Finetune Once: Decoupling General & Domain Learning with Dynamic Anchor Annealing

Anonymous ACL submission

Abstract

Fine-tuning large language models on domain data is a standard practice, yet vanilla pipelines rely on a costly data mixture sweep to avoid catastrophic forgetting of general capabilities. We propose Dynamic Anchor Annealing (DAA), a full-parameter fine-tuning framework that decouples general and domain learning at the gradient level rather than at the data level. DAA estimates a global gradient anchor once on general data through zero learning rate training, anchors the domain gradient toward this anchor during fine-tuning, modulates step sizes by the cosine similarity between the two gradients, and combines this with annealing learning. Under this design, fine-tuning relies solely on domain data and does not lose general capability. Across four domains and three base models, DAA improves the joint domain and general harmonic mean by 5.8 points on average over vanilla data mixture fine-tuning. Our headline 91% reduction in GPU hours is a workflow level saving from removing the mixture ratio sweep rather than a per run speedup, since the per run cost of DAA is comparable to direct fine-tuning. The code is available at https://anonymous.4open.science/r/Finetune-once-5C13.

1 Introduction

Large Language Models (LLMs) show significant promise in various applications due to their ability to understand and generate human-like text. Fine-Tuning (FT) LLMs on domain-specific tasks has become a common approach to enhance their performance in targeted applications (Yang et al., 2023; Zhou et al., 2024; Chen et al., 2024; Huang et al., 2023). However, empirical evidence suggests that fine-tuned LLMs frequently demonstrate significant degradation of their original performance (Chen et al., 2020; Luo et al., 2025; Lin et al., 2023; Korbak et al., 2022). Therefore, mitigating catastrophic forgetting during fine tuning has become a crucial focus for LLMs. We term this phenomenon collapse, defined as a sharp drop in general task accuracy after domain fine tuning (Tab. 1, row 1).

Data Mixture (DM) strategy was the basic and vanilla solution (Wen et al., 2023; Wu et al., 2023; Zhang et al., 2024a; Wu et al., 2024; Held et al., 2025) to solve catastrophic problem. It combines general and domain-specific data in fine-tuning datasets to mitigate forgetting of general capabilities. Due to the coupling between data from different domains, each fine-tuning requires repeated experimentation to adjust the data mixture in order to achieve satisfactory performance (Tab. 1, row 2). As shown in Fig. 1, the effectiveness of DM heavily depends on the mixing ratio, necessitating extensive empirical validation to determine optimal proportions for each domain. Alternative approaches, such as Low-Rank Adaptation (LoRA) (Hu et al., 2021; Yang et al., 2023; Cui et al., 2023), have demonstrated some success in preserving general capabilities, yet they face inherent limitations in achieving peak domain-specific performance (Tab. 1, row 3). This ad-hoc process of data mixing is not only computationally prohibitive but also lacks scalability, as the optimal ratio for one domain rarely transfers to another. Consequently, an ideal fine-tuning framework must decouple domain adaptation from the costly cycle of data mixture experiments, while still effectively balancing specialization with the preservation of general knowledge.

To address the above challenges, we propose **Dynamic Anchor Annealing (DAA)**, a streamline fine-tuning framework that eliminates the requirements for data mixture and repeated experiments. First, to effectively isolating the contributions of general-domain and domain-specific data, we propose **Global Gradient Anchoring (GGA)**. Here, Anchoring refers to augmenting the domain-specific gradient with a pre-computed global one.

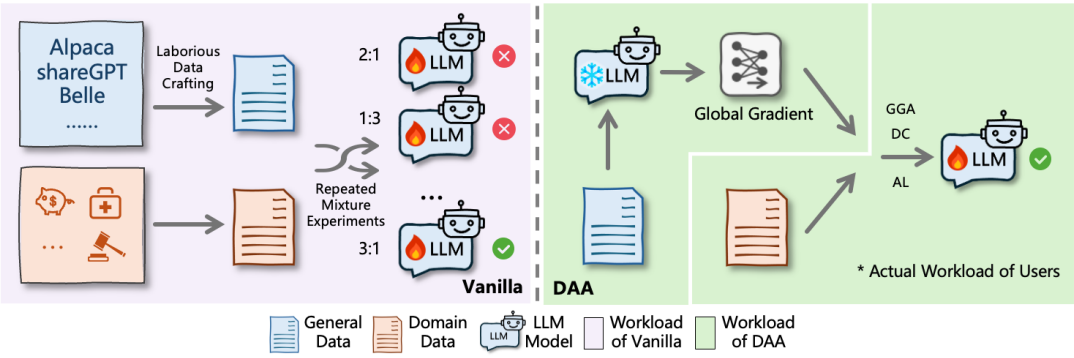


Figure 1: Comparison between vanilla and DAA. [*] is the part that users need to perform in SFT.

Method	No Data Mixture	No Repeated Exps	Reduce Cost	No Collapse	SOTA
Direct FT	✔	✔	✔	✗	✗
Vanilla FT	✗	✗	✗	✔	✗
LoRA-like FT	✔	✗	✔	✔	✗
DAA (Ours)	✔	✔	✔	✔	✔

Table 1: Comparison of methods. Repeated Exps indicates the method requires hyper-parameter tuning or recipe adjustment for data mixture ratios to achieve optimal results. Collapse means losing generalization.

This method initially estimates the global gradient in the general domain through zero-learning-rate learning. During fine-tuning, the global gradient is used as guidance, combined with annealing learning, to mitigate catastrophic forgetting. Second, to achieve global optimal performance in specifics, we propose a domain similarity-guided **Dynamic Correction (DC)** strategy. This adaptive parameter update strategy modulates the optimization steps based on the gradient similarity between specific and general domains. It is important to emphasize that the fundamental goal of DAA extends beyond the reduction of fine-tuning expenses to specifically address the mitigation of catastrophic forgetting. The efficiency of this method emerges as a significant inherent benefit of its design, as it delivers comparable or superior performance while avoiding the substantial costs associated with data mixture search. As demonstrated in Tab. 1, DAA achieves superior performance compared to conventional fine-tuning approaches, while significantly reducing workload by eliminating the need for data mixing and repeated experiments. Our contributions are summarized as follows:

- We explore the impact of data mixture on both fine-tuning performance and workload, and propose new fine-tuning schemes.
- We propose DAA, a novel training framework designed to efficiently fine-tuning by gradient-based domain decoupling and similarity-guided adaptation.
- We conduct empirical evaluations across various tasks, demonstrating that our method effectively balances domain-specific performance while maintaining general capabilities with low workload.

2 Motivation

2.1 Related Work

Recent fine-tuning approaches (Xie et al., 2023; Zhang et al., 2023b; Bao et al., 2023; Yue et al., 2023; Chen et al., 2024; Zhou et al., 2024; Yang et al., 2023; Cui et al., 2023) aim to balance domain adaptation with the mitigation of catastrophic forgetting. Direct fine-tuning (Xie et al., 2023; Zhu et al., 2024) is cost-effective but prone to collapsing general capabilities. LoRA (Yang et al., 2023; Chen et al., 2023; Cui et al., 2023) reduces computational overhead but often underperforms full-parameter tuning in specific domains. Vanilla fine-tuning with data mixtures (Bao et al., 2023; Yue et al., 2023; Chen et al., 2024; Deng et al., 2023) offers a balanced performance trade-off; however, it incurs substantial experimental costs. Determining optimal mixing ratios is domain-dependent (Wen et al., 2023), and the data volume scales quadratically. Specifically, sweeping ratios from $1:1$ through $1:N$ results in a complexity of $\sum_{n=1}^{N}(1+n) = O(N^2)$ relative to the domain dataset size. This inefficiency necessitates more scalable methodologies.

To optimize data mixtures, recent studies employ Reinforcement Learning (RL) (Wu et al., 2024) or sample-level curriculum learning (Liang et al., 2025). While effective, these methods introduce complex online search dynamics or sample-level

granularity. In contrast, DAA adopts a domain-level Global Gradient Anchor, abstracting the general prior into a stable optimization direction. This approach prioritizes computational efficiency and facilitates a streamlined *Finetune Once* workflow. Furthermore, unlike gradient surgery methods such as PCGrad (Yu et al., 2020) and GradNorm (Chen et al., 2018), which incur prohibitive memory costs by storing individual task gradients, DAA utilizes a memory-efficient, pre-computed global anchor to mitigate forgetting.

Theoretically, our setting aligns with single-task Continual Learning (CL), where a model adapts to an instruction task while preserving pre-trained capabilities. This differs fundamentally from Multi-Task Learning (MTL). While MTL typically addresses imbalances between competing specific tasks or cross-domain transfer (Sener and Koltun, 2018; Liu et al., 2019), DAA addresses the tension between the massive general prior and the smaller target dataset. By explicitly regulating this imbalance via the Global Gradient Anchor and decay coefficient γ_t, DAA ensures general-to-specific retention, thereby avoiding the negative transfer often observed in disjoint domain optimization.

2.2 Role of Gradient in fine-tuning

During stochastic optimization, gradient variance strongly affects convergence and generalization (Gurbuzbalaban et al., 2021). High variance slows convergence and complicates optimization (Agarwal et al., 2022; Xia et al., 2024), which can hinder domain adaptation. We provide qualitative and quantitative analyses of fine-tuning across general and specific domains and expose drawbacks of multi domain optimization.

First, to qualitatively analyze the divergence in convergence trajectories between general and specific domains, we visualize the loss landscape by constructing a two-dimensional slice via orthogonalized interpolation between the Qwen3-1.7B (Yang et al., 2025) base weights θ_0 and the fully fine-tuned weights θ_D exhibits a tortuous path characterized by numerous local optima, whereas the specific news QA domain demonstrates a more stable trajectory with fewer optima, suggesting that general domain training may impose constraints on domain-specific fine-tuning.

