QUERY-ADAPTIVE NEURON STEERING FOR LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Effectively eliciting the capabilities of large language models (LLMs) at inference time remains an important challenge. Existing studies address this challenge through context engineering, which shapes user queries, or fixed steering, which directly intervenes in model computation. However, predefined contexts and fixed intervention strategies may not adequately accommodate the diverse knowledge and reasoning requirements of individual queries. This raises a natural question: **Can internal steering in LLMs be made query-adaptive?** In this work, we investigate whether query-induced internal activations can guide adaptive steering in LLMs. Our pilot studies reveal subject- and layer-dependent neuron activation patterns and show that amplifying subject-relevant neurons can improve model performance. Based on these findings, we propose **Q-NeuS**, a query-adaptive neuron steering framework that dynamically determines which neurons to modulate and by how much for each input. Specifically, Q-NeuS constructs a compact subject-guided candidate neuron space and employs a lightweight Query-Adaptive Gate to generate neuron-wise modulation coefficients. To learn effective intervention strategies, we introduce **Neu-GRPO**, which adapts group-relative optimization to train the Query-Adaptive Gate using the relative performance of multiple neuron intervention configurations. The base LLM remains frozen throughout training and inference, and no task or subject labels are required at inference time. Extensive experiments across multiple LLMs and benchmarks demonstrate that Q-NeuS consistently improves model performance and exhibits broad applicability across model scales and strong cross-benchmark transferability.

1 INTRODUCTION

Exploring and eliciting the latent capabilities of Large Language Models (LLMs) is crucial for understanding and improving their latent potential. (Greenblatt et al., 2024; Donoway et al., 2025; Hofstätter et al., 2025). Existing studies largely follow two directions (As illustrated in Fig. 1). The first operates on model inputs through context engineering (Wang et al., 2025b; Ramnath et al., 2025; Zhao et al., 2026). While effective, these methods often rely on manually designed or task-specific contexts and are sensitive to how well the provided context matches each query. The second directly manipulates internal representations through steering (Rimsky et al., 2024; Stolfo et al., 2025; Wang et al., 2025a). However, existing steering methods typically target predefined tasks, capabilities, or behaviors, with the steering target or direction identified in advance. Such predefined interventions are difficult to generalize to heterogeneous queries that may require different knowledge and reasoning processes. Despite operating at different levels, both paradigms still rely on an intervention strategy specified in advance rather than adapting it to the current query. This raises a natural question: **Can internal steering in LLMs be made query-adaptive?**

To investigate this question, we first examine whether the internal activations of LLMs provide informative signals for query-adaptive intervention. We conduct a series of preliminary experiments to characterize neuron activation patterns across various domains and evaluate the effects of intervening on highly activated neurons. We observe that 1) highly activated neurons exhibit clear subject- and layer-dependent structures, and 2) amplifying subject-relevant neurons can improve model performance, while the intervention effects vary substantially across layers and subjects. These findings indicate that neuron activations encode task-relevant information that can guide internal intervention, while revealing the limitations of fixed steering strategies for heterogeneous queries.

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

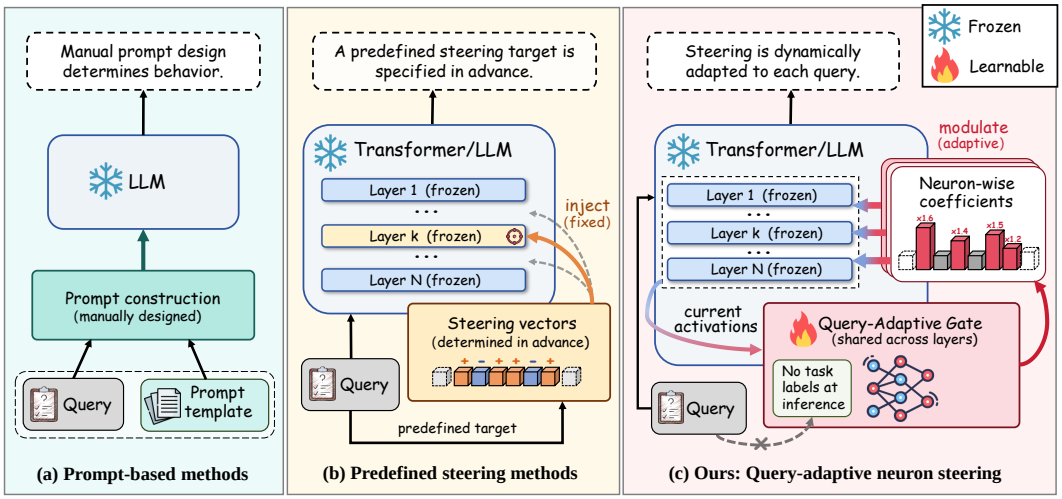


Figure 1: Comparison of paradigms for eliciting latent capabilities in LLMs. Prompt-based methods modify model inputs, predefined steering applies fixed internal interventions, while Q-NeuS dynamically adapts neuron-level interventions to each query.

Based on these pilot results, we introduce Q-NeuS, a query-adaptive neuron steering framework that identifies and modulates neurons according to the internal state induced by each query. Specifically, we first construct a subject-guided candidate neuron space from training data to constrain the intervention search space. We then introduce a lightweight Query-Adaptive Gate that takes the internal activations as input and dynamically generates neuron-wise modulation coefficients, allowing different queries to induce different steering patterns. To learn effective intervention strategies, we further propose Neu-GRPO, which adapts GRPO to the neuron intervention setting and optimizes the Query-Adaptive Gate by comparing the relative performance of multiple neuron intervention configurations. During inference, the method requires no task or domain labels, and the modulation pattern is determined solely by the internal activation state of the current query. Extensive experiments on seven LLMs ranging from 3B to 32B parameters across MMLU-Pro, SuperGPQA, and WildSci show that Q-NeuS improves performance across all evaluated models and benchmarks. Notably, Q-NeuS trained only on MMLU-Pro transfers effectively to multiple unseen benchmarks, demonstrating broad applicability across model scales and strong cross-benchmark transferability.

2 Related Work

Activation Steering for LLMs. Activation steering has emerged as an effective approach for controlling LLM behavior by directly modifying internal activations without updating the base model parameters (Li et al., 2023; Rimsky et al., 2024; Stolfo et al., 2025). Prior work has demonstrated its effectiveness in behavior control and reasoning enhancement (Højer et al., 2025), typically using techniques such as Contrastive Activation Addition (CAA) (Rimsky et al., 2024) or Sparse Autoencoders (SAEs) (Cho & Hockenmaier, 2025) to construct or identify steerable representations. More recent studies explore conditional and adaptive steering (Lee et al., 2025; Ye et al., 2026; Zheng et al., 2026). However, these methods remain constrained by a preconstructed intervention space, where the available steering directions or primitives are identified in advance. Unlike these methods, Q-NeuS performs query-adaptive neuron modulation based on query-induced activations, enabling fine-grained intervention beyond predefined steering spaces.

