.78±0.05**
LOS (s2)	0.82±0.00 → 0.82±0.00	0.82±0.00 → 0.82±0.00	0.82±0.00 → 0.81±0.01	0.81±0.01 → 0.81±0.00	0.82±0.00 → 0.80±0.01
BIRADS (s2)	0.81±0.00 → 0.81±0.00	0.79±0.01 → 0.79±0.01	0.77±0.03 → 0.76±0.03	0.78±0.01 → 0.77±0.00	0.80±0.00 → 0.79±0.01
MOR (s2)	0.82±0.00 → 0.82±0.00	0.82±0.00 → 0.82±0.00	0.82±0.00 → 0.82±0.00	0.83±0.00 → 0.83±0.00	0.82±0.00 → **0.78±0.02**
IHM (s3)	0.81±0.01	0.81±0.00	0.81±0.02	0.80±0.00	0.81±0.00
RISK (s3)	0.74±0.00	0.74±0.01	0.74±0.02	0.73±0.01	0.75±0.00
RAD (s3)	0.73±0.00	0.66±0.01	0.68±0.00	0.67±0.01	0.75±0.00

Table 16: Setup 4 per-task forgetting F and summary metrics (3 timestamp-generated seeds, mean ± std). $F = -BWT$ in every cell, so BWT is reported only in the summary row.

	FLAME-CL	TRGE	SAME	LwF	EWC
F: PHENO (s1)	0.000±0.000	0.048±0.009	0.054±0.007	0.028±0.002	0.043±0.005
F: DENSITY (s1)	0.000±0.000	0.171±0.061	0.058±0.011	0.272±0.109	0.143±0.054
F: LOS (s2)	0.000±0.000	0.002±0.000	0.006±0.005	0.001±0.001	0.020±0.011
F: BIRADS (s2)	0.000±0.000	0.005±0.004	0.011±0.010	0.004±0.002	0.009±0.011
F: MOR (s2)	0.000±0.000	0.001±0.001	0.003±0.002	0.001±0.000	0.041±0.021
F (avg) ↓	**0.000±0.000**	0.046±0.010	0.026±0.007	0.061±0.022	0.051±0.009
BWT (avg) ↑	**0.000±0.000**	−0.046±0.010	−0.026±0.007	−0.061±0.022	−0.051±0.009
AA (8 tasks) ↑	**0.793±0.002**	0.757±0.007	0.764±0.006	0.743±0.013	0.767±0.006

based and distillation baselines forget little (F≤0.009) but learn new tasks less well: they trail FLAME-CL by 0.03–0.05 on LOS and IHM in Setup 1, 0.04–0.07 on RAD in Setup 2, and 0.06–0.11 on BIRADS in Setup 3. EWC is about as plastic as FLAME-CL but forgets (F up to 0.083).

Table 17: Setups 1–3: AUROC when first learned → after the final stage for FLAME-CL and the MoE-based / distillation baselines (seed 42; Setup 4 is in Tabs. 15, 16). Drops of at least 0.02 in bold; FLAME-CL's first value is before rank-32 truncation; F against the model deployed at the stage where each task was learned. Parameter rows give inference-time totals after the first → last stage: TRGE, SAME and LwF adapt only the MoE and keep the full encoder.

Task (stage)	FLAME-CL	TRGE	SAME	LwF	EWC
Setup 1: PHENO → LOS → IHM					
PHENO (s1)	0.714 → 0.703	0.714 → 0.713	0.712 → 0.701	0.716 → 0.713	0.716 → **0.668**
LOS (s2)	0.822 → 0.821	0.782 → 0.782	0.790 → 0.782	0.786 → 0.780	0.825 → 0.805
IHM (s3)	0.809	0.776	0.756	0.760	0.814
AA ↑	**0.777**	0.757	0.747	0.751	0.762
F ↓	0.000	0.001	0.009	0.004	0.034
Enc. params, at inference (M)	49.6→65.1	150.6	150.6	150.6	150.6
MoE params, at inference (M)	0.17→0.52	0.84→1.26	0.42	0.42→0.43	0.42
Setup 2: MOR → RAD					
MOR (s1)	0.830 → 0.826	0.830 → 0.829	0.828 → 0.832	0.831 → 0.830	0.830 → **0.793**
RAD (s2)	0.737	0.665	0.694	0.678	0.750
AA ↑	**0.782**	0.747	0.763	0.754	0.771
F ↓	0.000	0.000	0.000	0.001	0.037
Enc. params, at inference (M)	1.71→1.79	1.84	1.84	1.84	1.84
MoE params, at inference (M)	0.18→0.35	0.84	0.42	0.42→0.43	0.42
Setup 3: DENSITY → BIRADS → RISK					
DENSITY (s1)	0.926 → **0.902**	0.922 → 0.915	0.906 → 0.911	0.923 → 0.921	0.923 → **0.761**
BIRADS (s2)	0.813 → 0.809	0.756 → 0.759	0.708 → 0.712	0.753 → 0.748	0.813 → 0.809
RISK (s3)	0.750	0.724	0.663	0.724	0.748
AA ↑	**0.820**	0.799	0.762	0.798	0.772
F ↓	0.000	0.003	0.000	0.003	0.083
Enc. params, at inference (M)	0.13→0.39	0.49	0.49	0.49	0.49
MoE params, at inference (M)	0.17→0.52	0.84→1.26	0.42	0.42→0.43	0.42

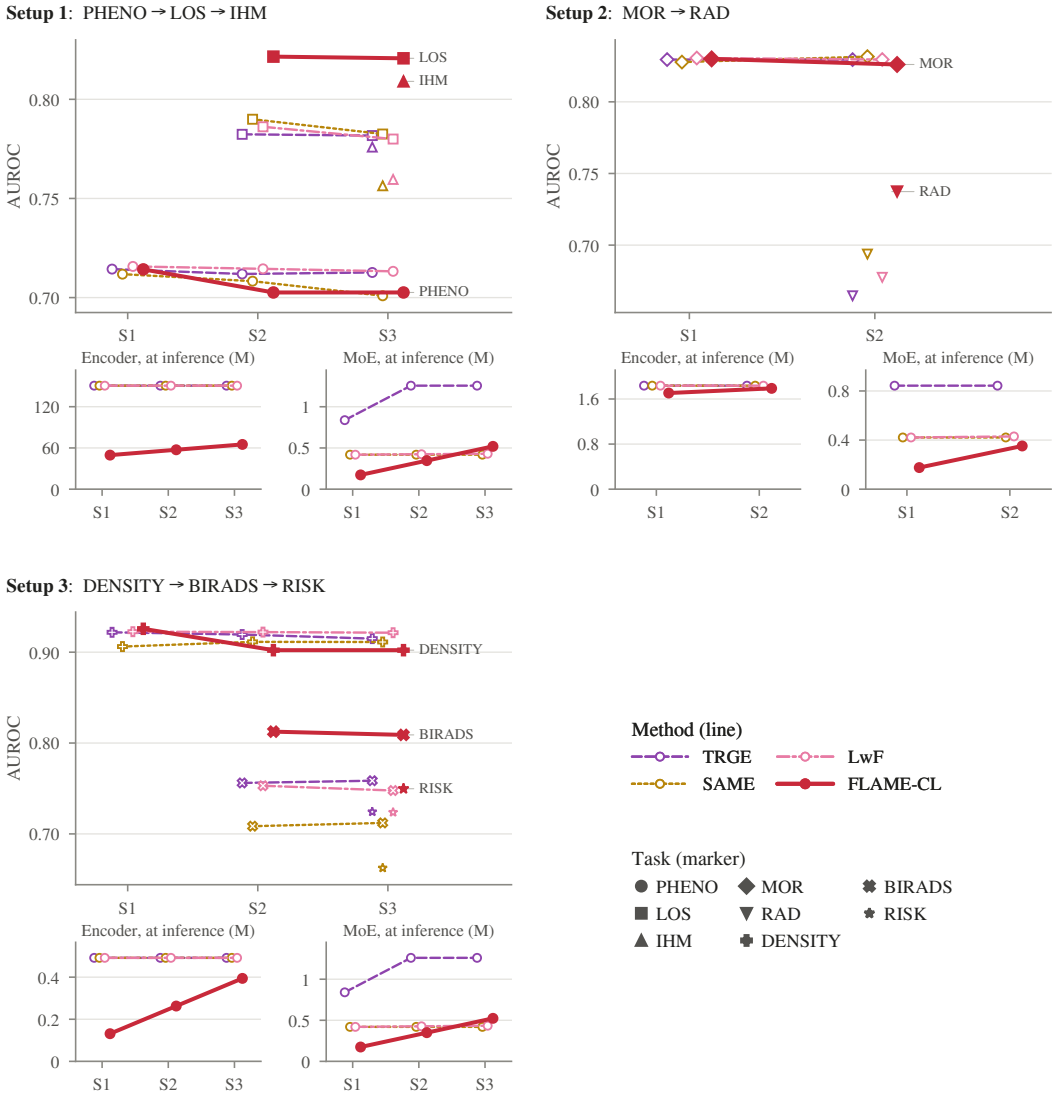


Figure 10: FLAME-CL against the MoE-based (TRGE, SAME) and distillation (LwF) baselines on Setups 1–3 (seed 42; Setup 4 in Tabs. 15, 16). Parameter panels are inference-time totals: TRGE, SAME and LwF adapt only the MoE and keep the full encoder.

D.3.4 Retention and Plasticity at a Fixed Budget

A natural concern is that baselines with a flat parameter budget could match FLAME-CL at lower cost as stages accumulate. Tab. 18 tests this on Setup 2 (MOR→RAD). The methods that hold their budget flat (SIMPLE FT, EWC) match FLAME-CL's plasticity on the new task but forget the old one by 0.03–0.04 AUROC, while every baseline that retains MOR caps at 0.69 on RAD, against 0.74 for FLAME-CL. No baseline is simultaneously retentive and plastic at any budget we measured.

Table 18: Setup 2 (MOR→RAD): AUROC when first learned → after the final stage.