Second, employing stochastic optimization methodology (Ghadimi and Lan, 2013) to analyze average gradient norms and noise scales on 1,000 sampled instances, we observe in Fig. 2 that the general domain exhibits nearly twice the noise scale of the specific domain. Since such disparities and gradient conflicts in multi-domain optimization are known to degrade efficiency (Yu et al., 2020; Hadsell et al., 2020; Liu et al., 2021), we draw upon prior approaches like projection removal (Yu et al., 2020) and automatic balancing (Liu et al., 2021) to propose a mechanism that balances specific and general domain gradients, thereby preserving generalization while effectively learning specific domain distributions.

3 Method

In this section, we formally introduce the Dynamic Anchor Annealing (DAA) illustrated in Fig. 3, which is based on annealing learning. It is worth to note that DAA is a full-parameter fine-tuning framework in which decoupling refers to the optimization-level separation of general and domain-specific signals within the gradient space rather than the division of model components. Initially, the global gradient is independently estimated in the general domain through zero-learning-rate learning. During the fine-tuning stage, DAA anchors the gradient to preserve the general capability. Subsequently, the similarity between the global gradient and specific gradient adaptively selects the magnitude of parameter update. Finally, the learning rate with the annealing strategy suppresses degradation effectively.

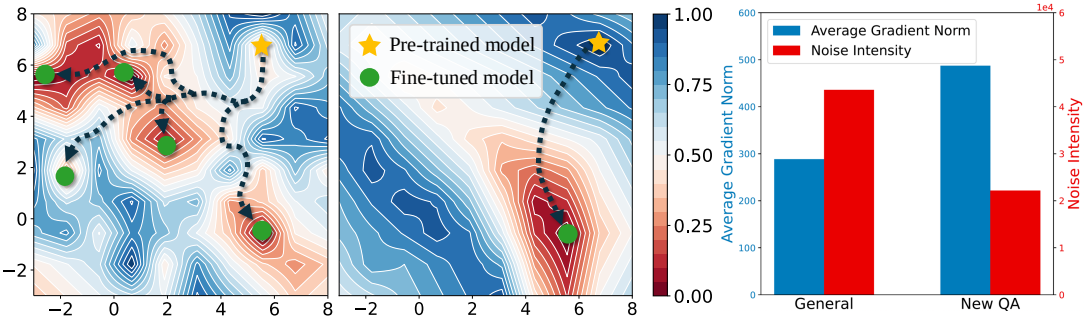
3.1 Global Gradient Anchoring

The global gradient serves as a stable optimization anchor. Our design is motivated by two key factors. 1) Variance Reduction. $\hat{g}_G$ acts as a low-variance regularizer for the high-variance domain gradient $g_{D,t}$, preventing severe oscillations (see Eqn. 4). 2) Trajectory Smoothing. As shown in Fig. 2, $\hat{g}_G$ points towards regions of good generalization, helping the optimizer bypass poor local optima in the general domain landscape.

The term Anchoring is metaphorical. We anchor or augment the domain-specific gradient at each step with this pre-computed stable anchor. This approach is distinct from traditional Gradient Boosting Machines.

In the joint learning of general and specific domains, the gradient is a weighted sum of the general gradient and the specific gradient, with weights determined by the data mixing ratio λ, that is:

$$g_{M,t} = \lambda g_{G,t} + (1 - \lambda) g_{D,t}. \quad (1)$$



(a) General domain loss landscape (b) Specific domain loss landscape (c) Statistic comparison result

Figure 2: Qualitative and quantitative analysis result of general and specific domain.

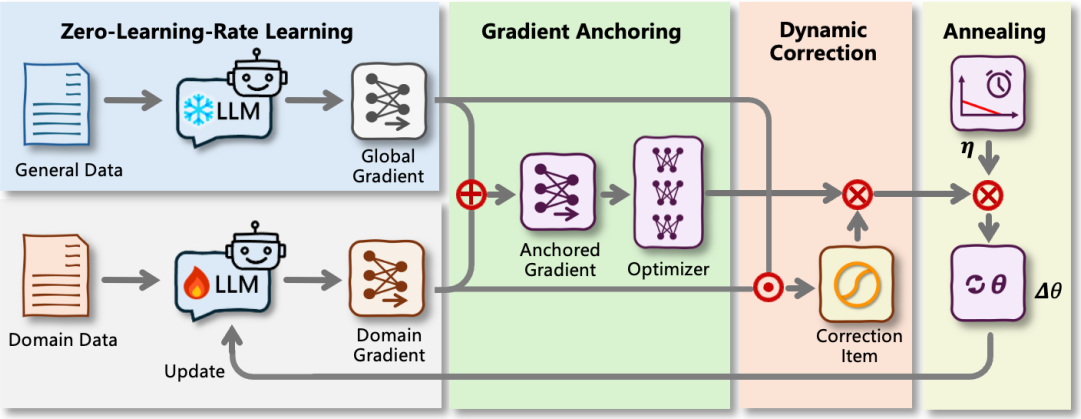


Figure 3: Overview of Dynamic Anchor Annealing. Our approach consists of two stages. In the first stage, global gradient is estimated in the general domain through zero-learning-rate learning, which serves as an independent preprocessing stage. In the second stage, the fine-tuning step, global gradient anchors the specific gradient to preserve general capability, while the similarity between global and specific gradients adaptively determines the parameter update magnitude. The learning rate with annealing strategy suppresses degradation.

We define $\hat{g}_G$ as a fixed estimator of $g_{G,t}$ in joint training to diminish the volatility of the combined gradient.

$$g_{B,t} = \gamma_t \hat{g}_G + (1 - \gamma_t) g_{D,t}, \tag{2}$$

where γ_t is the anchoring magnitude. The expectation and variance of $g_{B,t}$ are given by

$$\mathbb{E}[g_{B,t}] = \gamma_t \hat{g}_G + (1 - \gamma_t)\mathbb{E}[g_{D,t}], \tag{3}$$

$$\mathbb{E}[\|g_{B,t} - \mathbb{E}[g_{B,t}]\|^2] = (1-\gamma_t)^2 \mathbb{E}[\|g_{D,t} - \mathbb{E}[g_{D,t}]\|^2]. \tag{4}$$

By fixed $\hat{g}_G$, we can significantly mitigate the randomness of parameter update while maintaining the regularization effect on optimization.

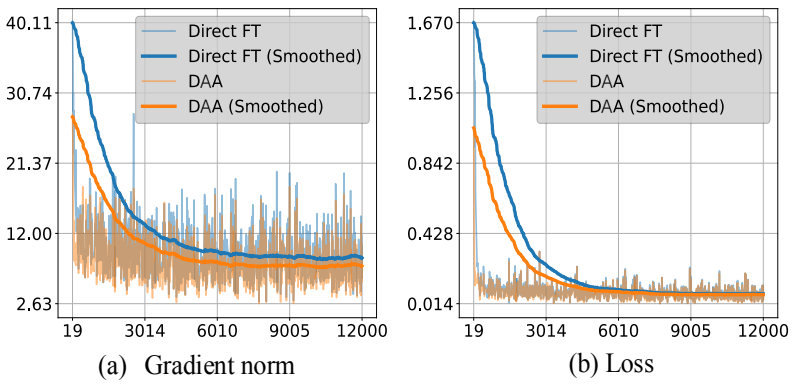
To estimate the global gradient $\hat{g}_G$, according to the derivation of Adam (Kingma and Ba, 2017), when the exponential decay rate for the 1st momentum estimates $\beta_1 \to 1$, the momentum $m_{G,t}$ approximates the expectation of the gradient.

$$\hat{g}_G = \mathbb{E}\left[g_{G,i}\right] = s^{-1} \sum_{i=1}^{s} g_{G,i} = \lim_{\beta_1 \to 1} m_{G,s}. \tag{5}$$

To derive the global anchor $\hat{g}_G$, we process the general domain with a learning rate of zero and a decay rate $\beta_1 \to 1$, storing the final momentum after s steps. Crucially, this single anchor is universally applicable across all target domains. This methodology aligns with LIMA (Zhou et al., 2023) by prioritizing data diversity and quality over quantity. Provided the general dataset possesses sufficient diversity, the direction of $\hat{g}_G$ stabilizes, indicating that massive datasets are not strictly necessary for robust anchoring.

We refer to this method as Global Gradient Anchoring (GGA). The stored m_G from the general domain supplies a stable guidance signal during specific domain optimization and steers updates toward a joint optimum. As shown in Fig. 4, GGA markedly reduces gradient norm and loss, especially early in fine-tuning, which indicates improved training stability on the specific domain with preservation of general capability.

The anchoring magnitude can be adapted over training. We set $\gamma_t = k_0 \left(1 - \frac{t}{T}\right)$, so that the



(a) Gradient norm (b) Loss

Figure 4: Comparative results of global gradient anchoring learning.

