FINETUNE ONCE: DECOUPLING GENERAL & DOMAIN LEARNING WITH DYNAMIC BOOSTED ANNEALING

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) fine-tuning shows excellent implications. However, vanilla fine-tuning methods often require intricate data mixture and repeated experiments for optimal generalization. To address these challenges and streamline the training process, we propose an efficient and universal solution, Dynamic Boosted Annealing (DBA). We obtain a global gradient through zero-learning-rate training on general data, which is subsequently employed for gradient boosting and dynamic training step correction during domain training. In conjunction with annealing learning, we end up establishing a fine-tuning pipeline that relies solely on domain data without collapse. By evaluating both general and domain-specific performance across multiple tasks on several popular base models, DBA achieves an average improvement of 5.8% in joint performance over vanilla fine-tuning. Furthermore, since general data is no longer involved in annealing, repeated experiments led by data mixture are also eliminated. According to our tests, the DBA method can reduce GPU hours by 91.0% compared to the vanilla method.

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 in the fine-tuning process has emerged as a crucial research focus for LLMs (Table 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 (Table 1, row 2). As shown in Figure 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 (Table 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 Boosted Annealing (DBA)**, 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 Boosted learning (GGB)**. Here, "boosted" refers to augmenting the domain-specific gradient with a pre-computed global one, rather than sequentially fitting models to residuals as in methods like XGBoost. This method initially estimates the global gradient in the general

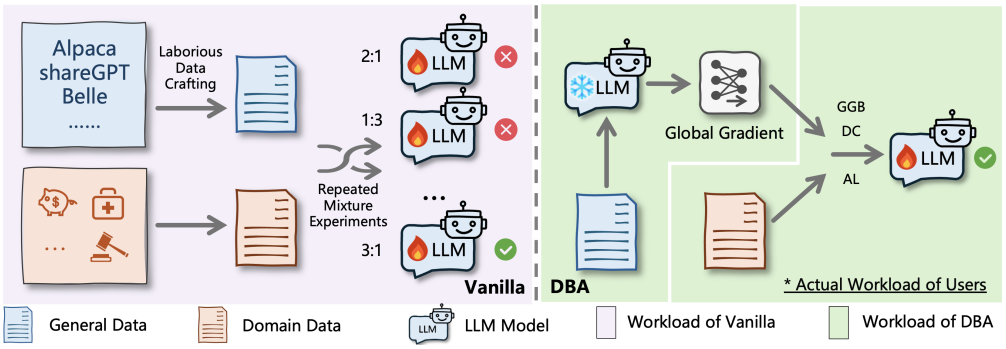


Figure 1: Comparison between vanilla and DBA. [*] is the part that users need to perform in SFT.

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

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

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. As demonstrated in Table 1, DBA 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 DBA, 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 work on fine-tuning 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) including direct fine-tuning Xie et al. (2023); Zhu et al. (2024), vanilla fine-tuning Bao et al. (2023); Yue et al. (2023); Chen et al. (2024); Deng et al. (2023) and LoRA Yang et al. (2023); Chen et al. (2023); Cui et al. (2023) seeks to mitigate catastrophic forgetting while controlling cost. As shown in Table 1, direct fine-tuning is inexpensive yet the general performance of LLMs can collapse. Vanilla fine tuning employs data mixtures to suppress forgetting and often preserves general capability, although the cost rises sharply. LoRA is effective in reducing both forgetting and cost, but performance in unfamiliar specific domains remains below that of full-parameter fine-tuning.

Among these options, vanilla fine-tuning with data mixture offers the best balance between general and domain-specific performance, yet its experimental cost is substantial. Mixing ratios tend to be

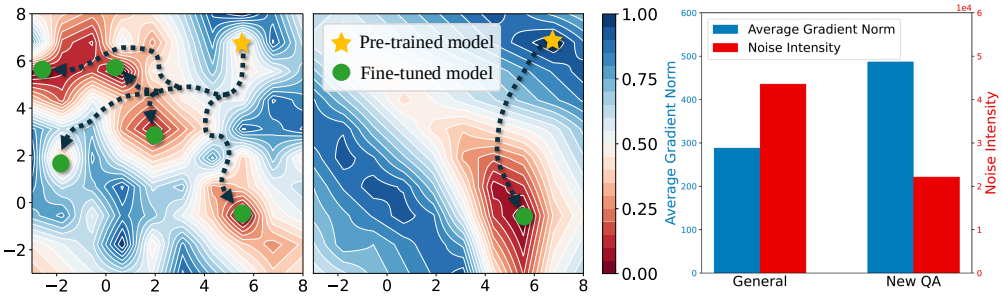
domain- dependent and therefore require repeated experimentation for each target domain Wen et al. (2023). In addition, when the ratio is swept from $1:1$ through $1:N$, the total volume of processed data scales as $\sum_{n=1}^{N}(1+n) = O(N^2)$ times the size of the specific domain set, which becomes prohibitive as the domain dataset grows. This computational inefficiency motivates more efficient and more generalizable fine-tuning methodology.

