Bayesian Sparse Low-Rank Adaptation for Large Language Model Uncertainty Estimation

Anonymous ACL submission

Abstract

Large language models (LLMs) exhibit remarkable reasoning capabilities, but their task-specific fine-tuning is notoriously plagued by overconfidence, severely hindering trustworthy deployment. We propose Data-Adaptive Lower-Rank Adaptation (DALorRA), a simple and effective variational Bayesian sparse framework that shifts the paradigm of uncertainty quantification from the dense parameter space to the lightweight rank level of low-rank adaptation (LoRA). With the insight that LoRA essentially aggregates multiple rank-one components that may provide superfluous model capacity, DALorRA imposes stochastic masking on rank dimensions, enabling Bayesian regularization of model capacity during training and ensemble-like calibration during inference. Extensive experiments demonstrate DALorRA's excellent calibration of LLMs without compromising reasoning accuracy.

1 Introduction

Large language models (LLMs) exhibit remarkable reasoning capabilities, but their reliability remains a central obstacle to trustworthy deployment (Xiong et al., 2024; Shelmanov et al., 2025; Bouchard et al., 2026). When adapted to downstream tasks via deterministic fine-tuning, LLMs can memorize the target distribution, yielding overconfident predictions even for incorrect answers (Yang et al., 2024; Li et al., 2025). This miscalibration weakens the interpretability of predictive confidence and highlights an urgent need for scalable uncertainty quantification. While Bayesian neural networks and Deep Ensembles provide rigorous uncertainty quantification (Blundell et al., 2015; Lakshminarayanan et al., 2017), performing Bayesian inference over billion-parameter weight spaces or maintaining multiple model copies is computationally prohibitive for LLMs (Nemani et al., 2025; Xiang et al., 2026).

Parameter-efficient fine-tuning, particularly Low-Rank Adaptation (LoRA, Hu et al., 2022), circumvents full weight updates and has emerged as a promising foundation for scalable uncertainty quantification (e.g., Wang et al., 2024; Shi et al., 2026). However, LoRA-based calibration largely focuses on the uncertainty of the LoRA adapter, incurring significant computational overhead and compromising the lightweight and efficiency that make LoRA so appealing (Marszałek et al., 2025). Moreover, these approaches predominantly face a structural bottleneck: The LoRA rank r is treated as a static, globally fixed hyperparameter (Gu et al., 2025; Zhou et al., 2026). Since the true intrinsic dimensionality varies across downstream tasks, rigidly fixing r can introduce superfluous model capacity that exacerbates overfit in low-data regimes (Zhang et al., 2023; Liu et al., 2024; Cheon and Paik, 2026). As frontier LLMs tend to require minimal parameter adjustments for specialized tasks, there is a growing need for adaptation that can dynamically prune unnecessary rank components without sacrificing core capabilities (Huang et al., 2025; Kumaravelu et al., 2026; Deng et al., 2026).

In this work, we circumvent the computational difficulty of Bayesianizing the full LoRA adapter and relax the fixed LoRA rank by directly modeling the rank uncertainty. Given rank r, LoRA essentially utilizes the combination of r rank-one components that may provide superfluous capacity. We propose Data-Adaptive Lower-Rank Adaptation (DALorRA). Specifically, we inject a stochastic diagonal mask matrix $\mathbf{D} \in \mathbb{R}^{r\times r}$, formulating the weight update as $\Delta\mathbf{W} = \mathbf{BDA}$. By treating the diagonal elements as latent variables governed by learnable Bernoulli distributions via variational inference, DALorRA captures discrete structural uncertainty over which adaptation dimension should be active. As $\mathbf{D}$ deactivates components of $\mathbf{B}$ and $\mathbf{A}$ with deactivation probabilities depending on downstream tasks, we are essentially implementing

a lower-rank adaptation data-adaptively.

Despite its simplicity, DALorRA achieves outstanding performance in uncertainty quantification and reasoning accuracy by bridging Bayesian neural networks and deep ensembles in a LoRA framework. Analogous to Bayesian neural networks, DALorRA learns the posterior distribution over the mask $\mathbf{D}$, capturing epistemic uncertainty and pruning unnecessary model complexity to avoid overfit. It also mirrors a deep ensemble by stochastically aggregating the predictions of diverse rank-one configurations, maintaining outstanding prediction accuracy and high efficiency. More intuitive justifications are deferred to the end of Section 4. We summarize three primary contributions:

- **Methodology innovation.** We shift the paradigm of uncertainty quantification from the dense LLM parameter space or LoRA adapter ($\mathbf{B}$ and/or $\mathbf{A}$) space to the lightweight LoRA rank level, introducing minimal computational overhead.