Method	MOR (s1)	RAD (s2)
FLAME-CL	$0.83 \rightarrow 0.83$	0.74
SIMPLE FT	$0.83 \rightarrow 0.80$	0.75
EWC	$0.83 \rightarrow 0.79$	0.75
SAME	$0.83 \rightarrow 0.83$	0.69
TRGE	$0.83 \rightarrow 0.83$	0.67
Lifelong-PT	$0.83 \rightarrow 0.83$	0.62

FLAME-CL's growth is governed by the truncation rank, which we hold fixed across stages. Tab. 19 shows that this budget is not tightly constrained: halving the rank from 32 to 16 costs at most 0.01 AUROC on every task, which roughly halves the growth rate and doubles the number of stages absorbed at equal cost. Since the required rank falls with stage index, a decreasing rank schedule, or a per-task allocation guided by each task's functional-energy concentration, is a direct extension that we have not explored.

D.3.5 CL-Phase Rank Sweep

Because forgetting is zero by construction (Proposition 4), every degradation visible in Fig. 5 and Tab. 15 is truncation cost rather than interference, so a sweep over the reserved rank measures exactly that cost. Tab. 19 reports it for Setup 4: each column truncates the active component of the stage at which that task is learned to rank r, with all earlier stages frozen at rank 32. (i) Truncation is close to functionally lossless at the reserved rank: at $r{=}32$ the reserved slices retain only 57–69% of their weight-spectral energy, yet every task is within 0.01 AUROC of full rank. (ii) The required rank decreases with stage index: stage-3 tasks saturate by $r{=}2$–8 (IHM reaches its full-rank value at $r{=}2$), stage-2 tasks by $r{=}8$–16, and only stage-1 tasks keep improving up to $r{=}32$. Later stages reuse capacity reserved by earlier ones, so the per-stage budget can shrink as the stream progresses, and a single $r{=}32$ is conservative. (iii) Forward transfer is substantial: for stage-2 and stage-3 tasks the $r{=}0$ row is not chance, since it removes the new stage's capacity while keeping the earlier frozen components, and so measures what the shared pool already supports. With no new expert parameters, IHM reaches 0.77 and RISK 0.74 (vs. 0.82 and 0.75 at full rank): the per-modality routers reuse prior capacity rather than relearning each task.

Table 19: Full CL-phase rank sweep (Setup 4, AUROC). Each column truncates the active component of the stage at which that task is learned to rank r, with earlier stages frozen at rank 32; $r{=}0$ removes the new stage's expert capacity. Default $r{=}32$ shaded.

	Stage 1		Stage 2			Stage 3		
r	PHENO	DENSITY	LOS	BIRADS	MOR	IHM	RISK	RAD
0	0.50	0.50	0.72	0.60	0.60	0.77	0.74	0.59
2	0.58	0.72	0.81	0.77	0.80	0.82	0.74	0.71
4	0.61	0.85	0.82	0.78	0.81	0.81	0.74	0.73
8	0.65	0.88	0.82	0.80	0.82	0.81	0.75	0.73
16	0.69	0.91	0.82	0.80	0.83	0.81	0.74	0.73
32	0.70	0.91	0.82	0.80	0.83	0.81	0.75	0.73
64	0.71	0.92	0.82	0.80	0.83	0.82	0.75	0.73
full	0.71	0.92	0.82	0.80	0.83	0.82	0.75	0.73

D.3.6 Component Ablation Protocol

The ablation rows of Fig. 6a separate the two continual-learning components. $-$*cursor* re-evaluates the trained FLAME-CL models with the cursor removed ($\tau = T$ for every task, i.e., summing all stages' components) using the same trained weights. $-$*low-rank* reserves each stage's additive component at full rank instead of truncating it to rank r_t, with everything else unchanged, so the parameters grow by a full-shape component per stage (380.0M against 74.6M after three stages). Removing the cursor costs 0.034 AA and raises F from 0.000 to 0.055 at both rank 32 and full rank, so retention is attributable to cursor-based inference regardless of the rank of the frozen components. Removing compression leaves F at exactly 0.000 and changes AA by $+0.003$, within seed noise, at $5.1\times$ the parameter count. Low-rank compression therefore makes the parameter count scale economically with the number of stages at no measurable cost in accuracy, while retention comes from the cursor together with the freezing of each stage's reserved component and router head.

D.3.7 Training Cost

At stage t, FLAME-CL trains only the additive components $\{\widetilde{W}_i^{(t)}\}$, the new router heads, the new task head, and any new modality encoder. The pretrained base, all prior reserved components, router heads and task heads are frozen, and no prior-stage data is revisited (App. E). The per-stage cost is therefore comparable to LoRA fine-tuning on the same backbone rather than to retraining from scratch, and the SVD runs once per expert sublayer at the end of each stage, which is negligible against training time. We initialize $\widetilde{W}_i^{(t)}$ at full rank; for large MoEs it can instead be parameterized in factored form at a moderate rank and truncated afterwards, which removes the full-rank training step, as in the OLMoE experiment of App. D.7.2. At the system level, per-task baselines require a separate run per task, whereas one FLAME backbone is trained across all of them; since modalities recur across tasks, per-task training also duplicates the same encoder repeatedly, while FLAME instantiates one encoder per modality type and one shared expert pool.

D.3.8 Comparison Against Lifelong-PT

Fig. 11 compares FLAME-CL against Lifelong-PT [9], an MoE-based CL method that grows the expert pool with each new task, on the same four sequences and axes as Fig. 5. Lifelong-PT forgets where FLAME-CL does not: in Setup 4 it loses PHENO ($0.71\rightarrow0.57$) and DENSITY ($0.92\rightarrow0.51$), for AA 0.70 against 0.79 and F 0.107 against 0, and it trails FLAME-CL on every newly introduced task. Expanding the pool does not by itself protect earlier tasks; FLAME-CL's rank-reserved stack does so inside a fixed-size pool.

D.3.9 Per-Stage AUPRC Trajectories

For completeness, Fig. 12 reports per-stage AUPRC with the same layout as Fig. 5. The picture mirrors AUROC: FLAME-CL's largest margin is on the earliest Setup 4 task, DENSITY (0.72 against at most 0.48 for the baselines, which forget it), and on most other tasks it is within 0.03 of the best baseline. The exceptions are Setup 3's DENSITY, where the one-time truncation costs 0.06 AUPRC, and IHM in Setup 4 (0.46 vs. 0.52 for EWC).

D.4 Spectral Validation of Proposition 1

This section reports the empirical spectra that motivate the spectral-compression step of FLAME (Eq. (3)) and validate Proposition 1 across every task and joint training configuration considered in our experiments. For each configuration we plot, in the first cross-modal MoE block, the cumulative top-K energy of all 10 expert sublayers (5 experts $\times$ {`fc1`, `fc2`}) under three lenses:

- **Input spectrum** (C_i eigenvalues): the eigenvalue spectrum of the routed-input covariance $C_i = \mathbb{E}_{\boldsymbol{z}}[\boldsymbol{z}\boldsymbol{z}^\top]$, estimated from the test-set tokens that each expert actually receives.
- **Weight-only** (Frobenius): the singular-value energy $\{\sigma_{i,k}^2\}$ of the expert's weight matrix W_i, ignoring the input distribution.
- **Data-aware** (test activations): the per-rank functional energy $\mathcal{E}_{i,k} = \sigma_{i,k}^2 \boldsymbol{v}_{i,k}^\top C_i \boldsymbol{v}_{i,k}$ from Eq. (2), which couples weight and input geometry.

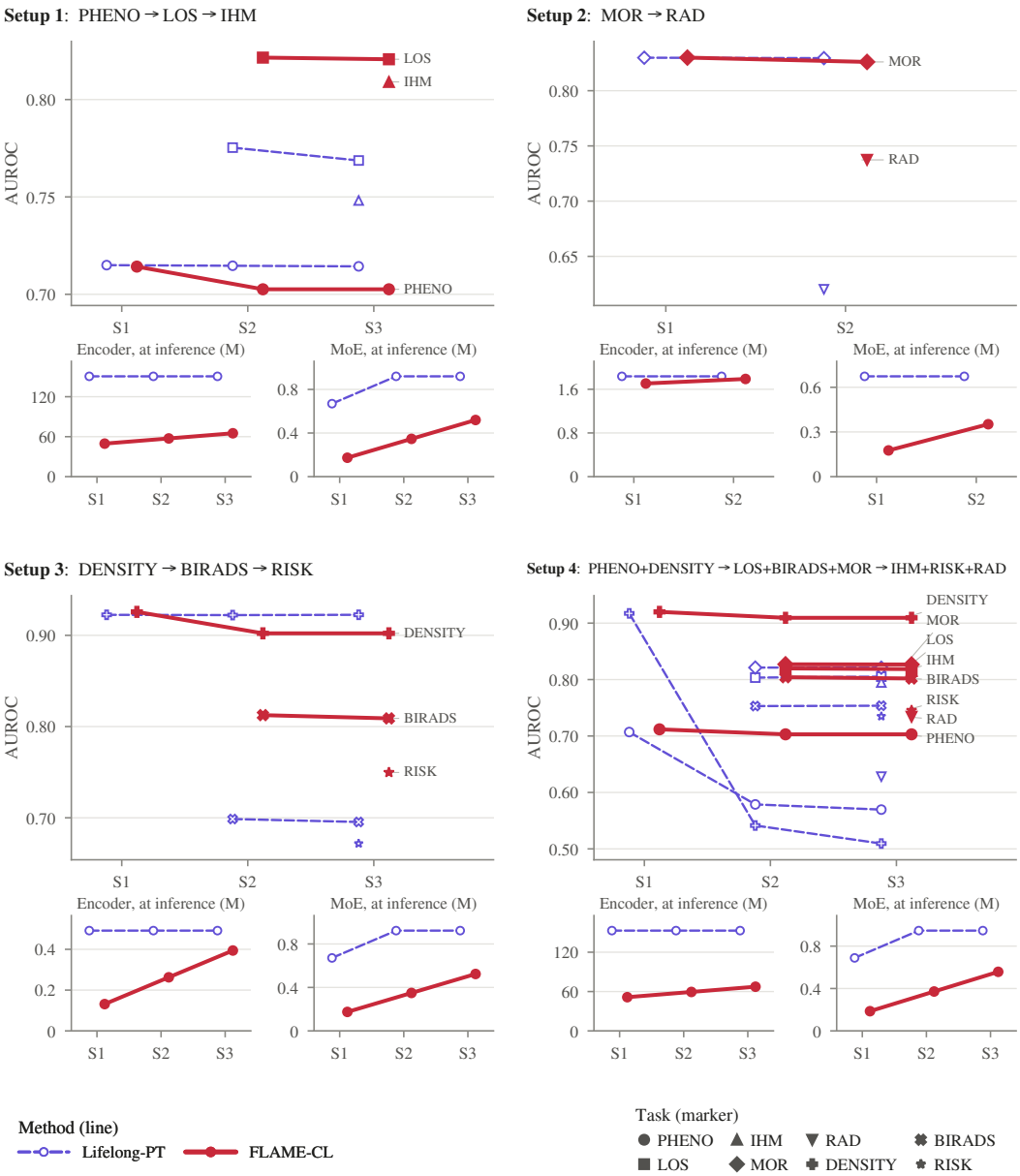


Figure 11: Comparison against Lifelong-PT [9], an MoE-based continual-learning method that grows the expert pool with each new task (seed 42; layout as in Fig. 5). Lifelong-PT forgets PHENO and DENSITY in Setup 4 and trails FLAME-CL on every new task.

