Neuron-Guided Fine-Tuning: Unlocking Efficient Alignment Mechanisms for Large Language Models

Zeyu Wu[1,*] **Junchao Wu**[1,2,*] **Shudong Liu**[1] **Runzhe Zhan**[1]
Xin Chen[3] **Shu Yang**[4] **Yichao Du**[2,5] **Longyue Wang**[2]
Weihua Luo[2] **Jinsong Su**[6] **Derek F. Wong**[1,†]
[1]NLP[2]CT Lab, Faculty of Information Science and Computing, University of Macau
[2]Alibaba Group [3]Nanjing University [4]KAUST [5]Wuhan University [6]Xiamen University
nlp2ct.{zeyu,junchao}@gmail.com, derekfw@um.edu.mo

Abstract

Existing Supervised Fine-Tuning paradigms, particularly Full Parameter Fine-Tuning are often plagued by parameter redundancy, inconsistent data quality, and catastrophic forgetting, which current methods typically address in isolation and lack a unified optimization signal to bridge data selection, parameter updates, and knowledge preservation. To address this, we propose **N**euron-**G**uided **F**ine-**T**uning **(NGFT)**, a holistic framework that leverages neuron activation patterns as a universal proxy to unify the fine-tuning lifecycle. NGFT operates via three synergistic mechanisms: (1) Adaptive Task-Specific Neuron Selection, which identifies essential neurons in a single forward pass to concentrate updates and reduce redundancy; (2) Activation-Based Data Selection, which prioritizes information-dense samples that maximize contribution to key neurons; and (3) Neuron Activation Alignment, a novel loss function that anchors activations to pre-trained states, deepening representation learning and preserving general knowledge. Experimental results across three models across both domain-specific and general benchmarks demonstrate that NGFT significantly outperforms existing mainstream fine-tuning methods in both efficiency and performance, while effectively mitigating catastrophic forgetting. Codes are publicly available at: https://github.com/NLP2CT/NGFT.

1 Introduction

Supervised Fine-Tuning (SFT) remains the primary approach to adapting Large Language Models (LLMs) from general pre-training to specific downstream tasks (Dong et al., 2024; Hu et al., 2023; Harada et al., 2025). While Parameter-Efficient Fine-Tuning (PEFT) methods reduce memory overhead, Full Parameter Fine-Tuning (FPFT) retains its status as the golden standard for achieving strong performance (Lv et al., 2024; Ding et al., 2022; Sun et al., 2020). However, FPFT's practical deployment is hindered by a fundamental bottleneck: the lack of a unified optimization signal that bridges data selection, parameter updates, and knowledge preservation across the fine-tuning lifecycle. This deficiency manifests in three intertwined challenges: (1) Parameter Redundancy: LLM capabilities are encoded by sparse active neurons, yet FPFT forces updates on all neurons, causing computational resources waste and induces negative transfer (Xu et al., 2025; Leng and Xiong, 2025; Wang et al., 2024; Chen et al., 2025a); (2) Inconsistent Data Quality: Instruction data often lack a universal utility selection criterion, which allows low-quality data to dilute the effective training signal and reduce data efficiency (Zhang et al., 2025; Li et al., 2024b); (3) Catastrophic Forgetting: Domain-specific dense optimization overwrites pre-trained LLM general knowledge, reducing model versatility (Li et al., 2022; Huang et al., 2024; Wang et al., 2020).

Existing approaches typically address these issues in isolation, resulting in fragmented and heterogeneous strategies. Sparse fine-tuning (Xu et al., 2025) targets parameter redundancy but ignores data quality, data selection (Wang et al., 2025) improves data utility but decouples from parameter updates, and anti-forgetting techniques (Li et al., 2024a) are seldom integrated with the other two. As a result, fine-tuning still lacks a unified objective that jointly governs data selection, parameter optimization, and knowledge preservation.

To bridge this gap, we identify **Neuron Activation Patterns** as a universal proxy for fine-tuning. As a shared signal for data characteristics, parameter computation, and knowledge representation, they link model behavior to internal parameters and enable a unified framework for aforementioned three key dimensions in fine-tuning lifecycle.

[*]Equal contribution.
[†]Corresponding author.

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 33268–33291
October 24-29, 2026 ©2026 Association for Computational Linguistics

Based on this, we propose Neuron-Guided Fine-Tuning (NGFT), a framework that utilizes neuron activation patterns as a universal proxy to construct a holistic full life-cycle optimization mechanism across three dimensions: (1) **Adaptive Task-Specific Neuron Selection**: We challenge the convention of fixed neuron selection ratios by addressing the heterogeneity of required knowledge across domains. We introduce a novel adaptive mechanism that identifies Task-Knowledge Neurons and dynamically calibrates optimal sparsity; (2) **Activation-Based Data Selection**: We pioneered a data selection criterion based on neuron activation intensity. By quantifying sample contributions to key neurons, we identify high-quality data located at the center of information density, improving the performance while reducing computational costs; (3) **Neuron Activation Alignment**: We propose a novel loss function for aligning neuron activation patterns. Introduced after cross-entropy loss stabilizes, it preserves general knowledge via pre-computed activation anchors while deepening task-specific representation learning, effectively mitigating catastrophic forgetting.

Experimental results on Llama-3.1 (Dubey et al., 2024), Qwen2.5 (Yang et al., 2024), and Mistral (Jiang et al., 2023) across three domain-specific datasets (GSM8K (Cobbe et al., 2021), BioInstruct (Tran et al., 2024), DialogSum (Chen et al., 2021)), and three general benchmarks (MMLU (Hendrycks et al., 2020), BBH (Suzgun et al., 2023), TyDiQA (Clark et al., 2020)) demonstrate that NGFT consistently outperforms FPFT and mainstream fine-tuning frameworks. Averaged across all models, NGFT achieves a 4.66 point improvement in domain performance and mitigates forgetting-induced performance degradation by 9.89 point compared to FPFT. Further analysis shows that each component of NGFT remains effective when used independently, establishing strong baselines for different stages of the fine-tuning lifecycle.

2 Related Work

2.1 Data Selection

Data selection is pivotal in the instruction tuning of LLMs, aiming to extract high-quality subsets that enhance instruction-following capabilities while minimizing redundancy (Zhang et al., 2025). Existing mainstream selection paradigms primarily include: (1) Powerful LLM Scoring: leveraging powerful closed-source models (e.g., GPT-4) as external scorers to evaluate data quality and diversity (Li et al., 2024b,c; Chen et al., 2024); (2) Indicator-based Method: relying on heuristic metrics (e.g., syntactic complexity, perplexity) to measure the data diversity and importance (Cao et al., 2023; Marion et al., 2023; Zheng et al., 2023). (3) Gradient-based Approaches: quantifying data value by monitoring the gradient trajectories of the targeted model on specific data (Xia et al., 2024; Wang et al., 2025). However, these methods face inherent limitations: (1) the unreliability of indicator-based methods in high-dimensional spaces, and (2) the high cost of accessing closed-source APIs or gradient computation. While recent efforts explored neuron-aware data selection (Chen et al., 2026b), these methods remain decoupled from parameter-level optimization and knowledge preservation, leaving the full potential of neuron activation patterns as a unifying signal untapped.

2.2 Neuron Attribution and Fine-Tuning

Feed-Forward Networks (FFNs), accounting for approximately two-thirds of LLM parameters, have emerged as a focal point in interpretability research. Widely regarded as the locus of knowledge storage, they are essential for encoding facts and integrating representations (Geva et al., 2021; Meng et al., 2022; Kim et al., 2025). Numerous neuron attribution studies aim to identify critical neurons within FFNs to gain deeper insights into LLM internals. Research on task-knowledge neurons primarily involves two paradigms: (1) Activation-based Attribution: localizing skill or language-specific neurons by analyzing activation offsets under soft prompts or cross-lingual patterns (Wang et al., 2022; Tang et al., 2024; Christ et al., 2025). (2) Gradient-based Attribution: employing techniques like integrated gradients to compute neuron-level contribution scores, pinpointing specific knowledge neurons (Dai et al., 2022a; Song et al., 2024; Zhang et al., 2024). The proven impact of these neurons on model behavior has inspired neuron-level parameter-efficient fine-tuning. For instance, (Xu et al., 2025) demonstrated that updating only sensitive neurons in machine translation outperforms FPFT. Similarly, (Leng and Xiong, 2025) showed that fine-tuning task-relevant neurons selected via gradient attribution improves performance while mitigating catastrophic forgetting in multi-task settings. However, existing neuron-based fine-tuning methods often lack adaptive mechanisms to dynamically adjust the ratio of

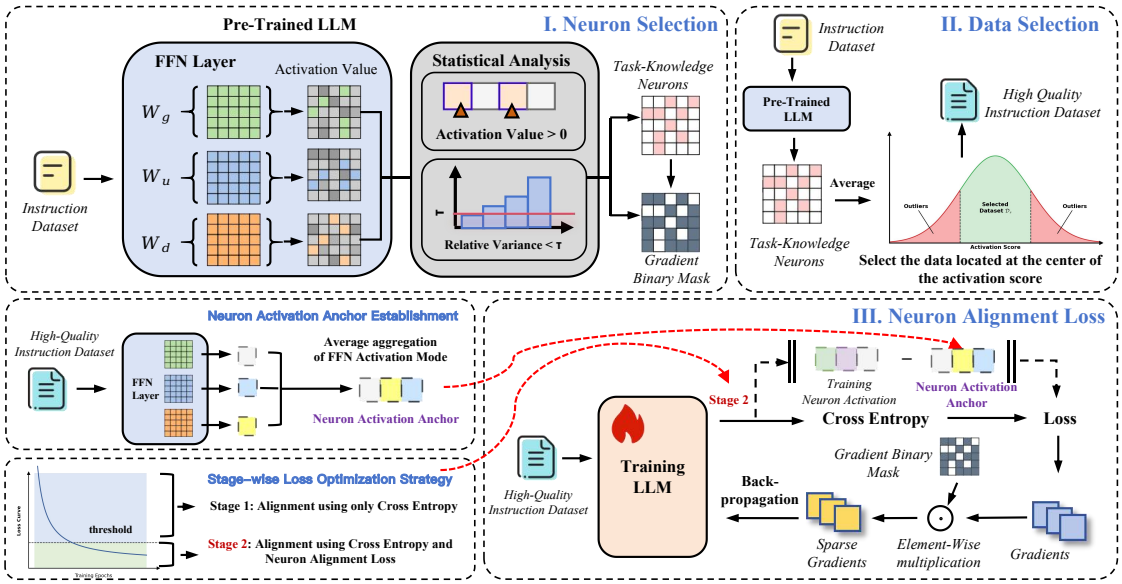

Figure 1: Overview of the proposed NGFT framework. It consists of three components: (I) Task-Knowledge Neuron Selection, which identifies sparse, stably activated neurons via coefficient of variation analysis; (II) Activation-Based Data Selection, which filters representative samples based on neuron activation intensity; and (III) Neuron-Level Alignment, which updates only critical neurons using gradient masking and incorporates a Neuron Alignment Loss to achieve deep alignment, thereby improving task performance while mitigating catastrophic forgetting.

key neurons based on specific domain or knowledge, thereby limiting model flexibility.

3 Method

We propose an efficient neuron-guided fine-tuning framework named NGFT, illustrated in Figure 1, which consists of three components: (I) **Task-Knowledge Neuron Selection**: We adaptively localize a sparse set of neurons that are consistently active and stably activated across the entire training dataset. These neurons are regarded as the critical units with the most significant impact on model performance; (II) **Activation-Based Data Selection**: We select data samples whose activation patterns are more aligned with the distribution centroids of the task-knowledge neurons. These samples tend to best represent the core characteristics of the data; (III) **Neuron-Level Alignment**: Once the cross-entropy (CE) loss reaches a specific convergence stage, we introduce a new objective that aligns the activation patterns of model responses with those induced by the ground truth. This guides the model to internalize deeper neuronal knowledge and achieve better representation alignment.

3.1 Task-Knowledge Neuron Selection

Mainstream LLMs (e.g., Llama 3, Qwen 2.5) predominantly adopt Transformer-based architectures that employ gated activation mechanisms within their FFNs. For the l-th layer, let the input hidden be denoted as $\tilde{\boldsymbol{h}}^i \in \mathbb{R}^d$. The computation of the FFN module output $\boldsymbol{h}^i$ is formalized as:

$$\boldsymbol{h}^i = \left(\sigma(\tilde{\boldsymbol{h}}^i W_g) \odot (\tilde{\boldsymbol{h}}^i W_u)\right) W_d \qquad (1)$$

where $W_g, W_u \in \mathbb{R}^{d\times 4d}$ and $W_d \in \mathbb{R}^{4d\times d}$ represent the Gate, Up, and Down projection matrices, respectively. $\sigma(\cdot)$ denotes the activation function, and $\odot$ represents element-wise multiplication. Structurally, we regard the column vectors of these projection matrices as neurons, and denote the activation of the j-th neuron in the l-th layer as $h_{l,j}$.