RL-based Policy Optimization for LLMs. Reinforcement learning (RL) has been widely used to optimize LLM generation policies (Ouyang et al., 2022; Shao et al., 2024; Guo et al., 2025). PPO-style RLHF typically relies on a learned value function for advantage estimation, introducing additional memory and computational overhead (Schulman et al., 2017; Ouyang et al., 2022). GRPO removes the critic by estimating advantages from the relative rewards of multiple outputs for the same prompt, thereby reducing training overhead (Shao et al., 2024). Recent studies have extended GRPO to adaptive routing and other settings (Ye et al., 2026). Different from methods that optimize

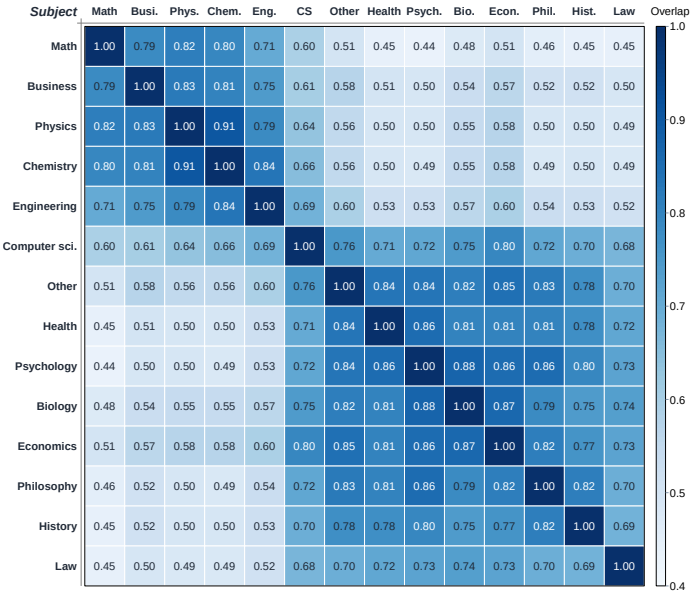


Figure 2: Overlap matrix of different subjects' Top-500 activation neurons in layer 25.

Table 1: Subject-wise performance on MMLU-Pro with Qwen2.5-3B-Instruct.

Subject	Baseline	Fix	Δ
Math	20.66	21.40	+0.74
Busi.	20.75	20.13	-0.63
Phys.	19.54	22.61	+3.07
Chem.	12.78	18.06	+5.29
Eng.	28.21	33.33	+5.13
CS	30.49	34.15	+3.66
Other	36.02	35.48	-0.54
Health	44.85	46.06	+1.21
Psych.	49.69	52.17	+2.48
Bio.	59.31	57.93	-1.38
Econ.	51.76	51.18	-0.59
Phil.	33.66	28.71	-4.95
Hist.	46.75	46.75	0.00
Law	22.62	23.53	+0.90
Overall	**31.56**	**32.92**	**+1.36**

the LLM policy itself, we keep the base LLM frozen and adapt GRPO to optimize the Query-Adaptive Gate according to the relative performance of different neuron intervention configurations.

3 Preliminary Study

In this section, we investigate whether LLM activations support query-adaptive intervention by relating subject-level queries to neuron activations. We conduct experiments with Qwen2.5-3B-Instruct on all 14 subjects of MMLU-Pro, splitting each subject into training/validation/test sets (70%/10%/20%) for neuron identification, layer selection, and final evaluation, respectively. For each layer, we extract the MLP activations at the final prompt token immediately before the output projection and rank neurons by their mean activation over the training examples. We then multiply the activations of the top 100 neurons by 1.5 and those of the next 400 neurons by 1.2, while leaving the remaining neurons unchanged. We independently intervene on each layer and retain those that improve validation accuracy. Finally, all selected layers are jointly intervened on and evaluated on the held-out test set. More details are provided in Appendix A.

Observation 1. Different subjects exhibit distinct neuron activation patterns. We examine the organization of highly activated neurons across subjects by selecting the top 500 neurons in each layer and computing pairwise overlap. As shown in Fig. 2, semantically related subjects exhibit higher overlap and form clear clusters, while substantial sharing remains even across dissimilar subjects. This suggests that LLMs rely on both shared computational components and subject-specific neuron subsets. As shown in Appendix A, this organization varies across depth: shallow layers are largely shared across subjects, intermediate layers exhibit stronger subject-specific structure, and the distinction weakens again in deeper layers. These results indicate that neuron organization is both subject-dependent and layer-dependent, motivating layer-aware and query-adaptive intervention.

Observation 2. Amplifying subject-relevant neurons improves model performance. As shown in Tab. 1, jointly intervening on the 21 positive-gain layers selected on the validation set improves test accuracy from 31.56% to 32.92% (+1.36 pp), demonstrating that selectively amplifying subject-relevant neurons can effectively improve model performance. However, the gains vary substantially across layers and subjects, with some subjects even experiencing performance degradation under joint intervention. These results suggest that while neuron-level intervention is effective, a fixed intervention strategy cannot consistently accommodate the heterogeneous requirements of different queries, motivating query-adaptive neuron steering.

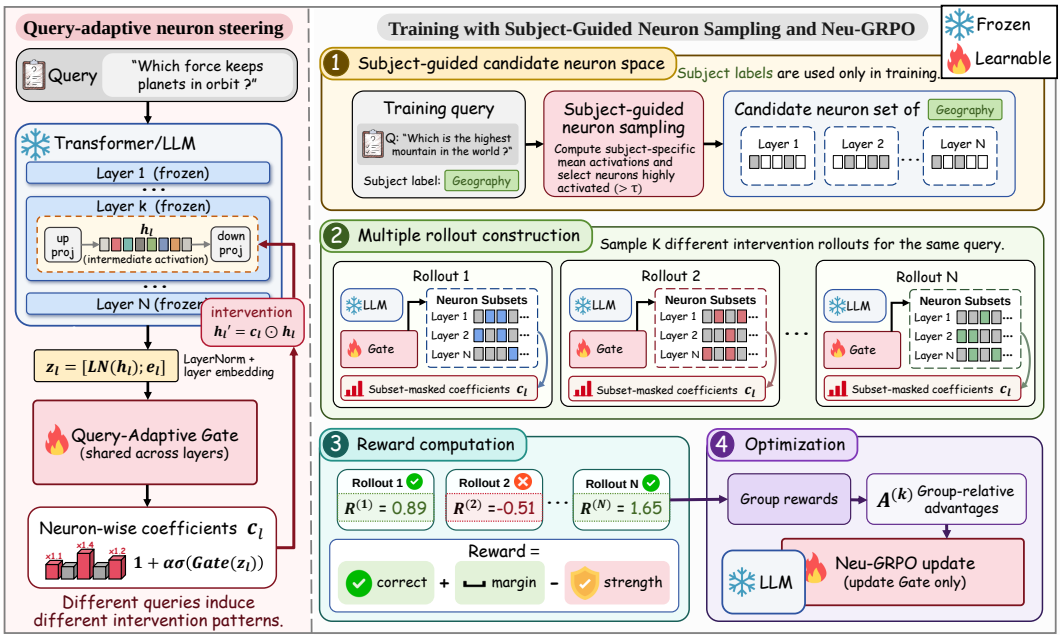


Figure 3: Overview of Q-NeuS. Q-NeuS generates query-adaptive neuron-wise modulation coefficients from query-induced activations. During training, subject-guided neurons form intervention rollouts evaluated by correctness, margin, and intervention-strength rewards. Neu-GRPO optimizes the Query-Adaptive Gate with group-relative feedback while keeping the base LLM frozen.