- **Principled Approach.** We bridge theoretically rigorous variational Bayesian estimation with empirically sound ensemble-like inference. This principled approach is empirically proven to outperform a rank dropout that similarly induces uncertainty and sparsity over LoRA rank.

- **Comprehensive empirical validation and excellent performance.** We demonstrate through extensive experiments on diverse reasoning benchmarks that DALorRA consistently performs exceptional calibration without sacrificing reasoning accuracy.

2 Related Work

We review LLM uncertainty quantification literature from two perspectives: output uncertainty and uncertainty based on parameter-efficient fine-tuning. The proposed DALorRA contributes to the latter, and we discuss our distinct contribution.

2.1 LLM Output Uncertainty

Reliable uncertainty estimation is critical for mitigating hallucinations and overconfident predictions in large language models (LLMs) (Shelmanov et al., 2025; Bouchard et al., 2026). Existing research has approached this from the output space, utilizing heuristics such as verbalized confidence (Lin et al., 2022), prompt-based ensembling (Jiang et al., 2023), and semantic entropy across multiple generations (Kuhn et al., 2023; Shelmanov et al., 2025; Bouchard et al., 2026). While highly appealing due to their plug-and-play nature, output-space metrics inherently bypass the underlying probability distributions of model parameters. Consequently, when adapting LLMs to specialized downstream tasks in low-data regimes, such output-centric heuristics do not fully capture epistemic uncertainty during fine-tuning, leaving models prone to overconfident predictions and overfitting (Niu et al., 2024; Bakman et al., 2025). This limitation necessitates a shift toward theoretically grounded, parameter-level uncertainty quantification during fine-tuning.

2.2 Uncertainty via Parameter-Efficient Fine-Tuning.

Classical uncertainty estimation methods, such as Monte Carlo Dropout (Gal and Ghahramani, 2016) and Deep Ensembles (Lakshminarayanan et al., 2017), approximate predictive uncertainty through stochastic inference or model averaging, but can be computationally prohibitive when directly applied to LLMs. To circumvent operations on full-parameter, parameter-efficient fine-tuning (PEFT), particularly LoRA (Hu et al., 2022), has emerged as a practical foundation for uncertainty-aware adaptation. Recent work has investigated how to Bayesianize or calibrate only the lightweight PEFT modules rather than the full LLM. Post-hoc Laplace approximation has been applied to fine-tuned LoRA parameters to obtain approximate posterior uncertainty (Wang et al., 2024; Shi et al., 2026; Marszałek et al., 2025; Rahmati et al., 2026). Advancing this paradigm towards end-to-end inference, Bayesian low-rank adaptation by backpropagation (BLoB, Wang et al., 2024) moves posterior learning directly into the fine-tuning stage by jointly optimizing the variational mean and covariance of LoRA adapters. Training-Free Bayesianization (TFB, Shi et al., 2026) further converts trained LoRA adapters into Bayesian ones by searching over low-rank isotropic Gaussian posteriors. More recently, Contextual LoRA (C-LoRA, Rahmati et al., 2026) introduces modules to capture sample-dependent uncertainty to complement global parameter uncertainty.

Our work contributes to this stream of uncertainty quantification but is distinct from extant research, which largely estimates the variances of LoRA adapters $\mathbf{B}$ and/or $\mathbf{A}$, potentially compromising LoRA's efficiency. Instead, the proposed