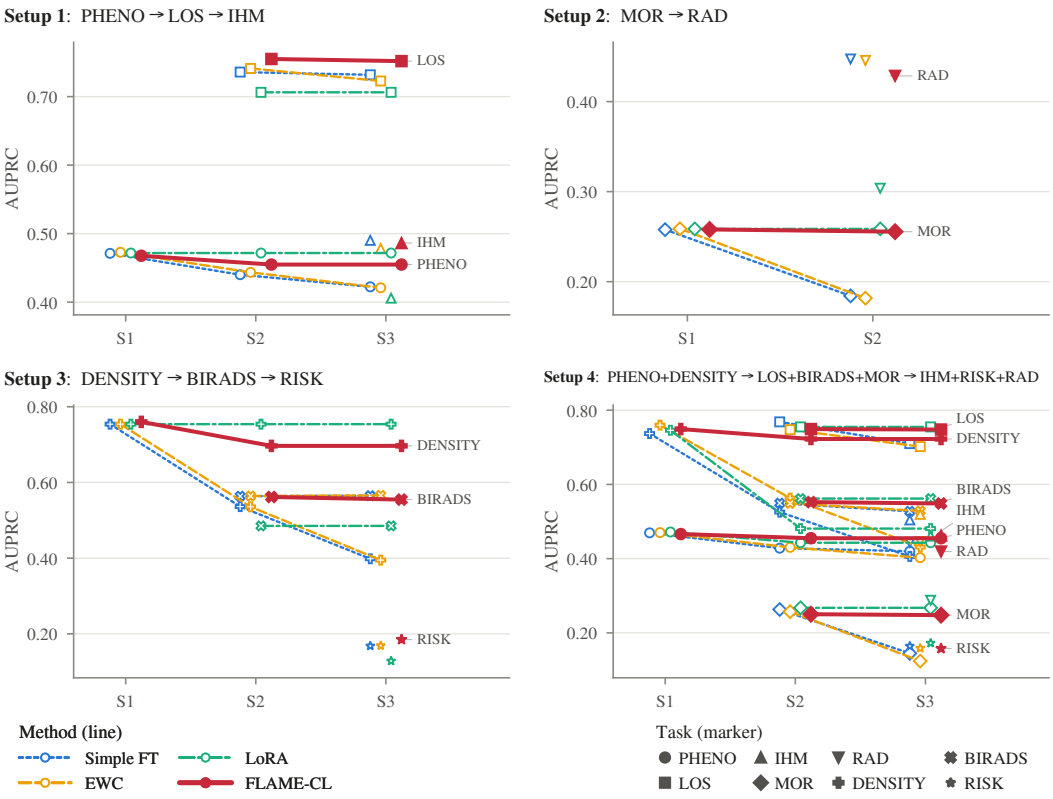


Figure 12: Per-stage **AUPRC** for the four continual-learning sequences (seed 42); layout as in Fig. 5.

The dashed red lines mark the 90% and 99% cumulative-energy thresholds. Across every benchmark, the input spectrum and the data-aware curve saturate by rank ≈20–40 while the weight-only spectrum remains near full rank, exposing the idle capacity that FLAME's continual-learning step reallocates.

D.4.1 Single-Task Pretraining Spectra

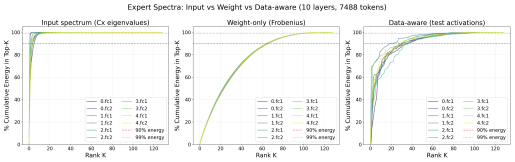
Fig. 13 shows the three-lens spectra for each of the nine tasks individually trained. The pattern is consistent across modality types: time-series ICU tasks (panels a–e), high-resolution mammography views (f–h), and multi-modal Alzheimer biomarkers (i) all show data-aware energy curves that saturate well before the ambient rank, despite the wide variation in input statistics. This consistency is what makes per-task rank reservation viable: the spectral-rank budget does not need to be re-tuned per modality, and the truncation step in Eq. (3) discards directions with negligible functional energy regardless of which clinical task contributes them.

D.4.2 Multi-Task Pretraining Spectra

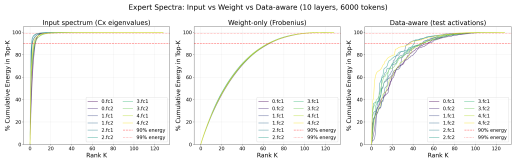
Fig. 14 reports the same three-lens analysis under joint pretraining, both within MIMIC-IV ({IHM, LOS, PHENO}) and across all nine tasks (MIMIC-IV ∪ eICU ∪ EMBED ∪ ADNI). Even with a larger and more heterogeneous routed-input distribution, the data-aware functional energy still saturates well before the ambient rank—only marginally later than the single-task spectra above. Multitask routing therefore does not break the low-rank structure that Proposition 1 relies on; it preserves it. This is the empirical foundation for using a single rank budget r_t during continual stages regardless of how many tasks were jointly pretrained at stage 0.

D.4.3 Continual-Stage Spectra and Alignment

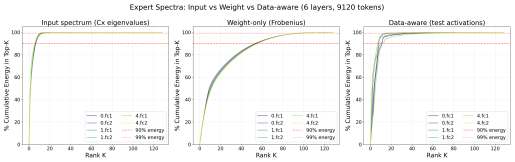
Fig. 15 repeats the three-lens diagnostic in the continual phase for all four sequences, for the active component of each stage under that stage's routed inputs, and adds the singular-vector alignment of Proposition 2; Tab. 20 gives the per-stage numbers. The routed-input spectrum reaches 90% of its



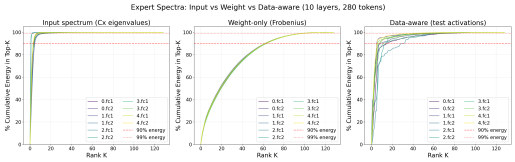
(a) 48-IHM, MIMIC-IV.



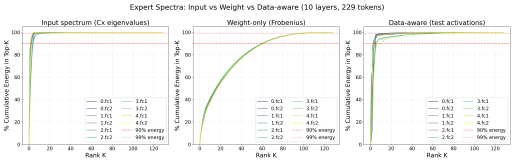
(b) LOS, MIMIC-IV.



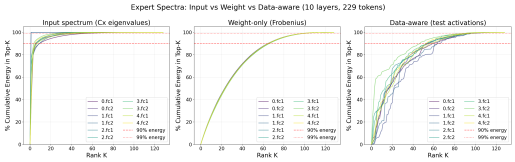
(c) 25-PHENO, MIMIC-IV.



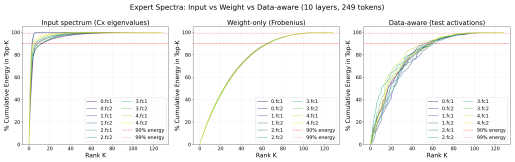
(d) MOR, eICU.



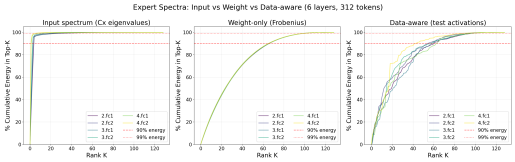
(e) RAD, eICU.



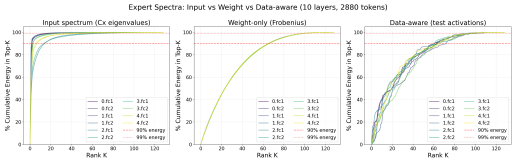
(f) BIRADS, EMBED.



(g) DENSITY, EMBED.



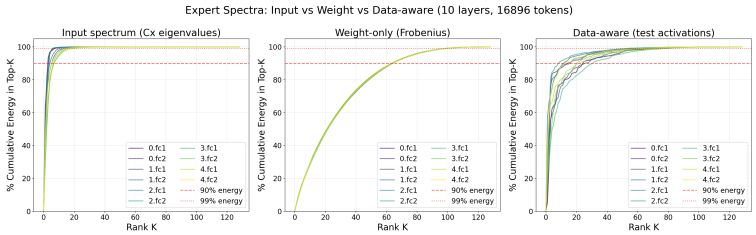
(h) RISK, EMBED.



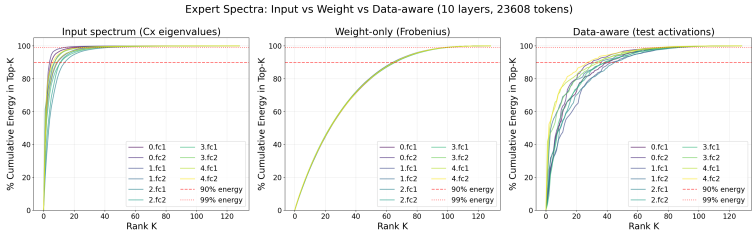
(i) DIAG, ADNI.