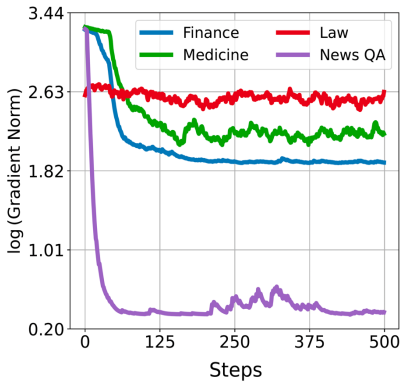


Figure 5: Norm of gradient projection.

model increasingly emphasizes the specific domain near the end of fine tuning. Accounting for exponential averaging in Adam (Kingma and Ba, 2017), the exponential average of $\gamma_t m_G$ yields an effective coefficient α:

$$\alpha_t = k_0 \left(1 - \frac{t}{T}\right) + \frac{k_0 \beta_1 \left(1 - t\beta_1^{t-1} + (t-1)\beta_1^t\right)}{T(1-\beta_1)(1-\beta_1^t)}. \tag{6}$$

The nonlinear term is monotonically increasing and is bounded by $\frac{k_0\beta_1\left(1-t\beta_1^{t-1}+(t-1)\beta_1^t\right)}{T(1-\beta_1)(1-\beta_1^t)}$. Since the cumulative contribution of the global gradient should be comparable across domains, it suffices to choose k_0 inversely proportional to T. With $T \gg k_0$, the nonlinear term becomes negligible and we use the approximation $\alpha_t \approx k_0 \left(1 - \frac{t}{T}\right)$. This schedule is simple to implement and robust. Hyperparameter details and sensitivity analyses are provided in the appendix Sec. A.

For deployment efficiency, storing m_G in 32 bit precision is memory intensive. We therefore apply singular value decomposition to m_G and retain a rank $r = 512$ approximation. During training, we reconstruct the low rank estimate of the global gradient and add it to each step. This saves memory and emphasizes the most informative components of the global signal.

Why a fixed anchor remains valid. The fixed anchor design is consistent with three established formulations. First, it aligns with trust region or shrinking step regularization, as the annealing schedule ensures that $\|\theta_t - \theta_0\| \leq \eta_0^{\text{a}} T/2$. Second, it utilizes gradient space anchoring toward a fixed reference similar to L2 SP. Third, it incorporates staleness tolerance comparable to SVRG under Lipschitz ∇L_G. Furthermore, the annealing weight $\gamma_t \to 0$ approaching zero attenuates any residual mismatch in late training. Empirically, the similarity $\cos(\hat{g}_G, g_G(\theta_t))$ stays above 0.9 throughout fine tuning, as detailed in the Appendix Sec. I.

3.2 Dynamic Correction

While pre-trained language models benefit from diverse data, they often underperform on private or temporally distinct domains. Consequently, applying uniform adaptation strategies across domains with varying distributions yields suboptimal results. To address this, we propose a Dynamic Correction (DC) mechanism that modulates the magnitude of parameter updates based on gradient similarity.

Intuition: direction $\neq$ step size. A counterintuitive design choice in DC is that we take smaller steps when $g_{D,t}$ aligns with $\hat{g}_G$ and larger steps when they are nearly orthogonal. The reason is that the gradient direction tells us where to move instead of how far. Under high alignment in familiar domains, the domain knowledge is largely covered by the base model, meaning large updates risk overshooting an already good equilibrium. Conversely, under low alignment in temporal out of distribution tasks like News QA, genuinely novel knowledge requires larger steps to acquire. Fig. 5 demonstrates that similarity for familiar domains is an order of magnitude higher than for News QA, and DC translates this gap into adaptive step sizes.

To quantify the alignment between general domain and specific domain, we introduce a gradient similarity metric based on the L2 norm of the normalized projection of specific gradient $g_{D,t}$ onto the estimation of the global gradient m_G:

$$s_t = \frac{||g_{D,t} \cdot \hat{g}_G||}{||g_{D,t}|| \cdot ||\hat{g}_G||}, \tag{7}$$

where $g_{D,t}$ denotes gradients for the specific domain at time step t, and $\hat{g}_G$ represents global gradient of the general domain. Our empirical analysis encompasses both familiar domains (finance (Yang et al., 2023), medicine (Wang et al., 2024), and

law (Fei et al., 2023)) and a temporally restrictive domain (news QA, details shown in section 4.2). As shown in Fig. 5, each domain maintains a characteristic similarity range with the general domain. Notably, familiar domains exhibit similar magnitude, while the unfamiliar domain demonstrates significantly lower similarity values, differing by more than an order of magnitude.

Leveraging this similarity measurement, we introduce a dynamic correction coefficient:

$$c_t = s_t + c_0, \tag{8}$$

where c_0 represents a base coefficient that prevents excessive parameter updates and potential overfitting when similarity values are minimal. We set $c_0 = 0.01$ in practice. The parameter update rule incorporating the dynamic correction is:

$$\Delta\theta = -\eta \frac{\hat{m}_t}{\sqrt{c_t \hat{v}_t} + \varepsilon}. \tag{9}$$

3.3 Annealing Learning

In the training of MiniCPM (Hu et al., 2024) and Llama3 (Grattafiori et al., 2024), Annealing Learning (AL) is applied at the final stage of pre-training. Using learning rate with minimal initialization and decay strategy, LLMs can learn downstream task knowledge from high-quality domain data without forgetting. Suppose the conventional initialization of learning rate is η_0, and η_0^{a} for annealing, the parameter updates for both schemes are:

$$\begin{aligned} \Delta\theta_t &= -\eta_0 \left(1 - \frac{t}{T}\right) \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \varepsilon}, \\ \Delta\theta_t^{\mathrm{a}} &= -\eta_0^{\mathrm{a}} \left(1 - \frac{t}{T}\right) \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \varepsilon}. \end{aligned} \tag{10}$$

Via a detailed comparative analysis of Eqn. 10, we can measure the influence of annealing on the parameter updates:

$$\Delta\theta_t - \Delta\theta_t^{\mathrm{a}} = -\left(\eta_0 - \eta_0^{\mathrm{a}}\right) \left(1 - \frac{t}{T}\right) \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \varepsilon}. \tag{11}$$

As shown in Eqn. 11, annealing effectively suppresses the learning of specific domains. Smaller parameter updates thus reduce the risk of catastrophic forgetting. Therefore, we use the annealing learning scheme in DAA. And we set $\eta_0^{\mathrm{a}} = 1e^{-7}$ in the our current experiment.

3.4 Summary

After integrating the above learning strategies, we obtain the complete parameter update of DAA:

$$\Delta\theta_t^{\mathrm{DAA}} = -\eta_0^{\mathrm{a}} \left(1 - \frac{t}{T}\right) \frac{\hat{m}_{B,t}}{\sqrt{c_t \hat{v}_{B,t}} + \varepsilon}. \tag{12}$$

The integration of GGA, DC and AL facilitates the adaptation to specific domains while simultaneously mitigating forgetting.

4 Experiments

4.1 Experiment Settings

We evaluate DAA across finance, medicine, and law within English and Chinese contexts. The general-domain training data comprises multi-task corpora detailed in appendix Sec. B, Tab. 7. We utilize FinGPT (Yang et al., 2023), CMB (Wang et al., 2024), and Fuzi-Mingcha (Deng et al., 2023) for evaluation. To mitigate benchmark contamination (Schaeffer, 2023; Jain et al., 2024; Zhang et al., 2024b), we introduce a temporal out-of-distribution (OOD) benchmark, News QA (Sec. 4.2). We compare DAA with baselines including direct fine-tuning, vanilla fine-tuning, LoRA (Hu et al., 2021), DoRA (Liu et al., 2024), and Galore (Zhao et al., 2024). Following (Wen et al., 2023), vanilla fine-tuning employs three data mixture ratios (specific to general: 1:1, 1:3, 1:5), reporting the optimal performance for each domain. Validations are conducted on foundational models including Llama3.1-8B (Grattafiori et al., 2024), Phi4-14B (Abdin et al., 2024), and Qwen3-8B (Yang et al., 2025).

4.2 News QA Benchmark

We build News QA as a temporal out-of-distribution factual-verification stress test rather than a general-purpose generative QA benchmark. Its purpose is narrow but well-defined, given that all evaluated base models pre-train before December 2024, we ask whether a fine-tuned model can absorb post-cutoff factual knowledge while retaining its general capability. The benchmark is constructed by extracting news articles published after December 2024 and using Qwen2.5-72B (Yang et al., 2024) to generate balanced True/False statements, see Tab. 2. The dataset contains 30,613 news titles across three categories (Politics, Economics, and Culture). All foundational models score only 31.06% average accuracy on the test set (Tab. 2, row 3), confirming the temporal-OOD

nature of the task. Because the labels are LLM-generated, generation prompts, label-balancing rules, topic and temporal metadata are released with the code so that potential stylistic artifacts can be inspected, and a topic/temporal robustness split is reported in Appendix Sec. H.

Split	Politics	Econ	Culture	Total
Train set	9823	10120	8670	28613
Test set	700	700	600	2000
S_D	29.05	30.70	33.43	31.06

Table 2: Details of news QA benchmark. The first two rows show the data distribution. The third row presents performance (S_D) of Qwen3-8B on each category.

4.3 Metrics

To evaluate general capabilities, we employ widely recognized benchmarks including MMLU (Hendrycks et al., 2021a), MMLU-Pro, GSM8K (Cobbe et al., 2021), MATH (Hendrycks et al., 2021b), and M3Exam (Zhang et al., 2023a). These datasets assess multidisciplinary knowledge and mathematical reasoning. Additionally, we use CMMLU to evaluate Chinese specific knowledge. All raw scores lie within the interval [0, 1], and we compute the aggregate metrics S_D, S_G, and S without external normalization. For vertical domains, we select representative public benchmarks. In the financial domain, we utilize the average weighted F1 score across the FPB (Malo et al., 2013), FiQA (Maia et al., 2018), TFNS (Zer, 2024), and NWGI (Yang, 2024) sentiment analysis datasets. For the medical and legal domains, we employ the Chinese CMB-Exam (Wang et al., 2024) and LawBench (Fei et al., 2023) respectively. The overall metric design is structured as follows:

$$S_G = \text{Mean}(\{S_x \mid x \in \mathcal{X}\}), \tag{13}$$

$$S = \text{HarmonicMean}(S_D, S_G), \tag{14}$$

where $\mathcal{X}$ includes MMLU, MMLU-Pro, GSM8k, MATH, and M3Exam. S_D is the score of the model's vertical domain performance, S_G is the average score of the model's general performance. S is the harmonic mean of S_D and ΔS_G, meaning the model scores high only if both are large. If fine-tuning boosts domain performance but reduces general capability significantly, the score nears 0.0. Conversely, if it enhances domain performance while preserving general capability, the score approaches 1.0.