To identify Task-Knowledge Neurons, we define an instruction dataset $\mathcal{D} = \{x_1, x_2, \ldots, x_m\}$, and a target LLM $\mathcal{M}$ with L layers. We analyze the neuron activations of the final token of input x_i, which aggregates global semantic information (Zou et al., 2023; Chen et al., 2025b). Based on this, we apply two constraints to identify Task-Knowledge Neurons: (1) Activation State, where a neuron is "activated" if its activation value for input x_i is positive (i.e., $h_{l,j}^{(i)} > 0$), indicating an effective response to the input; (2) Distribution Stability, which filters out noisy neurons via the Coefficient of Variation. Neurons with low activation distribution Coefficient of Variation are stably activated and critical to model performance, consistent with Wang et al. (2022); Dai et al. (2022b) studies. Key neurons

feature similarity and predictability. Specifically, for $h_{l,j}$, we calculate its mean activation value $\mu_{l,j}$ and standard deviation $\sigma_{l,j}$ across $\mathcal{D}$:

$$\mu_{l,j} = \frac{1}{m} \sum_{x_i \in \mathcal{D}} h_{l,j}^{(i)} \tag{2}$$

$$\sigma_{l,j} = \sqrt{\frac{1}{m-1} \sum_{x_i \in \mathcal{D}} \left(h_{l,j}^{(i)} - \mu_{l,j} \right)^2} \tag{3}$$

We then obtain the Coefficient of Variation $R_{l,j} = \frac{\sigma_{l,j}}{\mu_{l,j}+\epsilon}$ for each neuron's distribution and identify a set of Task-Knowledge Neurons, denoted as $\mathcal{N}_k$. This subset comprises neurons $n_{l,j}$ that demonstrate high stability and satisfy the activation condition, determined by a threshold τ:

$$\mathcal{N}_k = \{ n_{l,j} \mid R_{l,j} < \tau \text{ and } h_{l,j}^{(i)} > 0 \}. \tag{4}$$

During fine-tuning, we apply gradient masking to the $\mathcal{N}_k$ to enable structured parameter updates while freezing non-essential neurons. Let θ represent the model parameters and $\mathcal{L}$ be the loss function. We construct a binary mask $\boldsymbol{M}$ in one-to-one corresponding to $\mathcal{N}_k$:

$$M_{l,j} = \begin{cases} 1 & , n_{l,j} \in \mathcal{N}_k \\ 0 & , \text{otherwise} \end{cases} \tag{5}$$

In the backpropagation phase, constraints are imposed on the gradients via element-wise multiplication:

$$\nabla_{\boldsymbol{\theta}} \mathcal{L}^{\text{mask}} = \nabla_{\boldsymbol{\theta}} \mathcal{L} \odot \boldsymbol{M} \tag{6}$$

This operation restricts updates to the parameters of key neurons, maintaining the pre-trained state of the remainder (Leng and Xiong, 2025). It effectively decouples precise fine-tuning from global parameter freezing, enabling the model to adapt to domain-specific data with significantly reduced computational costs.

3.2 High-Quality Instruction Data Selection

Upon identifying the Task-Knowledge Neurons $\mathcal{N}_k$, we evaluate each instruction instance by its activation profile on these neurons. To quantify this, we define the *Task-Relevant Activation Score* for each instance x_i. Let $\mathcal{A}_{l,j}^{(i)}$ denote the activation value of neuron $n_{l,j} \in \mathcal{N}_k$ given input x_i. The score S_i is:

$$S_i = \frac{1}{|\mathcal{N}_k|} \sum_{n_{l,j} \in \mathcal{N}_k} \mathcal{A}_{l,j}^{(i)}. \tag{7}$$

However, selecting solely by magnitude can be misleading, as extreme activation spikes introduce skewness. Drawing on the information bottleneck framework and representation learning theory (Achille and Soatto, 2018; Kawaguchi et al., 2023), instances proximal to the class centroid (mean value) are highly compressible yet lack representational diversity, whereas those distant from the centroid retain diversity but are disproportionately susceptible to corrupted samples. We thus propose a median-centered interval selection strategy that optimally resolves the information bottleneck trade-off between two key criteria: Sufficiency (preserving task-relevant information) and Minimality (eliminating redundant noise).

Specifically, let $r_i \in [r_1, \ldots, r_m]$ denote the rank of instance x_i sorted by S_i. We select a subset $\mathcal{D}_s$ based on a sampling ratio $\rho \in (0, 1]$:

$$\mathcal{D}_s = \left\{ x_i \in \mathcal{D} \mid \left| r_i - \frac{m}{2} \right| \leq \frac{\rho \cdot m}{2} \right\} \tag{8}$$

The continuous median neighborhood criterion naturally absorbs the core samples of all subclusters, thereby ensuring robustness to mild intra-class multimodal distribution while retaining the holistic distribution of the class (Xia et al., 2022).

3.3 Neuron Alignment Loss Optimization

While CE loss serves as the standard paradigm for SFT, it only optimizes the next-token probability distribution at the output layer. As gradient signals weaken during late convergence, models often hit performance plateaus. To address this, we propose a novel loss function based on neuron-level alignment. Focusing on internal model mechanisms, it guides deep knowledge learning and representation alignment in later optimization, easing convergence bottlenecks and enhancing model performance.

Formally, given a dataset $\mathcal{D}_s$ with $|\mathcal{D}_s| = n$, we sequentially feed samples into the target model $\mathcal{M}$ to capture domain-specific activation patterns. For each input x_i, we extract the activation vector $\omega_{pred}^{(i)}$ from the FFNs. This vector concatenates the neuron activations associated with the Gate, Up, and Down projection weights across layers. We then aggregate these vectors over $\mathcal{D}_s$ to construct the *Neuron Activation Anchor*, denoted as $\mathbf{C}_{pred}$. This anchor characterizes the typical neuronal response pattern under the specific domain distribution:

$$\mathbf{C}_{pred} = \frac{1}{n} \sum_{i=1}^{n} \omega_{pred}^{(i)} \tag{9}$$

During the fine-tuning phase, let $\omega_{train}^{(i)}$ denote the real-time activation value of the FFNs given input x_i. By aligning the dynamic $\omega_{train}^{(i)}$ with the pre-computed activation anchor $\mathbf{C}_{pred}$, we explicitly guide the model to internalize domain-specific knowledge at a fine-grained neuron level. This alignment is achieved by minimizing the Mean Squared Error (MSE) between the current activation patterns and the anchor:

$$\mathcal{L}_{\text{neu}} = \left\| \omega_{train}^{(i)} - \mathbf{C}_{pred} \right\|^2 \tag{10}$$

To balance training efficiency with model performance, we introduce an adaptive gating mechanism. Based on the hypothesis that incorporating Neuron Alignment Loss is most effective after the CE loss has stabilized, we utilize the CE loss $\mathcal{L}_{\text{ce}}$ as a proxy for training maturity. Specifically, the Neuron Alignment Loss $\mathcal{L}_{\text{neu}}$ is activated only when $\mathcal{L}_{\text{ce}}$ falls below a predefined threshold γ. The final objective combines both terms dynamically:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{ce}} + \mathbb{I}(\mathcal{L}_{\text{ce}} < \gamma) \cdot \mathcal{L}_{\text{neu}} \tag{11}$$

This strategy ensures a coarse-to-fine optimization process, where the model first achieves precise semantic alignment before further exploring deep neural knowledge and representations.

4 Experiments

4.1 Experimental Setup

Instruction Tuning Datasets Following (Wang et al., 2025), we evaluate model capabilities across distinct domains using three representative real-world instruction tuning datasets: GSM8K (Cobbe et al., 2021) for mathematical reasoning, BioInstruct (Tran et al., 2024) for biomedical QA, and DialogSum (Chen et al., 2021) for summarization. These datasets serve as benchmarks for both training and evaluation. We employ ROUGE-L to evaluate model performance on the BioInstruct and DialogSum datasets, while adopting Accuracy as the metric for the GSM8K. Detailed descriptions of the datasets can be found in Appendix A.1.

Datasets for Anti-Forgetting Testing To assess the model's generalization ability and resistance to forgetting, we adopt MMLU (Hendrycks et al., 2020), BBH (Suzgun et al., 2023), and TyDiQA (Clark et al., 2020) as held-out benchmarks, aiming to verify whether the model retains its general-purpose knowledge question answering, reasoning, and multilingual capabilities after downstream task fine-tuning. For ease of subsequent analysis, we report the average score across these three benchmarks as the CF (Catastrophic Forgetting Evaluation) score, which quantifies the extent to which the model preserves its general capabilities throughout fine-tuning. Detailed descriptions of each dataset are provided in Appendix A.2.

Models To comprehensively evaluate the NGFT framework, we employ models from the Llama-3 (Dubey et al., 2024), Qwen2.5 (Yang et al., 2024), and Mistral (Jiang et al., 2023) families. Our primary experiments utilize three representative models: Llama-3.1-8B-Instruct, Qwen2.5-7B-Instruct, and Mistral-7B-Instruct-v0.3. Furthermore, to verify the scalability and robustness, we also examine variants with diverse parameter sizes across these families (including 0.5B to 14B). Detailed results are provided in Appendix A.3.

Baselines To evaluate the effectiveness of our proposed framework, we perform comparative experiments from a modular perspective. In terms of data selection, we benchmark our method against random selection and established baselines such as Nuggets (Li et al., 2024c), CCS (Zheng et al., 2023), LESS (Xia et al., 2024), and Data Whisperer (Wang et al., 2025). Regarding neuron selection, we compare our approach with random selection, MathNeuro (Christ et al., 2025), NeFT (Xu et al., 2025), and NCFT (Leng and Xiong, 2025). For the loss function, we compare against the standard Cross-Entropy loss. Finally, at the framework level, we compare our method against FPFT (with random data selection) and a composite Strong Baseline (Data Whisperer + NCFT + CE Loss). Detailed descriptions of these methods are provided in Appendix A.4.

4.2 Experimental Results

4.2.1 Main Results

Superior Performance Across Datasets. As shown in Table 1, experimental results demonstrate that NGFT exhibits both robustness and efficiency across various domains and model architectures. In the full data setting, NGFT consistently outperforms FPFT on the Qwen2.5-7B-Instruct, Llama3.1-8B-Instruct, and Mistral-7B-Instruct-v0.3, achieving average gains of 2.14, 2.11, and 2.80 points across the three datasets, respectively. Furthermore, it surpasses Strong Baseline

Model	Method	Metric	GSM8K				DialogSum				BioInstruct			
			1%	5%	10%	Full	1%	5%	10%	Full	1%	5%	10%	Full
Qwen2.5-7B-Instruct	Zero-shot	Test	42.91				23.27				30.68			
		CF	–				68.62				–			
	FPFT	Test	77.48	78.99	80.81	84.91	34.49	35.17	36.68	37.78	36.12	37.53	36.92	40.21
		CF	57.24	56.25	56.31	58.09	65.96	64.33	64.33	57.88	62.18	65.85	62.30	58.10
	Strong Baseline	Test	80.81	83.01	84.22	85.75	33.70	36.24	36.98	38.24	36.45	38.34	38.96	41.21
		CF	65.46	65.35	66.10	67.20	68.81	67.81	67.95	64.59	69.34	68.79	67.18	62.53
	NGFT	Test	**82.18**	**84.61**	**86.13**	**87.64**	**34.83**	**37.15**	**38.16**	**39.41**	**38.79**	**39.60**	**40.97**	**42.27**
		CF	65.75	66.27	65.73	67.10	69.99	68.50	69.34	64.21	69.68	70.31	70.46	65.34
Llama3.1-8B-Instruct	Zero-shot	Test	36.22				16.66				27.53			
		CF	–				66.93				–			
	FPFT	Test	62.77	67.37	69.67	77.71	32.77	34.77	35.64	38.10	30.70	33.79	34.82	40.57
		CF	63.23	62.53	62.50	63.48	47.57	48.10	49.35	38.27	51.22	54.93	54.95	49.04
	Strong Baseline	Test	61.11	73.10	75.51	77.64	34.84	**36.53**	37.27	38.58	34.34	35.83	36.63	40.84
		CF	68.15	67.80	68.56	68.25	56.21	58.22	58.47	48.30	62.16	61.85	60.03	55.05
	NGFT	Test	**66.94**	**73.85**	**78.47**	**80.75**	**34.92**	36.33	**37.85**	**39.79**	**35.36**	**38.37**	**38.87**	**42.19**
		CF	67.90	68.61	69.33	68.86	62.14	60.62	63.70	54.43	63.46	63.98	64.44	56.80
Mistral-7B-Instruct	Zero-shot	Test	25.24				20.46				21.40			
		CF	–				60.56				–			
	FPFT	Test	39.19	39.95	39.04	57.24	28.15	29.59	31.33	36.42	22.61	26.61	30.15	38.14
		CF	51.26	50.70	49.88	47.20	26.20	18.12	15.54	14.02	25.18	28.75	26.33	12.80
	Strong Baseline	Test	37.14	43.22	45.94	57.85	33.19	34.87	35.43	36.87	23.60	28.29	32.12	38.84
		CF	58.64	57.89	58.55	56.02	29.61	30.56	33.45	21.66	38.99	35.06	27.83	26.91
	NGFT	Test	**42.46**	**47.61**	**52.01**	**62.55**	**33.96**	**34.95**	**36.51**	**38.06**	**28.50**	**32.42**	**33.99**	**39.60**
		CF	58.30	58.29	59.02	58.07	33.97	32.17	39.44	27.71	39.39	47.08	33.11	30.77

Table 1: Performance Comparison of NGFT with Zero-Shot, FPFT, and Strong Baseline (Data Whisperer + NCFT + CE Loss, represent the most SOTA baseline for data selection, neuron selection, and loss function optimization) when training with top 1%, 5%, 10%, and full GSM8K, DialogSum, and BioInstruct data. The Test Metric measures in-domain performance, while the CF Metric assesses average performance on MMLU, BBH, and TyDiQA to reflect forgetting mitigation. **Bold** and underline indicates best Test and CF performance.

(Data Whisperer + NCFT + CE Loss) by average margins of 1.37, 1.89, and 2.21 points, respectively. Notably, NGFT excels in low-resource regimes. For examples, using Mistral-7B-Instruct-v0.3 on the GSM8K with 1%, 5%, and 10% data rates, NGFT outperforms FPFT by 3.72, 7.66, and 12.97 points, respectively, while exceeding Strong Baseline by 5.32, 4.39, and 6.07 points. These results highlight its capability to extract high-quality features from sparse data. In addition, the cost analysis in Appendix B.7 demonstrates that NGFT can achieve performance comparable to FPFT with only 20.63% of the computational cost. With an additional 11.3% of the FPFT cost, it can consistently outperform FPFT by 1.5–3 points, validating the superior efficiency of the proposed framework.

Robust Resistance to Catastrophic Forgetting. NGFT demonstrates exceptional efficacy in mitigating catastrophic forgetting and maintaining model generalization, even inducing significant positive transfer phenomena in certain settings. Specifically, experiments on Qwen2.5-7B-Instruct and Llama3.1-8B-Instruct show that, across varying proportions of the BioInstruct dataset, NGFT achieves average improvements of 6.84 points and 9.63 points compared to FPFT, and 1.98 points and 2.39 points over Strong Baseline. Notably, when fine-tuning Qwen2.5-7B-Instruct with only 5% and 10% of the BioInstruct data, the model's general capability scores (70.31 and 70.46) surpassed that of the Zero-shot (un-finetuned) baseline (68.62). This validates the superiority and practicality of the NGFT framework in preserving pre-trained knowledge while acquiring new domain expertise.