Figure 13: Expert input spectra after single-task FLAME pretraining on each of the nine tasks: MIMIC-IV (time series, clinical text, chest X-ray; a–c), eICU (d–e), EMBED (2D-CC, 2D-MLO, CC and MLO mammography views; f–h) and ADNI (multi-modal tabular and imaging biomarkers; i). Panels as in Fig. 3.

energy by rank 4 (median over all 110 stage sublayers, at most 22) at every stage of every sequence, so Proposition 1's premise holds throughout each stream. The weight-only and data-aware spectra split by initialization. At stages ≥ 2 the output sublayer `fc2` of each new component is zero-initialized, the setting of Proposition 2: its data-aware energy reaches 90% by rank 3 (range 1–8), the rank-32 truncation keeps $\geq 99.8\%$ of it, and its right singular vectors align with the eigenvectors of C_i well above chance ($A(16) = 0.41$, range 0.22–0.51, against 0.125 for random subspaces). The input sublayer `fc1` keeps its default random initialization, as do both sublayers at stage 1, which trains the base experts from scratch. `fc1` stays near chance alignment ($A(16) = 0.16$) and accumulates data-aware energy gradually (90% at rank 66), so the truncation keeps a median 65% of its data-aware energy; the stage-1 sublayers sit in between (90% at rank 46, 82% kept, $A(16) = 0.18$). Setup 4 gives the same picture on all four seeds: at stages 2 and 3, `fc2` keeps $\geq 99.9\%$ of its data-aware energy at rank 32 with per-seed median $A(16)$ between 0.39 and 0.44. Functional rank therefore cannot be read off the weight spectrum alone, and the decisive test is behavioral: by Proposition 5 the truncation error is exactly the discarded data-aware tail, and the CL-phase rank sweep (Tab. 19) shows that truncating to $r_t = 32$ stays within 0.01 AUROC of the untruncated component on all eight tasks, with the stage-2 and stage-3 tasks already within 0.01 at $r_t = 8$.



(a) Joint MIMIC-IV pretraining over {IHM, LOS, PHENO}.



(b) Joint pretraining over all 9 tasks (MIMIC-IV ∪ eICU ∪ EMBED ∪ ADNI).

Figure 14: Expert input spectra under joint FLAME pretraining. Despite the larger and more heterogeneous routed-input distribution, the data-aware functional energy still saturates well before the ambient rank, supporting per-task low-rank reservation in continual learning.

Table 20: Continual-phase spectra and singular-vector alignment of the active component of each stage (FLAME-CL, seed 42), median over the 5 experts, reported as `fc2/fc1`. r_{90}: rank at 90% cumulative energy (ambient dimension 128). Kept: data-aware energy inside the top-32 singular directions, i.e., what survives the $r_t{=}32$ truncation. $A(16)$: alignment as in Fig. 15; random subspaces give 0.125. `fc2` is zero-initialized at stages ≥ 2; `fc1`, and both sublayers at stage 1, start from random initialization.

		r_{90} (`fc2/fc1`)			Kept at $r_t{=}32$	$A(16)$
Setup	Stage	input	weight-only	data-aware	data-aware	(random 0.125)
1	1: PHENO	4 / 4	65 / 66	37 / 45	0.87 / 0.84	0.21 / 0.20
	2: LOS	5 / 3	9 / 65	2 / 74	1.00 / 0.51	0.46 / 0.14
	3: IHM	4 / 2	9 / 65	3 / 69	1.00 / 0.65	0.44 / 0.15
2	1: MOR	3 / 7	66 / 66	47 / 50	0.83 / 0.78	0.17 / 0.17
	2: RAD	8 / 7	10 / 66	3 / 62	1.00 / 0.64	0.36 / 0.15
3	1: DENSITY	6 / 12	66 / 65	50 / 55	0.77 / 0.74	0.16 / 0.15
	2: BIRADS	5 / 6	9 / 66	6 / 62	1.00 / 0.69	0.41 / 0.16
	3: RISK	2 / 2	7 / 66	2 / 69	1.00 / 0.56	0.32 / 0.12
4	1: PHENO+DENSITY	4 / 4	66 / 66	41 / 41	0.86 / 0.85	0.21 / 0.23
	2: LOS+BIRADS+MOR	7 / 5	12 / 66	1 / 68	1.00 / 0.69	0.44 / 0.16
	3: IHM+RISK+RAD	4 / 5	11 / 65	3 / 61	1.00 / 0.76	0.41 / 0.16

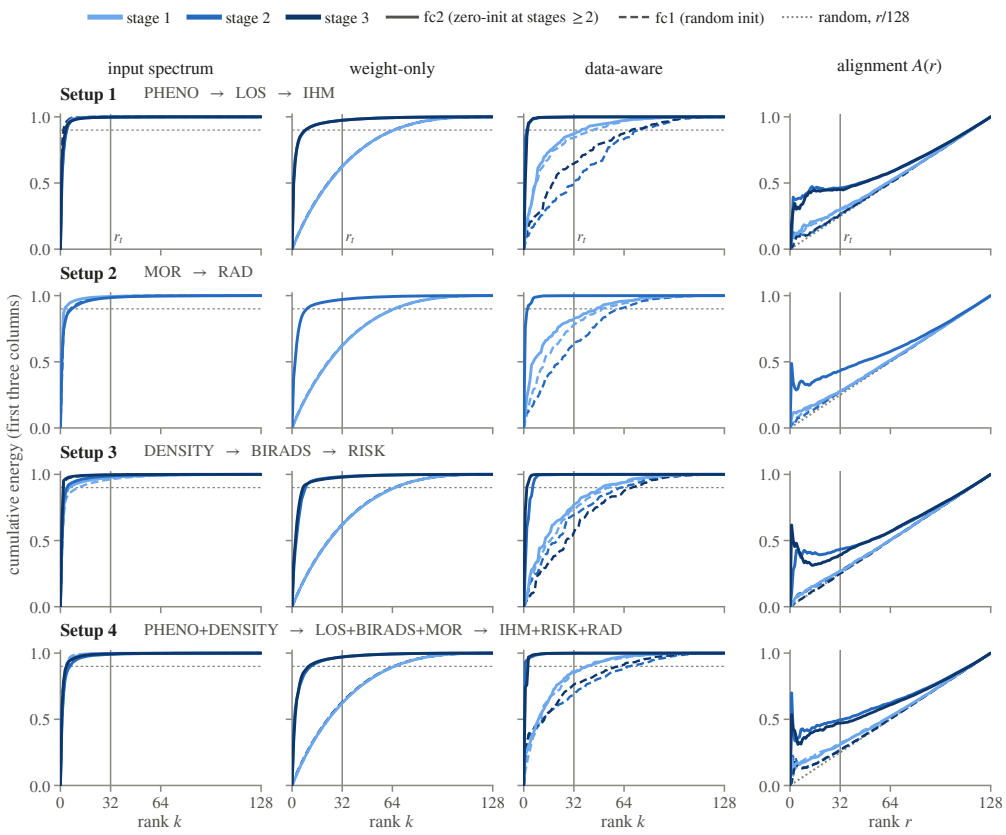


Figure 15: Continual-stage spectra and singular-vector alignment for the four CL sequences (rows; FLAME-CL, seed 42), for the active component of each stage under that stage's routed inputs. Columns: input spectrum of C_i, weight-only spectrum, cumulative data-aware energy $\mathcal{E}_{i,k}$ (as in Fig. 3), and the alignment $A(r) = \|V_r^\top Q_r\|_F^2/r$ between the top-r right singular vectors of the component and the top-r eigenvectors of C_i (dotted: random subspaces, $r/128$). Lines are medians over the 5 experts; solid `fc2`, dashed `fc1`; color marks the stage. `fc2` is zero-initialized at stages ≥ 2 (the setting of Proposition 2); `fc1`, and both sublayers at stage 1, start from random initialization. Vertical line: $r_t = 32$. C_i is estimated on the first ≤ 512 validation samples of each task in the stage.

D.5 Per-Modality Routing Distributions

This section reports the routing patterns learned by FLAME's per-modality routers in the first cross-modal MoE block. For every configuration we plot, for each modality (and each task in the multi-task panels):

- **Activation ratio** (left): the fraction of modality-m tokens for which expert e is among the top-K activated experts.
- **Mean gate weight** (right): the average sparse gate weight assigned to expert e, conditioned on e being activated.

Modality-specific routers concentrate each modality on a small subset of experts, while different modalities reuse overlapping experts only when their representations are complementary—the empirical basis for the per-modality routing design of Sec. 2.1. In the multi-task panels, comparing rows reveals which task pairs share modality-conditioned experts and which carve out disjoint capacity.

D.5.1 Single-Task Routing

Fig. 16 shows per-modality routing for each task trained individually. The MIMIC-IV triple (panels a–c) gives a clear illustration of the design's intent: chest X-ray, clinical text, and time series each

concentrate on a distinct pair of experts, with mild reuse on a shared expert when the modalities are complementary for the task. The eICU pair (d–e) follows the same structure on its two-modality input. EMBED's four mammography views (f–h) split cleanly between view-specific experts, confirming that the router exploits the same-modality-different-view signal supplied by breast laterality. ADNI's five biomarker modalities (i) occupy expert positions that are essentially disjoint from the other datasets', reflecting its unique modality pool.

D.5.2 Multi-Task Routing

Figs. 17, 18 report the per-task, per-modality routing under joint training. Within MIMIC-IV (Fig. 17), IHM and LOS share an almost identical routing fingerprint—the structural reason their multitask AUROC reinforces—whereas 25-PHENO carves out a distinct subset via the `cxr_pheno`, `text_pheno`, and `ts_pheno` per-task router heads. In the nine-task panel (Fig. 18), the EMBED triple BIRADS/DENSITY/RISK collapses onto two shared experts; the eICU pair MOR/RAD settles on another; ADNI's biomarker tasks occupy expert positions that no other dataset reaches. This expert-level task clustering is the data-driven evidence behind FLAME's joint-training pairing recommendations and reads consistently with the pairwise heatmap in Sec. D.2.2.

D.5.3 Continual-Stage Routing

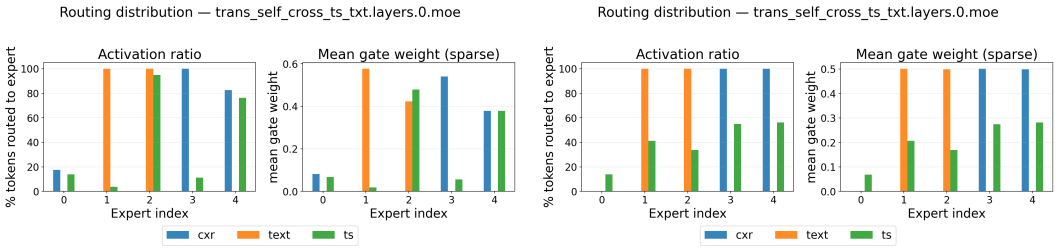
This subsection extends the routing analysis from joint multi-task pretraining (Sec. D.5.2) to the continual-stage setting: Figs. 19–22 visualize how the per-stage router heads $\{G_m^{(k)}\}_m$ (Algorithm 1) deploy each modality across the shared expert pool at every stage of the four continual sequences from Sec. D.3. Each panel row corresponds to one stage with the tasks introduced at that stage labeled; columns within a row are the modalities used at that stage; the heatmap inside each block is the per-expert activation ratio. Two structural properties of FLAME-CL read directly off the panels.