Another related area of research is Continual Learning (CL). CL is defined as a model learning from a dynamic data distribution Wen et al. (2023). Our setting can be viewed as single task continual learning in which, after adapting a pretrained model to one instruction task, we aim to mitigate degradation of its general capabilities.

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, we qualitatively analyze the differences in convergence trajectory between general and specific domain by visualizing the loss landscape. Following the methodology in Lucas et al. (2021), we interpolate between the weights θ_0 of Qwen3-1.7B Yang et al. (2025) base model and the fully fine-tuned weights θ_D, constructing a two-dimensional slice of the loss landscape. To ensure independence, we apply orthogonalization to the interpolation direction.



(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(a) and 3(b) show that the general domain has many local optima and a tortuous path details in appendix B, while the specific domain news QA details in Section 4.2 shows fewer optima and a more stable trajectory. General domain training can therefore constrain domain specific fine-tuning.

Second, we follow stochastic optimization methodology Ghadimi & Lan (2013) to compare average gradient norms and noise scale across the two domains. As shown in Figure 3(c), the general domain has nearly twice the noise scale of the specific domain. We randomly sample 1,000 instances from each domain. On the full general domain the gap may be larger. We therefore attempt to freeze general domain gradients to limit their impact on training.

In multi domain optimization, conflicts between domain gradients degrade efficiency Yu et al. (2020); Hadsell et al. (2020); Liu et al. (2021). Prior work reduces negative interactions by removing projection components between domain gradients Yu et al. (2020) or by automatic gradient balancing Liu et al. (2021). This motivates a balancing mechanism between specific and general domain gradients that preserves generalization while learning specific domain distributions.

3 Method

In this section, we formally introduce the Dynamic Boosted Annealing (DBA) illustrated in Figure 3, which is based on annealing learning. Initially, the global gradient is independently estimated in the general domain through zero-learning-rate learning. During the fine-tuning stage, DBA boosts the

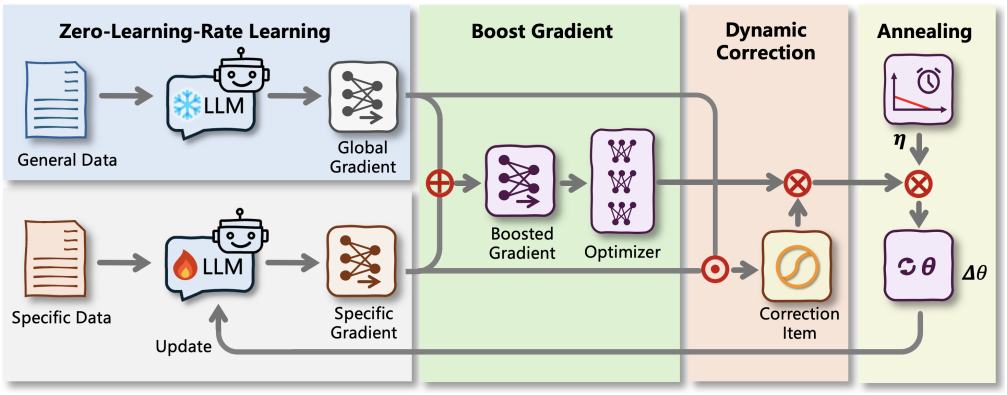


Figure 3: Overview of Dynamic Boosted Annealing. Our approach consists of two stages. In the first stage, global gradient is estimated in the general domain through zero-learning-rate learning, which serves as an independent preprocessing stage. In the second stage, the fine-tuning step, global gradient boosts 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.

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 Boosted Learning

The global gradient serves as a stable optimization anchor rather than an instantaneous mean gradient on the general dataset. Our design is inspired by Johnson & Zhang (2013), which shows that a fixed global average gradient can reduce stochastic gradient variance and remains effective with update intervals up to five epochs. Using a fixed global gradient over a single epoch of fine tuning thus acts as a practical regularizer. The term "boosted" is metaphorical. We boost or augment the domain-specific gradient at each step with this pre-computed stable anchor. This approach is distinct from traditional Gradient Boosting Machines (e.g. XGBoost) that sequentially fit models to residuals.

