DEAD WEIGHTS, LIVE SIGNALS: FEEDFORWARD GRAPHS OF FROZEN LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

We present a feedforward graph architecture in which heterogeneous frozen large language models serve as computational nodes, communicating through a shared continuous latent space via learned linear projections. Recent work has demonstrated geometric compatibility between independently trained LLM latent spaces, suggesting that cross-architecture activation translation is possible via linear projection. This work extends that finding from static two-model steering to end-to-end trainable multi-node graphs, where projection matrices are optimized jointly via backpropagation through residual stream injection hooks. Three small frozen models (Llama-3.2-1B, Qwen2.5-1.5B, Gemma-2-2B) project into a shared latent space injected into two larger frozen models (Phi-3-mini, Mistral-7B), whose representations feed a cross-attention output node. With only 17.6M trainable parameters against approximately 12B frozen, the architecture achieves 87.3% on ARC-Challenge, 82.8% on OpenBookQA, and 67.2% on MMLU, outperforming the best single constituent model by 11.4, 6.2, and 1.2 points and outperforming majority vote of the same five models by 8.2, 8.0, and 10.4 points — benchmarks where output-level ensembling actively degrades performance. Gradient flow through frozen model boundaries is empirically verified to be tractable, and the output node develops selective routing toward Phi-3-mini without supervision, validated by a 7.8pp performance collapse upon removal.

1 INTRODUCTION

The prevailing paradigm for improving language model performance is scaling: larger models, more data, more compute. Yet smaller models trained on specialized corpora routinely outperform larger general models on narrow tasks Abdin et al. (2024); Qwen et al. (2025), and models trained with different objectives or architectures exhibit systematically different strengths at equivalent parameter counts. The information needed to solve a given problem may already exist, distributed unevenly across a population of existing models — the challenge is aggregating it.

Current ensemble and multi-agent approaches pursue this at the output level: routing queries to specialized models Jiang et al. (2024), ensembling token distributions Qiu et al. (2025), or having models critique each other's responses Wu et al. (2024); Moura (2023). These methods treat each model as a black box, and their limitations are concrete: majority vote over five heterogeneous models actually *degrades* performance on MMLU and OpenBookQA relative to the best single model, because weaker models corrupt the vote. The geometric structures in LLM latent spaces that encode semantic relationships, reasoning patterns, and factual knowledge remain inaccessible to output-level approaches.

Recent work Armstrong et al. (2026) showed this need not be the case: independently trained LLMs converge to geometrically compatible latent spaces, and a linear projection suffices to translate activations between heterogeneous architectures. We exploit this compatibility in a new direction — not for inference-time correction between two models, but as a *differentiable communication medium* for a feedforward graph of frozen LLMs. Learned projection matrices translate each node's hidden state into a shared latent space; the aggregate signal is injected into downstream nodes via residual stream hooks; and the entire system is trained end-to-end via backpropagation through the

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

frozen model boundaries. Only the projection matrices and a lightweight cross-attention output node are updated — approximately 17.6M parameters against 12B frozen.

We make four contributions:

1. **Tractable gradient flow.** Gradients flow cleanly through multiple frozen LLM boundaries, with layer-1 projection matrices receiving approximately 13% of the gradient signal at the output node — attenuated but sufficient for learning, with skip connections providing no additional benefit.
2. **Strong benchmark performance.** The graph achieves 87.3% on ARC-Challenge, 82.8% on OpenBookQA, and 67.2% on MMLU, outperforming the best single constituent model by 11.4, 6.2, and 1.2 percentage points respectively, and outperforming majority vote of the same five models by 8.2, 8.0, and 10.4 points — benchmarks where output-level ensembling is actively harmful.
3. **Comprehensive ablations.** The graph outperforms parameter-matched learned classifiers by 9.1, 5.2, and 6.7 points across three benchmarks. Systematic ablations confirm robustness across a $10\times$ range of injection coefficient and $8\times$ range of shared dimension, and a prompt ablation establishes that performance is stable within 1pp across all framing strategies — representational diversity derives from model heterogeneity, not prompt engineering.
4. **Emergent selective routing.** Without supervision, the output node develops asymmetric attention over the two layer-2 nodes, preferentially weighting Phi-3-mini — a pattern consistent with its synthetic training corpus inducing geometrically regular representations, and validated by a 7.8pp performance collapse when Phi-3-mini is removed.

2 Related Work

Activation steering and residual stream intervention. Prior work has shown that transformer behavior can be controlled by directly manipulating internal activations without modifying weights. Turner et al. (2025) demonstrated that fixed bias vectors reliably shift model behavior across behavioral dimensions, and Zou et al. (2023) showed that interpretable concept directions generalize across tasks and model families. These methods establish the residual stream as a writable medium, a foundational assumption of our injection mechanism. Recent work Armstrong et al. (2026) extended this to cross-architecture settings, showing that linear projections suffice to translate activations between independently trained LLMs, with dense projections substantially outperforming sparse alternatives ($R^2 \approx 0.50$ vs. ≈ 0.01), and that domain-specific projection matrices motivate end-to-end training against a task loss rather than offline fitting.

Model composition and ensemble methods. Mixture-of-experts architectures Jiang et al. (2024) route tokens to specialized subnetworks within a single model, while multi-agent frameworks Wu et al. (2024); Moura (2023) coordinate LLMs through natural language. Both operate at the token level; internal representations remain inaccessible across model boundaries. Model stitching Bansal et al. (2021) demonstrated that independently trained networks can be composed via a learned linear adapter, establishing affine compatibility as an empirically grounded prior. We extend this from pairwise adapters to a full graph topology with jointly optimized projection matrices. The Platonic Representation Hypothesis Huh et al. (2024) provides theoretical grounding: neural networks converge to a shared underlying representation of reality, with architecture-specific differences reducible to coordinate transformations.