Frozen prior fingerprints. Once stage k converges, the router heads $\{G_m^{(k)}\}_m$ and the rank-r_k truncations $\{W_i^{(k)}\}$ are frozen (Sec. 2.2); the routing fingerprints assigned at stage k therefore appear unchanged in every later stage's panels — rows do not shift, swap experts, or reweight when subsequent tasks are added. This is the routing-level reading of the cursor identity in Eq. (4): under cursor $\tau = k(t)$, task t at inference sees only its own stage's router heads, and new stages cannot perturb them.

Within-dataset reuse vs. cross-dataset isolation. For sequences whose stages share modality semantics (S1: PHENO→LOS→IHM on MIMIC-IV in Fig. 19; S2: MOR→RAD on eICU in Fig. 20; S3: DENSITY→BIRADS→RISK on EMBED in Fig. 21), each new stage's router lands on the same modality-conditioned experts that prior stages occupied, with at most mild reweighting — knowledge transfer at the architectural level rather than at the parameter level. In the mixed cross-dataset sequence (S4, Fig. 22), tasks from different datasets occupy disjoint expert subsets at every stage, mirroring the all-nine-task pretraining panel (Fig. 18); cross-dataset interference is bounded structurally, with no replay buffer or alignment loss. Together with the AUROC and parameter trajectories of Sec. D.3, these visualizations explain *how* FLAME-CL retains every earlier task while training 5–15× fewer encoder parameters than SIMPLE FT and EWC: shared-modality stages share experts, disjoint-modality stages occupy disjoint experts, and prior fingerprints are never overwritten.

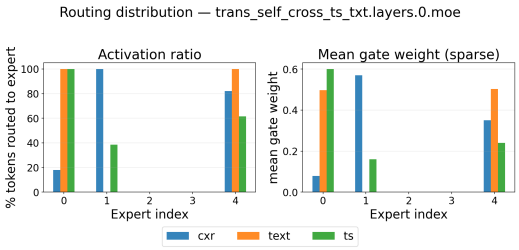
D.6 Robustness to Modality Corruption, Disagreement, and Missingness

Protocol. We re-evaluate all eight tasks of Setup 4 with one modality corrupted at a time, injected after its encoder and before its router. There are six conditions per modality: additive Gaussian noise at 0.5, 1, 2 and 4 times the modality's own standard deviation; *shuffling*, which permutes that modality's embeddings across samples, preserving its marginal distribution while destroying its association with the label; and removal. This gives 186 evaluations over 31 (task, modality) pairs. Shuffling is a controlled form of cross-modal disagreement: a shuffled imaging embedding is a well-formed image representation belonging to a different patient, so the corrupted modality keeps supplying confident evidence that now conflicts with the remaining modalities, as when imaging contradicts laboratory values and notes.

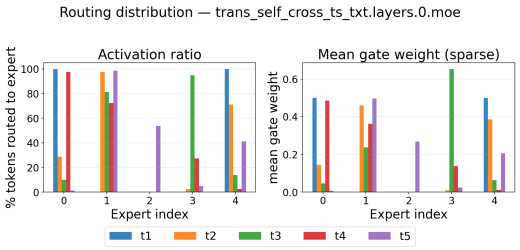


(a) 48-IHM, MIMIC-IV.

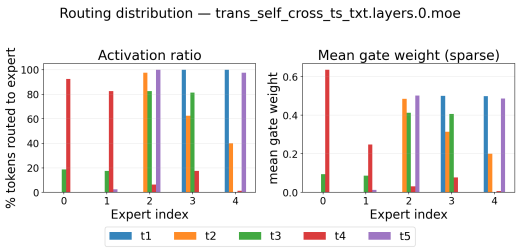
(b) LOS, MIMIC-IV.



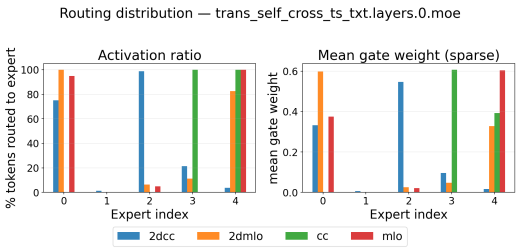
(c) 25-PHENO, MIMIC-IV.



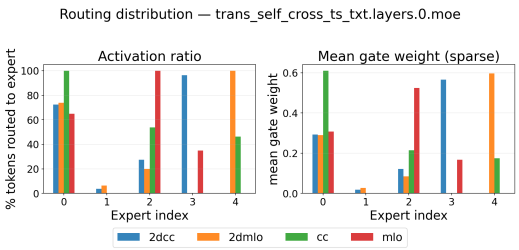
(d) MOR, eICU.



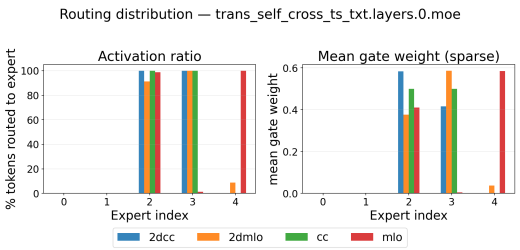
(e) RAD, eICU.



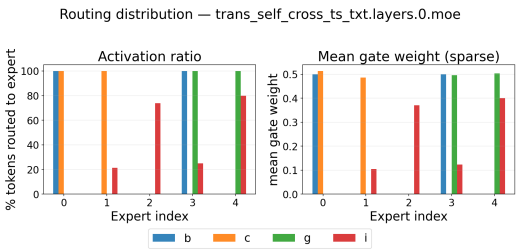
(f) BIRADS, EMBED.



(g) DENSITY, EMBED.



(h) RISK, EMBED.



(i) DIAG, ADNI.

Figure 16: Per-modality routing for each task trained individually. MIMIC-IV (a–c): chest X-ray (`cxr`), clinical text (`text`) and time series (`ts`) each concentrate on a distinct pair of experts, with mild reuse on a shared expert. eICU (d–e). EMBED (f–h): mammography views (`2dcc`, `2dmlo`, `cc`, `mlo`) split between view-specific experts. ADNI (i): T1–T5 imaging biomarkers.



Figure 17: Per-task, per-modality routing under joint MIMIC-IV pretraining over {IHM, LOS, PHENO} (one row per task). IHM and LOS share a routing fingerprint; PHENO's per-task router heads carve out a distinct subset of experts.

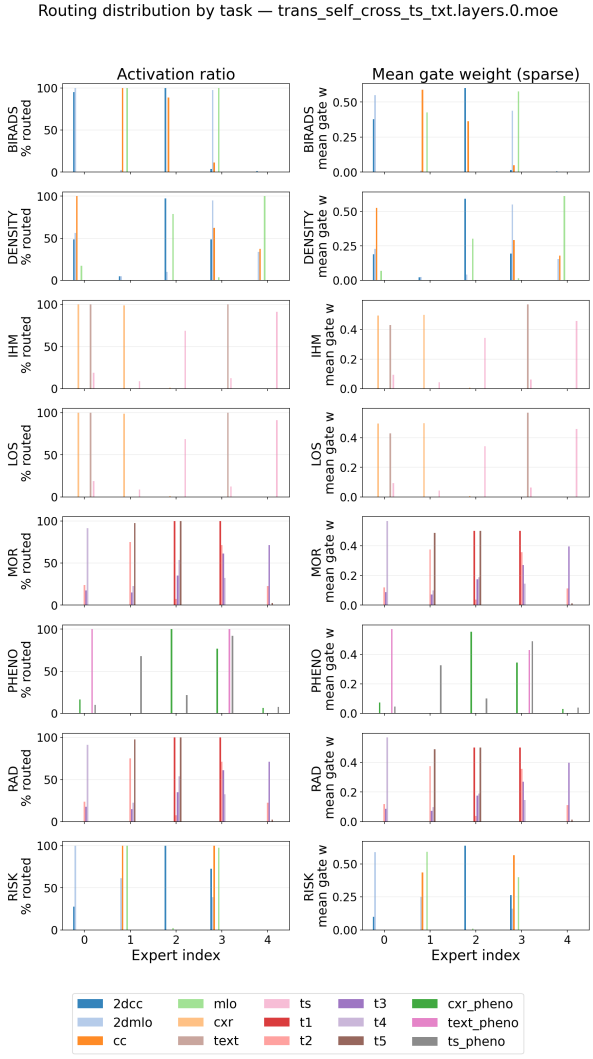


Figure 18: Per-task, per-modality routing under joint pretraining over all nine tasks (rows: tasks; columns: the five experts). Tasks that share modality semantics (the EMBED triple, the eICU pair) collapse onto overlapping expert subsets, while tasks with disjoint modality pools occupy disjoint experts.

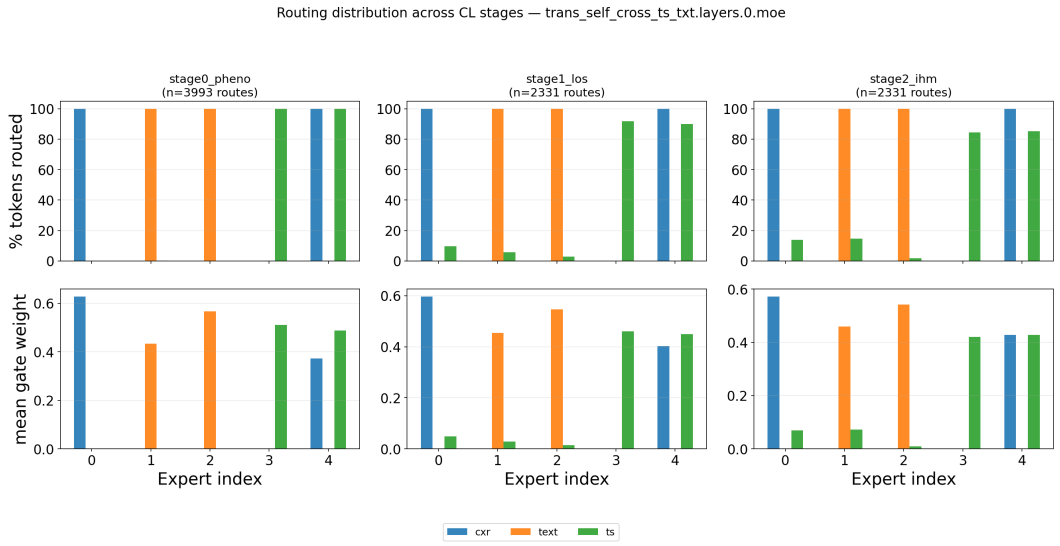


Figure 19: S1: per-modality, per-stage routing under MIMIC-IV continual learning PHENO→LOS→IHM. Modalities at every stage: time series (TS), clinical text (Text), chest X-ray (CXR). Each subsequent stage's fingerprint overlaps heavily with stage 0's, which is the structural reason FLAME-CL retains prior-task AUROC within 0.01 on this sequence (Fig. 5, S1).