Motivation. These results show that modulating subject-relevant neurons can effectively improve model performance, while the optimal intervention varies across layers and subjects. However, our pilot study relies on manually specified neuron counts, fixed modulation coefficients, and predetermined layer selection, limiting its ability to adapt to individual queries. To address this limitation, we introduce Q-NeuS, which employs a learnable gating mechanism to dynamically determine which neurons to modulate and by how much according to the internal activation state of each query.

4 Q-NeuS

Based on these pilot findings, we introduce Q-NeuS, a query-adaptive neuron steering framework. During training, subject-guided candidate neurons are explored through multiple rollouts, and a Query-Adaptive Gate is optimized with Neu-GRPO using their relative performance. At inference time, the Query-Adaptive Gate dynamically modulates all FFN neurons across different Transformer layers based on the internal activations induced by each query, without requiring subject labels.

4.1 Query-Adaptive Neuron Gating

Neuron Specification. Most LLMs employ a feed-forward network (FFN) in each Transformer layer. Given the hidden state $\mathbf{x}_l$ of the last input token at layer l, the FFN can be written as:

$$\mathbf{h}_l = \mathrm{SiLU}\left(W_l^{\mathrm{gate}}\mathbf{x}_l\right) \odot \left(W_l^{\mathrm{up}}\mathbf{x}_l\right), \qquad \mathbf{y}_l = W_l^{\mathrm{down}}\mathbf{h}_l, \tag{1}$$

where W_l^{gate}, W_l^{up}, and W_l^{down} denote the gate, up, and output projections, respectively. The vector $\mathbf{h}_l \in \mathbb{R}^{d_{\mathrm{ff}}}$ denotes the FFN intermediate activation before the output projection, where each dimension is treated as an FFN neuron. Q-NeuS directly modulates $\mathbf{h}_l$ before the output projection.

Query-Adaptive Neuron Gating. Motivated by our preliminary findings, we introduce a lightweight Query-Adaptive Gate that generates neuron-wise modulation coefficients from query-induced internal activations. To capture layer-specific information, the shared Query-Adaptive Gate takes the normalized activation $\tilde{\mathbf{h}}_l$ together with a learnable layer embedding $\mathbf{e}_l$:

$$\tilde{\mathbf{h}}_l = \mathrm{LayerNorm}(\mathbf{h}_l), \qquad \mathbf{z}_l = [\tilde{\mathbf{h}}_l; \mathbf{e}_l]. \tag{2}$$

The Query-Adaptive Gate produces gating logits through a two-layer feedforward network:

$$\mathbf{g}_l = W_2 \,\mathrm{SiLU}\,(W_1 \mathbf{z}_l + \mathbf{b}_1) + \mathbf{b}_2, \tag{3}$$

where W_1, W_2 and $\mathbf{b}_1$ and $\mathbf{b}_2$ are learnable weights and biases, respectively. The resulting logits are mapped to neuron-wise modulation coefficients:

$$\mathbf{c}_l = \mathbf{1} + \alpha\sigma(\mathbf{g}_l), \tag{4}$$

where $\sigma(\cdot)$ denotes the sigmoid function and α controls the maximum modulation strength. The intermediate activations are then modulated as:

$$\hat{\mathbf{h}}_l = \mathbf{c}_l \odot \mathbf{h}_l, \tag{5}$$

the modulated activation $\hat{\mathbf{h}}_l$ is passed to the original output projection.

The Query-Adaptive Gate is shared across Transformer layers, while the layer embedding $\mathbf{e}_l$ enables layer-specific modulation. Conditioned on query-induced internal activations, it generates neuron-wise coefficients that adapt both the intervention pattern and modulation strength to each query, without requiring task or subject labels. It provides fine-grained and parameter-efficient adaptation to heterogeneous queries, with detailed parameter analysis provided in Appendix B.

4.2 NEU-GRPO

To learn query-adaptive neuron modulation policies of Q-NeuS, we introduce **Neu-GRPO**, which adapts GRPO to neuron-level intervention. For each training query, we construct multiple neuron intervention configurations and evaluate their relative task performance. The resulting group-relative feedback is then used to optimize Q-NeuS, while keeping the base LLM frozen.

Subject-Guided Neuron Sampling. Our preliminary study shows that highly activated neurons exhibit subject-specific patterns and provide effective targets for intervention. We leverage this property to construct a compact candidate neuron space, reducing the search space for subsequent query-adaptive modulation. Given N_c training examples from subject c, the mean activation of neuron j in layer l is computed as:

$$\bar{h}_{c,l,j} = \frac{1}{N_c} \sum_{i=1}^{N_c} h_{l,j}^{(i)}. \tag{6}$$

We define the subject- and layer-specific candidate set $S_{c,l}$ as:

$$S_{c,l} = \left\{ j \mid \bar{h}_{c,l,j} > \tau \right\}, \tag{7}$$

where τ is an activation threshold. Unlike the fixed-size neuron selection used in our preliminary study, this formulation allows the candidate set to adapt to different subjects and layers.

For each training query from subject c, we construct K intervention rollouts. In rollout k, a subset of candidate neurons $M_l^{(k)}$ is sampled at each layer:

$$M_l^{(k)} \subseteq S_{c,l}, \qquad |M_l^{(k)}| = \lfloor \rho |S_{c,l}| \rfloor, \qquad k = 1, \ldots, K, \tag{8}$$

where ρ denotes the neuron sampling ratio. We sample diverse neuron subsets across rollouts, while Q-NeuS adaptively determines their modulation strengths from query-induced activations.

Group-Relative Gate Optimization. For each rollout k, Q-NeuS multiplicatively generates neuron-wise modulation coefficients and applies them to the sampled neurons $M_l^{(k)}$, resulting in a query-specific intervention configuration. Let $\pi_\phi^{(k)}(\cdot \mid q)$ denote the output distribution of the modulated model for query q, where ϕ denotes the parameters of Q-NeuS. We evaluate each rollout using a reward that considers task correctness, prediction confidence, and intervention strength:

$$R^{(k)} = R^{(k)}_{\text{correct}} + \lambda_m R^{(k)}_{\text{margin}} - \lambda_s R^{(k)}_{\text{strength}}, \tag{9}$$

where λ_m and λ_s control the contributions of the margin reward and intervention-strength penalty, respectively. The correctness reward indicates whether the predicted answer is correct. To capture the model's preference for the correct answer, we further define:

$$R^{(k)}_{\text{margin}} = \log \pi_\phi^{(k)}(y \mid q) - \max_{a' \neq y} \log \pi_\phi^{(k)}(a' \mid q), \tag{10}$$

where y is the ground-truth answer. We additionally penalize unnecessarily strong interventions:

$$R_{\text{strength}}^{(k)} = \frac{1}{|\mathcal{V}^{(k)}|} \sum_{l \in \mathcal{V}^{(k)}} \frac{1}{|M_l^{(k)}|} \sum_{j \in M_l^{(k)}} \left(c_{l,j}^{(k)} - 1 \right)^2, \tag{11}$$

where $\mathcal{V}^{(k)}$ is the set of layers containing sampled neurons in rollout k. We optimize the Query-Adaptive Gate using GRPO and the relative rewards of the K intervention rollouts.