4.2.2 Effectiveness of Adaptive Task-Knowledge Neuron Selection

As showed in Table 3, NGFT_{NS} significantly surpasses baseline methods through its adaptive neuron selection. On Qwen2.5-7B-Instruct, it outperforms random selection by margins of 25.40, 8.89, and 9.84 points across three datasets. Moreover, compared to the SOTA method (NCFT), NGFT_{NS} achieves consistent improvements in both task performance (up to 1.56 points) and knowledge retention (up to 3.86 points) across multiple model architectures. The key driver of this improvement is elucidated in Appendix B.1. The adaptive activation variance threshold adopted by NGFT effectively resolves the inherent plasticity-stability dilemma in fixed-ratio methods, and the derived selection ratio generalizes well to other baselines. Moreover, as detailed in Appendix B.4, ablation studies demonstrate that a Coefficient of Variation threshold within the range of 0.10 to 0.50 can con-

Method	GSM8K			DialogSum			BioInstruct		
	1%	5%	10%	1%	5%	10%	1%	5%	10%
					Qwen2.5-7B-Instruct				
Random	77.48 / 55.95	78.99 / 55.91	80.81 / 55.44	34.49 / 65.72	35.17 / 62.61	36.68 / 65.02	36.12 / 63.60	37.53 / 63.52	36.92 / 62.33
CCS	80.14 / 55.53	81.20 / 56.31	82.11 / 56.89	34.44 / 66.21	36.00 / 63.28	36.74 / 64.63	37.23 / 64.02	37.99 / 62.87	38.51 / 61.68
Nuggets	78.47 / 55.44	80.51 / 55.34	82.87 / 57.06	34.65 / 66.34	36.02 / 62.12	35.99 / 65.37	36.96 / 63.83	36.75 / 63.20	37.67 / 62.91
LESS	80.14 / 55.89	81.05 / 56.11	82.03 / 56.27	**34.96** / 65.98	35.78 / 63.99	36.62 / 65.12	36.88 / 64.55	37.21 / 63.77	38.09 / 62.15
Data Whisperer	79.70 / 56.37	81.65 / 55.49	82.64 / 56.83	34.51 / 66.45	**36.14** / 63.42	37.04 / 65.51	36.79 / 64.07	37.85 / 63.19	38.63 / 62.68
NGFT$_{DS}$	**80.41** / 56.93	**81.86** / 56.13	**83.26** / 56.31	34.77 / 67.21	36.05 / 64.20	**37.14** / 66.28	**38.35** / 63.39	**38.60** / 63.83	**39.64** / 63.32
					Llama3.1-8B-Instruct				
Random	62.78 / 59.18	67.39 / 63.53	69.68 / 61.50	32.77 / 49.15	34.71 / 45.29	35.64 / 49.15	30.70 / 47.83	33.79 / 55.83	34.82 / 51.80
CCS	62.47 / 61.68	70.81 / 62.81	72.79 / 62.47	34.08 / 48.95	35.19 / 45.96	36.35 / 51.83	32.20 / 47.46	35.77 / 56.40	36.59 / 52.82
Nuggets	63.00 / 60.26	71.65 / 62.95	72.33 / 62.41	33.00 / 47.93	35.46 / 46.09	35.55 / 50.17	31.41 / 48.14	34.12 / 54.25	35.60 / 53.27
LESS	62.85 / 60.91	71.57 / 63.87	**74.15** / 61.85	32.93 / 48.66	**36.21** / 47.25	36.38 / 50.12	31.75 / 49.98	34.99 / 55.47	35.89 / 54.36
Data Whisperer	63.14 / 61.46	71.27 / 63.19	73.54 / 62.43	33.48 / 48.09	35.53 / 46.71	**36.73** / 50.88	32.27 / 49.51	34.66 / 56.13	36.47 / 53.84
NGFT$_{DS}$	**63.50** / 62.60	**72.32** / 63.60	73.26 / 62.81	**34.74** / 47.65	35.62 / 47.57	36.64 / 50.03	**34.02** / 51.39	**36.96** / 55.26	**37.76** / 54.33
					Mistral-7B-Instruct-v0.3				
Random	39.19 / 50.35	39.95 / 50.88	39.04 / 48.30	28.15 / 21.33	29.59 / 16.65	31.33 / 13.34	21.61 / 23.46	26.61 / 27.38	30.15 / 25.92
CCS	**41.02** / 48.17	43.67 / 50.67	45.20 / 46.45	30.28 / 24.82	32.63 / 16.08	33.14 / 15.07	23.03 / 24.63	28.93 / 28.48	31.01 / 24.46
Nuggets	40.19 / 49.07	43.67 / 52.57	44.28 / 47.05	28.95 / 23.42	33.34 / 17.28	33.96 / 13.27	22.71 / 25.33	28.37 / 27.98	29.91 / 25.46
LESS	40.48 / 49.31	43.29 / 52.63	45.34 / 45.95	30.15 / 24.85	33.31 / 14.84	33.73 / 14.75	25.12 / 23.12	28.72 / 29.48	31.69 / 25.68
Data Whisperer	40.80 / 49.82	43.89 / 52.18	45.79 / 46.68	30.19 / 22.24	**33.52** / 15.22	**34.28** / 14.46	24.85 / 23.61	29.53 / 28.95	31.27 / 26.34
NGFT$_{DS}$	40.96 / 51.00	**43.98** / 51.03	**45.85** / 49.10	**30.45** / 25.18	33.38 / 17.33	34.09 / 15.14	**25.42** / 24.96	**30.32** / 28.55	**32.62** / 26.50

Table 2: Performance (Test / CF) Comparison of NGFT with Data Selection Baselines when training with top 1%, 5%, and 10% of the GSM8K, DialogSum, and BioInstruct data. The Test Metric measures in-domain performance, while the CF Metric assesses average performance on MMLU, BBH, and TyDiQA to reflect the model's ability to mitigate forgetting. **Bold** indicates the best Test performance, and underline indicates the best CF performance.

sistently and stably improve performance. To ensure stability and experimental consistency, this work uniformly adopts the well-performing and stable value of 0.10 as the default Coefficient of Variation threshold across all experiments.

Method	GSM8K	DialogSum	BioInstruct
		Qwen2.5-7B-Instruct	
Random	62.17 / 63.31	29.87 / 63.08	32.12 / 62.90
MathNeuro	82.87 / 67.05	36.47 / 62.01	38.61 / 61.78
NeFT	83.57 / 65.09	37.93 / 61.84	40.47 / 62.52
NCFT	85.75 / 67.20	38.24 / 62.39	41.21 / 62.37
NGFT$_{NS}$	**87.57** / 67.28	**38.76** / 68.29	**41.69** / 66.68
		Llama3.1-8B-Instruct	
Random	64.37 / 65.81	30.75 / 50.17	29.41 / 52.51
MathNeuro	75.97 / 66.70	37.48 / 49.34	36.61 / 52.27
NeFT	74.37 / 67.61	38.52 / 51.59	40.29 / 53.35
NCFT	77.64 / 66.25	38.58 / 48.30	40.84 / 55.05
NGFT$_{NS}$	**79.91** / 68.64	**38.84** / 55.98	**41.46** / 58.70
		Mistral-7B-Instruct-v0.3	
Random	42.31 / 53.41	26.81 / 20.64	25.04 / 24.26
MathNeuro	53.30 / 53.26	35.69 / 26.85	34.23 / 25.51
NeFT	56.71 / 52.92	36.41 / 25.62	37.58 / 23.17
NCFT	57.85 / 56.02	36.87 / 21.66	38.84 / 26.91
NGFT$_{NS}$	**58.00** / 56.02	**37.30** / 26.97	**39.18** / 29.50

Table 3: Performance (Test / CF) Comparison of NGFT$_{NS}$ with Neuron Selection Baselines on full data.

4.2.3 Effective Data Selection with Reduced Computational Costs

As shown in Table 2, the NGFT$_{DS}$ method exhibits superior performance. Compared to the SOTA baseline Data Whisperer, NGFT improves average performance on the BioInstruct dataset by 1.28, 1.78, and 0.90 points for Qwen2.5-7B-Instruct, Llama3.1-8B-Instruct, and Mistral-7B-Instruct-v0.3, respectively. Furthermore, models trained on subsets selected by NGFT consistently outperform other baselines in general capability metrics. While these gains are incremental, they serve as robust evidence of our method's capacity to mitigate catastrophic forgetting. Beyond performance, our approach offers a decisive advantage in computational efficiency. As detailed in Appendix B.2, NGFT requires only a single forward pass with no auxiliary operations, achieving the lowest overhead among all compared methods. Notably, it delivers a 25.7× speedup over the Nuggets algorithm on the DialogSum dataset.

4.2.4 Effectiveness and Intervention Timing of Neuron Alignment Loss

Figure 2 demonstrates the effectiveness of Neuron Alignment Loss On Qwen2.5-7B-Instruct and Llama3.1-8B-Instruct, in-domain performance improves by averages of 1.43 and 1.48 points, respectively, while general task performance increases by 7.11 and 7.29 points. These results indicate that Neuron Alignment Loss effectively captures neuron-level features to enhance domain capabilities and mitigate catastrophic forgetting during FPFT. Mechanistically, by anchoring initial activation patterns, Neuron Alignment Loss restricts the variation of non-critical neurons more strictly than CE loss. This effectively regularizes overfitting and achieves a superior trade-off between preserving prior knowledge and learning new tasks (see Appendix B.6 for detailed analysis). Furthermore, regarding the intervention timing, as shown in Appendix B.5, introducing the neuron alignment loss

when the CE loss converges to 0.03–0.10 can consistently improve the model performance. Based on ablation results, this work uniformly adopts the well-performing value of 0.04 as the intervention threshold across all experiments in this study.

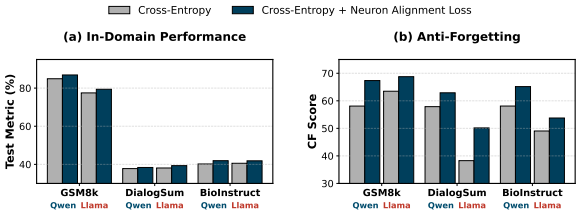

Figure 2: Impact of Neuron Alignment Loss on model performance and ablation on intervention timing.

4.3 Analysis

Neuron Selection is Critical for Effective Data Selection. We validate the necessity of data selection based on Task-Knowledge Neurons by comparing our strategy against selection based on the entire FFN. As shown in Figure 3, expanding the selection scope to the full FFN consistently degrades performance, resulting in drops of 1.03, 0.63, and 0.85 points on GSM8K, DialogSum, and BioInstruct, respectively. This decline is attributed to the noise introduced by the vast number of silent or randomly activated neurons within the full FFN. NGFT mitigates this by focusing exclusively on the Task-Knowledge Neuron subspace, effectively filtering noise and ensuring accurate data valuation.

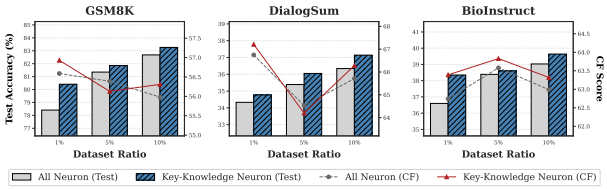

Figure 3: Impact of neuron analysis scope (all FFN neurons or Task-Knowledge Neurons) on data selection performance.

Data Selection Helps Optimize Neuron Alignment Loss. We evaluated three data selection strategies for extracting anchor activation vectors for neuron alignment loss: random sampling, NGFT framework, and full-data. As illustrated in Figure 4, random sampling consistently underperforms compared to the full dataset. In contrast, NGFT demonstrates superior efficiency; utilizing only 5% of the data selected by NGFT surpasses the full-data baseline. Interestingly, increasing this ratio to 50% degrades performance, suggesting that an excessive inclusion of low-quality samples introduces noise and distorts the feature distribution. These results confirm that NGFT data selection successfully isolates high-value representative samples, ensuring superior neuron feature alignment with minimal data usage.

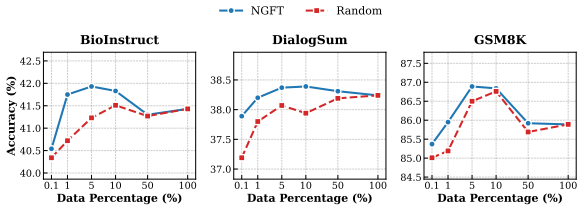

Figure 4: Comparison of anchor activation vector effectiveness for optimizing Neuron Alignment Loss across three data selection methods: random selection, NGFT selection, and the full dataset.

Scalability Across Different Model Sizes. To assess the generalization and efficiency of NGFT on different scales, we extended our evaluation to other models in the Qwen2.5 series. As shown in Table 11, NGFT consistently achieves superior performance in all parameter sizes. For example, on Qwen2.5-14B-Instruct, it outperforms FPFT by average margins of 2.26 points in in-domain performance and 7.62 points in anti-forgetting capability across three datasets, and surpasses the strong baseline by 1.21 and 1.14 points, respectively. These results show that NGFT effectively adapts to diverse computational budgets and deployment scenarios, with exceptional scalability and robustness.

5 Conclusion

In this paper, we propose NGFT, a LLM fine-tuning framework guided by neuron activation patterns acting as a universal proxy. The framework integrates three modules: Task-Knowledge Neuron Selection, High-Quality Data Selection, and Neuron Alignment Loss Optimization, aiming to efficiently stimulate specific model capabilities for downstream tasks. Extensive experiments across three model families on both domain-specific and general benchmarks demonstrate that NGFT not only achieves superior in-domain performance compared to FPFT and strong baselines while effectively preventing catastrophic forgetting. The framework demonstrates excellent robustness and flexibility, offering a scalable solution for customizing LLMs in diverse scenarios.