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

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

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 boosted magnitude. The expectation and variance of $g_{B,t}$ are given by

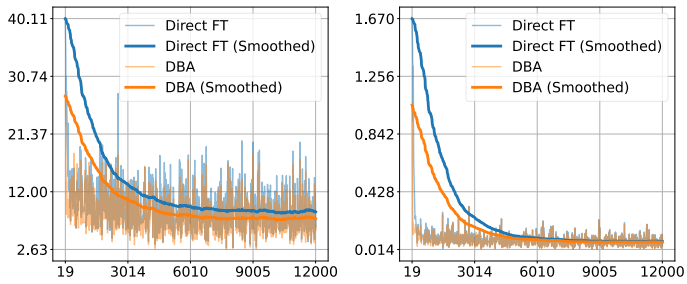
$$\mathbb{E}[g_{B,t}] = \gamma_t \hat{g}_G + (1-\gamma_t)\mathbb{E}[g_{D,t}], \tag{3}$$

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

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 & 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.

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



(a) Gradient norm (b) Loss

Figure 4: Comparative results of global gradient boosted learning.

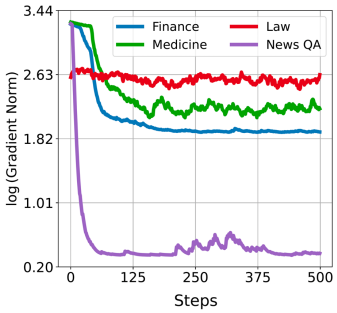


Figure 5: Norm of gradient projection.

Therefore, we trained the LLM on the general domain with a learning rate of 0 and a decay rate $\beta_1 \to 1$, then stored the final momentum after s training steps. Notably, $\hat{g}_G$ can be applied across all domains, rather than being obtained per domain.

We refer to this method as Global Gradient Boosted learning (GGB). 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 Figure 4, GGB 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 boosted magnitude can be adapted over training. We set $\gamma_t = k_0\left(1 - \frac{t}{T}\right)$, so that the model increasingly emphasizes the specific domain near the end of fine tuning. Accounting for exponential averaging in Adam Kingma & 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 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.

3.2 Dynamic Correction

Pre-trained language models have demonstrated remarkable capabilities by incorporating data from diverse domains during pre-training. However, these models often struggle with domains that are either private or temporally distinct from the pre-training distribution, necessitating extensive experiments for optimal performance. Applying uniform strategies across domains with varying degrees of familiarity can lead to suboptimal outcomes. To address this challenge, we propose a Dynamic Correction (DC) mechanism that modulates the magnitude of parameter update based on gradient similarity.

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 Figure 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 resulting 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:

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

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

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

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

3.4 Summary

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

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

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

4 Experiment

4.1 Experiment Settings

This study evaluates the effectiveness of DBA across diverse vertical domains in both English and Chinese contexts, including finance, medicine, and law. The general-domain data used in our experiments comprises Chinese and English corpora covering multiple tasks. Its detailed composition can be found in Appendix B, Table 6. The evaluation utilizes multiple datasets: FinGPT Yang et al. (2023), CMB Wang et al. (2024) and Fuzi-Mingcha Deng et al. (2023). To avoid the potential contamination or overfitting of evaluation benchmarks during pre-training as new and improved LLMs are developed Schaeffer (2023); Jain et al. (2024); Zhang et al. (2024b), we constructed a temporal out-of-distribution (OOD) evaluation benchmark named News QA (details in Section 4.2).

For comparative analysis, we selected several representative fine-tuning methods. In addition to direct fine-tuning and vanilla fine-tuning, we also compared the performance of LoRAHu et al.

(2021), DoRALiu et al. (2024), GaloreZhao et al. (2024), and our proposed DBA across diverse vertical domains. Especially, for vanilla fine-tuning, we followed Wen et al. (2023) and combined our vertical domain fine-tuning experience to choose three distinct data mixture ratios (specific data : general data = 1:1, 1:3, 1:5). We ensured that the vanilla fine-tuning results presented in the experimental tables all represent the optimal performance in the specific domain.

In addition, we have validated the effectiveness of the proposed method across multiple 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 constructed a benchmark comprising QA pairs extracted from news articles. As shown in Table 2, the dataset contains 30,613 news titles across three categories (Politics, Economics, and Culture), with corresponding true/false questions designed to evaluate factual verification capabilities. The task requires binary responses ("true" or "false") for each statement. To ensure minimal overlap with foundational models' pre-training corpus Yang et al. (2024); Grattafiori et al. (2024); Abdin et al. (2024), we specifically selected news articles published after December 2024. As shown in Table 2, row 3, all the foundational models exhibits limited factual verification capabilities , achieving only 31.06% average accuracy.

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 Yang et al. (2025) on each category.