During training, the base LLM remains frozen and only the Query-Adaptive Gate is optimized, while subject labels are used solely to construct candidate neuron sets. At inference time, the learned Query-Adaptive Gate generates neuron-wise modulation coefficients from query-induced internal activations without requiring task or subject labels. This enables subject-guided policy learning during training while retaining fully query-adaptive, label-free intervention at inference time.

5 EXPERIMENT

5.1 EXPERIMENTAL SETUP

Models. We evaluate Q-NeuS on seven LLMs spanning different model sizes and families: Qwen2.5-3B (Yang et al., 2025b), Qwen3-4B, Qwen3-8B, Qwen3-14B, Qwen3-32B (Yang et al., 2025a), Ministral3-8B, and Ministral3-14B (Liu et al., 2026a).

Datasets and Evaluation. We use MMLU-Pro (Wang et al., 2024) , SuperGPQA (Du et al., 2026) , and WildSci (Liu et al., 2026b) as our primary benchmarks. For in-domain evaluation, we partition each benchmark into training (70%), validation (10%), and test (20%) splits, train Q-NeuS on the training split, and report results on the held-out test split. To assess cross-benchmark generalization, we directly evaluate the model trained on MMLU-Pro on GPQA-Main, GPQA-Diamond (Rein et al., 2023) , SuperGPQA, LiveBench (White et al., 2025) , AGIEval (Zhong et al., 2024) , GSM8K (Cobbe et al., 2021) , and ARC-Challenge (Clark et al., 2018) , without using any training data from these target benchmarks. Accuracy is used as the primary evaluation metric.

Baselines. We compare Q-NeuS with three baselines. **Base** is the original LLM without intervention. **Fix** follows our preliminary study, selecting highly activated neurons from training-set mean activations, applying predefined modulation coefficients, and selecting intervention layers on the validation set. **GRPO** optimizes LoRA parameters (Hu et al., 2021) with GRPO (Shao et al., 2024). All methods use the same base model and evaluation protocol whenever applicable.

Implementation Details. We set the candidate neuron threshold to $\tau = 0.1$. The Query-Adaptive Gate uses a hidden dimension of $2{,}048$ and a layer embedding dimension of 32, with a maximum modulation coefficient of 1.7. For each query, we construct $K = 4$ rollouts and sample 60% of the candidate neurons in each layer. In Neu-GRPO, we set $\lambda_m = 1.0$ and $\lambda_s = 0.1$ for the margin reward and intervention strength penalty, respectively. Q-NeuS is trained with Adam at a learning rate of 1×10^{-4} for up to 3 epochs. All main experiments are conducted on NVIDIA A800 GPUs.

5.2 MAIN RESULTS

Overall Performance across Models and Benchmarks. We train and evaluate Q-NeuS separately on MMLU-Pro, SuperGPQA, and WildSci using seven LLMs. As shown in Tab. 2, Q-NeuS improves performance across all evaluated models and benchmarks, with gains spanning model scales from 3B to 32B parameters. The improvements extend beyond the Qwen family to the Ministral family, indicating that the effectiveness of Q-NeuS is not tied to a specific model family or task. This broad applicability is attributed to the query-adaptive design of Q-NeuS, which generates neuron-wise modulation from query-induced internal activations without relying on fixed steering patterns or modifying the base model parameters. These results demonstrate the broad applicability of query-adaptive FFN neuron modulation across model scales, architectures, and task domains.

Cross-Benchmark Generalization. We train Q-NeuS only on MMLU-Pro and directly evaluate it on the other datasets in Tab. 3. Across most benchmarks, performance improves or remains comparable to the base model, suggesting that the learned neuron modulation policy can transfer to other knowledge-intensive reasoning tasks. However, transfer gains on LiveBench are relatively limited,

Table 2: Performance of Q-NeuS across different LLMs on MMLU-Pro, SuperGPQA, and WildSci.

Model	MMLU-Pro			SuperGPQA			WildSci		
	Base	Q-NeuS	Δ	Base	Q-NeuS	Δ	Base	Q-NeuS	Δ
Qwen2.5-3B	31.43	38.37	+6.94	19.58	24.37	+4.79	33.95	39.51	+5.56
Qwen3-4B	44.86	45.93	+1.07	26.29	29.79	+3.50	43.20	45.52	+2.32
Qwen3-8B	44.20	48.53	+4.33	26.83	30.95	+4.12	42.91	45.78	+2.87
Qwen3-14B	50.89	54.03	+3.14	30.35	35.46	+5.11	46.91	49.58	+2.67
Qwen3-32B	54.56	59.11	+4.55	33.97	38.73	+4.76	49.64	52.56	+2.92
Ministral3-8B	44.82	49.03	+4.21	24.50	28.68	+4.18	45.05	50.01	+4.96
Ministral3-14B	50.48	53.45	+2.97	27.45	32.23	+4.78	49.71	53.22	+3.51

Table 3: In-domain performance and cross-benchmark generalization of Q-NeuS trained on MMLU-Pro. Each cell reports accuracy (%), with the change over the base model shown in parentheses.

Model	In-Domain	Cross-Benchmark Generalization						
	MMLU-Pro	GPQA-M	GPQA-D	SuperGPQA	LiveBench	AGIEval	GSM8K	ARC-C
Qwen2.5-3B	38.37 (+6.94)	33.04 (+2.91)	36.87 (+4.55)	19.39 (+0.13)	24.15 (-0.48)	52.32 (+0.31)	81.20 (-2.27)	82.76 (+0.17)
Qwen3-4B	45.93 (+1.07)	36.61 (0.00)	39.90 (+2.53)	27.72 (+1.30)	55.06 (-0.26)	58.29 (-0.63)	90.75 (+0.30)	89.59 (-0.60)
Qwen3-8B	48.53 (+4.33)	36.38 (+0.22)	36.87 (+3.03)	29.54 (+1.11)	51.96 (+0.63)	59.28 (-0.37)	93.25 (+0.60)	91.64 (+0.68)
Qwen3-14B	54.03 (+3.14)	40.62 (+2.45)	42.93 (+4.04)	32.71 (+3.23)	57.57 (+0.36)	65.55 (+0.14)	94.54 (+0.23)	92.66 (+0.42)
Qwen3-32B	59.11 (+4.55)	42.41 (-1.12)	43.43 (0.00)	33.46 (+4.14)	58.08 (-0.61)	69.41 (+0.04)	95.45 (+0.53)	94.80 (+0.26)
Ministral3-8B	49.03 (+4.21)	38.62 (+1.34)	39.90 (+5.05)	27.75 (+2.82)	38.32 (-1.80)	54.62 (+0.10)	90.37 (0.00)	91.72 (+0.25)
Ministral3-14B	53.45 (+2.97)	37.72 (+1.11)	38.38 (+4.04)	30.25 (+1.66)	42.05 (0.00)	58.93 (+0.12)	93.40 (-0.08)	92.92 (+0.60)
Win / Tie / Loss	**7 / 0 / 0**	**5 / 1 / 1**	**6 / 1 / 0**	**7 / 0 / 0**	**2 / 1 / 4**	**5 / 0 / 2**	**4 / 1 / 2**	**6 / 0 / 1**

Table 4: Cross-benchmark transfer comparison between Q-NeuS and GRPO on Qwen3-8B.