Limitations

While NGFT exhibits efficiency and robustness across models and benchmarks, our approach has inherent limitations. Grounded in neuron activation pattern analysis, NGFT requires access to model weights and internal states, making it applicable only to open-weights LLMs and inapplicable to proprietary black-box models such as GPT-5 with limited internal accessibility. Additionally, computational constraints precluded large-scale evaluations. With sufficient resources, future work will conduct larger-scale experiments to validate the generalization to more complex scenarios.

Acknowledgments

This work was supported in part by the Science and Technology Development Fund of Macau SAR (Grant Nos. FDCT/0007/2024/AKP, EF2024-00185-FST), the UM and UMDF (Grant Nos. MYRG-GRG2024-00165-FST-UMDF, MYRG-GRG2025-00236-FST), the Tencent AI Lab Rhino-Bird Research Program (Grant No. EF2023-00151-FST), the Stanley Ho Medical Development Foundation (Grant No. SHMDF-AI/2026/001), and the National Natural Science Foundation of China (Grant No. 62266013).

References

Alessandro Achille and Stefano Soatto. 2018. Emergence of invariance and disentanglement in deep representations. *Journal of Machine Learning Research*, 19(50):1–34.

Yihan Cao, Yanbin Kang, and Lichao Sun. 2023. Instruction mining: High-quality instruction data selection for large language models. *arXiv preprint arXiv:2307.06290*, 1(3):6.

Guanhua Chen, Yutong Yao, Ci-Jun Gao, Lidia S. Chao, Feng Wan, and Derek F. Wong. 2025a. Not all lora parameters are essential: Insights on inference necessity. *CoRR*, abs/2503.23360.

Lichang Chen, Shiyang Li, Jun Yan, Hai Wang, Kalpa Gunaratna, Vikas Yadav, Zheng Tang, Vijay Srinivasan, Tianyi Zhou, Heng Huang, and Hongxia Jin. 2024. Alpagasus: Training a better alpaca with fewer data. In *The Twelfth International Conference on Learning Representations*.

Xin Chen, Feng Jiang, Yiqian Zhang, Hardy Chen, Shuo Yan, Wenya Xie, Min Yang, and Shujian Huang. 2026a. Reasoning while asking: Transforming reasoning large language models from passive solvers to proactive inquirers. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 35069–35090. Association for Computational Linguistics.

Xin Chen, Junchao Wu, Shu Yang, Runzhe Zhan, Zeyu Wu, Ziyang Luo, Di Wang, Min Yang, Lidia S Chao, and Derek F Wong. 2025b. Repreguard: Detecting llm-generated text by revealing hidden representation patterns. *Transactions of the Association for Computational Linguistics*, 13:1812–1831.

Xin Chen, Junchao Wu, Shu Yang, Runzhe Zhan, Zeyu Wu, Min Yang, Shujian Huang, Lidia S. Chao, and Derek F. Wong. 2026b. Neuron-aware data selection in instruction tuning for large language models. *arXiv preprint arXiv:2603.13201*.

Yulong Chen, Yang Liu, Liang Chen, and Yue Zhang. 2021. Dialogsum: A real-life scenario dialogue summarization dataset. *arXiv preprint arXiv:2105.06762*.

Bryan R Christ, Zachary Gottesman, Jonathan Kropko, and Thomas Hartvigsen. 2025. Math neurosurgery: Isolating language models' math reasoning abilities using only forward passes. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 24803–24840, Vienna, Austria. Association for Computational Linguistics.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics*, 8:454–470.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022a. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8493–8502.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022b. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8493–8502, Dublin, Ireland. Association for Computational Linguistics.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. In *Advances in Neural Information Processing Systems*, volume 36, pages 10088–10115. Curran Associates, Inc.

Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zonghan Yang, Yusheng Su, Shengding Hu, Yulin Chen, Chi-Min Chan, Weize Chen, Jing Yi, Weilin Zhao, Xiaozhi Wang, Zhiyuan Liu, Hai-Tao Zheng, Jianfei Chen, Yang Liu, Jie Tang, Juanzi Li, and Maosong Sun. 2022. Delta tuning: A comprehensive study of parameter efficient methods for pre-trained language models. *Preprint*, arXiv:2203.06904.

Guanting Dong, Hongyi Yuan, Keming Lu, Chengpeng Li, Mingfeng Xue, Dayiheng Liu, Wei Wang, Zheng Yuan, Chang Zhou, and Jingren Zhou. 2024. How abilities in large language models are affected by supervised fine-tuning data composition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 177–198, Bangkok, Thailand. Association for Computational Linguistics.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yuto Harada, Yusuke Yamauchi, Yusuke Oda, Yohei Oseki, Yusuke Miyao, and Yu Takagi. 2025. Massive supervised fine-tuning experiments reveal how data, layer, and training factors shape LLM alignment quality. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 22360–22381, Suzhou, China. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

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

Zhiqiang Hu, Lei Wang, Yihuai Lan, Wanyu Xu, Ee-Peng Lim, Lidong Bing, Xing Xu, Soujanya Poria, and Roy Lee. 2023. LLM-adapters: An adapter family for parameter-efficient fine-tuning of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5254–5276, Singapore. Association for Computational Linguistics.

Jianheng Huang, Leyang Cui, Ante Wang, Chengyi Yang, Xinting Liao, Linfeng Song, Junfeng Yao, and Jinsong Su. 2024. Mitigating catastrophic forgetting in large language models with self-synthesized rehearsal. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1416–1428, Bangkok, Thailand. Association for Computational Linguistics.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Kenji Kawaguchi, Zhun Deng, Xu Ji, and Jiaoyang Huang. 2023. How does information bottleneck help deep learning? In *International conference on machine learning*, pages 16049–16096. PMLR.

Sohee Kim, Soohyun Ryu, Joonhyung Park, and Eunho Yang. 2025. Unveiling the response of large vision-language models to visually absent tokens. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 21546–21568, Suzhou, China. Association for Computational Linguistics.

Yongqi Leng and Deyi Xiong. 2025. Towards understanding multi-task learning (generalization) of LLMs via detecting and exploring task-specific neurons. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 2969–2987, Abu Dhabi, UAE. Association for Computational Linguistics.

Dingcheng Li, Zheng Chen, Eunah Cho, Jie Hao, Xiaohu Liu, Fan Xing, Chenlei Guo, and Yang Liu. 2022. Overcoming catastrophic forgetting during domain adaptation of seq2seq language generation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5441–5454, Seattle, United States. Association for Computational Linguistics.

Hongyu Li, Liang Ding, Meng Fang, and Dacheng Tao. 2024a. Revisiting catastrophic forgetting in large language model tuning. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 4297–4308, Miami, Florida, USA. Association for Computational Linguistics.

Ming Li, Yong Zhang, Zhitao Li, Jiuhai Chen, Lichang Chen, Ning Cheng, Jianzong Wang, Tianyi Zhou, and Jing Xiao. 2024b. From quantity to quality: Boosting LLM performance with self-guided data selection for instruction tuning. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7602–7635, Mexico City, Mexico. Association for Computational Linguistics.

Yunshui Li, Binyuan Hui, Xiaobo Xia, Jiaxi Yang, Min Yang, Lei Zhang, Shuzheng Si, Ling-Hao Chen, Junhao Liu, Tongliang Liu, Fei Huang, and Yongbin Li. 2024c. One-shot learning as instruction data prospector for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4586–4601, Bangkok, Thailand. Association for Computational Linguistics.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. 2024. Dora: Weight-decomposed low-rank adaptation. In *Forty-first International Conference on Machine Learning*.

Kai Lv, Yuqing Yang, Tengxiao Liu, Qipeng Guo, and Xipeng Qiu. 2024. Full parameter fine-tuning for large language models with limited resources. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8187–8198, Bangkok, Thailand. Association for Computational Linguistics.

Max Marion, Ahmet Üstün, Luiza Pozzobon, Alex Wang, Marzieh Fadaee, and Sara Hooker. 2023. When less is more: Investigating data pruning for pretraining llms at scale. *arXiv preprint arXiv:2309.04564*.

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

Ran Song, Shizhu He, Shuting Jiang, Yantuan Xian, Shengxiang Gao, Kang Liu, and Zhengtao Yu. 2024. Does large language model contain task-specific neurons? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7101–7113, Miami, Florida, USA. Association for Computational Linguistics.

Xiao Sun, Naigang Wang, Chia-Yu Chen, Jiamin Ni, Ankur Agrawal, Xiaodong Cui, Swagath Venkataramani, Kaoutar El Maghraoui, Vijayalakshmi (Viji) Srinivasan, and Kailash Gopalakrishnan. 2020. Ultra-low precision 4-bit training of deep neural networks. In *Advances in Neural Information Processing Systems*, volume 33, pages 1796–1807. Curran Associates, Inc.

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

Tianyi Tang, Wenyang Luo, Haoyang Huang, Dongdong Zhang, Xiaolei Wang, Wayne Xin Zhao, Furu Wei, and Ji-Rong Wen. 2024. Language-specific neurons: The key to multilingual capabilities in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5701–5715.

Hieu Tran, Zhichao Yang, Zonghai Yao, and Hong Yu. 2024. Bioinstruct: instruction tuning of large language models for biomedical natural language processing. *Journal of the American Medical Informatics Association*, 31(9):1821–1832.

Shaobo Wang, Xiangqi Jin, Ziming Wang, Jize Wang, Jiajun Zhang, Kaixin Li, Zichen Wen, Zhong Li, Conghui He, Xuming Hu, and Linfeng Zhang. 2025. Data whisperer: Efficient data selection for task-specific LLM fine-tuning via few-shot in-context learning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 23287–23305, Vienna, Austria. Association for Computational Linguistics.

Xiaozhi Wang, Kaiyue Wen, Zhengyan Zhang, Lei Hou, Zhiyuan Liu, and Juanzi Li. 2022. Finding skill neurons in pre-trained transformer-based language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11132–11152, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yong Wang, Longyue Wang, Shuming Shi, Victor O. K. Li, and Zhaopeng Tu. 2020. Go from the general to the particular: Multi-domain translation with domain transformation networks. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 9233–9241. AAAI Press.

Zihan Wang, Deli Chen, Damai Dai, Runxin Xu, Zhuoshu Li, and Yu Wu. 2024. Let the expert stick to his last: Expert-specialized fine-tuning for sparse architectural large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 784–801, Miami, Florida, USA. Association for Computational Linguistics.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. 2024. LESS: Selecting influential data for targeted instruction tuning. In *International Conference on Machine Learning (ICML)*.

Xiaobo Xia, Jiale Liu, Jun Yu, Xu Shen, Bo Han, and Tongliang Liu. 2022. Moderate coreset: A universal method of data selection for real-world data-efficient deep learning. In *The Eleventh International Conference on Learning Representations*.

Haoyun Xu, Runzhe Zhan, Yingpeng Ma, Derek F Wong, and Lidia S Chao. 2025. Let's focus on neuron: Neuron-level supervised fine-tuning for large language model. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 9393–9406.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, and 1 others. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Bolin Zhang, Jiahao Wang, Qianlong Du, Jiajun Zhang, Zhiying Tu, and Dianhui Chu. 2025. A survey on data selection for llm instruction tuning. *Journal of Artificial Intelligence Research*, 83.

Zhihao Zhang, Jun Zhao, Qi Zhang, Tao Gui, and Xuan-Jing Huang. 2024. Unveiling linguistic regions in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6228–6247.

Haizhong Zheng, Rui Liu, Fan Lai, and Atul Prakash. 2023. Coverage-centric coreset selection for high pruning rates. In *The Eleventh International Conference on Learning Representations*.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, and 1 others. 2023. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*.

A Dataset Details

A.1 Instruction Tuning Dataset Details

To comprehensively evaluate model performance, this experiment selects three instruction-tuning datasets with significant domain differences for training and testing. GSM8K (Cobbe et al., 2021) serves as a high-quality benchmark for grade school math word problems, covering various topics ranging from basic arithmetic operations to fractions and percentages. The inclusion of this dataset aims to rigorously examine the mathematical logical reasoning capabilities of LLMs. BioInstruct (Tran et al., 2024) is an instruction-tuning dataset specifically customized for the biomedical domain. It contains 25,005 pairs of instructions and demonstration samples, encompassing a wide range of biomedical tasks. This dataset is utilized to verify the model's ability to process information and apply knowledge within specific medical fields. We split the dataset into training and test sets in a 9:1 ratio. DialogSum (Chen et al., 2021) is a dialogue summarization dataset oriented toward real-life scenarios, involving diverse topics from daily life. This dataset is primarily used to test the model's comprehension of long texts as well as its capacity for extracting key information from dialogues and generating summaries.

A.2 Additional Dataset Details

To validate the model's resistance to catastrophic forgetting and assess its generalization capabilities, we employ three additional datasets for evaluation. MMLU (Hendrycks et al., 2020) serves as a comprehensive benchmark for evaluating multi-task accuracy. It encompasses 57 diverse subjects, ranging from elementary to professional levels. Utilizing multiple-choice questions, MMLU is designed to rigorously assess the model's breadth of knowledge, cross-domain integration, and problem-solving abilities under both zero-shot and few-shot settings. BBH (Suzgun et al., 2023) is a curated subset of 23 challenging tasks selected from the BIG-Bench suite. It focuses on complex reasoning capabilities, such as symbolic manipulation, logical deduction, and commonsense reasoning. TyDiQA (Clark et al., 2020) is a multilingual question-answering benchmark covering 11 typologically diverse languages. This dataset evaluates the model's capacity for information retrieval and reading comprehension across diverse language families, making it critical for assessing multilingual generaliza-

tion, particularly in non-English and low-resource contexts.

A.3 Training Model Details

In this paper, we employed 8 distinct LLMs for training and evaluation. We designed three mainstream LLM frameworks covering a parameter range from 0.5B to 14B, specifically: Qwen2.5-0.5B-Instruct, Qwen2.5-3B-Instruct, Qwen2.5-7B-Instruct, Qwen2.5-14B-Instruct, Llama3.2-1B-Instruct, Llama3.2-3B-Instruct, Llama3.1-8B-Instruct, and Mistral-7B-Instruct-v0.3.