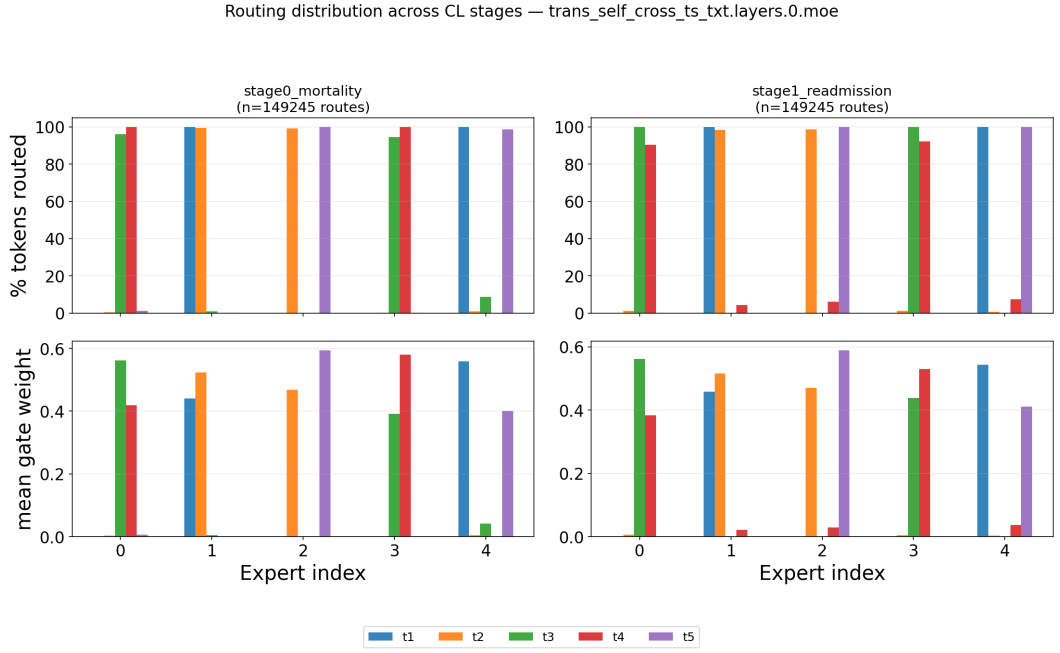


Figure 20: S2: per-modality, per-stage routing under eICU continual learning MOR→RAD. The two tasks share the same multivariate time-series modality bank (T1–T5), and the corresponding routing fingerprints overlap almost identically across stages.

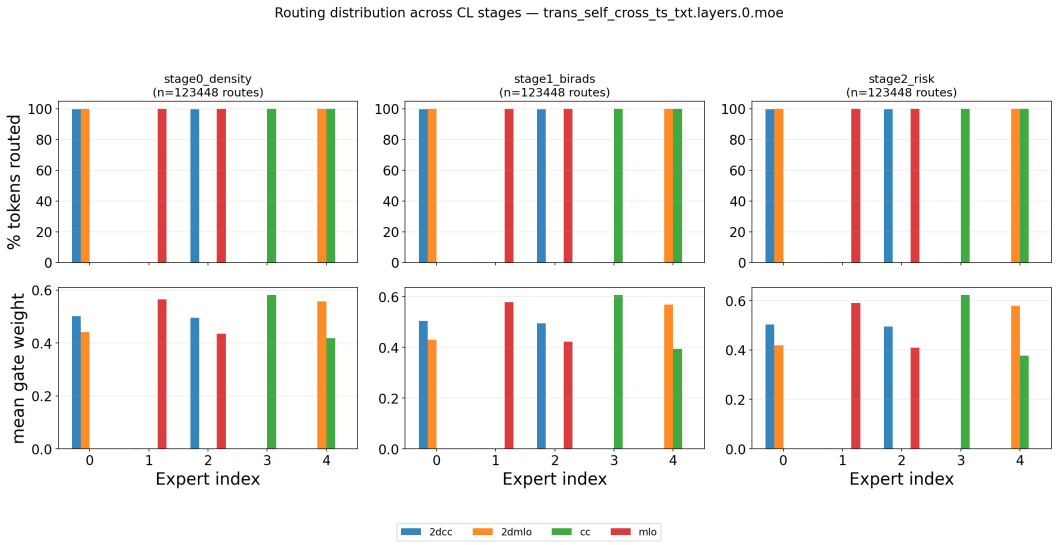


Figure 21: S3: per-modality, per-stage routing under EMBED continual learning DENSITY→BIRADS→RISK. The four mammography views (`cc`, `mlo`, `2dcc`, `2dmlo`) keep their view-specific expert assignments across all three stages with negligible drift, providing the routing-level analogue to the single-task EMBED panels of Fig. 16.

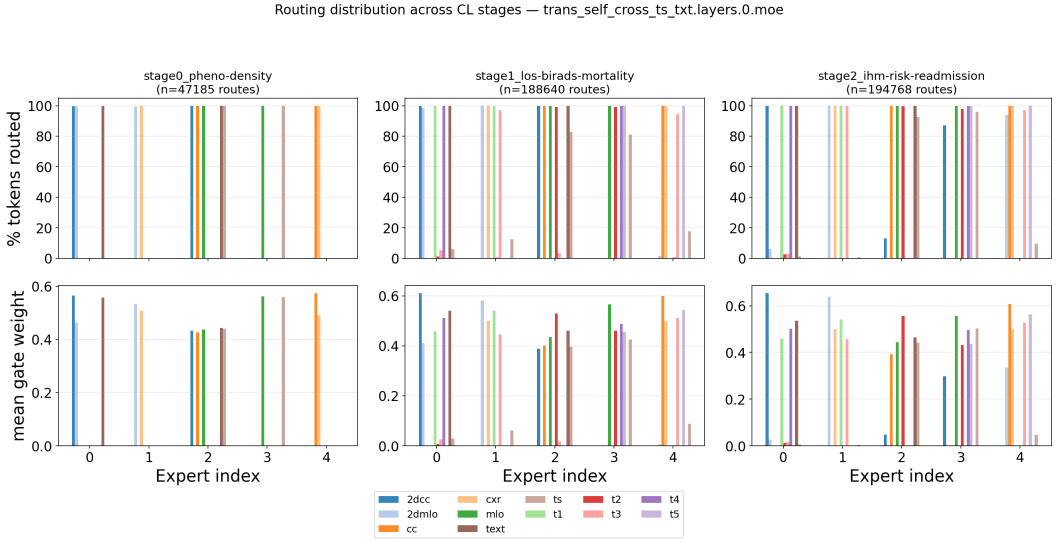


Figure 22: S4: per-modality, per-stage routing under the mixed cross-dataset continual sequence {PHENO, DENSITY}→{LOS, BIRADS, MOR}→{IHM, RISK, RAD}. MIMIC-IV (`TS`/`Text`/`CXR`), EMBED (`cc`/`mlo`/`2dcc`/`2dmlo`), and eICU (`T1`–`T5`) modalities occupy disjoint expert subsets that persist across stages; cross-dataset interference is therefore bounded structurally rather than by an auxiliary loss.

Routing isolates modalities. Each router reads only its own modality's summary (Eq. (1)), so its gradient with respect to any other modality is zero and no router can observe a cross-modal conflict at routing time. Empirically, cross-modality routing spillover is exactly zero in all 186 conditions: corrupting one modality leaves every other modality's gate unchanged. Concatenation-based fusion cannot provide this, since all modalities compete under one softmax for one top-K budget and a high-norm modality can capture capacity and displace the others. In the continual setting the isolation is stronger still, because prior router heads are frozen. Routers do respond to atypicality *within* a modality, since an ambiguous image shifts its own summary and can alter its routing.

Routers do not downweight unreliable modalities; the fusion head does. Each gate is normalized within its modality and sums to one however corrupted the input is, so routing changes *which* experts a modality uses, never *how much* it contributes. Routing does react to corruption, with the Jensen–Shannon divergence from the clean gate rising monotonically with the noise level for all 31 (task, modality) pairs, but that reaction is uncorrelated with the modality's importance (Spearman $\rho = +0.14$, $p = 0.44$, $n = 31$). Reliability weighting instead resides in the fusion head: each modality's learned share of the fusion logit correlates with its measured importance (the AUROC drop on removal) at $\rho = +0.73$ ($p = 2.7 \times 10^{-6}$, $n = 31$), and per task at $\rho = +1$ on 5 of 8 tasks, although with 3–5 modalities per task only MOR is significant on its own (Tab. 22). FLAME thus learns which modalities to trust, amortized over training, but does not detect that a particular input is degraded.

Sensitivity. Tab. 21 corrupts each task's dominant modality, the one carrying the largest learned share of that task's fusion output. The median relative AUROC drop is 4.8% under $4\times$ noise against 15.1% under shuffling: zero-mean noise is largely absorbed by the temporal attention and the experts, whereas a shuffled modality supplies confidently wrong evidence that the fusion head keeps weighting at full strength. The ordering reverses on BIRADS and RISK, where noise directly destroys the signal carried by the mammography embeddings, so the relative severity of the two corruptions is task-dependent.

Table 21: Relative AUROC change when each task's dominant modality is corrupted (Setup 4 models).

Corruption	48-IHM	LOS	25-PHENO	MOR	DENSITY	RAD	BIRADS	RISK
$4\times$ noise	-0.1%	-0.7%	-2.5%	-3.6%	-6.0%	-18.3%	-9.3%	-17.1%
Shuffled	-28.3%	-31.4%	-18.3%	-12.0%	-10.9%	-21.0%	-6.8%	-9.4%

Table 22: Rank correlations behind App. D.6 (Setup 4 FLAME-CL, 31 (task, modality) pairs, 186 corrupted evaluations). Importance is the AUROC lost when the modality is removed; routing reaction is the Jensen–Shannon divergence of the gate distribution of a modality from its clean one. Per-task p-values are exact permutation tests ($n = 3$–5 modalities).