Frozen backbone adaptation. Parameter-efficient methods such as LoRA Hu et al. (2021) and prefix tuning Li and Liang (2021) adapt a single frozen model with minimal parameters, but do not address how multiple frozen models might communicate. Bai et al. (2025) showed that frozen LLM blocks act as gradient coherence rectifiers, producing stable gradient flow through frozen boundaries — directly addressing the theoretical concern about our architecture's trainability. Yin and Wang (2025) studied multi-node LLM computation graphs and recommended skip connections and auxiliary losses to mitigate gradient attenuation; our empirical analysis finds neither necessary at two-layer depth (Section 5). Our work differs from all of the above by composing *multiple* frozen

models into a graph where inter-node communication occurs in a shared continuous latent space learned from the task signal.

3 METHOD

We define a feedforward graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where each node $v \in \mathcal{V}$ is a frozen pretrained LLM and each directed edge $(u, v) \in \mathcal{E}$ carries a learned projection matrix $W_{uv} \in \mathbb{R}^{d_s \times d_u}$. Only the projection matrices and a lightweight output node are updated during training; all LLM weights θ_v remain frozen throughout.

3.1 ARCHITECTURE

Layer 1: Diverse Encoding. Three heterogeneous frozen LLMs — Llama-3.2-1B (d=2048), Qwen2.5-1.5B (d=1536), Gemma-2-2B (d=2304) — each receive the input with a distinct framing prefix ("analyze the factual content", "analyze the reasoning structure", "analyze the language and framing"). Each node produces a hidden state at relative depth $l_T = 0.90$, extracted at the final token position and L2-normalized:

$$\mathbf{h}_i = \frac{\text{hidden}(x \oplus p_i,\, \theta_i,\, l_T)}{\|\text{hidden}(x \oplus p_i,\, \theta_i,\, l_T)\|_2} \quad i \in \{1, 2, 3\} \tag{1}$$

We extract at depth 0.90 because deeper layers yield more fully resolved semantic representations; L2 normalization ensures the projection learns directional alignment rather than magnitude differences. As established empirically in Section 4, the specific framing content is non-critical, performance varies by less than 1pp across all prompt strategies including identical prompts for all three nodes; representational diversity derives from the architectural heterogeneity of the frozen models rather than from the prefix choice.

Shared Latent Space. Each hidden state is projected and averaged into a shared vector:

$$\mathbf{z}_1 = \tfrac{1}{3} \sum_{i=1}^{3} W_i \, \mathbf{h}_i \qquad W_i \in \mathbb{R}^{d_s \times d_i}, \quad d_s = 1024 \tag{2}$$

$d_s = 1024$ is deliberately neutral — not equal to any constituent model's hidden dimension — so every projection must perform genuine geometric work rather than approximate an identity mapping.

Layer 2: Injection and Refinement. $\mathbf{z}_1$ is injected into the residual stream of each layer-2 node — Phi-3-mini (d=3072) and Mistral-7B (d=4096) — at relative depth $l_S = 0.75$. Since $d_s = 1024 \neq d_j$, we linearly interpolate $\mathbf{z}_1$ to the target dimension before injection; this interpolation is differentiable, preserving gradient flow to the layer-1 matrices. The blend follows the magnitude-rescaling formulation of Armstrong et al. (2026):

$$\mathbf{h}' = (1 - \alpha)\, \mathbf{h}_{\text{node}} + \alpha(\tilde{\mathbf{z}}_1 \cdot \|\mathbf{h}_{\text{node}}\|_2)\,, \quad \alpha = 0.25 \tag{3}$$

where $\tilde{\mathbf{z}}_1$ is the interpolated injection vector. Magnitude rescaling ensures the injected signal is compatible with the residual stream regardless of dimensional differences. We use $\alpha = 0.25$; our ablation confirms representations become fragile at higher values ($\alpha = 0.75$ collapses to 24.4% on ARC-Challenge, Section 4), consistent with prior work on injection fragility Armstrong et al. (2026). Each layer-2 node completes its forward pass and its hidden state is projected:

$$\mathbf{z}_2^{(j)} = W_{j+3}\, \mathbf{h}'_j \quad j \in \{4, 5\} \tag{4}$$

Output Node. A cross-attention output node aggregates both layer-2 representations:

$$\mathbf{o} = \text{LayerNorm}\Big(\text{MHA}\Big(\mathbf{q},\, [\mathbf{z}_2^{(4)}, \mathbf{z}_2^{(5)}],\, [\mathbf{z}_2^{(4)}, \mathbf{z}_2^{(5)}]\Big)\Big)\,, \quad \hat{y} = \text{softmax}(W_{\text{cls}}\, \mathbf{o}) \tag{5}$$

where $\mathbf{q} \in \mathbb{R}^{d_s}$ is a learned query, MHA uses 4 heads, and $W_{\text{cls}} \in \mathbb{R}^{4 \times d_s}$ maps to the four answer choices. The attention weights implicitly route the prediction between the two layer-2 nodes based on their representations' compatibility with the learned query. Figure 1 illustrates the complete architecture.

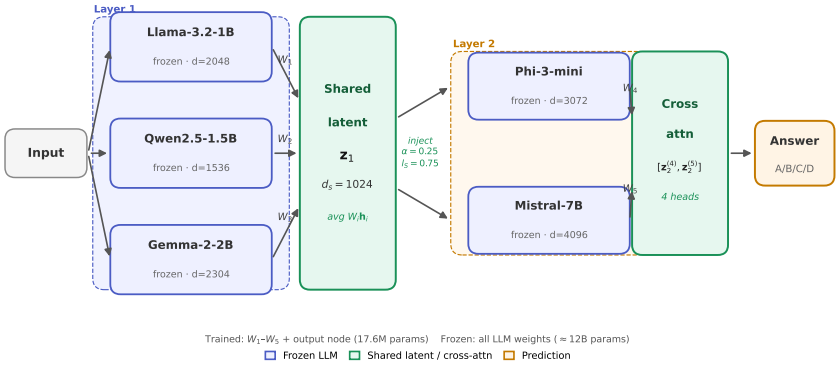

Figure 1: **Frozen LLM Graph Architecture.** Three heterogeneous frozen layer-1 models encode the input with distinct framings; their hidden states are projected via W_1, W_2, W_3 into a shared latent space and averaged to form $\mathbf{z}_1$, which is injected into two frozen layer-2 models at l_S=0.75. Layer-2 representations are projected via W_4, W_5 into a second shared space, over which a cross-attention output node produces the final prediction. Only the five projection matrices and output node (17.6M params) are trained.