All experiments were conducted using the LLAMA-Factory training framework on servers equipped with 8 NVIDIA A100 or H200 GPUs, leveraging DeepSpeed Zero-2.0 for training acceleration. To ensure fair comparison, we maintained consistent hyperparameter configurations across models for each task. Specifically, the learning rate was set to 2e-5 for all models on the DialogSum and BioInstruct datasets, while the learning rate was set to 1e-5 for the GSM8K dataset. Additionally, we utilized a batch size of 4 and a warmup ratio of 0.1 for all models. The training steps were uniformly set to 4 for all experiments. Detailed results for all models running within the NGFT framework can be found in Appendix C.

A.4 Baseline Details

Data Selection Baseline. This paper selects four representative data selection methods as baselines for comparative study:

- **Coverage-centric Coreset Selection** (Zheng et al., 2023): The core principle of this method is to maximize coverage within the feature space. By analyzing the distribution of data features, it aims to identify the most representative subset of samples. The goal is to ensure this subset faithfully preserves the topological structure and diversity of the original dataset.

- **Nuggets** (Li et al., 2024c): This method leverages the few-shot learning capabilities of LLMs to explore the quality of instruction data. By constructing a multi-task predefined anchor set and calculating the impact of candidate samples on the perplexity of anchor set tasks, it generates a "Golden Score" to quantify sample quality.

- **LESS** (Xia et al., 2024): This is a data selection strategy based on gradient information. Utilizing low-rank gradient projection technology, this method calculates the influence of training data on specific target tasks, thereby precisely locating and selecting the high-quality data subset that contributes most significantly to model optimization.

- **Data Whisperer** (Wang et al., 2025): This method is based on the intrinsic connection between In-Context Learning and fine-tuning. By evaluating the contribution of samples when used as demonstrations and combining this with attention weight calibration, it achieves efficient data selection without the need for additional training.

Neuron Selection Baseline. We conduct a comparative analysis using four representative neuron selection techniques as baselines.

- **MathNeuro** (Christ et al., 2025): This method employs the Wanda metric (defined as the product of the weight magnitude and the activation norm) to evaluate parameter importance across both mathematical and general domains. It identifies math-specific parameters by selecting the top-$K\%$ highest-scoring parameters on mathematical tasks and subsequently filtering out those that also rank within the top-$K\%$ for general tasks via a set difference operation.

- **NeFT** (Xu et al., 2025): This approach identifies key neurons by contrasting the pre-trained model with its fully fine-tuned counterpart. Specifically, it computes the cosine similarity between the weight vectors of corresponding neurons before and after fine-tuning. The top-$K\%$ neurons exhibiting the lowest similarity scores, which indicate the most significant directional shift in weights, are designated as task-sensitive neurons.

- **NCFT** (Leng and Xiong, 2025): This method adopts a gradient attribution technique to detect task-specific neurons. It approximates the impact of each neuron on the model's loss function using Taylor expansion to derive a relevance score. Based on these scores, the neurons are ranked, and the top-$k\%$ are selected as the key neurons.

B Experiment Details

B.1 NGFT Adaptive Neuron Selection Precisely Enhances Model Performance.

In this section, we investigate the neuron retention ratios of the Qwen2.5-7B-Instruct, Llama3.1-8B-Instruct, and Mistral-7B-Instruct-v0.3 models under the NGFT framework, and we further explore the potential benefits of adaptive ratio strategies for other neuron selection methods. As shown in Table 4, we report the proportion of neurons selected by NGFT. Although the absolute number of activated Task-Knowledge neurons varies across datasets due to the heterogeneity of pre-training distributions, all three models exhibit a highly consistent trend: BioInstruct shows the lowest activation ratio, followed by DialogSum, with GSM8K demonstrating the highest. This phenomenon strongly validates the superior generalizability and robustness of the NGFT_{NS} method. Unlike fixed-ratio strategies, NGFT_{NS} dynamically adjusts the number of neurons based on the model's intrinsic assessment of task complexity and cognitive demands. This task-aware adaptive mechanism ensures an optimal trade-off between compression rate and performance across tasks of varying types and difficulties.

Model	BioInstruct	DialogSum	GSM8K
Qwen2.5-7B-Instruct	7.20%	9.33%	14.67%
Llama3.1-8B-Instruct	5.28%	8.23%	12.18%
Mistral-7B-Instruct-v0.3	1.19%	14.42%	17.03%

Table 4: Neuron selection ratios of NGFT_{NS} across different models and tasks.

In subsequent experiments, we incorporate the adaptive ratios identified by NGFT_{NS} into the strong baseline NCFT as prior knowledge. As shown in Table 5, compared to conventional fixed ratios, these adaptive ratios effectively boosts the accuracy of NCFT across most scenarios, despite not outperforming the original NGFT_{NS}. These results not only demonstrate the generalizability of the learned ratios but also further corroborate the superiority of NGFT in characterizing neuron importance distributions.

B.2 Comparison of Time Cost of Different Data Selection Methods.

Taking the Qwen2.5-7B-Instruct model as a case study, we further analyze the time overhead of different data selection methods. To quantitatively evaluate the efficiency of each method, we conducted data selection experiments using four A100 GPUs (simulating a resource-constrained setting) and employed the STR formula (Wang et al., 2025) for analysis. As shown in Table 6, our proposed method requires the least amount of time. Notably, on the DialogSum dataset, it achieves a 25.7× speedup compared to the Nuggets, which incurs the highest computational cost. This is primarily because our method only requires a single forward pass for each data sample without the need for additional operations, thereby significantly reducing runtime.

Model	BioInstruct	DialogSum	GSM8K
Qwen2.5-7B-Instruct	41.83 (↑0.61)	38.45 (↑0.21)	86.81 (↑1.05)
Llama3.1-8B-Instruct	40.95 (↑0.11)	38.77 (↑0.19)	80.60 (↑2.96)
Mistral-7B-Instruct-v0.3	38.80 (↓0.04)	36.96 (↑0.09)	58.41 (↑0.56)

Table 5: Experimental results of NCFT method on three models with neuron selection ratios from NGFT_{NS} using full dataset.

Method	GSM8K	DialogSum	BioInstruct
Nugget	1.73	2.83	2.35
CCS	0.93	1.16	1.08
LESS	1.54	2.07	2.21
Data Whisperer	0.41	0.64	0.81
NGFT	0.08	0.11	0.12
Speedup	21.6×	25.72×	19.58×

Table 6: STR metrics of different data selection methods on the Qwen2.5-7B-Instruct model. A lower STR indicates lower time complexity for the method.

B.3 NGFT Outperforms Other PEFT Methods

The motivation of this work is to pursue a more efficient FPFT paradigm. In the main experimental phase, we primarily conduct core performance comparisons against the advanced FPFT frameworks NCFT (Xu et al., 2025) and NeFT (Leng and Xiong, 2025). To further comprehensively and systematically evaluate the overall performance and efficiency of NGFT framework, we additionally conduct extended comparisons with some Parameter-Efficient Fine-Tuning (PEFT) methods, specifically covering a suite of LoRA variants, including LoRA (Hu et al., 2022), QLoRA (Dettmers et al., 2023), and DoRA (Liu et al., 2024). All experiments are performed on the Qwen2.5-7B-Instruct model, and the full dataset is uniformly adopted for comparison, as these PEFT methods do not involve data selection.

Detailed experimental results are presented in Table 7. From the experimental results, we derive the following key findings:

(1) In terms of task performance, NGFT achieves state-of-the-art results across all benchmarks. On GSM8K, NGFT attains an accuracy of 87.64, representing an improvement of 1.74 points over the best-performing PEFT method DoRA, 4.24 points over LoRA, and 4.85 points over QLoRA. On DialogSum and BioInstruct, NGFT achieves ROUGE-L scores of 39.41 and 42.27, respectively, comprehensively surpassing all PEFT baselines. This demonstrates that the inherent constraints of low-rank fine-tuning restrict LoRA series methods from fully exploiting the model's capabilities, whereas FPFT frameworks can unlock the full potential of the model within a larger optimization space.

(2) In terms of resistance to catastrophic forgetting, NGFT also exhibits highly competitive performance. FPFT performs drastically worse than all PEFT methods on the CF metric (only 58.09 on GSM8K), reflecting its inherent defect of severely impairing pre-trained knowledge in the absence of regularization constraints. By updating task-knowledge neurons and adopting the neuron alignment loss, NGFT effectively mitigates forgetting, achieving a CF score of 67.10 on GSM8K, which is marginally superior to the best PEFT method DoRA. It also exhibits the best catastrophic forgetting performance on DialogSum and BioInstruct.

(3) From the perspective of the comprehensive trade-off between performance and forgetting, the LoRA series delivers reasonable anti-forgetting performance owing to the structural prior of low-rank decomposition. However, restricting weight updates to a low-rank subspace inevitably limits the upper bound of task performance. Although DoRA partially alleviates this bottleneck via sparse selection and magnitude-direction decoupling, it remains constrained by the inherent limitations of parameter-efficient frameworks. In contrast, NGFT achieves state-of-the-art performance in both performance and anti-forgetting dimensions through neuron-level knowledge-aware gradient regulation, fully validating the necessity and superiority of the proposed method.

Method	GSM8K		DialogSum		BioInstruct	
	Accuracy	CF	ROUGE-L	CF	ROUGE-L	CF
Full Parameter	84.91	58.09	37.78	57.88	40.21	58.10
LoRA	83.40	65.83	37.61	63.31	39.57	64.74
QLoRA	82.79	66.41	37.10	64.10	38.94	65.12
DoRA	85.90	66.47	38.25	63.73	41.15	65.01
NGFT	**87.64**	**67.10**	**39.41**	**64.21**	**42.27**	**65.34**

Table 7: Performance comparison between NGFT and PEFT methods on Qwen2.5-7B-Instruct.

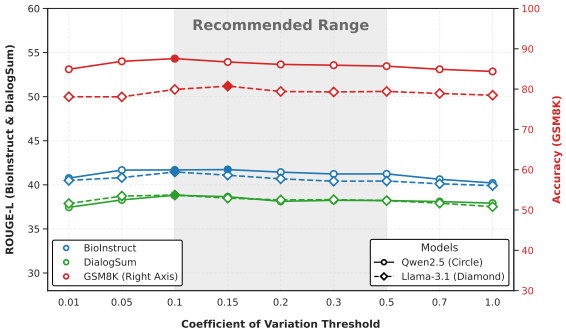

Figure 5: Performance comparison of neurons selected by different Coefficient of Variation thresholds on Qwen2.5-7B-Instruct and Llama3.1-8B-Instruct.

B.4 Performance Comparison of Different Coefficient of Variation Thresholds.

In NGFT, neuron selection is determined by a threshold derived from the coefficient of variation of activation distributions. To identify the optimal threshold, we perform ablation studies on Qwen2.5-7B-Instruct and Llama3.1-8B-Instruct across three diverse downstream tasks: GSM8K, DialogSum, and BioInstruct. Specifically, we evaluate coefficient of variation thresholds sampled from the set {0.01, 0.05, 0.10, 0.15, 0.20, 0.30, 0.40, 0.50, 0.70, 1.00}, with only the parameters of selected neurons updated during fine-tuning.

Figure 5 illustrates a consistent trend across all experimental settings: as the coefficient of variation threshold increases, task performance first improves, plateaus, and ultimately degrades. This trajectory underscores an inherent trade-off between the granularity of neuron selection and model performance. Specifically, a conservative threshold (coefficient of variation < 0.10) yields an sparse number of selected neurons, which fails to adequately capture task-relevant features and thus constrains performance improvements. Conversely, an overly permissive threshold (coefficient of variation > 0.50) incorporates some task-irrelevant redundant neurons, thereby degrading model performance.

Empirical results demonstrate that thresholds in the range of 0.10–0.50 consistently deliver stable performance gains, establishing this interval as a practical reference for real-world deployment. Optimal thresholds depend on model architecture and task characteristics: Llama3.1-8B-Instruct peaks at 0.15 (GSM8K) and 0.10 (DialogSum), while Qwen2.5-7B-Instruct performs best on GSM8K at 0.10. Thus, coefficient of variation thresholds are not universal, and task-specific tuning within

0.05–0.50 is recommended. Furthermore, from the perspective of mitigating catastrophic forgetting, a smaller coefficient of variation threshold correlates with fewer updated neurons, implying that smaller thresholds should be prioritized. Across all experimental configurations, a threshold of 0.10 consistently ranks among the top two performers, exhibiting strong robustness across both models and tasks. Consequently, to ensure reproducibility and experimental consistency, we adopt 0.10 as the recommended default coefficient of variation threshold for NGFT in this work.

B.5 Comparison of Different Neuron Alignment Loss Intervention Timing.

To investigate the impact of Neural Alignment Loss on model performance enhancement across diverse intervention timings, we conduct ablation studies using the Qwen2.5-7B-Instruct and Llama3.1-8B-Instruct models on three downstream tasks. Specifically, we evaluate the performance gains of the models by adjusting the intervention timing of Neural Alignment Loss at CE Loss values of {0.02, 0.03, 0.04, 0.05, 0.06, 0.10, 0.20, 0.40}.

Figure 6 reveals a consistent pattern across all settings: as the intervention timing of Neural Alignment Loss increases, task performance initially improves, then plateaus, and eventually degrades. This trend uncovers the distinct trade-offs of Neural Alignment Loss at different intervention points. Specifically, introducing Neural Alignment Loss too late (CE < 0.03) prevents the model from learning deep-level neuronal knowledge, thereby limiting the potential of performance improvement. Conversely, an overly early introduction (CE > 0.10) fails to enable effective alignment at the semantic level, which in turn weakens the model's performance enhancement.

Empirical results demonstrate that the optimal threshold varies with model architectures and task characteristics. For instance, the optimal threshold for Qwen2.5-7B-Instruct on the GSM8K task is CE=0.03, while that for Llama3.1-8B-Instruct on the same task is CE=0.04. Furthermore, both models achieve sustained and stable performance improvements within the range of 0.03 to 0.10 with negligible performance gaps, making this interval a recommended reference for practical deployment. Across all experimental configurations, setting the threshold to CE=0.04 consistently yields superior model performance and robustness against catastrophic forgetting, demonstrating strong cross-model and cross-task generalization. Therefore, to ensure the reproducibility and consistency of experiments, this work uses CE=0.04 as the default intervention threshold for Neural Alignment Loss in this study.