Pooled over all pairs	Spearman ρ	p	Pearson r
Routing JSD at $2\times$ noise vs. importance	$+0.14$	0.44	-0.14
Routing JSD at $4\times$ noise vs. importance	$+0.06$	0.75	-0.19
Routing JSD under shuffling vs. importance	$+0.12$	0.53	$+0.27$
Routing JSD at $2\times$ noise vs. contribution share	$+0.13$	0.48	-0.21
Fusion contribution share vs. importance	$+0.73$	2.7×10^{-6}	$+0.80$
Fusion magnitude share vs. importance	$+0.53$	2.4×10^{-3}	$+0.79$

Share vs. importance, per task	PHENO	DENSITY	LOS	BIRADS	MOR	IHM	RISK	RAD
Modalities n	3	4	3	4	5	3	4	5
Spearman ρ	$+1.00$	$+0.80$	$+1.00$	$+1.00$	$+1.00$	$+1.00$	$+0.00$	$+0.20$
Exact p	0.33	0.33	0.33	0.08	0.02	0.33	1.00	0.78
Smallest attainable p	0.33	0.08	0.33	0.08	0.02	0.33	0.08	0.02

Missingness. Removal is the missing-modality case, which our training data already contains: MIMIC-IV stays without linked imaging and ADNI's modality-stratified subsets are retained with the modality marked absent (App. C). An absent modality simply does not invoke its router, with no effect on the other gates and no imputation or missing-modality bank, unlike FuseMoE and Flex-MoE.

Sensitivity varies sharply within a task: on DENSITY, removing the `cc` view costs 0.0% while removing `2dcc` costs 5.5%; on RAD, removing `T4` costs 22.7% while `T1`, `T3` and `T5` cost nothing. A single modality-robustness number is therefore dominated by which modality is dropped.

Design trade-off. FLAME contains modality conflict rather than arbitrating it. Its stability and its lack of per-sample adaptivity are the same property: a router keyed to modality type cannot condition on which other modalities are present, which is exactly what makes it reusable across heterogeneous modality pools and robust to modality shift. A reliability gate conditioned jointly on all present modalities would restore per-sample adaptivity but reintroduce dependence on the modality combination.

D.7 Beyond Healthcare

D.7.1 Video–Audio–Text Benchmarks

Healthcare supplies the modality and task diversity our setting requires: every signal ties to a unique patient, and each patient supports several downstream tasks over different modality subsets. General multimodal corpora rarely offer both, being built mainly for contrastive pretraining on a single modality pair rather than for downstream prediction. To test the mechanisms outside clinical data, we interleave CMU-MOSEI [81] (7-class sentiment and multi-label emotion intensity) and UR-FUNNY [26] (binary humor), both video–audio–text, with MIMIC-IV tasks in the sequence PHENO→sentiment→humor→LOS→intensity→IHM, at reserved rank 32 (Tab. 23). Every row is flat after the task's own stage to within 0.01 AUROC, including across the switches between clinical and non-clinical modalities, so the routing and compression mechanisms carry over beyond healthcare.

Table 23: FLAME-CL on a stream interleaving MIMIC-IV with CMU-MOSEI and UR-FUNNY. Rows: tasks; columns: evaluation after each stage. AUROC: binary for LOS, IHM and humor; macro-averaged over labels (ml) for PHENO and intensity and over classes (mc) for 7-class sentiment. Last row: 7-class sentiment accuracy. Rank 32; standard deviations are at most 0.01.

Task	PHENO	→sentiment	→humor	→LOS	→intensity	→IHM
PHENO (ml)	0.71	0.70	0.70	0.70	0.70	0.70
Sentiment (mc)	–	0.82	0.82	0.82	0.82	0.82
Humor	–	–	0.69	0.68	0.68	0.68
LOS	–	–	–	0.82	0.82	0.82
Intensity (ml)	–	–	–	–	0.73	0.73
IHM	–	–	–	–	–	0.80
Sentiment (Acc-7)	–	0.50	0.50	0.50	0.50	0.50

D.7.2 Scaling to OLMoE-1B-7B

Setup. We apply FLAME-CL to OLMoE-1B-7B [52] (7B parameters, 64 experts $\times$ 16 layers) and fine-tune it over a sequential stream of five generative tasks spanning different domains: dialogue summarization (SAMSum [17]), mathematical reasoning (GSM8K [10]), program synthesis (MBPP [4]), news summarization (CNN/DailyMail [62, 28]), and constrained commonsense generation (CommonGen [48]). For every method we attach rank-8 LoRA adapters to the `gate_proj`, `up_proj` and `down_proj` of every expert and to the 16 routers, keeping the base weights frozen. FLAME-CL trains a fresh adapter on each incoming task and truncates it to rank 2 before freezing it under the cursor. IND trains independent MoE-LoRA adapters per task, and SEQ stacks all adapters without cursor-restricted inference. All methods are evaluated after the fifth stage on 500 held-out examples per task, with ROUGE-L for SAMSum, CNN/DailyMail and CommonGen, exact match for GSM8K, and executed pass@1 for MBPP. This experiment is text-only with five stages: it tests whether the continual-learning mechanism survives scale, not multimodality.

Results. Tab. 3 shows three effects. *Cursor isolation buys retention*: SEQ falls 8.2 mean points below FLAME-CL, with SAMSum and GSM8K ending below their zero-shot values; this is the cursor ablation of Fig. 6a at 7B scale, with a far larger effect on generative metrics than on closed-set classification. FLAME-CL *matches independently trained adapters* while storing $3.7\times$ fewer parameters (104.9M vs. 388.0M, i.e., 1.5% vs. 5.6% of the base parameters); SEQ stores $1.35\times$ fewer parameters than FLAME-CL but trails it on every task. *The relative footprint shrinks with*

scale: the per-stage cost is linear in the reserved rank while the base grows with width and depth, and the capacity bound of Remark 8 loosens with the ambient dimension.

Functional energy at scale. Fig. 23 and Tab. 24 repeat the diagnostic of Fig. 3 for each stage's LoRA update ΔW (median over experts, layers 8 and 15). At $r = 2$ the update keeps 74–82% of its weight-only energy but 96–99% of its data-aware energy along routed activations, and already 92–97% at $r = 1$: the discarded directions carry weight mass but almost no function, which is the gap Proposition 1 predicts. The backbone is not degenerate: the effective weight $W_{\text{base}} + \Delta W$ reaches only about 56–59% data-aware energy at rank 128, so low rank is a property of the update rather than of OLMoE. The pattern is unchanged from the first stage to the last, so functional rank does not inflate as reserved components accumulate. Proposition 1 was motivated by experts trained from scratch on multimodal clinical data, yet the same signature appears for LoRA updates on a frozen pretrained text MoE at roughly $100\times$ the parameter count.

Table 24: OLMoE-1B-7B: share of each stage's LoRA update energy retained at the reserved rank (median over experts, layers 8 and 15).

Stage	Task	Weight-only @ r=2	Data-aware @ r=2	Data-aware @ r=4
0	SAMSum	74%	98%	99%
1	GSM8K	80%	99%	100%
2	MBPP	75%	96%	99%
3	CNN/DailyMail	78%	99%	100%
4	CommonGen	82%	99%	100%

E Additional Methodological Details

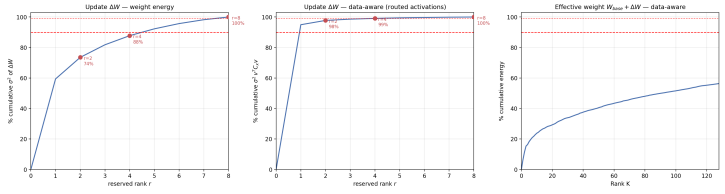
Stage-t trainable and frozen parameter sets. At each continual stage t, the trainable parameters are: the additive expert components $\{\widetilde{W}_i^{(t)}\}_{i=1}^N$; the analogous additive components on the variable-length attention layers inside each modality encoder ϕ_m, which are compressed and stacked under the same scheme as the experts; the new per-modality router heads $\{G_m^{(t)}\}_{m\in\mathcal{M}_t}$; the new task head $h^{(t)}$; and any genuinely new modality encoder instantiated at stage t. All other parameters are frozen: the multitask-pretrained base $W_i^{(0)}$, every prior reserved component (both expert and encoder), every prior router head $\{G_m^{(s)}\}_{s<t}$, prior task heads, and the static parts of encoders for modalities already seen. After stage-t convergence we apply Eq. (3) to obtain $W_i^{(t)}$ together with the analogous rank-r_t truncations on the encoder attention layers, then freeze them along with $G_m^{(t)}$ and $h^{(t)}$. The per-stage memory footprint is $r_t(p + d + 1)$ scalars per stacked weight (factored $U, \Sigma, V^\top$ form) plus the fixed router and head overheads, so the cumulative reserved rank $\sum_t r_t$ controls how many tasks fit within a fixed-size MoE. Algorithm 1 below describes the full procedure.

F Limitations

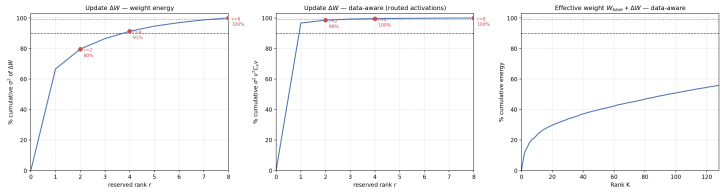
Scope of evaluation. Our multimodal evaluation is centered on healthcare. Video–audio–text benchmarks (App. D.7.1) and a 7B-parameter text MoE (App. D.7.2) indicate that the routing and compression mechanisms carry over, but large-scale vision–language continual learning and domains such as autonomous driving and robotics remain to be tested.

Task identity at inference. Our continual-learning protocol is task-incremental: task identity selects the output head, which is standard when evaluating CL methods with task-specific heads [67, 8, 12, 29]. Without it, FLAME can still emit all task outputs in one pass by enumerating heads, since each head is paired with a fixed cursor and router set; encoders and the expert pool are shared, so the marginal cost is forming $W_i^{\text{eff}}(\tau)$ (Eq. (4)) and evaluating the router heads. This suits clinical use, where questions such as mortality, length of stay and phenotype are asked together for the same patient rather than being mutually exclusive labels.