3.2 Training

We minimize cross-entropy loss $\mathcal{L} = -\log \hat{y}_{c^\star}$ where $c^\star$ is the correct answer index. Gradients flow backward through the output node, through W_4 and W_5, through the differentiable injection operations, and into W_1, W_2, W_3. Frozen LLM weights receive no gradients.

We use AdamW Loshchilov and Hutter (2017) with $\eta_W = 10^{-3}$ for projection matrices and $\eta_{\text{out}} = 10^{-4}$ for the output node, weight decay 10^{-4}, gradient clip 1.0, effective batch size 8, and cosine annealing. All models are loaded in `bfloat16` via HuggingFace Wolf et al. (2020); the full graph fits on a single 80GB A100. Injection hooks are registered immediately before each forward pass and removed afterward to prevent cross-example contamination. Full hyperparameters are in Table 3; trainable parameter counts are in Table 5 (Appendix).

4 Experiments

4.1 Datasets and Baselines

We evaluate on three multiple-choice benchmarks covering distinct reasoning demands. **MMLU** Hendrycks et al. (2021) provides 14,042 training and 1,531 validation questions across 57 subjects (STEM, humanities, social sciences). **ARC-Challenge** Clark et al. (2018) provides 1,119 training and 1,165 test questions drawn from standardized science tests. **OpenBookQA** Mihaylov et al. (2018) provides 4,957 training and 500 test questions requiring integration of a core science fact with commonsense inference. Together the three benchmarks cover broad factual recall, standardized scientific reasoning, and fact-integrated inference.

We compare against three baselines under the same zero-shot greedy decoding protocol. **Single-model greedy**: each constituent model evaluated individually with no learned components. **Majority vote**: output-level ensemble over all five constituent models with no learned parameters, representing the strongest parameter-free combination baseline. **Parameter-matched learned head**: an MLP classifier trained on the frozen hidden states of the strongest single-model baseline, with parameter budget matched to the graph network's 17.6M parameters (16.8M for Mistral-7B on MMLU; 22.8M for Qwen2.5-1.5B on ARC and OpenBookQA). This baseline isolates whether gains arise from the communication mechanism or from the learned classifier alone.

Table 1: Results on MMLU, ARC-Challenge (ARC), and OpenBookQA (OBQA). Best single-model result per benchmark is underlined. Majority vote requires no learned parameters. The learned head uses Mistral-7B for MMLU and Qwen2.5-1.5B for ARC and OBQA.

Method	**MMLU**	**ARC**	**OBQA**
Single-model greedy			
Llama-3.2-1B-Instruct	44.0%	50.8%	56.6%
Qwen2.5-1.5B-Instruct	59.0%	<u>75.9%</u>	<u>76.6%</u>
Gemma-2-2B-IT	60.0%	67.1%	60.8%
Phi-3-mini-4K-Instruct	43.0%	75.3%	62.2%
Mistral-7B-Instruct-v0.3	<u>66.0%</u>	73.9%	70.4%
Output-level ensemble (no learned parameters)			
Majority vote (5 models)	56.8%	79.1%	74.8%
Parameter-matched learned head on best single model			
Head on Mistral-7B (16.8M)	60.5%	—	—
Head on Qwen2.5-1.5B (22.8M)	—	78.2%	77.6%
Ours			
Frozen LLM graph (17.6M)	**67.2%**	**87.3%**	**82.8%**
Margins			
vs. best single model	+1.2pp	+11.4pp	+6.2pp
vs. majority vote	+10.4pp	+8.2pp	+8.0pp
vs. learned head	+6.7pp	+9.1pp	+5.2pp

4.2 Results

The graph network outperforms all baselines on every benchmark. The majority vote results are particularly revealing: while majority vote over five heterogeneous models improves over the best single model on ARC-Challenge (+3.2pp), it *degrades* performance on MMLU (−9.2pp) and OpenBookQA (−1.8pp), where the weaker constituent models (Llama-3.2-1B at 44%, Phi-3-mini at 43%) corrupt the vote. The graph network achieves +10.4pp over majority vote on MMLU and +8.0pp on OpenBookQA, benchmarks where output-level ensembling is actively harmful, demonstrating that latent-space communication extracts useful representational signal from weaker models without inheriting their incorrect output predictions. This is qualitatively different from output-level combination, not merely quantitatively better.

The margins against the parameter-matched learned head (6.7–9.1pp) confirm that the gains arise from the communication mechanism rather than the classifier. A notable secondary finding is that the learned head on Mistral-7B *underperforms* Mistral's greedy decoding on MMLU (60.5% vs. 66.0%), consistent with Mistral's instruction tuning optimizing its output distribution in a way that is not recoverable from the penultimate representation alone. The graph network circumvents this by using Mistral as a refinement node rather than a representation to classify directly.

4.3 Error Analysis

To characterize the gains mechanistically, we compare graph and individual model predictions question-by-question on the 1,165 ARC-Challenge test examples. Of these, 54.4% involve partial inter-model disagreement, and on these the graph consistently resolves disagreement toward the correct answer, the primary source of the performance gain. Ten questions (0.9%) are recovered by the graph despite all five individual models failing, while 95 regressions (8.2%) represent a real cost of injection. The majority vote comparisons sharpen the mechanistic interpretation: on ARC, where model quality is uniform, majority vote improves over the best single model (+3.2pp) yet the graph exceeds it by a further +8.2pp. On MMLU and OpenBookQA, majority vote actively degrades performance (−9.2pp and −1.8pp), while the graph improves on both — confirming that latent-space aggregation extracts useful signal from weaker models without inheriting their incorrect outputs.