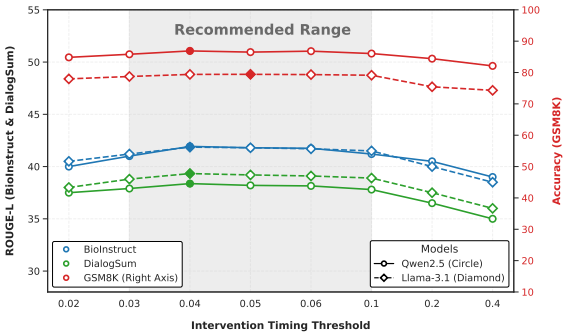

Figure 6: Performance comparison across different intervention timings of Neural Alignment Loss on Qwen2.5-7B-Instruct and Llama3.1-8B-Instruct.

Section	GSM8K	DialogSum	BioInstruct
	Only Cross-Entropy		
Task-Knowledge Neurons	2.98×10^{-4}	4.54×10^{-4}	6.81×10^{-4}
Non-Task-Knowledge Neurons	2.64×10^{-4}	4.22×10^{-4}	6.67×10^{-4}
	Cross-Entropy + Neuron Alignment Loss		
Task-Knowledge Neurons	2.71×10^{-4}	4.26×10^{-4}	6.36×10^{-4}
Non-Task-Knowledge Neurons	1.89×10^{-4}	3.77×10^{-4}	4.48×10^{-4}
$\Delta_{\text{Task-Knowledge Neurons}}$	9.06%	6.16%	6.60%
$\Delta_{\text{Non-Task-Knowledge Neurons}}$	28.40%	10.66%	32.83%

Table 8: Comparison of changes in Task-Knowledge and Non-Task-Knowledge Neurons of Qwen2.5-7B-Instruct after fine-tuning with different loss functions.

B.6 Catastrophic Forgetting Induced by Using Neuron Alignment Loss

With Qwen2.5-7B-Instruct as our baseline model, we conducted a comparative analysis of only CE loss and CE loss augmented with Neuron Alignment Loss across three datasets, specifically focusing on their impact on internal neuronal representations during FPFT. As presented in Table 8, experimental results indicate that while the adjustment ratios for Task-Knowledge Neurons were comparable between the two methods, significant discrepancies emerged regarding Non-Task-Knowledge Neurons. Notably, on the BioInstruct dataset, this divergence reached 32.83%. These findings suggest that Neuron Alignment Loss, by aligning with the activation patterns extracted from the initial model on the instruction dataset, effectively mitigates the drift of non-critical neurons during the later stages of fine-tuning, thereby equipping the model with robust resistance against catastrophic forgetting.

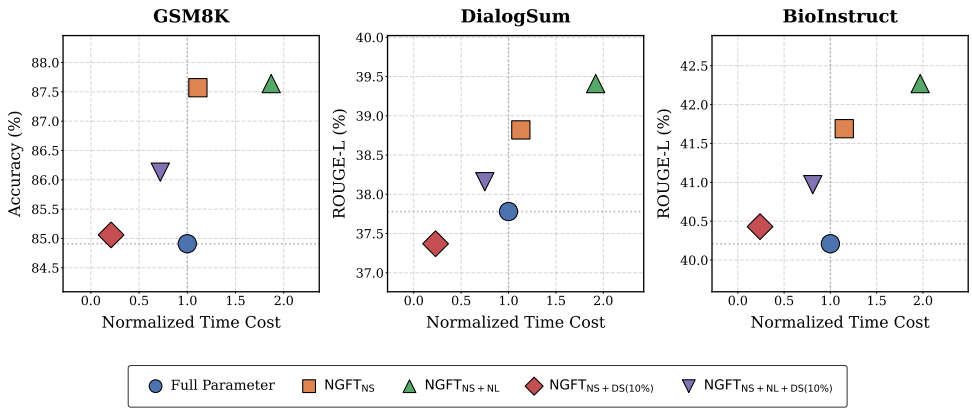

Figure 7: Analysis of In-Domain Performance and Time Overhead of the NGFT Framework on Qwen2.5-7B-Instruct Under Different Configurations. Specifically, NGFT$_{NS}$ fine-tunes only the Task-Knowledge Neurons; NGFT$_{NS+NL}$ incorporates a Neuron Alignment Loss on this basis; NGFT$_{NS+DS(10\%)}$ fine-tunes the aforementioned neurons using only 10% of the selected data; while NGFT$_{NS+NL+DS(10\%)}$ combines both the 10% data selection and the Neuron Alignment Loss.

B.7 Costs of Running of NGFT

In this section, we analyze the time efficiency of NGFT using Qwen2.5-7B-Instruct as a representative case study. To account for variations in absolute training duration across different tasks, we introduce *Normalized Time Cost* as a metric, defined as the ratio of a specific method's fine-tuning time to that of FPFT. This is calculated as:

$$T_{\text{norm}} = \frac{T_{\text{method}}}{T_{\text{full}}} \tag{12}$$

where T_{method} denotes the training duration of the target method and T_{full} denotes the duration of FPFT.

As shown in Figure 7 and Table 9, the experimental results demonstrate the dual advantages of NGFT under varying computational constraints:

First, in resource-constrained scenarios, NGFT exhibits superior computational efficiency. By synergizing Neuron Selection with Data Selection (utilizing only 10% of the data), NGFT not only matches or even surpasses the performance of full-data FPFT but also reduces the average training time to merely 20.67% of the latter, constituting a highly cost-effective fine-tuning baseline. Furthermore, while incorporating Neuron Alignment Loss introduces additional overhead due to autoregressive generation, when combined with efficient data selection strategies, the total time cost remains controlled at 76.14% of the full-data FPFT baseline, yet it consistently yields performance superior to that of FPFT. This indicates that high-quality data selection strategies effectively offset the temporal overhead incurred by the alignment loss, achieving an excellent balance between efficiency and performance.

Second, in performance-oriented scenarios, NGFT proves its capability to unleash the model's ultimate potential. Even with the full dataset, deploying the Neuron Selection module alone incurs a marginal average time overhead of 12.72% for neuron localization, yet it consistently outperforms FPFT, demonstrating its robustness as an efficient fine-tuning solution. To further explore the performance upper bound, combining Neuron Alignment Loss increases the full-data training time to 92.31% of the baseline; however, this configuration successfully achieves the maximum performance gain on in-domain tasks. This strategy of exchanging high computational investment for SOTA performance is well-suited for scenarios with ample computational budgets and strict requirements for model accuracy.

In summary, NGFT is not a rigid paradigm but a highly flexible, modular framework centered on neuron activation patterns. Each of its components (Neuron Selection, Data Selection, and Neuron Alignment Loss) has been independently verified to achieve advanced performance and features complete plug-and-play capability. This design empowers users to dynamically configure fine-tuning strategies according to specific computational resource budgets and performance metrics.

B.8 In-Depth Analysis of Catastrophic Forgetting in NGFT

This section details how FPFT induces performance degradation, specifically knowledge forget-

Method	GSM8K	DialogSum	BioInstruct
FPFT	84.91 / 1.00	37.78 / 1.00	40.21 / 1.00
NGFT_{NS}	87.57 / 1.11	38.82 / 1.13	41.69 / 1.15
$\text{NGFT}_{\text{NS+NL}}$	87.64 / 1.87	39.41 / 1.92	42.27 / 1.97
$\text{NGFT}_{\text{NS+DS}(10\%)}$	85.06 / 0.21	37.37 / 0.20	40.43 / 0.21
$\text{NGFT}_{\text{NS+NL+DS}(10\%)}$	86.13 / 0.72	38.16 / 0.75	40.97 / 0.81

Table 9: Comparison of in-domain performance and time cost (Score / Normalized Time Cost) for Qwen2.5-7B-Instruct under Different NGFT Configurations.

ting, style shifts, and position bias, and how the proposed NGFT framework resolves these issues.

(1) **Knowledge Forgetting and Negative Transfer in FPFT**: Evaluation on mathematical tasks (using GSM8K) reveals that FPFT causes negative transfer in weakly correlated out-of-domain areas. While MMLU elementary mathematics and abstract algebra remain stable, college mathematics and statistics suffer average performance drops of 7.4, 8.6, and 9.3 points across the three evaluated LLMs. By forcing global adaptation to GSM8K's elementary reasoning style, FPFT overwrites the distinct activation patterns required for complex computational logic and data modeling in higher-level mathematics.

(2) **Mitigation and Positive Transfer via NGFT**: NGFT effectively arrests this degradation through precise neuron regulation. Post-NGFT, the drop in college mathematics and statistics is constrained to a negligible 1.8 points. Furthermore, NGFT uniquely enables positive transfer by acquiring new skills while strictly preserving original knowledge structures: in elementary mathematics and abstract algebra, Qwen2.5-7B-Instruct (45.3 $\rightarrow$ 47.1) and Mistral-7B-v0.3 (34.7 $\rightarrow$ 35.9) demonstrate notable performance gains.

(3) **Disentangling Position Bias and Style Shifts**: Beyond forgetting, FPFT triggers a severe problem-solving style shift. After FPFT on the biomedical BioInstruct dataset, models experience a catastrophic 50.7 point drop on MMLU college biology. Our analysis confirms this is driven by an extreme position bias, with the probability of choosing option "B" surging from 23.9% to 71.5%. This proves that unconstrained global updates can distort a model's foundational reasoning style (Chen et al., 2026a). In contrast, NGFT actively resists this dataset-induced bias, limiting performance declines to within 2 points for two models (and a modest 5.3 points for Mistral-7B-Instruct), thereby maintaining the integrity of the model's problem-solving capabilities.

C Result Details

This section provides supplementary details on the experiments, specifically: (1) Table 10 reports the detailed anti-forgetting performance of the models from Table 1 across three benchmarks: MMLU, BBH, and TyDiQA; (2) Table 11 compares the performance of Qwen2.5 models of various sizes using NGFT against FPFT and strong baselines (Data Whisperer + NCFT + CE Loss); and (3) Tables 12 through 19 analyze the performance gains of eight models using individual and combined NGFT components, relative to zero-shot and full-data FPFT baselines.

Model	Method	Metric	GSM				DialogSum				BioInstruct			
			1%	5%	10%	Full	1%	5%	10%	Full	1%	5%	10%	Full
Qwen2.5-7B-Instruct	Zero-shot	Test	42.91				23.27				30.68			
		MMLU	–				74.14				–			
		BBH	–				65.74				–			
		TyDiQA	–				65.99				–			
	FPFT	Test	77.48	78.99	80.81	84.91	34.49	35.17	36.68	37.78	36.12	37.53	36.92	40.21
		MMLU	65.31	63.31	62.81	66.19	70.81	70.38	70.15	65.16	67.41	67.58	66.28	62.74
		BBH	54.68	54.87	55.40	56.53	66.85	64.16	62.22	52.22	61.83	62.85	61.72	51.66
		TyDiQA	51.74	50.58	50.73	51.56	60.22	58.45	60.62	56.26	57.31	67.13	58.90	59.91
	Strong Baseline	Test	80.81	83.01	84.22	85.75	33.70	36.24	36.98	38.24	36.45	38.34	38.96	41.21
		MMLU	74.87	73.28	75.26	73.68	71.78	71.97	71.36	68.43	73.89	73.16	72.91	65.37
		BBH	65.00	63.33	64.44	66.29	66.20	66.85	68.98	63.42	69.16	69.01	67.68	58.88
		TyDiQA	56.50	59.45	58.59	61.64	68.44	64.62	63.51	61.93	64.96	64.21	60.95	63.34
	NGFT	Test	82.18	84.61	86.13	87.64	34.83	37.15	38.16	39.41	38.79	39.60	40.97	42.27
		MMLU	74.05	74.23	74.16	74.46	74.24	74.12	73.62	70.33	74.44	73.81	73.90	67.72
		BBH	66.48	64.16	63.98	66.94	70.18	68.51	71.01	62.87	71.48	71.11	70.96	63.40
		TyDiQA	56.71	60.42	59.06	59.91	65.54	62.87	63.38	59.42	63.13	66.02	66.53	64.90
Llama3-8B-Instruct	Zero-shot	Test	36.22				16.66				27.53			
		MMLU	–				68.51				–			
		BBH	–				69.90				–			
		TyDiQA	–				62.37				–			
	FPFT	Test	62.77	67.37	69.67	77.71	32.77	34.77	35.64	38.10	30.70	33.79	34.82	40.57
		MMLU	62.92	61.56	62.11	63.23	60.87	57.23	56.05	54.13	58.18	57.89	57.97	50.32
		BBH	62.51	62.15	60.28	63.70	44.69	40.88	42.97	24.81	60.12	59.79	55.43	40.55
		TyDiQA	64.27	63.89	65.10	63.50	37.15	46.19	49.03	35.87	35.36	47.10	51.45	56.25
	Strong Baseline	Test	61.11	73.10	71.50	77.64	34.84	36.53	37.27	38.58	34.34	35.83	36.63	40.84
		MMLU	68.91	67.12	69.25	67.38	65.35	66.01	62.87	58.74	63.89	63.14	61.56	57.13
		BBH	69.07	68.51	68.33	69.90	58.05	62.31	60.18	45.83	65.42	63.33	63.61	46.66
		TyDiQA	66.46	67.76	68.10	67.48	45.22	46.33	52.36	40.32	57.17	59.07	54.91	61.35
	NGFT	Test	66.94	73.85	78.47	80.75	34.92	36.33	37.85	39.79	35.36	38.37	38.87	42.19
		MMLU	67.78	68.24	68.38	68.46	67.39	66.17	63.41	60.21	65.96	64.96	64.32	57.97
		BBH	67.40	68.51	70.64	70.83	69.26	63.50	63.20	53.11	70.46	69.90	66.50	49.99
		TyDiQA	68.52	69.09	68.97	67.30	49.78	52.20	64.49	49.98	53.96	57.07	62.50	62.43
Mistral-Instruct-V0.3	Zero-shot	Test	25.24				20.46				21.40			
		MMLU	–				61.67				–			
		BBH	–				65.74				–			
		TyDiQA	–				54.29				–			
	FPFT	Test	39.19	39.95	39.04	57.24	28.15	29.59	31.33	36.42	22.61	26.61	30.15	38.14
		MMLU	57.98	55.10	55.21	52.98	36.97	27.21	25.89	25.03	36.42	28.10	28.99	25.08
		BBH	45.05	46.12	42.03	39.25	10.92	11.05	10.87	10.18	8.69	28.74	22.31	9.35
		TyDiQA	50.76	50.88	52.39	49.37	30.71	16.10	9.85	6.85	30.44	29.41	27.68	4.10
	Strong Baseline	Test	37.14	43.22	45.94	57.85	33.19	34.87	35.43	36.87	23.60	28.29	32.12	38.84
		MMLU	62.81	60.92	61.95	60.33	42.31	37.72	47.21	34.98	49.73	46.99	35.58	33.99
		BBH	56.57	56.75	57.96	50.74	20.46	25.37	26.20	10.70	15.83	20.46	18.33	28.42
		TyDiQA	56.53	55.99	55.73	56.98	26.05	28.58	26.94	19.31	51.40	37.72	29.57	18.33
	NGFT	Test	42.46	47.61	52.01	62.55	33.96	34.95	36.51	38.06	28.50	32.42	33.99	39.60
		MMLU	61.94	61.64	62.00	61.19	41.45	36.89	48.08	34.06	48.86	47.84	34.71	34.86
		BBH	57.22	56.29	57.94	54.90	21.51	24.16	28.70	18.05	23.98	43.33	32.50	29.62
		TyDiQA	55.75	56.94	57.13	58.13	38.96	35.47	41.55	31.02	45.32	50.08	32.13	27.84