Method	In-Domain	Cross-Benchmark Generalization						
	MMLU-Pro	GPQA-M	GPQA-D	SuperGPQA	LiveBench	AGIEval	GSM8K	ARC-C
Base	44.20	36.16	33.84	28.43	51.33	59.65	92.65	90.96
GRPO	48.33	35.71	33.84	**30.77**	51.36	**60.07**	93.18	91.21
Q-NeuS	**48.53**	**36.38**	**36.87**	29.54	**51.96**	59.28	**93.25**	**91.64**

with only small improvements or occasional declines across models. This may be attributed to the larger distribution shift from MMLU-Pro: LiveBench includes tasks such as coding and data analysis, whose formats and required capabilities are less represented in the subject-oriented MMLU-Pro training data. As a result, both the candidate neuron space and the learned modulation policy may transfer less effectively to these different reasoning patterns. Meanwhile, Q-NeuS shows more consistent transfer on knowledge-intensive reasoning benchmarks such as GPQA and SuperGPQA, which are more closely aligned with the training distribution.

We further evaluate the transferability of Q-NeuS and GRPO when both are trained only on the MMLU-Pro training set and directly applied to unseen benchmarks without target-domain training. As shown in Tab. 4, both methods transfer beyond the source benchmark, while Q-NeuS achieves better performance on most target benchmarks. We attribute this advantage to their adaptation mechanisms: GRPO learns fixed parameter updates from the source domain, while Q-NeuS keeps the base LLM frozen and dynamically determines neuron-level interventions from the internal activations induced by each new query. It allows Q-NeuS to adapt its intervention pattern to the knowledge and reasoning requirements of individual queries at inference time, rather than relying on a fixed adaptation learned from the source domain, thereby improving generalization to unseen task distributions.

Comparison with Fixed Intervention and GRPO. Tab. 5 presents the subject-wise comparison on MMLU-Pro. The fixed intervention strategy yields limited and unstable gains. In comparison, Q-NeuS improves 13 of the 14 subjects and achieves an average gain of $+4.33$ percentage points.

Table 5: Subject-wise accuracy changes on MMLU-Pro with Qwen3-8B. The baseline row reports original accuracy, other rows show changes in percentage points (pp).

Method	Bio.	Bus.	Chem.	CS	Econ.	Eng.	Health	Hist.	Law	Math	Other	Phil.	Phys.	Psych.	Avg.
Base	70.34	33.96	25.11	50.00	60.59	43.59	53.94	58.44	32.58	36.53	42.47	38.61	41.38	60.25	44.20
Fix	-1.38	-1.26	-0.44	+1.22	+1.76	+3.59	+1.21	-1.30	-0.90	+2.58	+0.54	+0.00	-0.77	+0.00	+0.45
GRPO	+2.07	+13.84	+11.45	+3.66	+4.12	+1.54	+6.67	-1.30	-0.45	-0.74	+5.91	+2.97	+3.07	+4.35	+4.13
Ours	+3.45	+5.03	+7.49	+1.22	+5.88	+3.59	+13.94	+1.30	+4.07	-0.74	+4.30	+2.97	+2.30	+5.59	**+4.33**

Table 6: Neu-GRPO reward ablation on MMLU-Pro with Qwen3-8B. Baseline: 44.20%.

Variant	λ_c	λ_m	λ_s	Acc. (%)	Δ (pp)
w/o R_{margin}	1.0	0	0.1	48.04	+3.84
w/o R_{strength}	1.0	1.0	0	47.83	+3.63
w/o $R_{\text{margin}}, R_{\text{strength}}$	1.0	0	0	47.50	+3.30
Full Neu-GRPO	**1.0**	**1.0**	**0.1**	**48.53**	**+4.33**

Table 7: Effect of the neuron sampling ratio on MMLU-Pro with Qwen3-8B.

Sampling Ratio	Baseline (0%)	10%	20%	30%	40%	50%	**60%**	70%	80%	90%
Acc. (%)	44.20	46.76	47.13	47.38	48.24	48.16	**48.53**	48.29	48.29	46.80
Δ (pp)	+0.00	+2.56	+2.93	+3.18	+4.04	+3.96	**+4.33**	+4.09	+4.09	+2.60

It indicates that predefined neuron selection, modulation strengths, and layer selection may not adequately accommodate the heterogeneous requirements of different subjects, whereas query-adaptive modulation provides broader improvements. Compared with GRPO, Q-NeuS improves performance on a larger number of subjects. While GRPO learns a shared set of parameter updates across inputs, Q-NeuS keeps the base LLM frozen and dynamically adjusts neuron-level modulation according to each query. These results demonstrate that query-adaptive neuron steering achieves competitive overall performance with more robust improvements across heterogeneous subjects.

5.3 Ablation Studies

We evaluate two key design choices in Q-NeuS: the Neu-GRPO reward design and the neuron sampling ratio. All ablation experiments use Qwen3-8B on MMLU-Pro.

Effect of Reward Design. The Neu-GRPO reward consists of correctness, margin, and intervention-strength terms. To evaluate their contributions, we remove R_{margin}, R_{strength}, or both and compare the resulting variants with the full objective. As shown in Tab. 6, the correctness reward alone improves performance, while the full reward achieves the best result. This indicates that the auxiliary terms provide complementary optimization signals beyond correctness. Specifically, the margin reward distinguishes rollouts with identical correctness outcomes through their prediction margins, while the strength penalty discourages excessive neuron modulation. Together, they provide a more informative and stable learning signal for optimizing the neuron intervention policy.

Effect of Neuron Sampling Ratio. During Neu-GRPO training, we construct intervention rollouts by sampling different neuron subsets from the candidate neuron space. To study the effect of the sampling ratio, we vary it from 10% to 90%. As shown in Tab. 7, Q-NeuS outperforms the base model across all sampling ratios, demonstrating robustness to this hyperparameter. The accuracy improves as the sampling ratio increases and remains relatively stable between 40% and 80%, indicating that the benefit of increasing neuron coverage gradually saturates. The best result is achieved at 60%. However, increasing the ratio to 90% leads to a clear performance drop, suggesting that excessive neuron sampling may reduce the diversity of intervention configurations across rollouts.