4.4 Ablation Studies

We ablate the three primary design choices of the architecture on ARC-Challenge using 3-epoch runs (full 4-epoch results reported above). All ablations use fixed random seed 42.

Table 2: Ablation results on ARC-Challenge test set (3-epoch runs). Full architecture achieves 87.7% at 3 epochs vs. 87.3% at 4 epochs, consistent with run-to-run variance. Δ is relative to the full architecture within each experiment group. $\star$ denotes the reported configuration.

Experiment	Variant	Test	Δ
(A) Injection coefficient α			
	0.10	86.0%	−1.7pp
	0.25*	86.6%	—
	0.50	87.0%	+0.4pp
	0.75	24.4%	−62.2pp
(B) Shared latent dimension d_s			
	256	87.3%	−0.6pp
	1024*	87.9%	—
	2048	86.4%	−1.5pp
(C) Node removal			
	Full*	87.7%	—
	Drop Llama-3.2-1B	86.8%	−0.9pp
	Drop Qwen2.5-1.5B	86.1%	−1.6pp
	Drop Gemma-2-2B	86.5%	−1.2pp
	Drop Phi-3-mini	79.9%	−7.8pp
	Drop Mistral-7B	87.2%	−0.5pp
	2×L1 only	86.7%	−1.0pp
(D) L1 prompt strategy			
	Analytical*	87.8%	—
	Distinct neutral	87.3%	−0.5pp
	Identical neutral	87.2%	−0.6pp
	No prefix	86.8%	−1.0pp

Injection coefficient (A). Performance is stable within 1.7pp for $\alpha \in \{0.10, 0.25, 0.50\}$; at $\alpha = 0.75$ it collapses to near-random (24.4%), directly replicating the representation fragility reported by Armstrong et al. (2026).

Shared dimension (B). Performance is robust across an 8× range. $d_s = 256$ with 4× fewer projection parameters achieves −0.6pp, suggesting the shared space develops geometrically structured representations rather than requiring high capacity.

Node removal (C). Dropping Phi-3-mini causes a 7.8pp collapse, the functional validation of the routing finding. All three L1 nodes contribute comparably (0.9–1.6pp when dropped), confirming the averaging operation does not degenerate. Dropping Mistral-7B costs only 0.5pp, consistent with its weaker gradient signal throughout training.

Prompt strategy (D). Performance varies by less than 1pp across all strategies, including identical prompts for all three L1 nodes. Representational diversity derives from model heterogeneity, not prompt engineering.

4.5 Training Dynamics

Figure 2 shows validation accuracy over training. ARC-Challenge converges rapidly — 80.7% after just 50 gradient steps — reflecting the immediate utility of aggregated multi-node representations. OpenBookQA peaks at 85.4% validation mid-epoch 2 and plateaus; training stops at epoch 3.

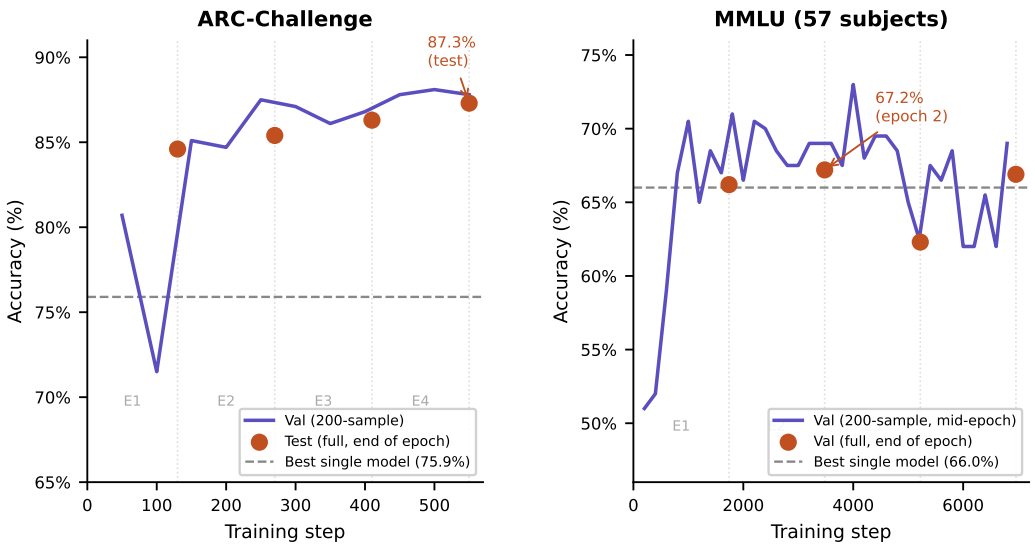

Figure 2: Validation accuracy over training on ARC-Challenge (left) and MMLU (right). Violet line: mid-epoch accuracy on 200-example subset. Coral markers: end-of-epoch full evaluation. Dashed line: best single-model baseline. ARC-Challenge reaches 80.7% after 50 steps; MMLU peaks at 67.2% at epoch 2.

MMLU converges more slowly over 14,042 training examples, peaking at 67.2% at epoch 2 under cosine annealing. Full hyperparameters are in Table 3.

Table 3: Training hyperparameters for all graph network experiments.