Table 10: Performance Comparison of NGFT with Zero-Shot, FPFT, and Strong Baselines (Data Whisperer + NCFT + CE Loss represent the SOTA baselines for data selection, neuron selection, and loss function optimization.) when training with top 1%, 5%, 10%, and full GSM8K, DialogSum, and BioInstruct data.

Model	Method	Metric	GSM				DialogSum				BioInstruct			
			1%	5%	10%	Full	1%	5%	10%	Full	1%	5%	10%	Full
Qwen2.5-0.5B-Instruct	Zero-shot	Test	24.03				15.95				17.22			
		CF	–				39.88				–			
	FPFT	Test	31.18	32.67	35.92	39.30	28.95	30.72	31.58	34.88	28.39	29.68	31.71	36.40
		CF	32.07	31.39	31.53	30.17	28.51	27.08	25.94	18.01	32.40	26.99	24.57	11.82
	Strong Baseline	Test	40.96	42.79	44.55	47.56	32.46	33.70	34.00	36.40	32.24	34.37	34.78	39.10
		CF	39.19	38.37	37.80	37.89	37.15	37.45	36.66	35.29	38.33	38.52	36.64	36.29
	NGFT	Test	**41.78**	**44.00**	**45.20**	**48.98**	**32.93**	**34.83**	**34.95**	**37.16**	**33.62**	**34.95**	**35.79**	**39.49**
		CF	40.54	40.25	39.92	39.55	39.50	38.53	38.40	37.71	39.85	39.48	38.91	37.70
Qwen2.5-3B-Instruct	Zero-shot	Test	37.44				21.70				34.50			
		CF	–				59.36				–			
	FPFT	Test	63.15	65.28	67.48	67.24	33.89	36.43	36.80	37.19	36.41	37.17	37.79	41.14
		CF	55.47	55.95	54.99	56.39	58.57	53.94	53.93	51.58	49.83	54.85	54.50	52.62
	Strong Baseline	Test	65.62	66.91	**70.70**	71.23	34.57	36.75	**37.34**	37.97	36.22	37.30	38.23	40.38
		CF	58.66	59.45	59.54	59.21	63.14	60.82	57.84	54.86	54.13	59.50	58.80	58.38
	NGFT	Test	**66.00**	**68.18**	70.58	**73.79**	**34.83**	**36.82**	37.22	**38.15**	**38.16**	**39.40**	**39.68**	**41.93**
		CF	59.51	60.35	60.64	60.41	63.09	61.57	61.73	58.64	55.81	59.00	59.95	59.08
Qwen2.5-14B-Instruct	Zero-shot	Test	54.81				22.72				38.93			
		CF	–				72.17				–			
	FPFT	Test	83.75	85.17	86.83	86.43	33.64	35.46	36.81	38.82	38.49	38.77	39.64	41.52
		CF	62.12	61.57	61.35	64.43	74.05	66.68	69.84	67.16	62.17	63.80	66.54	63.29
	Strong Baseline	Test	86.68	**87.27**	88.63	87.19	34.19	37.08	38.03	39.27	38.15	39.68	39.72	42.13
		CF	72.65	72.41	71.23	71.37	75.13	72.85	72.62	70.15	71.28	69.21	69.57	72.53
	NGFT	Test	**88.86**	**87.27**	**89.40**	**88.30**	**34.59**	**37.53**	**38.11**	**40.58**	**41.14**	**41.36**	**42.24**	**43.17**
		CF	72.08	72.66	72.07	73.42	74.90	74.33	73.58	72.49	71.98	71.00	72.44	73.68

Table 11: Performance (Test / CF) Comparison on Qwen2.5-0.5B-Instruct, Qwen2.5-3B-Instruct, and Qwen2.5-14B-Instruct. The Test Metric measures in-domain performance, while the CF Metric assesses average performance on MMLU, BBH, and TyDiQA to reflect the model's ability to mitigate forgetting. **Bold** indicates the best Test performance, and underline indicates the best CF performance.

Model	Method	Metric	GSM				DialogSum				BioInstruct			
			1%	5%	10%	Full	1%	5%	10%	Full	1%	5%	10%	Full
Qwen2.5-0.5B-Instruct	Zero-shot	Test	24.03				15.95				17.22			
		MMLU	–				46.87				–			
		BBH	–				32.31				–			
		TyDiQA	–				40.48				–			
	NGFT_{DS} + FPFT	Test	32.62	34.62	37.56	39.30	30.51	32.25	32.89	34.88	30.22	31.75	33.08	36.40
		MMLU	40.83	40.12	40.21	40.22	40.41	38.83	36.84	33.02	38.98	37.42	35.49	12.87
		BBH	24.29	24.03	24.22	21.16	26.25	25.14	24.59	15.09	25.14	24.87	19.96	12.87
		TyDiQA	34.83	32.88	34.35	29.13	21.42	20.56	20.28	5.93	31.27	22.28	20.65	9.72
	NGFT_{DS} + NGFT_{NS}	Test	38.03	40.37	41.67	46.25	30.76	32.88	33.44	35.23	32.41	33.70	34.26	37.90
		MMLU	49.96	46.80	46.81	46.56	47.03	47.05	46.66	44.91	46.68	46.90	46.47	44.25
		BBH	30.92	30.18	30.37	31.48	31.38	31.01	31.94	28.15	31.75	30.37	30.27	28.42
		TyDiQA	40.48	40.21	39.71	39.19	36.56	36.52	38.33	31.00	40.66	38.36	37.22	39.00
	NGFT_{DS} + NGFT_{NL}	Test	40.11	42.46	44.88	48.45	32.08	34.23	34.44	36.79	32.54	34.14	34.93	39.09
		MMLU	47.15	46.82	46.55	46.21	46.33	46.01	45.72	45.38	46.52	46.18	45.85	45.51
		BBH	31.58	31.25	30.92	30.59	33.24	31.66	30.38	29.05	29.69	28.61	28.88	19.16
		TyDiQA	40.85	40.52	40.18	39.85	39.12	38.79	38.45	38.12	40.42	37.12	32.66	29.24
	NGFT_{DS} + NGFT_{NS} + NGFT_{NL}	Test	41.78	44.00	45.20	48.98	32.93	34.83	34.95	37.16	33.62	34.95	35.79	39.49
		MMLU	48.25	47.91	47.53	47.12	46.88	46.55	46.12	45.87	47.05	46.72	46.31	45.98
		BBH	32.15	31.88	31.56	31.22	33.78	31.53	31.90	30.41	31.48	30.92	31.66	29.62
		TyDiQA	41.22	40.95	40.68	40.31	37.85	37.52	37.18	36.85	41.03	40.80	38.76	37.51

Table 12: Comparison of performance improvements on the Qwen2.5-0.5B-Instruct model using different NGFT components individually and in combination.

Model	Method	Metric	GSM				DialogSum				BioInstruct			
			1%	5%	10%	Full	1%	5%	10%	Full	1%	5%	10%	Full
Qwen2.5-3B-Instruct	Zero-shot	Test	37.44				21.70				27.50			
		MMLU	–				66.36				–			
		BBH	–				56.29				–			
		TyDiQA	–				55.44				–			
	NGFT_{DS} + FPFT	Test	62.03	64.30	65.24	67.24	34.45	36.13	36.60	37.19	37.51	38.68	39.13	41.14
		MMLU	61.35	61.99	61.81	58.06	61.71	61.61	61.13	58.33	61.32	59.90	60.36	57.43
		BBH	49.46	47.98	47.14	50.85	52.03	50.83	49.25	46.66	47.68	51.94	52.12	47.57
		TyDiQA	58.16	60.58	59.33	60.26	60.18	52.99	54.56	49.76	43.34	51.05	54.46	52.87
	NGFT_{DS} + NGFT_{NS}	Test	64.17	66.44	68.44	70.45	34.19	36.29	36.71	37.77	37.88	39.20	39.56	41.65
		MMLU	66.92	66.89	66.91	64.67	66.77	66.75	66.76	64.49	66.75	65.88	65.90	64.32
		BBH	54.62	53.88	54.62	55.46	56.75	56.20	55.09	54.90	54.35	55.64	56.57	53.70
		TyDiQA	57.00	59.16	59.67	59.74	64.59	60.97	61.70	60.82	47.97	55.79	61.32	62.63
	NGFT_{DS} + NGFT_{NL}	Test	65.37	68.85	71.25	72.45	34.92	36.22	36.78	37.44	37.94	39.06	39.49	41.67
		MMLU	66.65	66.61	66.49	63.82	66.48	66.47	66.26	63.67	66.31	65.62	65.71	62.54
		BBH	54.35	55.09	55.92	55.55	57.49	57.03	57.24	53.79	53.79	54.90	55.83	52.45
		TyDiQA	56.49	59.66	59.71	60.70	64.97	59.65	61.42	54.42	49.98	57.77	58.71	60.30
	NGFT_{DS} + NGFT_{NS} + NGFT_{NL}	Test	66.00	68.18	70.58	73.79	34.83	36.82	37.22	38.15	38.16	39.40	39.68	41.93
		MMLU	66.83	66.82	66.72	64.39	66.74	66.69	66.37	63.63	66.31	64.34	63.17	62.42
		BBH	54.81	54.44	55.83	56.29	58.18	58.24	57.87	55.19	51.38	55.37	57.59	51.38
		TyDiQA	56.89	59.78	59.37	60.56	64.34	59.79	60.94	57.10	49.74	57.29	59.09	63.44

Table 13: Comparison of performance improvements on the Qwen2.5-3B-Instruct model using different NGFT components individually and in combination.

Model	Method	Metric	GSM				DialogSum				BioInstruct			
			1%	5%	10%	Full	1%	5%	10%	Full	1%	5%	10%	Full
Qwen2.5-7B-Instruct	Zero-shot	Test	42.91				23.27				30.68			
		MMLU	–				74.14				–			
		BBH	–				65.74				–			
		TyDiQA	–				65.99				–			
	NGFT_{DS} + FPFT	Test	80.41	81.68	83.26	84.91	33.97	36.05	37.14	37.78	38.35	38.60	39.64	40.21
		MMLU	66.04	63.02	63.23	66.19	71.05	70.61	70.33	65.16	68.52	68.24	67.08	62.74
		BBH	55.61	54.31	55.24	56.53	66.38	64.90	67.87	52.22	61.66	61.48	62.03	51.66
		TyDiQA	49.14	51.07	50.47	51.56	64.19	57.10	60.64	56.26	59.98	61.76	60.85	59.91
	NGFT_{DS} + NGFT_{NS}	Test	81.81	83.95	85.06	87.57	34.75	36.62	37.37	38.82	38.76	39.30	40.43	41.69
		MMLU	74.11	74.14	74.15	74.25	74.42	74.55	74.39	71.74	74.33	73.75	73.37	69.93
		BBH	65.46	63.70	63.98	65.92	67.31	67.96	68.07	65.09	68.98	68.05	67.96	65.55
		TyDiQA	56.89	59.04	58.59	61.67	64.18	58.71	61.90	68.03	63.41	67.25	66.41	65.05
	NGFT_{DS} + NGFT_{NL}	Test	81.95	84.41	85.16	86.89	34.77	36.90	37.29	38.37	38.79	39.44	40.25	41.93
		MMLU	74.12	74.29	74.17	74.51	74.24	73.74	73.97	70.63	74.29	73.94	74.08	67.26
		BBH	65.92	64.07	64.53	67.59	70.18	69.53	69.62	60.37	71.94	72.50	70.40	63.79
		TyDiQA	56.60	60.34	59.39	59.96	65.38	62.86	62.16	57.75	62.81	67.02	66.64	64.44
	NGFT_{DS} + NGFT_{NS} + NGFT_{NL}	Test	82.18	84.61	86.13	87.64	34.83	37.15	38.16	39.41	38.79	39.60	40.97	42.27
		MMLU	74.05	74.23	74.16	74.46	74.24	74.12	73.62	70.33	74.44	73.81	73.90	67.72
		BBH	66.48	64.16	63.98	66.94	70.18	68.51	71.01	62.87	71.48	71.11	70.96	63.40
		TyDiQA	56.71	60.42	59.06	59.91	65.54	62.87	63.38	59.42	63.13	66.02	66.53	64.90

Table 14: Comparison of performance improvements on the Qwen2.5-7B-Instruct model using different NGFT components individually and in combination.