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

4.3 Metrics

To evaluate the general performance of the models, we selected four benchmarks commonly used across all LLMs: MMLU Hendrycks et al. (2021a), MMLU-Pro, GSM8K Cobbe et al. (2021), MATH Hendrycks et al. (2021b) and M3Exam Zhang et al. (2023a). MMLU tests general knowledge across multiple subjects, CMMLU focus on Chinese-specific knowledge and reasoning, while GSM8K and MATH tests mathematical problem-solving skills. To facilitate calculation, we normalized the changes in general performance across all studies within the same vertical domain. To evaluate various vertical performance of the models, we selected suitable public benchmarks for evaluation. For the financial domain, we utilized the weighted F1 score average across the English FPB Malo et al. (2013), FiQA Maia et al. (2018), TFNS Zer (2024), and NWGI Yang (2024) financial sentiment analysis test sets as the metric for this domain. For the medical domain, the accuracy score from the Chinese CMB-Exam Wang et al. (2024) test set served as the domain-specific metric. For the legal domain, we employed Chinese LawBench Fei et al. (2023) for a comprehensive evaluation. Our 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 $\mathcal{X} = \{\text{MMLU, MMLU-Pro, GSM8k, MATH, M3Exam}\}$. S_D is the normalized score of the model's vertical domain performance, S_G is the average normalized 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. Conversely, if it enhances domain performance while preserving general capability, the score approaches 1.

4.4 Cost Analysis

To demonstrate the efficiency of our approach, we compared GPU hours across fine-tuning methods using 16 Nvidia A100 GPUs. As shown in Table 3, DBA requires $T_{\text{DBA}} \approx$

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

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	2.09	**80.01**	54.69	64.97	3.83	**89.72**	77.92	**83.41**	2.08	**85.83**	63.74	73.15
	Vanilla FT	23.86	79.30	60.50	68.64	43.84	83.42	77.71	80.46	23.85	85.49	71.01	77.58
	LoRA	**1.54**	76.45	61.69	68.28	**2.84**	87.13	78.27	82.46	**1.54**	81.68	73.21	77.21
	DoRA	1.59	76.12	61.25	67.88	2.90	86.34	78.24	82.09	1.58	82.19	73.27	77.47
	Galore	3.13	77.31	60.78	68.06	5.75	86.23	**78.55**	82.21	3.13	84.37	74.59	79.18
	DBA (Ours)	2.12	79.84	**61.75**	**69.64**	3.91	87.73	78.50	82.86	2.10	85.32	**76.49**	**80.66**
Medicine	Direct FT	7.63	**89.23**	52.73	66.29	14.02	92.13	78.34	84.68	7.62	**92.67**	64.44	76.02
	Vanilla FT	87.27	87.32	59.47	**70.75**	160.36	91.24	**79.26**	84.83	87.27	81.81	68.74	74.71
	LoRA	**5.65**	81.76	59.97	69.19	**10.38**	90.30	79.02	84.28	**5.05**	84.00	69.51	76.07
	DoRA	5.73	81.21	59.07	68.40	10.49	90.31	78.74	84.13	5.15	84.74	69.67	76.47
	Galore	8.42	81.23	58.55	68.05	15.47	91.96	78.56	84.73	8.42	87.57	73.71	80.04
	DBA (Ours)	7.72	83.97	**60.33**	70.22	14.09	**92.61**	78.82	**85.16**	7.78	92.24	**77.97**	**84.51**
Law	Direct FT	2.70	**56.81**	48.94	52.58	4.95	**42.63**	71.06	53.29	2.70	**55.28**	70.13	61.83
	Vanilla FT	30.84	51.37	53.08	52.21	56.66	41.82	72.12	52.94	30.83	52.28	72.95	60.91
	LoRA	**2.00**	46.58	**58.06**	51.69	**3.67**	41.87	**73.38**	**53.32**	**2.00**	51.90	76.19	61.74
	DoRA	2.11	46.37	56.05	50.75	3.84	41.98	72.36	53.13	2.07	51.91	76.30	61.79
	Galore	3.77	47.80	56.13	51.63	6.93	40.12	71.53	51.41	3.77	52.76	77.53	62.79
	DBA (Ours)	2.79	49.93	56.68	**53.09**	5.03	41.95	73.12	53.31	2.83	52.79	**79.38**	**63.41**
News QA	Direct FT	0.78	**82.38**	36.34	50.44	1.43	87.38	26.40	40.55	0.78	79.37	3.83	7.31
	Vanilla FT	8.89	80.62	39.73	53.22	10.16	88.23	72.11	79.36	8.89	80.27	52.36	63.38
	LoRA	**0.58**	71.23	48.04	57.38	**1.06**	83.12	77.46	80.19	**0.57**	70.27	67.51	68.86
	DoRA	0.61	73.71	50.52	59.95	1.11	83.37	76.68	79.88	0.61	70.78	67.81	69.26
	Galore	1.24	79.13	**53.05**	63.52	2.28	85.02	76.27	80.41	1.25	**80.88**	68.43	74.14
	DBA (Ours)	0.81	82.37	52.00	**63.76**	1.49	**89.19**	**77.91**	**83.17**	0.79	80.82	**68.52**	**74.16**

4.2 GPU-hours per domain, similar to direct full-tuning ($T_{\text{Direct}} \approx 4.1$ GPU-hours) but without its drop in general-task performance. More importantly, DBA reduces training costs by over 90% compared to vanilla fine-tuning ($T_{\text{Vanilla}} \approx 46.7$ GPU-hours) while achieving notable gains in vertical (S_D) and general (S_G) scores.