Hyperparameter	**Value**
Optimizer	AdamW
LR (W_1–W_5) / output node	10^{-3} / 10^{-4}
Weight decay	10^{-4}
Effective batch size	8
LR schedule	Cosine annealing
Gradient clip	1.0
Epochs	4 (MMLU, ARC); 3 (OBQA)
l_S / l_T / α	0.75 / 0.90 / 0.25
d_s / output heads	1,024 / 4
Precision / hardware	bfloat16 / A100 80GB

5 Analysis

5.1 Gradient Flow Through Frozen Model Boundaries

We validate gradient flow with a minimal two-node graph: Llama-3.2-1B as source, Qwen2.5-1.5B as destination, connected by a single projection matrix $W \in \mathbb{R}^{1536\times2048}$ and a four-class output head. We measure gradient norms, the W-to-head gradient ratio, and a permutation control (target activations randomly shuffled to destroy semantic correspondence while preserving marginal statistics).

Results are summarized in Table 4. Gradients flow cleanly: W receives approximately 13% of the output head's gradient signal — attenuated but well above catastrophic collapse, and comparable to attenuation across equivalent depth in standard feedforward networks. The permutation R^2 of -0.243 versus Ridge R^2 of 0.299 confirms the projection captures genuine semantic geometry rather

than distributional artifacts. Skip connections produce a 1.00× improvement, confirming they are unnecessary at this depth and simplifying the final architecture.

Table 4: Gradient flow validation (Llama-1B → Qwen-1.5B). Negative permutation R^2 confirms genuine cross-architecture alignment.

Metric	Value
Ridge projection R^2	0.299
Permutation control R^2	-0.243
W gradient max	2.48×10^{-1}
W gradient mean	1.81×10^{-3}
Grad norm ratio (W / head)	0.130
Skip connection improvement	1.00× (neutral)

5.2 Projection Matrix Gradient Norms During Training

Figure 3 shows gradient norms for W_1–W_5 across 4 epochs of MMLU training. Three findings emerge.

No dead nodes. All five projection matrices receive nonzero gradients throughout training, confirming all communication channels remain active. Heterogeneous layer-1 nodes are essential here — identical models would produce symmetric gradients and degenerate to the same solution.

Layer-2 projections receive stronger gradients. W_4 and W_5 consistently exhibit larger norms than W_1–W_3, as expected: layer-2 matrices feed directly into the output node while layer-1 matrices must traverse two frozen model boundaries first. The ratio is consistent with the 0.130 measured in the validation experiment.

Layer-1 projections do not specialize. W_1, W_2, and W_3 maintain nearly identical gradient norms throughout training. We attribute this to the averaging operation: if two matrices produce similar projections, the third receives no gradient pressure to diverge. The prompt ablation (Section 4) establishes that this non-specialization is not a failure mode — performance is stable within 1pp across all prompt strategies, including identical prompts for all three L1 nodes. Representational diversity therefore arises from the architectural and training heterogeneity of the frozen constituent models themselves, not from prompt differentiation or from learned specialization of the projection matrices. The projection matrices learn to extract complementary geometric structure from models with different inductive biases regardless of input framing.

5.3 Emergent Routing in the Output Node

Without any supervision on routing, the output node develops a consistent and substantial asymmetry between W_4 (Phi-3-mini) and W_5 (Mistral-7B). On MMLU, W_4 gradient norms exceed W_5 by 10–30× in early training, settling to 5–10× by epoch 4. On ARC-Challenge the asymmetry is larger early and partially closes by the final epoch.

This asymmetry is a direct consequence of the cross-attention weights: a higher weight on W_4 means W_4 receives a stronger backward signal, so the gradient ratio reflects implicit routing. We attribute Phi-3-mini's dominant role to its synthetic training corpus, which prior work has shown produces geometrically regular representations more amenable to external redirection Armstrong et al. (2026). The network discovers this property autonomously, without any prior indicating a preference, because Phi-3-mini's representations generate more consistent gradient signals. The node removal ablation validates this interpretation directly: dropping Phi-3-mini causes a 7.8pp collapse on ARC-Challenge, the largest degradation of any single node, while dropping Mistral-7B costs only 0.5pp. W_5 remains active throughout training (an order of magnitude above zero), confirming Mistral-7B contributes a secondary but nonzero signal. The benchmark-specificity of the routing asymmetry (larger on ARC than MMLU) is consistent with domain-dependent steerability, though controlled ablations are needed to disentangle this from dataset size and training duration effects.

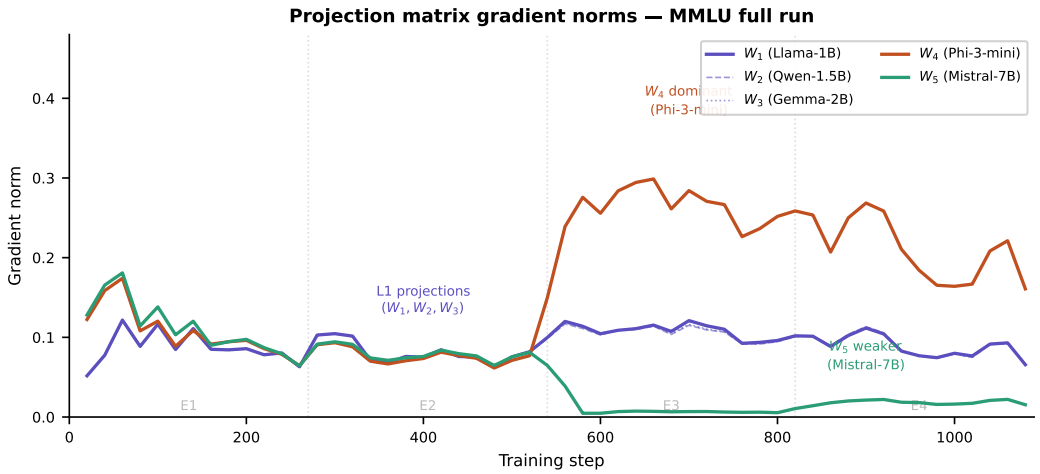

Figure 3: Projection matrix gradient norms over training (MMLU, 57 subjects). W_1–W_3 (violet) converge to nearly identical norms, showing layer-1 matrices do not specialize. W_4 (Phi-3-mini, coral) consistently dominates W_5 (Mistral-7B, teal), reflecting emergent selective routing. All five matrices remain active.