4.4 Cost Analysis

We distinguish two notions of cost. The per run cost is the GPU hours for a single fine tuning run. The total workflow cost is the GPU hours to obtain a deployment ready domain model in practice. Per run, DAA (T_{DAA}≈4.2 h) matches direct full tuning (T_{Direct}≈4.1 h) and is slower than LoRA (≈3.0 h) and DoRA (≈3.1 h), but is consistently better in joint S (Tab. 3). The headline 91% saving is therefore at the workflow level. Vanilla data mixture fine tuning costs ≈15.6 h per run, and the standard recipe (Wen et al., 2023) sweeps three ratios (1:1, 1:3, 1:5), so the workflow cost reaches ≈46.7 h per domain. DAA replaces this sweep with a single 4.2 h run plus a one time anchor pre-computation reused across all domains of the same base model, giving 46.7→4.2 h (∼91%) within +0.7 GB peak VRAM (Appendix Sec. F).

4.5 Main Results

Tab. 3 demonstrates that DAA consistently achieves the best balance between domain ability and general retention across four distinct domains and three base models. On familiar domains (Finance, Medicine, Law), pre training already covers most knowledge so the headroom is small, yet DAA still wins or ties the best S on every cell. The largest gains appear on Qwen3, where Direct and Vanilla fine tuning lose 7 to 8 points of S_G to alignment drift, while DAA preserves S_G and matches them on S_D (Finance S=80.66 vs. 77.58, Medicine S=84.51 vs. 74.71, Law S=63.41 vs. 60.91). On the temporal OOD News QA setting, post cutoff facts are absent from pre training. Direct fine tuning collapses on Qwen3 (S_G=3.83, S=7.31), whereas DAA retains S_G=68.52 at S_D=80.82 (S=74.16); on Phi4, DAA reaches the highest S_D=89.19 and S_G=77.91, beating the best baseline by 2.76 points on S. The gain is most pronounced on temporal OOD News QA, confirming that DAA enables genuine knowledge acquisition rather than mere style alignment. A more detailed Law analysis is in Appendix Sec. C.

4.6 Ablation Analysis

Tab. 4 shows that on News QA, GGA alone significantly improves S_G over direct fine-tuning, confirming gradient space anchoring as the primary anti-collapse mechanism. Sequentially adding DC and AL further boosts S, proving the gains extend beyond merely a small effective learning rate. Ad-

Domain	Method	Llama3.1				Phi4				Qwen3			
		$T\downarrow$	$S_D\uparrow$	$S_G\uparrow$	$S\uparrow$	$T\downarrow$	$S_D\uparrow$	$S_G\uparrow$	$S\uparrow$	$T\downarrow$	$S_D\uparrow$	$S_G\uparrow$	$S\uparrow$
Finance	Direct FT	3.40	**80.01**	54.69	64.97	6.22	**89.72**	77.92	**83.41**	3.38	**85.83**	63.74	73.15
	Vanilla FT	38.77	79.30	60.50	68.64	71.24	83.42	77.71	80.46	38.74	85.49	71.01	77.58
	LoRA	**2.50**	76.45	61.69	68.28	**4.61**	87.13	78.27	82.46	**2.50**	81.68	73.21	77.21
	DoRA	2.58	76.12	61.25	67.88	4.71	86.34	78.24	82.09	2.57	82.19	73.27	77.47
	Galore	5.09	77.31	60.78	68.06	9.34	86.23	**78.55**	82.21	5.09	84.37	74.59	79.18
	DAA (Ours)	3.45	79.84	**61.75**	**69.64**	6.35	87.73	78.50	82.86	3.41	85.32	**76.49**	**80.66**
Medicine	Direct FT	12.39	**89.23**	52.73	66.29	22.78	92.13	78.34	84.68	12.38	**92.67**	64.44	76.02
	Vanilla FT	141.81	87.32	59.47	**70.75**	260.58	91.24	**79.26**	84.83	141.81	81.81	68.74	74.71
	LoRA	**9.18**	81.76	59.97	69.19	**16.87**	90.30	79.02	84.28	**8.21**	84.00	69.51	76.07
	DoRA	9.31	81.21	59.07	68.40	17.05	90.31	78.74	84.13	8.37	84.74	69.67	76.47
	Galore	13.68	81.23	58.55	68.05	25.14	91.96	78.56	84.73	13.68	87.57	73.71	80.04
	DAA (Ours)	12.54	83.97	**60.33**	70.22	22.90	**92.61**	78.82	**85.16**	12.64	92.24	**77.97**	**84.51**
Law	Direct FT	4.39	**56.81**	48.94	52.58	8.04	**42.63**	71.06	53.29	4.39	**55.28**	70.13	61.83
	Vanilla FT	50.12	51.37	53.08	52.21	91.98	41.82	72.12	52.94	50.10	52.28	72.95	60.91
	LoRA	**3.25**	46.58	**58.06**	51.69	**5.97**	41.87	**73.38**	**53.32**	**3.25**	51.90	76.19	61.74
	DoRA	3.43	46.37	56.05	50.75	6.24	41.98	72.36	53.13	3.36	51.91	76.30	61.79
	Galore	6.13	47.80	56.13	51.63	11.26	40.12	71.53	51.41	6.13	52.76	77.53	62.79
	DAA (Ours)	4.53	49.93	56.68	**53.09**	8.17	41.95	73.12	53.31	4.60	52.79	**79.38**	**63.41**
News QA	Direct FT	1.27	**82.38**	36.34	50.44	2.32	87.38	26.40	40.55	1.27	79.37	3.83	7.31
	Vanilla FT	14.45	80.62	39.73	53.22	16.51	88.23	72.11	79.36	14.45	80.27	52.36	63.38
	LoRA	**0.94**	71.23	48.04	57.38	**1.72**	83.12	77.46	80.19	**0.93**	70.27	67.51	68.86
	DoRA	0.99	73.71	50.52	59.95	1.80	83.37	76.68	79.88	0.99	70.78	67.81	69.26
	Galore	2.02	79.13	**53.05**	63.52	3.70	85.02	76.27	80.41	2.03	**80.88**	68.43	74.14
	DAA (Ours)	1.32	82.37	52.00	**63.76**	2.42	**89.19**	**77.91**	**83.17**	1.28	80.82	**68.52**	**74.16**

Table 3: Performance metrics across different domains and models. T is the GPU hours. S_D is the score of the model's vertical domain performance, S_G is the average score of the model's general performance change. S is the harmonic mean of S_D and S_G.

Variant	Llama3.1-8B			Phi4-14B		
	S_D	S_G	S	S_D	S_G	S
AL	68.22	27.60	39.30	78.07	46.47	58.26
GGA	76.84	40.32	52.86	84.93	58.61	69.36
GGA+DC	78.51	45.18	57.36	85.71	64.92	73.88
AL+GGA	71.25	31.36	43.55	82.14	50.34	62.42
AL+DC	72.99	41.66	53.04	83.35	61.49	70.77
DAA	**75.65**	**47.09**	**58.05**	**86.24**	**67.36**	**75.64**

Table 4: Component isolation of Annealing Learning, Global Gradient Anchoring and Dynamic Correction on News QA. More experimental results are provided in Appendix Sec. J.

Method	ΔVRAM	T (h)	S
Direct FT	0	4.1	50.44
Vanilla FT	0	14.5	53.22
PCGrad	+5.4 GB	8.3	55.38
GradNorm	+5.6 GB	8.5	55.91
EWC	+5.2 GB	4.4	54.62
LwF	+0.6 GB	5.7	55.84
O-LoRA	0	3.6	56.27
DAA	**+0.7 GB**	**4.2**	**63.76**

Table 5: Optimization level and continual learning baselines evaluated on Llama3.1-8B / News QA at a matched compute budget.

ditionally, under equal compute budgets (Tab. 5), gradient surgery methods double costs and add over 5GB peak VRAM, while continual learning maintains S_G but trails in S_D. Ultimately, DAA achieves the optimal joint S at a cost comparable to direct fine-tuning.

Cost-wise (Tab. 5), DAA achieves unmatched Pareto efficiency. While traditional gradient surgery doubles training costs by computing multiple gradients and adds over 5GB peak VRAM, and continual learning maintains S_G but trails in S_D due to rigid parameter constraints, DAA outperforms the strongest baseline (O-LoRA) by 7.49 points in S. Ultimately, DAA secures the optimal joint performance with negligible overhead (+0.7GB VRAM, +0.1h) over standard direct fine-tuning without requiring complex projections.

5 Conclusion

We present Dynamic Anchor Annealing, a fine-tuning method that mitigates catastrophic forgetting in LLMs. Using global gradient anchoring with similarity guided dynamic correction, DAA improves performance while reducing computation cost over prior methods. Domain specific LLMs equip workers with specialized AI in their fields. We will explore broader applications of DAA to inspire research on domain specific training.