While LoRA ($T_{\text{LoRA}} \approx 3.0$ GPU-hours) and DoRA ($T_{\text{DoRA}} \approx 3.1$ GPU-hours) are faster, DBA consistently outperforms them and Galore ($T_{\text{Galore}} \approx 5.1$ GPU-hours) in harmonic-mean score S, justifying the modest additional GPU time with superior task performance.

4.5 Main Results

Table 3 demonstrates that our DBA method consistently achieves the best balance between vertical domain ability and general performance retention, as measured by the harmonic mean S, across four domains and three base models.

Finance. On Llama3.1, DBA again leads with $S_D = 79.84\%$, $S_G = 61.75\%$ and $S = 69.64\%$, surpassing all competitors. On Qwen3, direct fine-tuning and vanilla fine-tuning suffer considerable general-performance drops ($S_G = 63.74\%$ and 71.01%), whereas DBA attains $S_G = 76.49\%$ (an improvement of 1.90 points over the next best) while maintaining a high domain score $S_D = 85.32\%$. This yields the highest overall score $S = 80.66\%$, outperforming direct fine-tuning ($S = 73.15\%$) and vanilla fine-tuning ($S = 77.58\%$).

Medicine. On Llama3.1, DBA's $S_G = 60.33\%$ and $S_D = 83.97\%$ produce $S = 70.22\%$, again the best trade-off. On Phi4, DBA secures the highest domain accuracy ($S_D = 92.61\%$) and a strong general score ($S_G = 78.82\%$), leading to an overall $S = 85.16\%$, which exceeds every baseline. For Qwen3, direct fine-tuning and vanilla fine-tuning obtain only $S_G = 64.44\%$ and 68.74%, while DBA achieves $S_G = 77.97\%$ coupled with $S_D = 92.24\%$, resulting in the top harmonic mean $S = 84.51\%$.

Law. DBA attains the high S_G on Llama3.1 (56.68%), Phi4 (73.12%), and Qwen3 (79.38%), and achieves harmonic means 53.09%, 53.31%, and $S = 63.41\%$ respectively. These results outperform direct fine-tuning and vanilla fine-tuning, both of which incur larger general-performance regressions. A more detailed discussion regarding the performance on the Law dataset is provided in the Appendix C.

Table 4: Results of ablation on news QA benchmark. AL, GGB and DC are defined in Section 3.

AL	GGB	DC	Llama3.1			Phi4			Qwen3		
			$S_D \uparrow$	$S_G \uparrow$	$S \uparrow$	$S_D \uparrow$	$S_G \uparrow$	$S \uparrow$	$S_D \uparrow$	$S_G \uparrow$	$S \uparrow$
✔	✗	✗	68.22	27.60	39.30	78.07	46.47	58.26	75.50	44.06	55.65
✔	✔	✗	71.25	31.36	43.55	82.14	50.34	62.42	78.62	47.90	59.53
✔	✗	✔	72.99	41.66	53.04	83.35	61.49	70.77	78.73	57.26	66.30
✔	✔	✔	**75.65**	**47.09**	**58.05**	**86.24**	**67.36**	**75.64**	**80.82**	**68.52**	**74.16**

News QA. In this strictly leak-free benchmark, direct fine-tuning collapses on general performance ($S_G = 3.83\%$ on Qwen3), whereas DBA preserves general knowledge ($S_G = 68.52\%$) while matching—indeed slightly exceeding—direct fine-tuning's domain score ($S_D = 80.82\%$ vs. 79.37%), yielding $S = 74.16\%$ (versus 7.31%). On Phi4, DBA simultaneously achieves the highest $S_D = 89.19\%$ and $S_G = 77.91\%$, leading to $S = 83.17\%$, which outperforms the best baseline by 2.76 points.

Overall, across all domains and models, DBA delivers the strongest joint performance S, validating its effectiveness at vertical domain fine-tuning with minimal general-knowledge degradation.