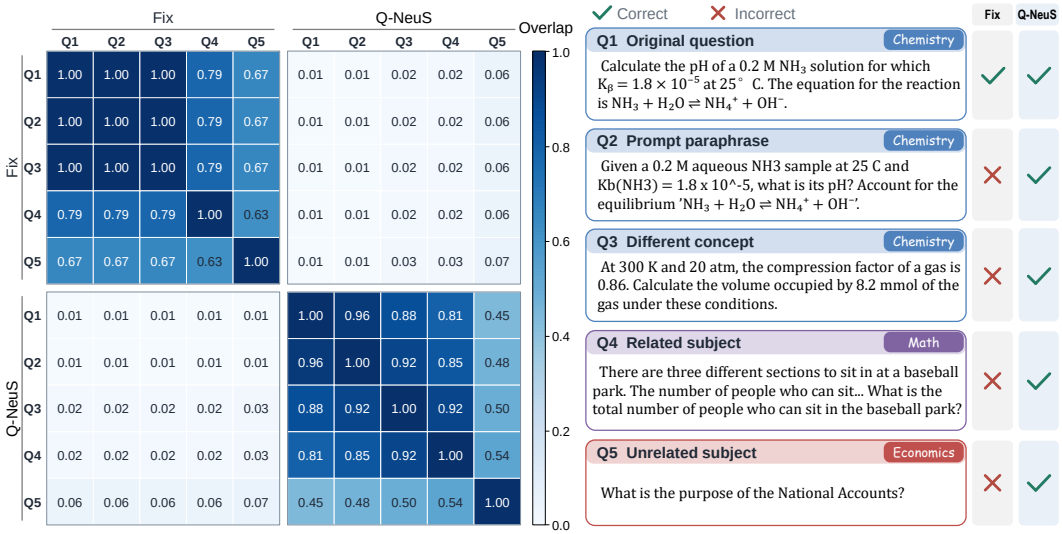


Figure 4: Case study of Fix and Q-NeuS across queries with different semantic relationships.

5.4 Case Study

We compare Fix and Q-NeuS on five queries: an original chemistry question (Q1), its paraphrase (Q2), a chemistry question involving a different concept (Q3), a question from a related subject, mathematics (Q4), and a question from an unrelated subject, economics (Q5). Fig. 4 presents the overlap among the top-100 most strongly modulated neurons together with the corresponding answer correctness. Fix applies a fixed subject-specific intervention, resulting in identical top-100 modulated neurons for Q1–Q3. Although related subjects exhibit higher overlap than unrelated ones, the differences remain limited. It indicates that Fix captures coarse subject-level relationships but cannot adapt its intervention to individual queries within the same subject.

Q-NeuS exhibits three notable properties: 1) It produces query-specific neuron patterns even within the same subject: the overlap between Q1 and Q3 is 0.88 despite both being chemistry questions. 2) It preserves a high overlap of 0.96 between Q1 and its paraphrase Q2 and answers both correctly, whereas Fix fails on Q2 despite succeeding on Q1, demonstrating robustness to prompt reformulation. 3) The selected neuron sets reflect semantic relationships across queries: relative to Q1, the overlap is 0.81 for the related-subject query Q4 but drops to 0.45 for the unrelated-subject query Q5, yielding a clearer distinction than Fix. Q-NeuS answers all five queries correctly, while Fix succeeds only on Q1. These results illustrate that Q-NeuS can adjust neuron-level interventions to individual queries while preserving similar modulation patterns for semantically equivalent inputs. Appendix C further shows that these query-specific overlap patterns vary across Transformer layers and exhibit clear layer-dependent structure, consistent with our preliminary findings.

6 Conclusion

In this work, we investigate query-adaptive neuron steering for LLMs. Our preliminary study shows that highly activated neurons exhibit subject- and layer-dependent patterns, and that amplifying subject-relevant neurons can improve model performance. Motivated by these findings, we propose Q-NeuS, which dynamically determines which neurons to modulate and by how much from query-induced internal activations. Q-NeuS combines a subject-guided candidate neuron space with a lightweight Query-Adaptive Gate and introduces Neu-GRPO to optimize neuron modulation through group-relative feedback while keeping the base LLM frozen. Extensive experiments across seven LLMs and multiple benchmarks demonstrate that Q-NeuS improves performance across model scales and architectures and transfers effectively to multiple unseen benchmarks with diverse knowledge and reasoning requirements. These results show that query-induced internal activations can provide effective signals for adaptive intervention, enabling LLMs to better elicit their existing capabilities without updating the base model parameters or requiring task labels at inference time.

AI USE STATEMENT

We used generative AI tools to assist with manuscript translation and improve the precision and readability of the text. These tools were not used to develop the research method, design or implement experiments, analyze experimental results, or draw research conclusions. We reviewed and verified all AI-assisted revisions and take full responsibility for the final content of this paper.

ETHICS STATEMENT

All experiments in this study were conducted using publicly available models and benchmark datasets. The study involved no human subjects and no collection of private or sensitive data. We have identified no significant ethical concerns arising from this research.

REPRODUCIBILITY STATEMENT

We provide implementation details, hyperparameter settings, and experimental configurations in the main text and appendix. We will publicly release our source code to facilitate the reproduction of our experimental results.

REFERENCES

Ikhyun Cho and Julia Hockenmaier. Toward efficient sparse autoencoder-guided steering for improved in-context learning in large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 28949–28961, 2025.

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

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Elizabeth Donoway, Hailey Joren, Arushi Somani, Henry Sleight, Julian Michael, Michael R DeWeese, John Schulman, Ethan Perez, Fabien Roger, and Jan Leike. Quantifying elicitation of latent capabilities in language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=Dkgx2pS4Ww.

Xeron Du, Yifan Yao, Kaijing Ma, Bingli Wang, Tianyu Zheng, Minghao Liu, Yiming Liang, Xiaolong Jin, Zhenlin Wei, Chujie Zheng, et al. Supergpqa: Scaling llm evaluation across 285 graduate disciplines. *Advances in Neural Information Processing Systems*, 38, 2026.

Ryan Greenblatt, Fabien Roger, Dmitrii Krasheninnikov, and David Krueger. Stress-testing capability elicitation with password-locked models. *Advances in Neural Information Processing Systems*, 37:69144–69175, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

Felix Hofstätter, Teun van der Weij, Jayden Teoh, Rada Djoneva, Henning Bartsch, and Francis Rhys Ward. The elicitation game: Evaluating capability elicitation techniques. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=kT0EVqL77E.

Bertram Højer, Oliver Jarvis, and Stefan Heinrich. Improving reasoning performance in large language models via representation engineering. In *International Conference on Learning Representations*, volume 2025, pp. 44746–44763, 2025.

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. *arXiv preprint arXiv:2106.09685*, 2021.

Bruce W Lee, Inkit Padhi, Karthikeyan Natesan Ramamurthy, Erik Miehling, Pierre Dognin, Manish Nagireddy, and Amit Dhurandhar. Programming refusal with conditional activation steering. In *International conference on learning representations*, volume 2025, pp. 90960–90985, 2025.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in neural information processing systems*, 36:41451–41530, 2023.