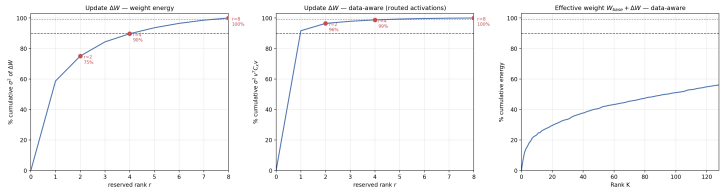
Capacity and parameter growth. Parameters grow with the number of stages at a rate set by the reserved rank, and a constant per-stage rank r admits about $\lfloor d/r \rfloor$ stages of fresh directions



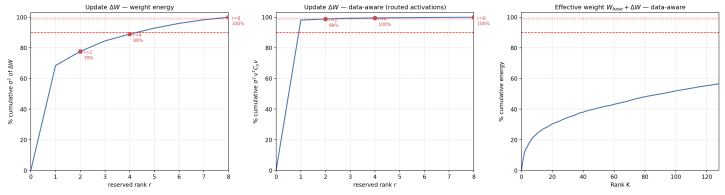
(a) Stage 0: SAMSum.



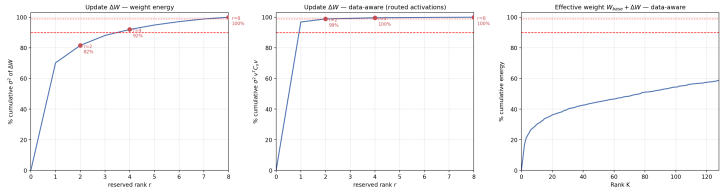
(b) Stage 1: GSM8K.



(c) Stage 2: MBPP.



(d) Stage 3: CNN/DailyMail.



(e) Stage 4: CommonGen.

Figure 23: OLMoE-1B-7B rank-reservation diagnostics, one row per stage (median over experts, layers 8 and 15; pretrained weights frozen, only the stage's update ΔW truncated). Left: cumulative σ^2 of ΔW (weight-only). Center: cumulative $\sigma^2 \boldsymbol{v}^\top C \boldsymbol{v}$ of ΔW along routed activations (data-aware). Right: data-aware energy of the effective weight $W_{\text{base}} + \Delta W$.

Algorithm 1 FLAME-CL: multi-task pretraining and continual adaptation.

Require: Stage data $\{\mathcal{D}_t\}_{t\geq 0}$, with tasks $\mathcal{T}_t$ and modalities $\mathcal{M}_t$ at stage t; per-stage rank r_t.

Stage 0 **— multi-task pretraining (Sec. 2.1).**

Initialize full-shape experts $\{\widetilde{W}_i^{(0)}\}_{i=1}^{N}$, per-modality routers $\{G_m^{(0)}\}_{m\in\mathcal{M}_0}$, encoders $\{\phi_m\}_{m\in\mathcal{M}_0}$, and task heads $\{h^{(t)}\}_{t\in\mathcal{T}_0}$.

Train: minimize $\mathcal{L}_0$ on $\mathcal{D}_0$ over all of the above.

Compress & stack: take the SVD $\widetilde{W}_i^{(0)} = U_i^{(0)}\Sigma_i^{(0)}V_i^{(0)\top}$ and initialize the expert stack with its rank-r_0 truncation,

$$W_i^{(0)} \;\leftarrow\; U^{(0)}_{i,\,1:r_0}\,\Sigma^{(0)}_{i,\,1:r_0}\,V^{(0)\top}_{i,\,1:r_0}, \qquad \Pi_i \;\leftarrow\; (\,W_i^{(0)}\,) \quad \text{(Eq. 3)}.$$

Apply the same compress-and-init to the variable-length attention layers inside each ϕ_m.

Freeze: $\{W_i^{(0)}\}$, $\{G_m^{(0)}\}$, $\{h^{(t)}\}_{t\in\mathcal{T}_0}$, and $\{\phi_m\}_{m\in\mathcal{M}_0}$.

Stage $t \geq 1$ **— continual adaptation (Sec. 2.2).**

for $t = 1, 2, \ldots$ **do**

  Expand: for each expert i, attach a fresh $\widetilde{W}_i^{(t)}$ of the same shape as W_i with its output layer zero-initialized; add routers $\{G_m^{(t)}\}_{m\in\mathcal{M}_t}$, task head $h^{(t)}$, and instantiate ϕ_m for any modality $m \in \mathcal{M}_t \setminus \bigcup_{s<t}\mathcal{M}_s$.

  Train: minimize $\mathcal{L}_t$ on $\mathcal{D}_t$ over $\{\widetilde{W}_i^{(t)}\}_i \cup \{G_m^{(t)}\}_{m\in\mathcal{M}_t} \cup \{h^{(t)}\}$ and any newly instantiated ϕ_m; all parameters from stages $s < t$ are held fixed.

  Compress & stack: take the SVD $\widetilde{W}_i^{(t)} = U_i^{(t)}\Sigma_i^{(t)}V_i^{(t)\top}$ and append its rank-r_t truncation to the expert stack,

$$W_i^{(t)} \;\leftarrow\; U^{(t)}_{i,\,1:r_t}\,\Sigma^{(t)}_{i,\,1:r_t}\,V^{(t)\top}_{i,\,1:r_t}, \qquad \Pi_i \;\leftarrow\; \Pi_i \,\|\, W_i^{(t)} \quad \text{(Eq. 3)}.$$

  Apply the same compress-and-stack to the variable-length attention layers inside each ϕ_m.

  Freeze: $\{W_i^{(t)}\}$, $\{G_m^{(t)}\}_{m\in\mathcal{M}_t}$, $h^{(t)}$, and any newly added encoder weights.

end for

Inference for task t **first trained at stage** $\tau = k(t)$**.**

Form the effective expert weights $W_i^{\text{eff}}(\tau) = \sum_{j=0}^{\tau} W_i^{(j)}$ (Eq. 4).

Route with $\{G_m^{(\tau)}\}_{m\in\mathcal{M}_t}$ and predict through head $h^{(t)}$.

(Remark 8). Very long streams therefore need a smaller or decreasing rank schedule; the falling required rank in Tab. 19 suggests this is viable, but we have not explored it.

Reliance on low-rank functional energy. Compression is near-lossless when the functional energy of each stage's update concentrates in a low-rank subspace (Propositions 1 and 5). Where it does not, the truncation cost in Eq. (20) grows and the rank must be raised at the expense of capacity; this cost is paid once per task and does not compound across stages.

Modality reliability. Per-modality routers contain cross-modal conflict but do not arbitrate it: they cannot downweight a degraded input per sample, and reliability weighting is amortized in the fusion head (App. D.6). A reliability gate conditioned on all present modalities would add per-sample adaptivity at the cost of reintroducing dependence on the modality combination.

G Broader Impact

FLAME's flexi-modal multi-task and continual learning capabilities are well-suited to clinical settings, where modality availability differs across institutions and new prediction tasks emerge after deployment. By enabling a single shared backbone to absorb new tasks while training 5–$15\times$ fewer encoder parameters than full fine-tuning, FLAME lowers the compute and validation overhead of maintaining multimodal clinical models. We caution, however, that healthcare deployment requires rigorous prospective validation, fairness auditing across patient subgroups, and institutional oversight; structural no-forgetting reduces but does not eliminate the risk of stale or miscalibrated predictions on shifting populations.

H Asset Licenses

All four datasets used in this paper are publicly available under data use agreements that require credentialed access and prohibit redistribution or re-identification. **MIMIC-IV** [37] and the linked **MIMIC-CXR** imaging are released by the MIT Laboratory for Computational Physiology under the PhysioNet Credentialed Health Data License 1.5.0 and require completion of CITI "Data or Specimens Only Research" training. **eICU** [55] is similarly distributed via PhysioNet under the same credentialed license. **EMBED** [34] is released by Emory University via the AWS Open Data program under a custom data use agreement that restricts use to non-commercial research and prohibits attempts at patient re-identification. **ADNI** [73] is distributed by the Laboratory of Neuro Imaging at USC and requires a signed ADNI Data Use Agreement; users must additionally acknowledge the ADNI funding sources in any publication. We accessed all four datasets in compliance with their respective terms, used them solely for non-commercial academic research, and do not redistribute any raw data or derivatives that could enable re-identification.

For baseline models, FuseMoE [24] is released under BSD 3-Clause license. HighMMT [46] is released under MIT license. The remaining code base assets do not have any license inforamtion.

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: The abstract and introduction accurately describe FLAME's three contributions (multitask pretraining, continual learning, interoperability), all of which are directly supported by experiments in Section 3. No claim exceeds what is demonstrated in the paper.

   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: Appendix F discusses the limitations of the work.

   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: [Yes]

   Justification: Yes, the detailed assumptions and proofs can be found at Appendix B.

   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 have included the code as part of supplementary material, a simple demo of our method in Appendix to ensure reproducibility.

   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: [Yes]

   Justification: All datasets are publicly accessible to credentialed researchers. Code is available at `anonymous.4open.science/r/flame-5992/README.md`

   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 have included full experiment details in Appendices C - E.

   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: [Yes]

   Justification: Error bars are included in the result tables from Section 3.

   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 discussed such information in Appendix D.

   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: This research does not involve human subjects or participant interaction. All experiments use de-identified clinical datasets accessed under data use agreements of corresponding platform, which we comply with fully, including not transmitting data to external APIs and not releasing personally identifiable information. We do not foresee harmful societal consequences from this work.

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: Appendix G discusses the potential societal impacts of the work.

    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: [N/A]

    Justification: The paper does not release a new pretrained model or dataset. The released code trains multimodal transformers for clinical prediction tasks on existing credentialed datasets, which pose no high risk for misuse.

    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: The dataset/code/models creators are properly cited in Section 3 and Appendix C. Licenses and terms of use for datasets and code bases are listed in Appendix H.

    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: [Yes]

    Justification: The experiment code is released via an anonymized repository. No new datasets or model weights are released. We included a README file in the repository.

    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: The paper does not involve crowdsourcing or research with human subjects. All experiments use pre-existing de-identified EHR datasets.

    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: The paper does not involve research with human subjects. All experiments use pre-existing de-identified EHR datasets, and IRB approval is not required.

    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: [N/A]

    Justification: LLMs are used only for writing, editing, and surveying assistance. This usage did not influence the scientific methodology, experimental design, or originality of the research.

    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.