Limitations

While Dynamic Anchor Annealing demonstrates robust performance across vertical domains such as finance and medicine, several limitations warrant future investigation. First, our current evaluation is primarily confined to textual modalities and dense language model architectures. The framework's applicability to other modalities, such as computer vision or speech processing, as well as its effectiveness within reinforcement learning paradigms, remains unverified. Furthermore, although the method scales linearly in theory, its stability and convergence behavior on extremely deep networks or models with hundreds of billions of parameters have not yet been empirically tested.

Second, the implementation of Global Gradient Anchoring introduces a requisite pre-computation stage involving general domain data. Although this process eliminates the need for repeated data mixture experiments and the resulting gradient artifact is reusable, it necessitates initial access to a high-quality general dataset to derive the stable anchor. Additionally, as indicated in our cost analysis, the method incurs a modest increase in peak VRAM usage during the fine-tuning phase to accommodate the sharded gradient anchors. This additional memory overhead, while minimal compared to the savings in experimental time, may still present constraints for hardware environments with strictly limited memory resources.

Ethics Statement

This work utilizes a combination of publicly available benchmarks and self-constructed datasets across diverse vertical domains. For specific domain evaluations, we employ established datasets including FinGPT (Yang et al., 2023) for finance, CMB-Exam (Wang et al., 2024) for medicine, and Fuzi-Mingcha (Deng et al., 2023) for legal reasoning, alongside standard general benchmarks such as MMLU (Hendrycks et al., 2020) and GSM8K (Cobbe et al., 2021). All public datasets are used in accordance with their original licenses. Regarding the self-constructed News QA benchmark, data was collected from publicly accessible news articles published after December 2024 to evaluate temporal out-of-distribution performance. We conducted an assessment to ensure that the data used, particularly in the medical and legal domains, consists of anonymized case studies, standardized exam questions, or public records, thereby avoiding the exposure of sensitive personal information or real-world private individuals.

We employ open-source large language models including Llama3.1-8B (Grattafiori et al., 2024), Phi4-14B (Abdin et al., 2024), and Qwen3-8B (Yang et al., 2025), accessed through standard model repositories. All models are used under their respective licenses which permit research and fine-tuning. We have reviewed and complied with all terms of use for these models. Furthermore, our proposed method, Dynamic Anchor Annealing (DAA), is designed to enhance these models' domain-specific capabilities without compromising their general knowledge. By mitigating catastrophic forgetting, our approach aims to preserve the safety alignment and reasoning capabilities present in the pre-trained models that might otherwise be degraded during aggressive domain adaptation. We have not observed the generation of offensive or harmful content during our evaluation, though we acknowledge the necessity of continuous monitoring in real-world deployments.

References

2024. Zeroshot/twitter-financial-news-sentiment · Datasets at Hugging Face.

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, and 1 others. 2024. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*.

Chirag Agarwal, Daniel D'souza, and Sara Hooker. 2022. Estimating example difficulty using variance of gradients. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10368–10378.

Zhijie Bao, Wei Chen, Shengze Xiao, Kuang Ren, Jiaao Wu, Cheng Zhong, Jiajie Peng, Xuanjing Huang, and Zhongyu Wei. 2023. DISC-MedLLM: Bridging General Large Language Models and Real-World Medical Consultation.

Junying Chen, Xidong Wang, Ke Ji, Anningzhe Gao, Feng Jiang, Shunian Chen, Hongbo Zhang, Dingjie Song, Wenya Xie, Chuyi Kong, Jianquan Li, Xiang Wan, Haizhou Li, and Benyou Wang. 2024. HuatuoGPT-II, One-stage Training for Medical Adaption of LLMs.

Sanyuan Chen, Yutai Hou, Yiming Cui, Wanxiang Che, Ting Liu, and Xiangzhan Yu. 2020. Recall and Learn: Fine-tuning Deep Pretrained Language Models with Less Forgetting.

Wei Chen, Qiushi Wang, Zefei Long, Xianyin Zhang, Zhongtian Lu, Bingxuan Li, Siyuan Wang, Jiarong Xu, Xiang Bai, Xuanjing Huang, and Zhongyu Wei. 2023. DISC-FinLLM: A Chinese Financial Large Language Model based on Multiple Experts Fine-tuning.

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

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

Jiaxi Cui, Zongjian Li, Yang Yan, Bohua Chen, and Li Yuan. 2023. ChatLaw: Open-Source Legal Large Language Model with Integrated External Knowledge Bases.

Wentao Deng, Jiahuan Pei, Keyi Kong, Zhe Chen, Furu Wei, Yujun Li, Zhaochun Ren, Zhumin Chen, and Pengjie Ren. 2023. Syllogistic Reasoning for Legal Judgment Analysis. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13997–14009. Association for Computational Linguistics.

Zhiwei Fei, Xiaoyu Shen, Dawei Zhu, Fengzhe Zhou, Zhuo Han, Songyang Zhang, Kai Chen, Zongwen Shen, and Jidong Ge. 2023. LawBench: Benchmarking Legal Knowledge of Large Language Models.

Saeed Ghadimi and Guanghui Lan. 2013. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM journal on optimization*, 23(4):2341–2368.

Aaron Grattafiori, Abhimanyu Dubey, and Jauhri et al. 2024. The Llama 3 Herd of Models.

Mert Gurbuzbalaban, Umut Simsekli, and Lingjiong Zhu. 2021. The heavy-tail phenomenon in sgd. In *International Conference on Machine Learning*, pages 3964–3975. PMLR.

Raia Hadsell, Dushyant Rao, Andrei A Rusu, and Razvan Pascanu. 2020. Embracing change: Continual learning in deep neural networks. *Trends in cognitive sciences*, 24(12):1028–1040.

William Held, Bhargavi Paranjape, Punit Singh Koura, Mike Lewis, Frank Zhang, and Todor Mihaylov. 2025. Optimizing Pretraining Data Mixtures with LLM-Estimated Utility. *Preprint*, arXiv:2501.11747.

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*.

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

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021b. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. LoRA: Low-Rank Adaptation of Large Language Models.

Shengding Hu, Yuge Tu, Xu Han, Chaoqun He, Ganqu Cui, Xiang Long, Zhi Zheng, Yewei Fang, Yuxiang Huang, Weilin Zhao, Xinrong Zhang, Zheng Leng Thai, Kaihuo Zhang, Chongyi Wang, Yuan Yao, Chenyang Zhao, Jie Zhou, Jie Cai, Zhongwu Zhai, and 6 others. 2024. MiniCPM: Unveiling the Potential of Small Language Models with Scalable Training Strategies.

Quzhe Huang, Mingxu Tao, Chen Zhang, Zhenwei An, Cong Jiang, Zhibin Chen, Zirui Wu, and Yansong Feng. 2023. Lawyer LLaMA Technical Report.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. 2024. Livecodebench: Holistic and contamination free evaluation of large language models for code. *arXiv preprint arXiv:2403.07974*.

Diederik P. Kingma and Jimmy Ba. 2017. Adam: A Method for Stochastic Optimization.

Tomasz Korbak, Hady Elsahar, German Kruszewski, and Marc Dymetman. 2022. Controlling Conditional Language Models without Catastrophic Forgetting. In *Proceedings of the 39th International Conference on Machine Learning*, pages 11499–11528. PMLR.

Xize Liang, Lin Yang, Jie Wang, Yiyang Lu, Runyu Wu, Hanzhu Chen, and Jianye Hao. 2025. Boosting multi-domain fine-tuning of large language models through evolving interactions between samples. In *Forty-second International Conference on Machine Learning*.

Yong Lin, Lu Tan, Hangyu Lin, Zeming Zheng, Renjie Pi, Jipeng Zhang, Shizhe Diao, Haoxiang Wang, Han Zhao, Yuan Yao, and Tong Zhang. 2023. Speciality vs Generality: An Empirical Study on Catastrophic Forgetting in Fine-tuning Foundation Models. *Preprint*, arXiv:2309.06256.

Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. 2021. Conflict-averse gradient descent for multi-task learning. *Advances in Neural Information Processing Systems*, 34:18878–18890.

Hong Liu, Mingsheng Long, Jianmin Wang, and Michael Jordan. 2019. Transferable adversarial training: A general approach to adapting deep classifiers. In *International conference on machine learning*, pages 4013–4022. PMLR.

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*.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. 2025. An Empirical Study of Catastrophic Forgetting in Large Language Models During Continual Fine-tuning. *Preprint*, arXiv:2308.08747.

Macedo Maia, Siegfried Handschuh, André Freitas, Brian Davis, Ross McDermott, Manel Zarrouk, and Alexandra Balahur. 2018. WWW'18 Open Challenge: Financial Opinion Mining and Question Answering. In *Companion of the The Web Conference 2018 on The Web Conference 2018 - WWW '18*, pages 1941–1942. ACM Press.

Pekka Malo, Ankur Sinha, Pyry Takala, Pekka Korhonen, and Jyrki Wallenius. 2013. Good Debt or Bad Debt: Detecting Semantic Orientations in Economic Texts.

Rylan Schaeffer. 2023. Pretraining on the test set is all you need. *arXiv preprint arXiv:2309.08632*.

Ozan Sener and Vladlen Koltun. 2018. Multi-task learning as multi-objective optimization. In *Advances in neural information processing systems*, volume 31.

Xidong Wang, Guiming Hardy Chen, Dingjie Song, Zhiyi Zhang, Zhihong Chen, Qingying Xiao, Feng Jiang, Jianquan Li, Xiang Wan, Benyou Wang, and Haizhou Li. 2024. CMB: A Comprehensive Medical Benchmark in Chinese.

Cheng Wen, Xianghui Sun, Shuaijiang Zhao, Xiaoquan Fang, Liangyu Chen, and Wei Zou. 2023. ChatHome: Development and Evaluation of a Domain-Specific Language Model for Home Renovation.