Alexander H Liu, Kartik Khandelwal, Sandeep Subramanian, Victor Jouault, Abhinav Rastogi, Adrien Sadé, Alan Jeffares, Albert Jiang, Alexandre Cahill, Alexandre Gavaudan, et al. Ministral 3. *arXiv preprint arXiv:2601.08584*, 2026a.

Tengxiao Liu, Deepak Nathani, Zekun Li, Kevin Yang, and William Yang Wang. Wildsci: Advancing scientific reasoning from in-the-wild literature. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 11677–11695, 2026b.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744, 2022.

Kiran Ramnath, Kang Zhou, Sheng Guan, Soumya Smruti Mishra, Xuan Qi, Zhengyuan Shen, Shuai Wang, Sangmin Woo, Sullam Jeoung, Yawei Wang, et al. A systematic survey of automatic prompt optimization techniques. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 33066–33098, 2025.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. Gpqa: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*, 2023.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. Steering llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15504–15522, 2024.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Alessandro Stolfo, Vidhisha Balachandran, Safoora Yousefi, Eric Horvitz, and Besmira Nushi. Improving instruction-following in language models through activation steering. In *International Conference on Learning Representations*, volume 2025, pp. 55790–55823, 2025.

Mengru Wang, Ziwen Xu, Shengyu Mao, Shumin Deng, Zhaopeng Tu, Huajun Chen, and Ningyu Zhang. Beyond prompt engineering: Robust behavior control in llms via steering target atoms. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 23381–23399, 2025a.

Xubin Wang, Jianfei Wu, Yuan Yichen, Deyu Cai, Mingzhe Li, and Weijia Jia. Demonstration selection for in-context learning via reinforcement learning. In *Forty-second International Conference on Machine Learning*, 2025b. URL https://openreview.net/forum?id=sugs65XoGg.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *Advances in Neural Information Processing Systems*, 37:95266–95290, 2024.

Colin White, Samuel Dooley, Manley Roberts, Arka Pal, Benjamin Feuer, Siddhartha Jain, Ravid Shwartz-Ziv, Neel Jain, Khalid Saifullah, Sreemanti Dey, et al. Livebench: A challenging, contamination-limited llm benchmark. In *International Conference on Learning Representations*, volume 2025, pp. 91595–91631, 2025.

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

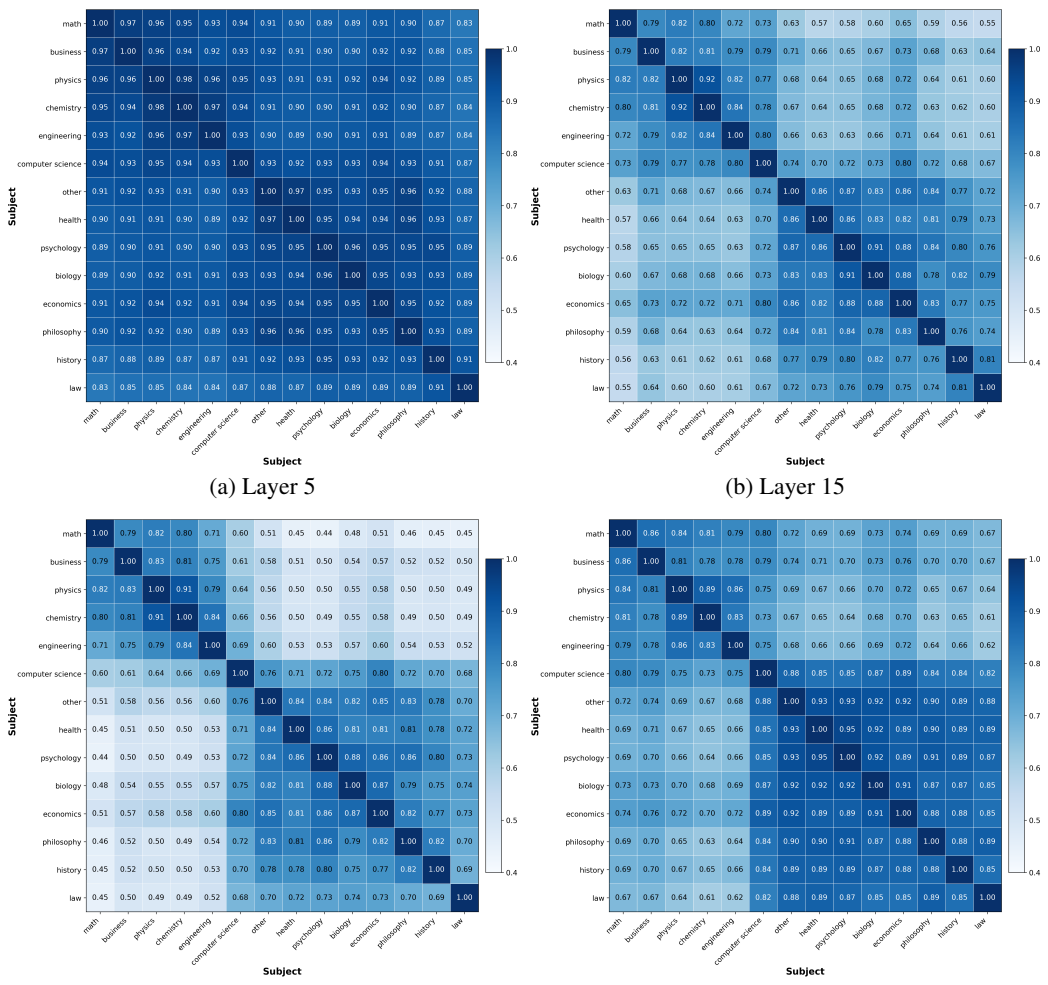
Qwen: An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025b. URL https://arxiv.org/abs/2412.15115.

Wencheng Ye, Xiaoyang Yuan, Yi Bin, Hengyu Jin, Liang Peng, Pengpeng Zeng, and Heng Tao Shen. Riser: Orchestrating latent reasoning skills for adaptive activation steering. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 4627–4644, 2026.

ChenZhuo Zhao, Xinda Wang, Pu Zhao, Yue Huang, Junting Lu, Ziqian Liu, Qingwei Lin, Saravan Rajmohan, and Dongmei Zhang. Gradient-guided multi-judge prompt optimization. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 23744–23773, 2026.

Lifan Zheng, Xue Yang, Jiawei Chen, Chenyan Wu, Jingyuan Zhang, Fanheng Kong, Xinyi Zeng, Xiang Chen, and Yu Tian. Dpn-le: Dual personality neuron localization and editing for large language models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 30567–30591, 2026.

Wanjun Zhong, Ruixiang Cui, Yiduo Guo, Yaobo Liang, Shuai Lu, Yanlin Wang, Amin Saied, Weizhu Chen, and Nan Duan. Agieval: A human-centric benchmark for evaluating foundation models. In *Findings of the association for computational linguistics: NAACL 2024*, pp. 2299–2314, 2024.



(a) Layer 5 (b) Layer 15

(c) Layer 25 (d) Layer 35

Figure 5: Neuron overlap matrices across representative Transformer layers. Each matrix reports the pairwise overlap rates of the Top-500 highly activated FFN neurons across MMLU-Pro subjects.