4.6 Ablation Analysis

Furthermore, we conducted ablation studies on individual modules within the proposed DBA to quantify their contributions. The results are shown in Table 4. When solely applying annealing learning (row 1), the model shows decreased domain-specific performance and improved general domain performance, yet fails to match the overall effectiveness of DM. This indicates that while the annealing strategy helps mitigate catastrophic forgetting, its effectiveness is limited in isolation. The incorporation of global gradient boosted learning with annealing learning (row 2) leads to enhanced performance in both domain-specific and general domains, demonstrating the significant impact of global gradient optimization.

Incorporating dynamic correction into annealing learning (row 3) leads to significant improvements in both domain-specific and general domain performance. This demonstrates that dynamic correction effectively optimizes the update step size, thereby enhancing the learning process. The combination of all three components (row 4) - annealing learning, global gradient boosted learning, and dynamic correction - yields optimal performance across both domains, achieving highest joint performance S of 58.05%, 75.64%, and 74.16% on Llama3.1, Phi4, and Qwen3. These results validate the synergistic effects of DBA components in enhancing the model's overall capabilities.

5 Conclusion

We present Dynamic Boosted Annealing, a fine-tuning method that mitigates catastrophic forgetting in LLMs. Using global gradient boosted learning with similarity guided dynamic correction, DBA improves performance while reducing compute cost over prior methods.

Limitations. DBA is designed for dense models used in vertical domain tasks. Our experiments cover a few domains such as medical and finance. Robustness across vision, speech, reinforcement learning, continual fine-tuning, and large scale language modeling remains unverified. Although DBA scales linearly in theory, extremely deep or wide networks with billions of parameters and web scale datasets may reveal stability or convergence issues not seen in our mid scale benchmarks.

Applicability Analysis. DBA relies on gradient boosted learning and magnitude adjustment of parameter updates, so it applies to other optimizers. In fine-tuning we focus on AdamW Loshchilov & Hutter (2019), which is widely used.

Future Work. Domain specific LLMs can equip workers with specialized AI in their fields. We will explore broader applications of DBA to inspire research on domain specific training. We will release code and associated global gradients, followed by additional global gradients matched to more base models for the community.

REFERENCES

Zeroshot/twitter-financial-news-sentiment · Datasets at Hugging Face. 2024. URL https://huggingface.co/datasets/zeroshot/twitter-financial-news-sentiment.

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

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

Zhijie Bao, Wei Chen, Shengze Xiao, Kuang Ren, Jiaao Wu, Cheng Zhong, Jiajie Peng, Xuanjing Huang, and Zhongyu Wei. DISC-MedLLM: Bridging General Large Language Models and Real-World Medical Consultation. 2023. doi: 10.48550/arXiv.2308.14346. URL http://arxiv.org/abs/2308.14346.

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. HuatuoGPT-II, One-stage Training for Medical Adaption of LLMs. 2024. doi: 10.48550/arXiv.2311.09774. URL http://arxiv.org/abs/2311.09774.

Sanyuan Chen, Yutai Hou, Yiming Cui, Wanxiang Che, Ting Liu, and Xiangzhan Yu. Recall and Learn: Fine-tuning Deep Pretrained Language Models with Less Forgetting. 2020. doi: 10.48550/arXiv.2004.12651. URL http://arxiv.org/abs/2004.12651.

Wei Chen, Qiushi Wang, Zefei Long, Xianyin Zhang, Zhongtian Lu, Bingxuan Li, Siyuan Wang, Jiarong Xu, Xiang Bai, Xuanjing Huang, and Zhongyu Wei. DISC-FinLLM: A Chinese Financial Large Language Model based on Multiple Experts Fine-tuning. 2023. doi: 10.48550/arXiv.2310.15205. URL http://arxiv.org/abs/2310.15205.

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

Jiaxi Cui, Zongjian Li, Yang Yan, Bohua Chen, and Li Yuan. ChatLaw: Open-Source Legal Large Language Model with Integrated External Knowledge Bases. 2023. doi: 10.48550/arXiv.2306.16092. URL http://arxiv.org/abs/2306.16092.

Wentao Deng, Jiahuan Pei, Keyi Kong, Zhe Chen, Furu Wei, Yujun Li, Zhaochun Ren, Zhumin Chen, and Pengjie Ren. Syllogistic Reasoning for Legal Judgment Analysis. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 13997–14009. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.864. URL https://aclanthology.org/2023.emnlp-main.864/.

Zhiwei Fei, Xiaoyu Shen, Dawei Zhu, Fengzhe Zhou, Zhuo Han, Songyang Zhang, Kai Chen, Zongwen Shen, and Jidong Ge. LawBench: Benchmarking Legal Knowledge of Large Language Models. 2023. doi: 10.48550/arXiv.2309.16289. URL http://arxiv.org/abs/2309.16289.