6 Conclusion

We introduced frozen LLM graphs: heterogeneous pretrained LLMs as computational nodes communicating through a shared continuous latent space via end-to-end learned projections. With only 17.6M trainable parameters against 12B frozen, the architecture achieves 87.3% on ARC-Challenge (+11.4pp over the best single model, +9.1pp over a parameter-matched head, +8.2pp over majority vote of the same five models), 82.8% on OpenBookQA (+6.2pp, +5.2pp, +8.0pp), and 67.2% on MMLU (+1.2pp, +6.7pp, +10.4pp). The majority vote comparisons are particularly informative: output-level ensembling of the same five models actively degrades performance on MMLU and OpenBookQA relative to the best single model, while latent-space communication consistently improves on all three — establishing that the gains arise from geometric aggregation in representation space rather than from output-level model combination. Gradient flow through frozen model boundaries is tractable at 13% of output-node signal strength; the output node develops asymmetric routing toward Phi-3-mini without supervision, validated by a 7.8pp collapse upon removal; and performance is stable within 1pp across all prompt strategies, confirming representational diversity arises from model heterogeneity.

Limitations. *Single training runs*: all results use one run per benchmark; the MMLU +1.2pp margin should be interpreted cautiously. *Scheduler instability*: cosine annealing interacts poorly with checkpoint resumption, causing oscillating MMLU validation accuracy; warmup-then-constant schedules should be preferred in future work. *Multiple-choice only*: generalization to open-ended generation and longer contexts is untested, as the current output node is a fixed-class classifier rather than a generative head. *Regression cost*: 8.2% of ARC-Challenge test questions are answered incorrectly by the graph despite at least one individual model being correct, representing a real cost of residual stream injection that selective injection strategies may mitigate.

Future directions. The most impactful near-term extensions are: replacing fixed average pooling with learned attention-weighted pooling over layer-1 nodes to enable dynamic per-query routing; scaling to deeper graphs via multi-GPU deployment; and warm-starting projection matrices from offline Ridge solutions to accelerate convergence. The frozen constraint ensures that alignment properties established during pretraining are preserved throughout composition — as the ecosystem of open-weight LLMs grows, treating them as composable graph nodes rather than monolithic systems to retrain becomes increasingly practical.

AI USE STATEMENT

The authors used Claude (Anthropic) as an AI assistant during the preparation of this work. Specifically, Claude was used to assist with the generation of training and evaluation code (including the graph network forward pass, residual stream injection hooks, baseline scripts, and diagnostic logging), and to assist with drafting and editing prose across multiple sections of the manuscript. All experimental results were produced and verified by the authors. All scientific claims, architectural decisions, experimental design, and interpretations are solely the work of the authors. No AI tools were used to generate or fabricate experimental data.

REPRODUCIBILITY STATEMENT

We have taken the following steps to facilitate reproducibility of our results. Full training hyperparameters are reported in Table 3. The trainable parameter breakdown is provided in Appendix A. All five constituent models are publicly available via HuggingFace Transformers Wolf et al. (2020) under their respective licenses: Llama-3.2-1B-Instruct (`meta-llama/Llama-3.2-1B-Instruct`), Qwen2.5-1.5B-Instruct (`Qwen/Qwen2.5-1.5B-Instruct`), Gemma-2-2B-IT (`google/gemma-2-2b-it`), Phi-3-Mini-4K-Instruct (`microsoft/Phi-3-mini-4k-instruct`), and Mistral-7B-Instruct-v0.3 (`mistralai/Mistral-7B-Instruct-v0.3`). All experiments used a fixed random seed of 42 and were conducted on a single NVIDIA A100 80GB GPU. Training and evaluation code, including the graph network forward pass, residual stream injection hooks, and all baseline scripts, is provided as supplementary material. Benchmark datasets are publicly available: MMLU via `cais/mmlu`, ARC-Challenge via `allenai/ai2_arc`, and OpenBookQA via `allenai/openbookqa` on the HuggingFace Datasets hub.

ETHICS STATEMENT

This work composes publicly available open-weight language models into a graph architecture for multiple-choice question answering. All constituent models are released under permissive research licenses and no proprietary models or private data are used. The work does not involve human subjects, does not collect or release new datasets, and does not introduce capabilities beyond those of the underlying pretrained models. The frozen constraint, under which all LLM weights remain unchanged throughout training, ensures that safety and alignment properties established during pretraining and instruction tuning are preserved in the composed system. We identify no foreseeable negative societal impacts specific to this work.

REFERENCES

M. Abdin, S. A. Jacobs, A. A. Awan, et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024.

M. Armstrong, N. Ayoobi, and A. Mukherjee. Thinking in different spaces: Domain-specific latent geometry survives cross-architecture translation. *arXiv preprint arXiv:2603.20406*, 2026.

L. Bai, Z. Xiong, H. Lin, G. Xu, X. Xie, R. Guo, Z. Kang, H.-T. Zheng, and H.-G. Kim. Frozen language models are gradient coherence rectifiers in vision transformers. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence and Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence and Fifteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'25/IAAI'25/EAAI'25. AAAI Press, 2025. ISBN 978-1-57735-897-8. doi: 10.1609/aaai.v39i2.32176. URL `https://doi.org/10.1609/aaai.v39i2.32176`.

Y. Bansal, P. Nakkiran, and B. Barak. Revisiting model stitching to compare neural representations. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, Red Hook, NY, USA, 2021. Curran Associates Inc. ISBN 9781713845393.

P. Clark, I. Cowhey, O. Etzioni, T. Khot, A. Sabharwal, C. Schoenick, and O. Tafjord. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song, and J. Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. LoRA: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

M. Huh, B. Cheung, T. Wang, and P. Isola. The platonic representation hypothesis, 2024. URL https://arxiv.org/abs/2405.07987.