Minghao Wu, Thuy Vu, Lizhen Qu, and Reza Haf. 2024. Mixture-of-skills: Learning to optimize data usage for fine-tuning large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 14226–14240.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. 2023. BloombergGPT: A Large Language Model for Finance.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. 2024. Less: Selecting influential data for targeted instruction tuning. *arXiv preprint arXiv:2402.04333*.

Chaojun Xiao, Xueyu Hu, Zhiyuan Liu, Cunchao Tu, and Maosong Sun. 2021. Lawformer: A pre-trained language model for Chinese legal long documents. 2:79–84.

Qianqian Xie, Weiguang Han, Xiao Zhang, Yanzhao Lai, Min Peng, Alejandro Lopez-Lira, and Jimin Huang. 2023. PIXIU: A Comprehensive Benchmark, Instruction Dataset and Large Language Model for Finance. 36:33469–33484.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

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

Hongyang Yang. 2024. AI4Finance-Foundation/FinGPT.

Hongyang Yang, Xiao-Yang Liu, and Christina Dan Wang. 2023. FinGPT: Open-Source Financial Large Language Models.

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

Shengbin Yue, Wei Chen, Siyuan Wang, Bingxuan Li, Chenchen Shen, Shujun Liu, Yuxuan Zhou, Yao Xiao, Song Yun, Xuanjing Huang, and Zhongyu Wei. 2023. DISC-LawLLM: Fine-tuning Large Language Models for Intelligent Legal Services.

Biao Zhang, Zhongtao Liu, Colin Cherry, and Orhan Firat. 2024a. When Scaling Meets LLM Finetuning: The Effect of Data, Model and Finetuning Method. *Preprint*, arXiv:2402.17193.

Hugh Zhang, Jeff Da, Dean Lee, Vaughn Robinson, Catherine Wu, Will Song, Tiffany Zhao, Pranav Raja, Dylan Slack, Qin Lyu, and 1 others. 2024b. A careful examination of large language model performance on grade school arithmetic. *arXiv preprint arXiv:2405.00332*.

Wenxuan Zhang, Mahani Aljunied, Chang Gao, Yew Ken Chia, and Lidong Bing. 2023a. M3exam: A multilingual, multimodal, multilevel benchmark for examining large language models. *Advances in Neural Information Processing Systems*, 36:5484–5505.

Xuanyu Zhang, Qing Yang, and Dongliang Xu. 2023b. XuanYuan 2.0: A Large Chinese Financial Chat Model with Hundreds of Billions Parameters.

Jiawei Zhao, Zhenyu Zhang, Beidi Chen, Zhangyang Wang, Anima Anandkumar, and Yuandong Tian. 2024. Galore: Memory-efficient llm training

by gradient low-rank projection. *arXiv preprint arXiv:2403.03507*.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, and 1 others. 2023. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36:55006–55021.

Zhi Zhou, Jiang-Xin Shi, Peng-Xiao Song, Xiao-Wen Yang, Yi-Xuan Jin, Lan-Zhe Guo, and Yu-Feng Li. 2024. LawGPT: A Chinese Legal Knowledge-Enhanced Large Language Model.

Jingwei Zhu, Minghuan Tan, Min Yang, Ruixue Li, and Hamid Alinejad-Rokny. 2024. CollectiveSFT: Scaling Large Language Models for Chinese Medical Benchmark with Collective Instructions in Healthcare.

A Hyper Parameters

This section will introduce the detailed process and hyperparameters involved in model training and testing. In the main experiments and ablation experiments, we chose Qwen2-7B as our base model. To obtain the gradient expectation estimation of the general domain, we set the learning rate of the general domain training $\eta_G = 0$, meaning no parameter updates are performed in the general domain. Additionally, $\beta_1 = 0.999$, so the momentum approximates the gradient expectation. The training batch size is 8, and only the gradient momentum is retained after training. Note that the computation in the general domain only needs to be done **once**, and the same momentum is used for different vertical domains subsequently. Since the original momentum is in F32 data format, loading it directly into the GPU memory would occupy a large space. We performed singular value decomposition on the momentum, retaining $r = 512$ dimensions of singular values and vectors. During the global gradient anchoring in training, the low-rank approximation of the original momentum is restored and then added to the gradient. In the fine-tuning phase, we set the initial learning rate $\eta_D = 1e-7$, which is much lower than the usual fine-tuning learning rate. We used a linear decay to zero learning rate schedule without warmup. The training batch size is 8, and we train for only one epoch. We use the AdamW optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.95$. For the global gradient anchoring coefficients defined in Eqn. 7 and 8, we chose a linear decay scheme with $k_0 = 200/T$, where T is the total number of steps in vertical domain fine-tuning. We propose the heuristic $k_0 \propto 1/T$ as a practical guideline.

k_0	S
$50/T$	57.23
$100/T$	60.13
$150/T$	61.98
$175/T$	62.92
$200/T$	63.76
$225/T$	63.28
$250/T$	63.50
$275/T$	63.12
$300/T$	63.39

Table 6: Extended sensitivity analysis of hyperparameter k_0 on the NewsQA benchmark.

This relationship maintains a consistent total contribution from global gradients irrespective of the training duration. Specifically, the total 'regularization influence' exerted by $\hat{g}_G$ can be approximated by the integral of the anchoringing magnitude γ_t over time: $\int_0^T [k_0(1 - t/T)]dt = k_0 \cdot T/2$. To ensure this total contribution remains a constant C independent of the total steps T, k_0 must be proportional to $1/T$. This ensures that whether fine-tuning for 1,000 or 5,000 steps, the total "pull" from the global anchor remains consistent.

This is analogous to tuning LoRA, where practitioners often fix the dropout rate and primarily experiment with the rank (r) and scaling factor (α). In our case, the core tuning effort is simplified to a single, well-behaved parameter governed by a clear rule. As shown in Tab. 6, the performance metric S of Llama3.1 on NewsQA improves as the hyperparameter k_0 increases. However, this growth plateaus after k_0 reaches $200/T$. Since there is no significant performance gain beyond this point, we select $k_0 = 200/T$ as the value for our experiments.

B Dataset Details

We obtained validated our proposed method across a wide range of vertical domains, covering finance, medicine, law and news QA.

General Data: Since the vertical domain tasks mainly cover Chinese and English languages and include multiple-choice and conversational tasks, the general data needs to fully cover similar data patterns. Therefore, we collected Chinese and English QA data, covering QA, conversations, and multiple-choice questions. Specifically, the general data includes 54,042 Chinese QA conversation pairs, 65,596 English QA conversation pairs, and 1,881 Chinese multiple-choice questions. Detailed data sources and quantities are shown in Tab. 7.

Name	Source	Quantity
Chinese QA Data	Self-built	54,042
English QA Data	Self-built	65,596
Chinese MCQs	Self-built	1,881

Table 7: General Data sources and quantities

Name	Source	Quantity
English Sentiment Data	FPB	12,122
	FiQA	26,532
	TFNS	12,731
	NWGI	25,387

Table 8: Finance Data sources and their quantities.

Finance: We referred to the training data and testing methods of FinGPT (Yang et al., 2023), selecting its sentiment analysis task as the financial vertical domain. This task requires the model to analyze the market sentiment of the input text as negative, neutral, or positive. According to (Yang et al., 2023), the training data was collected from FPB (Malo et al., 2013), FiQA (Maia et al., 2018), TFNS (Zer, 2024), and NWGI (Yang, 2024). FinGPT designed three types of instructions for each original data, resulting in a total of 76,772 training samples after filtering. Detailed data sources and quantities are shown in Tab. 8.

Medicine: We chose the CMB-Exam from the Chinese medicine Benchmark (CMB) (Wang et al., 2024) as the medical domain. This dataset includes 280,839 medicine multiple-choice questions, covering 124,926 physician questions, 16,919 nursing questions, 27,004 medicine technician questions, 33,354 pharmacist questions, 62,271 undergraduate exam questions, and 16,365 graduate entrance exam questions. We randomly selected 11,200 questions from each category as the test set, with a total of 269,359 questions in the training set. Detailed data sources and quantities are shown in Tab. 9.

Law: We referred to the data summarized by the Fuzi-Mingcha (Deng et al., 2023) to filter suitable legal vertical fine-tuning data. The fine-tuning data composition is as follows: 4,200 recall data and 5,750 summarization data from CAIL-Long (Xiao et al., 2021), 35,000 legal QA data from LawGPT (Zhou et al., 2024), 11,000 legal QA data from Lawyer Llama (Huang et al., 2023), 32,050 legal QA data and 11,237 syllogistic reasoning judgment data independently constructed by Fuzi-Mingcha (Deng et al., 2023). The total training data amounts to 99,237 samples. Detailed data Statistics are shown in Tab. 10.

Name	Source	Quantity
Physician	Physician Exam	124,926
Nursing	Nursing Exam	16,919
Technician	Technician Exam	27,004
Pharmacist	Pharmacist Exam	33,354
Undergraduate	medicine Exam	62,271
Graduate Entrance	medicine Exam	16,365

Table 9: Questions from various Chinese medicine exams.

Name	Source	Quantity
Fact Recall	CAIL-Long	4,200
Case Summarization	CAIL-Long	5,750
	LawGPT	35,000
Legal QA Data	Lawyer Llama	11,000
	Fuzi	32,050
Syllogistic Reasoning	Fuzi	11,237

Table 10: Law Data Statistics

News QA: To precisely evaluate the domain decoupling capabilities, we constructed a temporal out-of-distribution evaluation benchmark comprising QA pairs derived from news articles published after December 2024 for ablation study. We used Qwen2.5-72B (Yang et al., 2024) to extract three factual QA questions for each headline. We ensured that there is no overlap between the vertical domain data and the general data.