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

Aaron Grattafiori, Abhimanyu Dubey, and Jauhri et al. The Llama 3 Herd of Models. 2024. doi: 10.48550/arXiv.2407.21783. URL http://arxiv.org/abs/2407.21783.

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

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

William Held, Bhargavi Paranjape, Punit Singh Koura, Mike Lewis, Frank Zhang, and Todor Mihaylov. Optimizing Pretraining Data Mixtures with LLM-Estimated Utility, 2025. URL http://arxiv.org/abs/2501.11747.

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

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

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-Rank Adaptation of Large Language Models. 2021. doi: 10.48550/arXiv.2106.09685. URL http://arxiv.org/abs/2106.09685.

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, Ning Ding, Chao Jia, Guoyang Zeng, Dahai Li, Zhiyuan Liu, and Maosong Sun. MiniCPM: Unveiling the Potential of Small Language Models with Scalable Training Strategies. 2024. doi: 10.48550/arXiv.2404.06395. URL http://arxiv.org/abs/2404.06395.

Quzhe Huang, Mingxu Tao, Chen Zhang, Zhenwei An, Cong Jiang, Zhibin Chen, Zirui Wu, and Yansong Feng. Lawyer LLaMA Technical Report. 2023. doi: 10.48550/arXiv.2305.15062. URL http://arxiv.org/abs/2305.15062.

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

Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. *Advances in neural information processing systems*, 26, 2013.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. 2017. doi: 10.48550/arXiv.1412.6980. URL http://arxiv.org/abs/1412.6980.

Tomasz Korbak, Hady Elsahar, German Kruszewski, and Marc Dymetman. Controlling Conditional Language Models without Catastrophic Forgetting. In *Proceedings of the 39th International Conference on Machine Learning*, pp. 11499–11528. PMLR, 2022. URL https://proceedings.mlr.press/v162/korbak22a.html.

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

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

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

Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization, 2019. URL http://arxiv.org/abs/1711.05101.

James Lucas, Juhan Bae, Michael R Zhang, Stanislav Fort, Richard Zemel, and Roger Grosse. Analyzing monotonic linear interpolation in neural network loss landscapes. *arXiv preprint arXiv:2104.11044*, 1, 2021.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. An Empirical Study of Catastrophic Forgetting in Large Language Models During Continual Fine-tuning, 2025. URL http://arxiv.org/abs/2308.08747.

Macedo Maia, Siegfried Handschuh, André Freitas, Brian Davis, Ross McDermott, Manel Zarrouk, and Alexandra Balahur. 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*, pp. 1941–1942. ACM Press, 2018. ISBN 978-1-4503-5640-4. doi: 10.1145/3184558.3192301. URL http://dl.acm.org/citation.cfm?doid=3184558.3192301.

Pekka Malo, Ankur Sinha, Pyry Takala, Pekka Korhonen, and Jyrki Wallenius. Good Debt or Bad Debt: Detecting Semantic Orientations in Economic Texts. 2013. doi: 10.48550/arXiv.1307.5336. URL http://arxiv.org/abs/1307.5336.

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

Xidong Wang, Guiming Hardy Chen, Dingjie Song, Zhiyi Zhang, Zhihong Chen, Qingying Xiao, Feng Jiang, Jianquan Li, Xiang Wan, Benyou Wang, and Haizhou Li. CMB: A Comprehensive Medical Benchmark in Chinese. 2024. doi: 10.48550/arXiv.2308.08833. URL http://arxiv.org/abs/2308.08833.

Cheng Wen, Xianghui Sun, Shuaijiang Zhao, Xiaoquan Fang, Liangyu Chen, and Wei Zou. ChatHome: Development and Evaluation of a Domain-Specific Language Model for Home Renovation. 2023. doi: 10.48550/arXiv.2307.15290. URL http://arxiv.org/abs/2307.15290.

Minghao Wu, Thuy-Trang Vu, Lizhen Qu, and Gholamreza Haffari. Mixture-of-Skills: Learning to Optimize Data Usage for Fine-Tuning Large Language Models, 2024. URL http://arxiv.org/abs/2406.08811.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. BloombergGPT: A Large Language Model for Finance. 2023. doi: 10.48550/arXiv.2303.17564. URL http://arxiv.org/abs/2303.17564.

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

Chaojun Xiao, Xueyu Hu, Zhiyuan Liu, Cunchao Tu, and Maosong Sun. Lawformer: A pre-trained language model for Chinese legal long documents. 2:79–84, 2021. ISSN 2666-6510. doi: 10.1016/j.aiopen.2021.06.003. URL https://www.sciencedirect.com/science/article/pii/S2666651021000176.

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

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

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

Hongyang Yang. AI4Finance-Foundation/FinGPT. 2024. URL https://github.com/AI4Finance-Foundation/FinGPT.