A Additional Analysis of the Preliminary Study

A.1 Neuron Overlap Patterns across Layers

To further analyze the organization of subject-related neurons across different layers, we compute the overlap matrices of the Top-500 highly activated neurons across subjects at several representative layers, as shown in Fig. 5. Overall, different layers exhibit clearly distinct organizational patterns.

In shallower layers, such as layer 5, nearly all subject pairs exhibit relatively high overlap rates, suggesting that highly activated neurons in shallow layers largely reflect shared representations across subjects, with limited subject-specific differentiation. In intermediate layers, such as layers 15 and 25, subjects with similar semantic content or reasoning patterns tend to exhibit higher neuron overlap, whereas more dissimilar subjects show substantially lower overlap, forming clearer block-like structures. This differentiation is most pronounced at layer 25, indicating stronger subject-related neuron organization in intermediate layers. In deeper layers, such as layer 35, the overall overlap across subjects increases again, while the block structure becomes less distinct.

These observations further indicate that highly activated neurons exhibit different organizational structures across layers.

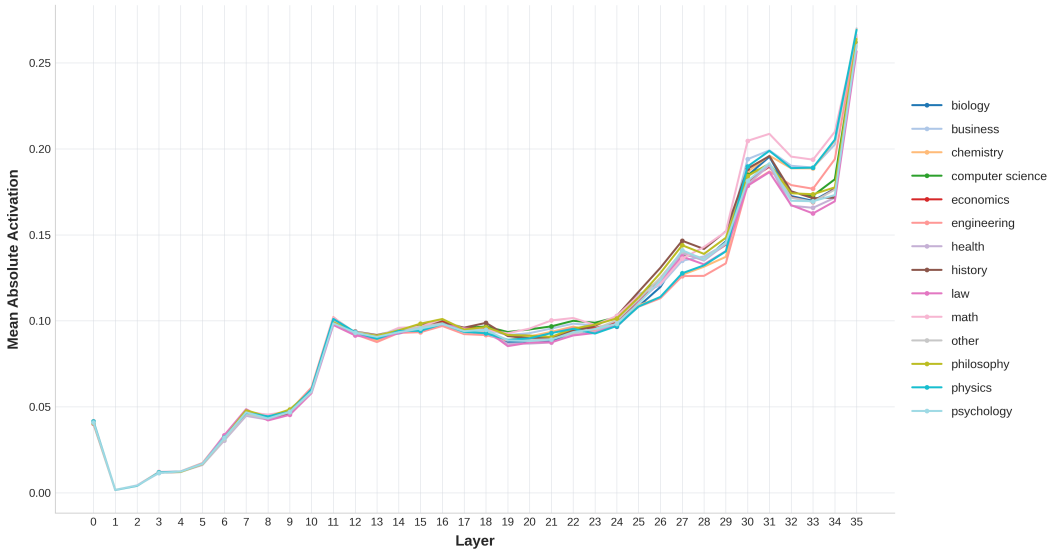


Figure 6: Mean absolute FFN neuron activations across Transformer layers for different MMLU-Pro subjects.

A.2 Overall Activation Magnitude across Layers

To further examine how internal activations vary with network depth, we compute the mean absolute activation of all FFN neurons at each Transformer layer for samples from different subjects. As shown in Fig. 6, different subjects exhibit highly similar layer-wise trends: the overall activation magnitude is relatively low in shallow layers, increases gradually through the intermediate layers, and becomes stronger in deeper layers.

In contrast, the differences in mean activation magnitude across subjects within the same layer are relatively small, with noticeable divergence appearing only in some deeper layers. This suggests that subject-related differences are not primarily reflected in the overall activation magnitude. Combined with the neuron overlap analysis in the previous subsection, these results suggest that subject-related differences are more likely to manifest in the specific composition and distribution patterns of activated neurons.

A.3 Layer-wise Intervention Effects

We further analyze the intervention effects across different layers in the preliminary study. During validation, the 36 Transformer layers exhibit substantially different responses to neuron intervention. Among them, 21 layers achieve positive accuracy gains, while the remaining layers show only limited improvements or performance degradation. Among all single-layer interventions, layer 18 yields the largest improvement, increasing accuracy by 1.17 percentage points.

In addition to the layer-selection results reported in the main text, we further examine the setting without layer selection. Specifically, we remove the validation-based layer selection and directly apply joint neuron intervention to all 36 Transformer layers. Under this setting, the model accuracy decreases from the baseline of 31.56% to 29.41%, corresponding to a drop of 2.15 percentage points. In contrast, jointly intervening on the 21 layers selected on the validation set achieves an accuracy of 32.92%.

This comparison shows that even though intervention on some individual layers can yield positive gains, indiscriminately extending the intervention to all layers can substantially degrade overall performance. Therefore, simply increasing the intervention coverage does not necessarily lead to better performance.

Table 8: Parameter overhead of the shared Query-Adaptive Gate across different backbone models. Weight storage is reported in BF16.

Model	Gate Params	Gate / Backbone	BF16 Weight Size
Qwen2.5-3B	45.17M	1.46%	90.34MB
Qwen3-4B	39.92M	0.99%	79.85MB
Qwen3-8B	50.41M	0.62%	100.83MB
Qwen3-14B	71.39M	0.48%	142.78MB
Qwen3-32B	104.95M	0.32%	209.91MB
Ministral3-8B	58.80M	0.69%	117.61MB
Ministral3-14B	67.19M	0.50%	134.39MB



(a) Layer 1 (b) Layer 15 (c) Layer 18

Figure 7: Top-100 neuron overlap matrices of Fix and Q-NeuS across different Transformer layers for the case-study queries.

B Parameter Efficiency of the Query-Adaptive Gate

Q-NeuS employs a single Query-Adaptive Gate shared across all Transformer layers, while using low-dimensional layer embeddings to preserve layer-specific information. As shown in Tab. 8, across the seven models evaluated in this work, the Query-Adaptive Gate contains only 39.92M–104.95M parameters, corresponding to 79.85–209.91MB of BF16 weights. This accounts for only 0.32%–1.46% of the parameters of the corresponding text backbone models. As the backbone model becomes larger, the relative parameter overhead generally decreases. For example, the Query-Adaptive Gate accounts for 0.99%, 0.62%, 0.48%, and 0.32% of the parameters of Qwen3-4B, Qwen3-8B, Qwen3-14B, and Qwen3-32B, respectively.

Cross-layer sharing is a key factor in reducing the additional parameter cost. If an independent Query-Adaptive Gate were introduced for each Transformer layer, the number of gating parameters would increase by approximately 34–64$\times$ across the evaluated models. Therefore, the current design substantially reduces the parameter and storage overhead of the additional module while retaining layer-specific conditioning.

C Additional Case Study Analysis

We further analyze how the neuron overlap patterns in the case study vary across different Transformer layers. As shown in Fig. 7, different layers exhibit distinct query-level overlap structures, indicating that the query-adaptive neuron patterns are clearly layer-dependent, consistent with the layer-dependent neuron organization observed in the preliminary study.