A. Q. Jiang, A. Sablayrolles, A. Roux, A. Mensch, B. Savary, C. Bamford, D. S. Chaplot, D. de las Casas, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

X. L. Li and P. Liang. Prefix-tuning: Optimizing continuous prompts for generation. In C. Zong, F. Xia, W. Li, and R. Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.353. URL https://aclanthology.org/2021.acl-long.353/.

I. Loshchilov and F. Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

T. Mihaylov, P. Clark, T. Khot, and A. Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2018.

J. Moura. CrewAI: Framework for orchestrating role-playing autonomous AI agents. https://github.com/crewaiinc/crewai, 2023.

J. Qiu, D. Guo, N. Papini, N. Peace, H. F. Fitterman-Harris, C. A. Levinson, T. Hartvigsen, and T. R. Henry. Labeling free-text data using language model ensembles. 2025. URL https://arxiv.org/abs/2501.08413.

Qwen, :, A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei, H. Lin, J. Yang, J. Tu, J. Zhang, J. Yang, J. Yang, J. Zhou, J. Lin, K. Dang, K. Lu, K. Bao, K. Yang, L. Yu, M. Li, M. Xue, P. Zhang, Q. Zhu, R. Men, R. Lin, T. Li, T. Tang, T. Xia, X. Ren, X. Ren, Y. Fan, Y. Su, Y. Zhang, Y. Wan, Y. Liu, Z. Cui, Z. Zhang, and Z. Qiu. Qwen2.5 technical report. 2025. URL https://arxiv.org/abs/2412.15115.

A. M. Turner, L. Thiergart, G. Leech, D. Udell, J. J. Vazquez, U. Mini, and M. MacDiarmid. Steering language models with activation engineering. 2025. URL https://openreview.net/forum?id=2XBPdPIcFK.

T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, et al. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2020.

Q. Wu, G. Bansal, J. Zhang, et al. AutoGen: Enabling next-gen LLM applications via multi-agent conversation. In *Conference on Language Modeling (COLM)*, 2024.

L. Yin and Z. Wang. LLM-AutoDiff: Auto-differentiate any LLM workflow. *arXiv preprint arXiv:2501.16673*, 2025.

A. Zou, L. Phan, S. Chen, J. Campbell, P. Guo, R. Ren, A. Pan, X. Yin, M. Mazeika, A.-K. Dombrowski, S. Goel, N. Li, M. J. Byun, Z. Wang, A. T. Mallen, S. Basart, S. Koyejo, D. Song, M. Fredrikson, Z. Kolter, and D. Hendrycks. Representation engineering: A top-down approach to ai transparency. *ArXiv*, abs/2310.01405, 2023. URL https://api.semanticscholar.org/CorpusID:263605618.

A TRAINABLE PARAMETER BREAKDOWN

Table 5 provides the full parameter count for each trainable component of the frozen LLM graph. All five LLM backbone weights are frozen throughout training; only the projection matrices and output node receive gradient updates.

Table 5: Trainable parameter breakdown. The five projection matrices and output node together constitute 17,580,036 parameters — approximately 0.15% of the total frozen parameter count across all five constituent models (≈12B).

Component	Dimensions	Parameters
W_1 (Llama-3.2-1B $\rightarrow$ shared)	1024×2048	2,098,176
W_2 (Qwen2.5-1.5B $\rightarrow$ shared)	1024×1536	1,573,888
W_3 (Gemma-2-2B $\rightarrow$ shared)	1024×2304	2,360,320
W_4 (Phi-3-mini $\rightarrow$ shared)	1024×3072	3,146,752
W_5 (Mistral-7B $\rightarrow$ shared)	1024×4096	4,195,328
Output node (attn + classifier)	—	4,205,572
Total trainable		**17,580,036**
Frozen (all LLM weights)	—	*≈12B*

The output node's 4,205,572 parameters break down as follows: the 4-head multi-head attention over $d_s = 1024$ contributes $4 \times (1024^2/4) \times 3 + 1024^2 \approx 4{,}194{,}304$ parameters for the key, query, and value projections plus output projection, and the final linear classifier $W_{\text{cls}} \in \mathbb{R}^{4\times 1024}$ contributes 4,096 parameters plus a learned query vector of 1,024 and LayerNorm parameters.

B GRADIENT FLOW VALIDATION: FULL DETAILS

Section 5 summarizes the gradient flow validation experiment. We provide full experimental details here for reproducibility.

B.1 SETUP

We construct a minimal two-node graph consisting of Llama-3.2-1B-Instruct (source, $d = 2048$) and Qwen2.5-1.5B-Instruct (destination, $d = 1536$), connected by a single projection matrix $W \in \mathbb{R}^{1536\times 2048}$ and a four-class linear output head $H \in \mathbb{R}^{4\times 1536}$.

The forward pass proceeds as follows: (1) encode a factual question with Llama-3.2-1B, extracting the final-token hidden state at layer 24 of 32 (depth 0.75); (2) L2-normalize the resulting vector $\mathbf{h}_{\text{src}} \in \mathbb{R}^{2048}$; (3) project via W to obtain $\mathbf{z} = W\mathbf{h}_{\text{src}} \in \mathbb{R}^{1536}$; (4) inject $\mathbf{z}$ into Qwen's residual stream at layer 20 of 28 (depth 0.75) via a forward hook using $\alpha = 0.25$; (5) pass the question through Qwen with the modified residual stream; (6) extract the final-token hidden state at layer 25 of 28 (depth 0.90); (7) pass through the output head to obtain four-class logits.

We initialize W and H with standard normal weights scaled by 0.01. Both are registered as trainable parameters with `requires_grad=True`. The Llama and Qwen weights are loaded with `requires_grad=False`.