Hongyang Yang, Xiao-Yang Liu, and Christina Dan Wang. FinGPT: Open-Source Financial Large Language Models. 2023. doi: 10.48550/arXiv.2306.06031. URL http://arxiv.org/abs/2306.06031.

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

Shengbin Yue, Wei Chen, Siyuan Wang, Bingxuan Li, Chenchen Shen, Shujun Liu, Yuxuan Zhou, Yao Xiao, Song Yun, Xuanjing Huang, and Zhongyu Wei. DISC-LawLLM: Fine-tuning Large Language Models for Intelligent Legal Services. 2023. doi: 10.48550/arXiv.2309.11325. URL http://arxiv.org/abs/2309.11325.

Biao Zhang, Zhongtao Liu, Colin Cherry, and Orhan Firat. When Scaling Meets LLM Finetuning: The Effect of Data, Model and Finetuning Method, 2024a. URL http://arxiv.org/abs/2402.17193.

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

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

Xuanyu Zhang, Qing Yang, and Dongliang Xu. XuanYuan 2.0: A Large Chinese Financial Chat Model with Hundreds of Billions Parameters. 2023b. doi: 10.48550/arXiv.2305.12002. URL http://arxiv.org/abs/2305.12002.

Jiawei Zhao, Zhenyu Zhang, Beidi Chen, Zhangyang Wang, Anima Anandkumar, and Yuandong Tian. Galore: Memory-efficient llm training by gradient low-rank projection. *arXiv preprint arXiv:2403.03507*, 2024.

Zhi Zhou, Jiang-Xin Shi, Peng-Xiao Song, Xiao-Wen Yang, Yi-Xuan Jin, Lan-Zhe Guo, and Yu-Feng Li. LawGPT: A Chinese Legal Knowledge-Enhanced Large Language Model. 2024. doi: 10.48550/arXiv.2406.04614. URL http://arxiv.org/abs/2406.04614.

Jingwei Zhu, Minghuan Tan, Min Yang, Ruixue Li, and Hamid Alinejad-Rokny. CollectiveSFT: Scaling Large Language Models for Chinese Medical Benchmark with Collective Instructions in Healthcare. 2024. doi: 10.48550/arXiv.2407.19705. URL http://arxiv.org/abs/2407.19705.

Appendix

This Appendix contains the following parts:

- **Hyper Parameters**. We delineate the specific hyperparameters for model training and evaluation, detailing the settings for gradient expectation estimation, momentum compression, the AdamW optimizer, and the empirical justification for the boosted learning coefficient k_0.
- **Dataset Details**. We provide a comprehensive description of the datasets utilized for both general and vertical domain fine-tuning, detailing the specific sources, composition, and quantities for the finance, medicine, law, and the constructed temporal out-of-distribution News QA domains.
- **Performance on the Law Dataset**. We provide a contextual analysis of the performance on the Law dataset, attributing the lower absolute scores to the domain's complex and heterogeneous task mixture while underscoring the robustness of the DBA method in achieving superior relative performance.
- **Practical Implementation Guide**. We outline a two-stage practical implementation guide for DBA, involving a one-time, reusable pre-computation of the global gradient and its subsequent integration into standard fine-tuning frameworks to ensure efficiency and ease of adoption.

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 boosted learning 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 boosted learning coefficients defined in equations (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.

Table 5: Performance impact of the hyperparameter k_0 on the NewsQA benchmark.

k_0	S
$50/T$	57.23
$100/T$	60.13
$150/T$	61.98
$200/T$	**63.76**
$250/T$	63.50
$300/T$	63.39

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 the Table 5, 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.

Table 6: Data sources and quantities

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

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.

Table 7: Data sources and their quantities.

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

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.

Table 8: Questions from various Chinese medicine exams.

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

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

Table 9: Law Data Statistics

NAME	SOURCE	QUANTITY
FACT RECALL	CAIL-LONG	4,200
CASE SUMMARIZATION	CAIL-LONG	5,750
LEGAL QA DATA	LAWGPT	35,000
	LAWYER LLAMA	11,000
	FUZI	32,050
SYLLOGISTIC REASONING	FUZI	11,237

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.

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 boosted annealin 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 DBA still consistently achieves the best or second-best harmonic mean score (S) across all three base models (Table 3). For instance, on Qwen2.5, DBA achieves the highest 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, DBA'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 DBA's robust performance on a more challenging and diverse task mixture. It showcases that DBA's benefits are not confined to simple classification tasks but extend to complex, mixed-task scenarios.

D PRACTICAL IMPLEMENTATION GUIDE

While Dynamic Boosted Annealing (DBA) 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 Boosting (GGB) 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 DBA via a simple command-line argument with minimal implementation overhead.

In summary, the initial setup cost of DBA 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.