The above datasets come from diverse sources, and the characteristics and distributions among the datasets vary significantly, providing ample and credible test scenarios for verifying the effectiveness of the Dynamic Anchor Annealing scheme.

C Performance on the Law Dataset

Task Diversity and Complexity: The Law fine-tuning data is a highly heterogeneous mixture of tasks, including not only multiple-choice questions but also complex generation tasks like **Case Summarization** and reasoning tasks like **Syllogistic Reasoning**. These generative and reasoning tasks are fundamentally more challenging and diverge more significantly from the pre-training objectives than the classification-style tasks that dominate the Finance, Medicine, and News QA datasets.

Performance Interpretation: While the absolute score on Law is lower across all methods, it is important to note that DAA still consistently achieves the best or second-best harmonic mean score (S) across all three base models (Tab. 3). For instance, on Qwen2.5, DAA achieves the highest

Method	Seed 42	Seed 43	Seed 44	Seed 45	Seed 46	**Mean** (S)	**Std Dev** (σ)
Vanilla FT	53.22	52.15	54.05	52.80	53.75	53.19	0.76
DAA (Ours)	**63.76**	64.25	63.15	63.88	63.45	**63.70**	**0.42**

Table 11: Significance test results on News QA (Llama3.1-8B).

Method	Anchor Memory	Peak VRAM	Parameters	Wall-clock	GPU Hour	S
Direct FT	-	37.1GB	7B	3.91h	3.40h	50.44
DAA	5.5GB	37.8GB	7B	4.20h	3.45h	63.76

Table 12: Training overhead comparison on Llama3.1-8B / News QA.

S score (59.85), significantly outperforming Direct FT (58.27) and Vanilla FT (57.35) by better preserving general capabilities (S_G). This demonstrates that even in this more complex, generation-heavy domain, DAA's regularization mechanism provides a tangible benefit over baselines by striking a better balance between domain specialization and knowledge retention.

Conclusion on Generality: The Law dataset does not necessarily indicate a weakness but rather highlights DAA's robust performance on a more challenging and diverse task mixture. It showcases that DAA's benefits are not confined to simple classification tasks but extend to complex, mixed-task scenarios.

D Practical Implementation Guide

While Dynamic Anchor Annealing (DAA) introduces steps beyond a standard fine-tuning script, it has been designed for high efficiency and straightforward integration. The methodology is intended to serve as a principal approach for domain specialization, analogous to the role of LoRA in parameter-efficient tuning. The practical implementation can be decomposed into two distinct stages.

The first stage is a one-time pre-computation of the global gradient $\hat{g}_G$, on general-domain data. This process is analogous to a standard training procedure but with the learning rate set to zero, representing a single, non-recurring computational cost. A critical feature of this approach is its reusability. The resulting gradient artifact is model-specific yet domain-agnostic, meaning that for a given foundation model like Llama3.1-8B, this computation is performed only once. The same global gradient can then be applied to fine-tuning tasks across any number of vertical domains, such as finance, law, or medicine. We propose that this pre-computation could become a standard practice, wherein foundation model developers release an official global gradient alongside model weights, leveraging their high-quality pre-training data. Such a community-driven effort would obviate this step entirely for downstream domain specialists.

The second stage is the integration of Global Gradient Anchoring (GGA) and Dynamic Correction (DC) into the fine-tuning loop. To facilitate seamless adoption, we have implemented our method within the LLaMA-Factory and DeepSpeed frameworks. We will release this implementation as open-source code and submit pull requests to these upstream projects, allowing practitioners to enable DAA via a simple command-line argument with minimal implementation overhead.

In summary, the initial setup cost of DAA is substantially offset by the elimination of repeated data mixing and extensive hyperparameter tuning. This modest, one-time investment yields significant and recurring savings in computational resources and engineering time during the iterative process of domain adaptation. A "Practical Implementation Guide" is provided in the Appendix to further detail these steps and emphasize the long-term efficiency benefits.

E Significance Test

As shown in Tab. 11, we conducted a rigorous significance test using the News QA benchmark on Llama3.1-8B. We chose this setup as it represents a challenging temporal OOD task. We repeated the experiments for both Vanilla FT (at its optimal mixture ratio) and DAA using 5 different random seeds $\{42, 43, 44, 45, 46\}$.

An independent t-test yields a p-value $\ll 0.001$, confirming that the performance improvement is significant and robust to random initialization.

Sample Ratio	**20% (Ours)**	40%	60%	80%	100%
S	**63.76**	63.55	63.68	63.71	63.12

Table 13: Performance sensitivity to general data sample ratio.

Anchor Source	S_D	S_G	S
General SFT Data (Ours)	82.37	52.00	63.76
Post-Pretrain Data	**82.51**	**53.85**	**65.17**

Table 14: Performance comparison of different anchor sources.

DAA also shows lower variance ($\sigma = 0.42$) compared to Vanilla FT ($\sigma = 0.76$), demonstrating that the Global Gradient Anchor effectively stabilizes the optimization process by reducing the impact of initialization randomness.

F Cost Analysis

We present the training overhead of DAA and Direct Fine-tuning (FT) for Llama3.1-8B on the News QA dataset using 16 A100 GPUs. As shown in Tab. 12, DAA's low-rank gradient anchors occupy 5.5 GB of memory. Relative to Direct Fine-tuning (FT), DAA exhibits a modest 0.7 GB increase in Peak VRAM, due to the sharded gradient anchors and the computational cost for gradient correction. Owing to the fact that the anchor gradients are loaded only once, the wall-clock time increased slightly. Crucially, DAA does not introduce additional memory or I/O overhead for *storage* and *repeated use* during training steps, as anchors are pre-loaded into VRAM. Crucially, DAA does not introduce additional memory or I/O overhead for *storage* and *repeated use* during training steps, as anchors are pre-loaded into VRAM.

G Additional Sensitivity Analysis

G.1 Sensitivity to General Data

The general dataset used in our paper was filtered from a raw pool of approximately 2.4M samples using the Platypus method. To evaluate sensitivity, we sampled various ratios from this 2.4M raw pool. As shown in Tab. 13, performance (S) remains remarkably stable as data volume increases. This confirms that $\hat{g}_G$ is robust and does not require massive amounts of raw SFT data to be effective. This aligns with findings from recent studies like LIMA (Zhou et al., 2023), suggesting that data quality and curation strategy outweigh sheer quantity. Once the general dataset achieves sufficient diversity to cover broad capabilities, the direction of $\hat{g}_G$ stabilizes, and the marginal utility of adding more data diminishes. Consequently, the cost of curating this data recipe is a one-time effort. We view $\hat{g}_G$ as a standard "Model Artifact" as critical as model weights, and advocate for developers to release these gradients to eliminate data collection costs for end-users.

G.2 Anchor Source

Deriving $\hat{g}_G$ from pre-training or post-pretraining data yields even better results, as these gradients align more closely with the base model's intrinsic distribution. We conducted a preliminary experiment where we curated 100B tokens of post-pretraining data to compute $\hat{g}_G$, comparing it against the anchor derived from standard SFT data used in our main paper. As shown in Tab. 14, the anchor derived from post-pretraining data achieved a superior joint performance score (S) on the News QA task.

H News QA Robustness Splits

To address the concern that gains on News QA may be artifacts of the LLM-generated label distribution, we report DAA vs. Direct FT on Llama3.1-8B over (i) topic splits and (ii) temporal-distance splits (months elapsed since the December 2024 pre-training cut-off). As shown in Tab. 15, DAA improves S uniformly across topics and time bins, indicating that the gain is not concentrated on a particular slice.

I Anchor-Staleness Analysis

We empirically validate the claim in Sec. 3.1 that $\hat{g}_G$ remains a good first-order approximation of the time-varying general-domain gradient. Every 200 steps during fine-tuning of Llama3.1-8B on

Split	Direct FT S	DAA S	Δ
Topic split			
Politics	49.12	62.38	+13.26
Economics	51.44	64.10	+12.66
Culture	50.81	64.85	+14.04
Temporal-distance split (months since pre-training cut-off)			
0–3 months since cut-off	52.07	64.61	+12.54
3–6 months since cut-off	50.32	63.82	+13.50
≥6 months since cut-off	49.18	62.94	+13.76

Table 15: News QA topic and temporal-distance splits on Llama3.1-8B. The temporal-distance rows report the number of months between the article's publication date and the December 2024 pre-training cut-off (e.g., "0–3 months since cut-off" covers articles published within three months after December 2024).

News QA, we recompute the instantaneous general gradient $g_G(\theta_t)$ on a held-out general batch and measure its cosine similarity with the fixed anchor $\hat{g}_G$. The similarity stays in $[0.91, 0.96]$ across the full training run, despite a parameter displacement $\|\theta_t - \theta_0\|$ that grows roughly linearly with t. This is consistent with the SVRG- and L2-SP-style locality argument: as long as θ_t stays within a bounded trust region of θ_0, $\hat{g}_G$ remains a faithful approximation, and the multiplier γ_t further attenuates any residual mismatch.

J Component-Isolation Ablation on Qwen3 and SVD Rank

Tab. 16 reports the same component-isolation ablation as Tab. 4 on Qwen3-8B / News QA, completing the per-base-model breakdown referenced in the main text. Tab. 17 sweeps the SVD rank r used to compress the anchor: rank 512 is the elbow point, retaining 99.7% of the spectral energy of the full-rank anchor and recovering performance within 0.05 of the uncompressed gradient on S. Tab. 18 reports the peak VRAM scaling of the anchor up to 70B parameters.