B.2 CROSS-ARCHITECTURE ALIGNMENT

Prior to the gradient test, we verify cross-architecture semantic alignment using Ridge regression. We collect 50 paired forward passes (simple factual questions) through Llama-3.2-1B and Qwen2.5-1.5B, extract final-token hidden states at depth 0.75 from both, and fit a Ridge regression mapping Llama representations to Qwen representations ($\lambda = 1.0$).

We fit a second Ridge regression with the rows of the Qwen activation matrix randomly permuted (permutation control), destroying semantic correspondence while preserving marginal statistics.

Results: Ridge $R^2 = 0.299$; Permutation $R^2 = -0.243$. The negative permutation R^2 confirms the alignment reflects genuine cross-architecture semantic geometry rather than distributional properties. This is lower than the $R^2 \approx 0.50$ reported by Armstrong et al. (2026) for similar-scale pairs; we attribute this to our smaller validation set (50 vs. 817 questions), smaller model scales (1B/1.5B vs. up to 8B), and cross-family pairing (Llama-Qwen vs. intra-family pairings which achieve $R^2 = 0.684$). Critically, Armstrong et al. (2026) find near-zero correlation ($r \approx -0.07$) between R^2 and behavioral correction rate, and our end-to-end training optimizes for task performance directly rather than R^2, so the lower offline R^2 does not imply inferior learned projections.

B.3 Gradient Norms

A single forward-backward pass through the two-node graph produces the following results. All values are computed under a cross-entropy loss against a dummy four-class target.

- W gradient maximum: 2.48×10^{-1}
- W gradient mean: 1.81×10^{-3}
- Output head H gradient maximum: 1.91×10^{0}
- Output head H gradient mean: 1.39×10^{-2}
- Gradient norm ratio $\|\nabla W\|_F / \|\nabla H\|_F$: 0.130

The ratio of 0.130 indicates that W, which must receive gradients through one frozen model boundary (Qwen), receives approximately 13% of the signal strength available at the output head. This is consistent with gradient attenuation in standard feedforward networks of equivalent depth and far above the catastrophic near-zero collapse sometimes hypothesized for frozen transformer boundaries.

B.4 Skip Connection Ablation

We test whether a direct skip connection from $\mathbf{z}$ to the output head (bypassing Qwen entirely) improves gradient flow to W. With skip connection: gradient norm ratio = 0.130 (identical). Without skip connection: gradient norm ratio = 0.130. Improvement factor: $1.00\times$ (neutral). Skip connections provide no benefit at two-layer depth and are omitted from the final architecture.

C Cross-Architecture Alignment Quality

This section provides extended discussion of our offline cross-architecture alignment quality relative to Armstrong et al. (2026), which was removed from the main paper for space.

Our Ridge R^2 of 0.299 for the Llama-1B $\rightarrow$ Qwen-1.5B pair is lower than the $R^2 \approx 0.50$ reported for similar-scale cross-architecture pairs in prior work. We attribute this to three factors. First, our validation set contains only 50 question-answer pairs, while prior work uses 817 TruthfulQA questions for fitting; larger fitting sets reliably improve R^2. Second, our models are relatively small (1B and 1.5B parameters); prior work finds alignment quality increases with model scale, with 8B-scale teachers achieving the strongest results. Third, our Llama-Qwen pairing is cross-family; intra-family pairings (e.g., Llama-8B $\rightarrow$ Llama-1B) achieve $R^2 = 0.684$, substantially above cross-family pairs.

Importantly, none of these factors undermine the validity of our end-to-end trained projections. Prior work reports a near-zero correlation ($r \approx -0.07$) between offline geometric alignment quality (R^2) and behavioral correction rate at test time, concluding that directional accuracy in relevant subspaces matters more than global variance explained. Our end-to-end training procedure optimizes projection matrices directly against task performance rather than R^2, and therefore implicitly optimizes for precisely the directional accuracy that matters for downstream behavior. The offline R^2 of 0.299 establishes that the cross-architecture geometry is real and non-trivial; the end-to-end training then refines the projection toward task-relevant directions that an offline proxy dataset may not capture.

D OpenBookQA Training Curve

Figure 4 shows the validation accuracy trajectory for the frozen LLM graph on OpenBookQA. The network reaches 74.0% validation accuracy within 30 gradient steps, crosses the best single-model baseline (Qwen2.5-1.5B, 76.6% test) before the end of epoch 1, and peaks at 85.4% mid-epoch 2. The curve plateaus between epochs 2 and 3 (test accuracy 82.4% $\rightarrow$ 82.8%), motivating our decision to stop training at epoch 3.

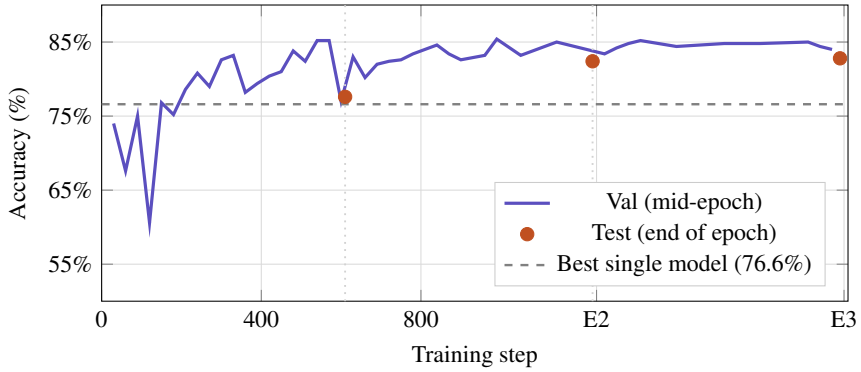

Figure 4: OpenBookQA validation accuracy over training. The network crosses the best single-model baseline (dashed) during epoch 1 and peaks at 85.4% mid-epoch 2. Test accuracy at end of epoch 3 is 82.8% (+6.2pp over best single model). Dotted vertical lines mark epoch boundaries.