Model	Method	Metric	GSM				DialogSum				BioInstruct			
			1%	5%	10%	Full	1%	5%	10%	Full	1%	5%	10%	Full
Qwen2.5-14B-Instruct	Zero-shot	Test	54.81				34.07				38.93			
		MMLU	–				79.80				–			
		BBH	–				69.25				–			
		TyDiQA	–				67.48				–			
	NGFT_{DS} + FPFT	Test	85.75	85.82	87.42	86.43	34.27	36.53	38.00	38.82	38.45	38.62	38.86	41.52
		MMLU	72.10	69.05	69.30	72.25	79.90	79.45	79.10	73.81	77.11	76.47	74.88	69.12
		BBH	62.50	61.20	62.15	63.40	77.03	72.96	75.37	65.64	74.90	73.42	73.14	56.20
		TyDiQA	55.20	57.15	56.50	57.65	66.13	51.83	58.19	62.03	38.24	44.81	54.45	64.56
	NGFT_{DS} + NGFT_{NS}	Test	87.11	87.34	88.18	88.10	34.30	36.53	37.89	39.64	38.69	39.38	40.44	42.10
		MMLU	80.15	80.18	80.20	80.30	81.50	81.65	81.45	78.80	77.83	78.11	77.52	73.46
		BBH	72.35	70.55	70.85	72.85	78.24	79.25	77.68	77.87	78.51	76.57	76.48	71.29
		TyDiQA	63.00	65.20	64.75	67.80	67.31	62.35	64.22	61.42	53.98	52.26	57.38	67.65
	NGFT_{DS} + NGFT_{NL}	Test	87.95	86.12	89.23	87.72	34.49	36.76	37.86	39.95	40.74	40.92	41.44	42.54
		MMLU	80.15	80.18	80.20	80.30	81.50	81.65	81.45	78.80	77.83	78.11	77.52	73.46
		BBH	72.80	71.00	71.45	74.50	78.24	79.35	76.75	77.73	78.70	78.42	77.59	74.44
		TyDiQA	62.75	66.50	65.55	66.10	67.96	62.05	64.14	61.75	55.71	53.66	53.13	67.42
	NGFT_{DS} + NGFT_{NS} + NGFT_{NL}	Test	88.86	87.27	89.40	88.30	34.59	37.53	38.11	40.58	41.14	41.36	42.24	43.17
		MMLU	80.10	80.28	80.20	80.40	79.84	79.70	79.20	75.90	81.30	81.15	80.75	77.50
		BBH	73.30	71.10	70.90	73.80	77.50	78.70	77.40	78.34	78.70	78.33	79.45	75.13
		TyDiQA	62.85	66.60	65.10	66.05	67.35	64.60	64.13	63.24	55.94	53.52	57.12	68.42

Table 15: Comparison of performance improvements on the Qwen2.5-14B-Instruct model using different NGFT components individually and in combination.

Model	Method	Metric	GSM				DialogSum				BioInstruct			
			1%	5%	10%	Full	1%	5%	10%	Full	1%	5%	10%	Full
Mistral-7B-Instruct-v0.3	Zero-shot	Test	25.24				20.46				21.40			
		MMLU	–				61.67				–			
		BBH	–				65.74				–			
		TyDiQA	–				54.29				–			
	NGFT_{DS} + FPFT	Test	40.96	43.98	45.85	57.24	30.45	33.38	34.09	36.42	25.42	30.32	32.62	38.14
		MMLU	57.09	56.24	54.47	52.98	35.82	26.45	26.32	25.03	35.84	27.56	29.51	25.08
		BBH	46.11	45.37	41.29	39.25	10.09	10.18	10.09	10.18	7.87	27.87	23.14	9.35
		TyDiQA	49.81	51.50	51.54	49.37	29.65	15.38	9.01	6.85	31.18	30.24	26.86	4.10
	NGFT_{DS} + NGFT_{NS}	Test	41.84	44.97	48.76	58.00	33.84	34.56	35.86	37.30	28.19	32.26	33.62	39.18
		MMLU	61.89	61.77	62.14	59.58	42.35	35.98	47.61	32.89	47.87	49.56	46.12	32.80
		BBH	57.77	56.01	57.59	51.94	15.45	17.96	23.98	17.70	25.55	44.44	39.79	28.42
		TyDiQA	55.83	56.56	57.34	56.53	39.30	32.17	39.04	29.31	39.94	49.45	46.67	27.28
	NGFT_{DS} + NGFT_{NL}	Test	42.64	45.04	48.93	58.00	34.01	34.14	36.18	37.85	28.00	32.74	33.82	39.18
		MMLU	62.02	61.72	61.97	59.98	36.65	32.12	47.17	33.08	46.75	48.28	46.65	32.80
		BBH	56.85	55.92	57.96	51.94	21.75	26.48	27.31	15.85	24.62	40.37	40.00	28.42
		TyDiQA	54.90	56.85	57.19	56.53	29.48	37.52	39.99	28.92	39.95	47.56	47.16	27.28
	NGFT_{DS} + NGFT_{NS} + NGFT_{NL}	Test	42.46	47.61	52.01	60.29	33.96	34.95	36.51	38.06	28.50	32.42	33.99	39.60
		MMLU	61.94	61.64	62.00	61.19	41.45	36.89	48.08	34.06	48.86	47.84	34.71	34.86
		BBH	57.22	56.29	57.94	54.90	21.51	24.16	28.70	18.05	23.98	43.33	32.50	29.62
		TyDiQA	55.75	56.94	57.13	58.13	38.96	35.47	41.55	31.02	45.32	50.08	32.13	27.84

Table 16: Comparison of performance improvements on the Mistral-7B-Instruct model using different NGFT components individually and in combination.

Model	Method	Metric	GSM				DialogSum				BioInstruct			
			1%	5%	10%	Full	1%	5%	10%	Full	1%	5%	10%	Full
Llama3.2-1B-Instruct	Zero-shot	Test	12.51				14.02				22.59			
		MMLU	–				46.00				–			
		BBH	–				38.42				–			
		TyDiQA	–				43.69				–			
	NGFT_{DS} + FPFT	Test	32.15	33.20	34.41	38.66	31.70	33.32	34.52	36.92	31.96	34.63	35.52	38.50
		MMLU	43.24	43.17	42.87	38.86	41.97	41.41	38.33	32.54	42.08	41.80	42.69	32.75
		BBH	35.87	34.38	33.09	28.46	29.44	24.62	25.00	22.75	35.33	32.64	32.75	27.07
		TyDiQA	42.97	43.59	42.79	34.68	35.89	35.23	31.52	29.08	40.06	43.95	41.46	31.23
	NGFT_{DS} + NGFT_{NS}	Test	35.70	36.76	36.39	40.87	32.14	33.55	34.83	36.69	32.73	34.93	35.69	39.34
		MMLU	45.61	46.46	45.82	44.99	46.04	45.58	45.09	42.10	47.28	46.35	47.08	42.03
		BBH	36.88	36.20	36.29	34.53	35.09	33.42	32.77	36.85	38.88	38.70	38.24	34.25
		TyDiQA	44.89	45.70	45.71	42.61	40.34	35.98	36.62	37.51	43.29	47.05	47.51	45.23
	NGFT_{DS} + NGFT_{NL}	Test	35.55	36.84	36.46	40.94	31.68	33.41	34.80	36.77	32.63	34.98	35.74	39.15
		MMLU	45.75	46.50	45.90	45.10	45.64	45.30	45.48	45.48	47.37	46.45	46.99	42.50
		BBH	37.05	36.40	36.55	34.80	37.03	36.20	37.40	38.05	39.90	39.62	41.20	35.10
		TyDiQA	45.10	45.85	45.90	43.15	38.78	36.58	37.96	37.96	43.21	47.34	47.40	45.50
	NGFT_{DS} + NGFT_{NS} + NGFT_{NL}	Test	36.10	37.22	37.15	41.47	31.72	33.80	34.81	36.77	32.79	34.95	35.62	39.49
		MMLU	45.80	46.65	46.10	45.25	45.71	45.25	45.49	41.98	47.27	46.30	47.02	41.81
		BBH	37.15	36.60	36.80	35.10	36.29	36.48	37.77	37.50	39.16	38.70	40.09	37.19
		TyDiQA	45.35	46.10	46.05	43.50	38.93	36.31	37.87	38.51	43.22	47.26	47.16	45.77

Table 17: Comparison of performance improvements on the Llama3.2-1B-Instruct model using different NGFT components individually and in combination.

Model	Method	Metric	GSM				DialogSum				BioInstruct			
			1%	5%	10%	Full	1%	5%	10%	Full	1%	5%	10%	Full
Llama3.2-3B-Instruct	Zero-shot	Test	29.45				15.12				26.88			
		MMLU	–				60.45				–			
		BBH	–				58.61				–			
		TyDiQA	–				62.50				–			
	NGFT_{DS} + FPFT	Test	62.25	67.40	68.46	70.97	33.31	36.13	37.09	38.21	35.47	37.56	38.53	41.49
		MMLU	59.45	59.80	60.10	58.90	60.39	59.28	58.34	54.62	60.09	58.66	58.63	53.43
		BBH	57.31	55.74	55.18	53.51	55.09	51.85	51.11	46.29	59.81	56.75	53.24	51.29
		TyDiQA	61.54	60.09	60.57	58.84	46.76	50.50	51.25	50.79	57.88	59.80	58.02	51.11
	NGFT_{DS} + NGFT_{NS}	Test	64.59	72.48	73.77	72.56	33.72	36.39	37.29	38.97	35.87	38.02	39.02	41.98
		MMLU	60.12	60.45	60.55	60.20	60.93	60.70	60.58	59.10	60.63	60.41	60.10	57.75
		BBH	58.33	57.13	56.66	54.35	56.66	55.55	56.20	52.68	60.64	59.72	59.44	45.18
		TyDiQA	61.50	61.29	61.50	60.35	58.42	56.62	57.07	57.38	58.82	60.99	59.82	62.39
	NGFT_{DS} + NGFT_{NL}	Test	65.13	72.79	73.47	73.09	33.68	36.34	37.10	38.30	35.54	37.96	40.29	43.17
		MMLU	60.20	60.50	60.48	60.15	60.85	60.60	60.45	58.90	60.55	60.30	60.05	57.50
		BBH	58.10	57.00	56.80	54.50	59.81	58.14	59.35	55.55	62.03	61.94	61.11	53.61
		TyDiQA	61.45	61.35	61.40	60.20	58.00	55.14	57.82	59.93	58.31	61.33	59.33	62.86
	NGFT_{DS} + NGFT_{NS} + NGFT_{NL}	Test	65.20	73.16	74.07	74.15	33.77	36.27	36.81	38.60	35.94	38.01	40.73	43.58
		MMLU	60.25	60.55	60.60	60.35	60.90	60.65	60.50	59.20	60.70	60.45	60.15	57.85
		BBH	58.50	57.40	57.10	55.00	59.35	56.75	59.62	55.00	62.86	62.40	62.03	54.81
		TyDiQA	61.60	61.50	61.65	60.80	57.87	55.25	57.76	59.81	58.57	60.94	59.68	62.93

Table 18: Comparison of performance improvements on the Llama3.2-3B-Instruct model using different NGFT components individually and in combination.

Model	Method	Metric	GSM				DialogSum				BioInstruct			
			1%	5%	10%	Full	1%	5%	10%	Full	1%	5%	10%	Full
Llama3.1-8B-Instruct	Zero-shot	Test	36.22				16.66				27.53			
		MMLU	–				68.51				–			
		BBH	–				69.90				–			
		TyDiQA	–				67.37				–			
	NGFT_{DS} + FPFT	Test	63.50	72.32	73.26	77.71	34.74	35.62	36.64	38.10	34.02	36.96	37.76	40.57
		MMLU	62.93	63.20	63.35	63.23	61.38	57.80	55.46	54.13	58.72	57.92	57.52	50.32
		BBH	61.38	62.96	61.11	63.70	43.24	39.72	44.81	24.81	61.38	59.90	55.27	40.55
		TyDiQA	63.49	64.64	63.98	63.50	38.33	45.20	50.12	35.87	34.08	47.96	50.21	56.25
	NGFT_{DS} + NGFT_{NS}	Test	64.17	73.26	78.21	79.91	34.68	36.08	37.52	38.84	35.32	38.26	39.15	41.46
		MMLU	67.86	68.16	68.39	68.40	67.44	65.83	64.60	62.49	66.14	64.59	64.60	59.84
		BBH	66.38	68.70	70.74	69.90	67.40	63.75	63.42	56.29	67.77	68.33	66.01	51.85
		TyDiQA	68.73	68.86	68.74	67.52	48.36	51.83	51.46	48.16	46.58	52.44	62.60	64.41
	NGFT_{DS} + NGFT_{NL}	Test	64.03	74.19	77.21	79.40	35.21	35.93	37.81	39.33	35.14	38.28	38.82	41.86
		MMLU	68.04	68.30	68.39	68.53	67.44	66.40	63.60	58.56	65.93	65.04	62.51	53.30
		BBH	68.14	69.16	69.72	70.64	68.98	63.42	63.18	46.48	70.74	69.90	64.35	46.57
		TyDiQA	68.44	68.98	68.80	67.11	49.13	42.85	61.40	45.45	43.42	55.72	62.04	61.16
	NGFT_{DS} + NGFT_{NS} + NGFT_{NL}	Test	66.94	73.85	78.47	80.75	34.92	36.33	37.85	39.79	35.36	38.37	38.87	42.19
		MMLU	67.78	68.24	68.38	68.46	67.39	66.17	63.41	60.21	65.96	64.96	64.32	57.97
		BBH	67.40	68.51	70.64	70.83	69.26	63.50	63.20	53.11	70.46	69.90	66.50	49.99
		TyDiQA	68.52	69.09	68.97	67.30	49.78	52.20	64.49	49.98	53.96	57.07	62.50	62.43

Table 19: Comparison of performance improvements on the Llama3.1-8B-Instruct model using different NGFT components individually and in combination.