AL	GGA	DC	S_D	S_G	S
✔	✗	✗	75.50	44.06	55.65
✗	✔	✗	79.84	56.10	65.92
✗	✔	✔	80.36	63.95	71.22
✔	✔	✗	78.62	47.90	59.53
✔	✗	✔	78.73	57.26	66.30
✔	✔	✔	**80.82**	**68.52**	**74.16**

Table 16: Component-isolation ablation on Qwen3-8B / News QA.

Rank r	64	128	256	512	1024	Full
S on News QA	60.18	62.45	63.51	**63.76**	63.79	63.81
Spec. energy	0.83	0.92	0.97	0.997	0.999	1.00

Table 17: SVD rank sensitivity for the compressed global anchor (Llama3.1-8B / News QA).

Model	Direct FT peak	Anchor	Extra peak	Extra/Peak
Llama3.1-8B	37.1 GB	5.5 GB	+0.7 GB	1.9%
Qwen2.5-32B	142.3 GB	14.8 GB	+2.6 GB	1.8%
Llama3.1-70B	310.2 GB	28.6 GB	+5.4 GB	1.7%

Table 18: Anchor memory and peak VRAM scaling. Extra peak VRAM stays below 2% of training peak even at 70B.

K Raw Experimental Results

As shown in Tab. 19, we provide the raw scores for experiments on general benchmarks here.

Domain	Method	Llama3.1									Phi4									Qwen3								
		$T\downarrow$	$S_D\uparrow$	S_{MMLU}	$S_{MMLU-Pro}$	S_{GSM8k}	S_{MATH}	S_{M3Exam}	$S_G\uparrow$	$S\uparrow$	$T\downarrow$	$S_D\uparrow$	S_{MMLU}	$S_{MMLU-Pro}$	S_{GSM8k}	S_{MATH}	S_{M3Exam}	$S_G\uparrow$	$S\uparrow$	$T\downarrow$	$S_D\uparrow$	S_{MMLU}	$S_{MMLU-Pro}$	S_{GSM8k}	S_{MATH}	S_{M3Exam}	$S_G\uparrow$	$S\uparrow$
Finance	Direct FT	3.40	**80.01**	46.67	48.90	55.38	56.99	65.50	54.69	64.97	6.22	**89.72**	65.57	66.63	80.00	85.12	92.27	77.92	**83.41**	3.38	**85.83**	52.89	59.37	59.00	70.24	77.19	63.74	73.15
	Vanilla FT	38.77	79.30	47.57	**58.49**	60.73	66.85	68.87	60.50	68.64	71.24	83.42	62.61	72.09	74.18	**88.65**	91.02	77.71	80.46	38.74	85.49	59.41	65.42	71.52	73.85	84.84	71.01	77.58
	LoRA	**2.50**	76.45	50.25	54.01	61.99	**71.10**	71.10	61.69	68.28	**4.61**	87.13	67.83	67.28	75.52	84.52	96.20	78.27	82.46	**2.50**	81.68	58.42	62.18	79.99	**78.43**	87.04	73.21	77.21
	DoRA	2.58	76.12	47.83	54.85	58.60	67.60	**77.38**	61.25	67.88	4.71	86.34	56.99	67.30	**82.48**	84.09	**100.00**	78.24	82.09	2.57	82.19	62.19	63.02	74.98	76.41	**89.76**	73.27	77.47
	Galore	5.09	77.31	49.93	56.04	55.89	68.25	73.79	60.78	68.06	9.34	86.23	**68.14**	73.86	73.35	83.67	93.73	**78.55**	82.21	5.09	84.37	60.78	69.49	74.68	78.29	89.72	74.59	79.18
	DAA (Ours)	3.45	79.84	**52.13**	51.00	**63.42**	64.97	77.23	**61.75**	**69.64**	6.35	87.73	61.89	**76.79**	74.85	85.24	93.73	78.50	82.86	3.41	85.32	**63.33**	**72.89**	**80.19**	77.96	88.09	**76.49**	**80.66**
Medicine	Direct FT	12.39	**89.23**	46.59	48.86	49.50	55.42	63.28	52.73	66.29	22.78	92.13	62.59	68.46	81.35	82.44	96.86	78.34	84.68	12.38	**92.67**	48.56	54.00	66.90	71.53	81.20	64.44	76.02
	Vanilla FT	141.81	87.32	45.02	**54.23**	58.12	**68.25**	71.73	59.47	**70.75**	260.58	91.24	59.32	71.71	**86.06**	85.68	93.53	**79.26**	84.83	141.81	81.81	51.22	61.47	65.90	76.68	88.43	68.74	74.71
	LoRA	**9.18**	81.76	47.52	51.10	**60.80**	67.61	72.83	59.97	69.19	**16.87**	90.30	**68.29**	69.54	78.50	86.56	92.21	79.02	84.28	**8.21**	84.00	55.69	65.32	65.18	**81.65**	79.71	69.51	76.07
	DoRA	9.31	81.21	45.40	54.15	60.67	65.61	73.29	59.07	68.40	17.05	90.31	60.76	72.28	82.27	84.50	93.89	78.74	84.13	8.37	84.74	59.14	66.57	63.86	80.46	78.32	69.67	76.47
	Galore	13.68	81.23	44.21	53.48	57.66	65.68	71.72	58.55	68.05	25.14	91.96	58.08	65.48	77.96	**91.53**	**99.77**	78.56	84.73	13.68	87.57	**63.91**	63.44	68.90	77.36	**94.93**	73.71	80.04
	DAA (Ours)	12.54	83.97	**48.85**	54.15	57.38	66.91	**74.37**	**60.33**	70.22	22.90	**92.61**	60.39	**74.66**	76.43	88.61	94.01	79.26	**85.16**	12.64	92.24	62.67	**75.23**	**75.17**	**89.45**	87.33	**77.97**	**84.51**
Law	Direct FT	4.39	**56.81**	40.27	44.22	47.20	56.13	56.88	48.94	**52.58**	8.04	**42.63**	57.94	65.11	64.46	82.20	85.59	71.06	53.29	4.39	**55.28**	58.46	62.10	70.84	74.57	84.68	70.13	61.83
	Vanilla FT	50.12	51.37	43.89	48.69	52.03	57.33	63.46	53.08	52.21	91.98	41.82	60.67	60.57	68.52	84.29	86.55	72.12	52.94	50.10	52.28	56.62	64.67	76.19	84.29	82.98	72.95	60.91
	LoRA	**3.25**	46.58	46.71	**54.68**	58.21	**65.42**	65.29	**58.06**	51.69	**5.97**	41.87	61.18	63.69	**71.58**	**86.92**	83.53	**73.38**	**53.32**	**3.25**	51.90	58.00	69.69	72.83	**88.14**	92.29	76.19	61.74
	DoRA	3.43	46.37	40.33	51.54	**59.77**	62.54	66.07	56.05	50.75	6.24	41.98	54.59	67.79	67.54	84.09	87.79	72.36	53.13	3.36	51.91	63.23	69.97	73.75	84.50	90.05	76.30	61.79
	Galore	6.13	47.80	**48.74**	54.16	54.99	59.29	63.48	56.13	51.63	11.26	40.12	53.81	**69.17**	70.79	81.86	82.02	71.53	51.41	6.13	52.76	59.69	**72.76**	79.33	83.11	**92.76**	77.53	62.79
	DAA (Ours)	4.53	49.93	42.56	52.79	53.99	64.28	**69.78**	56.68	53.09	8.17	41.95	**61.46**	66.30	69.39	78.05	**90.41**	73.12	53.31	4.60	52.79	**66.39**	68.63	**84.35**	86.94	90.59	**79.38**	**63.41**
News QA	Direct FT	1.27	**82.38**	30.50	34.28	35.56	40.76	40.60	36.34	50.44	2.32	87.38	20.21	25.35	27.52	27.33	31.59	26.40	40.55	1.27	79.37	3.06	3.54	3.95	4.24	4.36	3.83	7.31
	Vanilla FT	14.45	80.62	34.62	37.22	39.19	42.13	45.48	39.73	53.22	16.51	88.23	59.90	59.66	71.83	80.72	88.44	72.11	79.36	14.45	80.27	42.19	43.08	51.63	60.06	64.84	52.36	63.38
	LoRA	**0.94**	71.23	37.79	45.95	48.14	50.48	57.84	48.04	57.38	**1.72**	83.12	**62.67**	66.96	**81.24**	84.03	92.40	77.46	80.19	**0.93**	70.27	**55.48**	57.80	66.11	76.27	**81.90**	67.51	68.86
	DoRA	0.99	73.71	38.80	**48.29**	50.23	58.30	56.99	50.52	59.95	1.80	83.37	62.47	68.79	73.19	80.52	**98.51**	76.68	79.88	0.99	70.78	**55.48**	64.78	65.15	72.94	80.69	67.81	69.26
	Galore	2.02	79.13	**42.95**	46.20	53.13	**58.51**	**64.45**	**53.05**	63.52	3.70	85.02	60.55	66.79	74.46	**89.89**	89.66	76.27	80.41	2.03	**80.88**	51.70	59.68	66.43	**83.20**	81.14	68.43	74.14
	DAA (Ours)	1.32	82.37	40.84	44.16	**54.54**	56.67	63.80	52.00	**63.76**	2.42	**89.19**	59.08	**71.97**	77.04	85.05	96.41	**77.91**	**83.17**	1.28	80.82	51.32	**65.66**	**67.81**	76.73	81.08	**68.52**	**74.16**

Table 19: Raw scores for main comparison experiments.