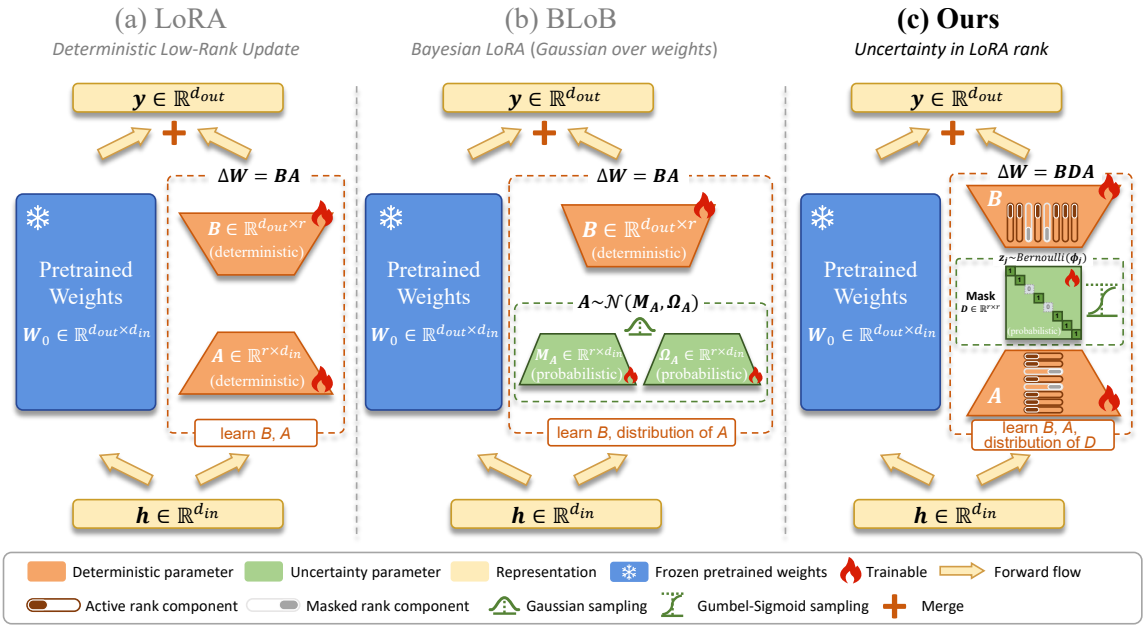

Figure 1: Comparison between LoRA, BLoB, and our DALorRA. Standard LoRA learns a deterministic low-rank update $\Delta\mathbf{W} = \mathbf{BA}$. BLoB models Gaussian uncertainty over LoRA adapter $\mathbf{A}$. In contrast, DALorRA introduces a stochastic diagonal mask $\mathbf{D}$, yielding $\Delta W = BDA$ to model rank-level uncertainty in LoRA.

DALorRA targets uncertainty at the rank level, introducing negligible computational overhead. Furthermore, our stochastic masking configuration imposes explicit sparsity on the already lightweight LoRA architecture, making it particularly suitable for frontier LLMs that require minimal adjustments and thus lower-rank adaptation for specialized tasks. Ultimately, our method bridges theoretically rigorous Bayesian neural networks with empirically robust deep ensembles, which underpin its excellent performance. We compare DALorRA with regular LoRA and BLoB in Figure 1; instead of learning the uncertainty of LoRA adapters, we keep them deterministic and learn the posterior distribution of the rank mask $\mathbf{D}$.

3 Preliminaries

3.1 Low-Rank Adaptation

Low-Rank Adaptation (LoRA) is a parameter-efficient fine-tuning paradigm that adapts a pre-trained model by freezing the original dense weights and optimizing a low-rank update (Hu et al., 2022). Given a linear layer with a frozen pre-trained weight $\mathbf{W}_0 \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$, LoRA parameterizes the task-specific update as $\Delta\mathbf{W} = \mathbf{BA}$, where $\mathbf{B} \in \mathbb{R}^{d_{\text{out}} \times r}$, $\mathbf{A} \in \mathbb{R}^{r \times d_{\text{in}}}$, and $r \ll \min(d_{\text{out}}, d_{\text{in}})$ is the adaptation rank. For an input representation $\mathbf{h} \in \mathbb{R}^{d_{\text{in}}}$, the forward pass is

$$\mathbf{o} = (\mathbf{W}_0 + \mathbf{BA})\mathbf{h},$$

where $\mathbf{o} \in \mathbb{R}^{d_{\text{out}}}$ is the output representation. Compared to full-parameter fine-tuning, LoRA introduces only $r \times (d_{\text{out}} + d_{\text{in}})$ trainable parameters, substantially reducing memory and computational costs while largely preserving adaptation efficacy.

3.2 Confidence Calibration in LLMs

Following standard protocols (Wang et al., 2024; Yang et al., 2024), we consider a downstream dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$ and formulate LLM reasoning and multiple-choice tasks as next-token classification over a candidate label set $\mathcal{Y}$. Given an input prompt $\mathbf{x}$, the LLM (parameterized by $\boldsymbol{\theta}$) yields a predictive distribution $p_{\boldsymbol{\theta}}(y \mid \mathbf{x})$. The predicted label and its confidence are

$$\hat{y} = \arg\max_{y \in \mathcal{Y}} p_{\boldsymbol{\theta}}(y \mid \mathbf{x}), \quad \hat{p} = \max_{y \in \mathcal{Y}} p_{\boldsymbol{\theta}}(y \mid \mathbf{x}).$$

When adapted to downstream tasks via deterministic fine-tuning, LLMs frequently overfit the target distribution, yielding overconfident predictions even for incorrect answers. To quantify this miscalibration, Expected Calibration Error (ECE) partitions predictions into M bins $\{B_m\}_{m=1}^{M}$:

$$\text{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{N} \left|\text{acc}(B_m) - \text{conf}(B_m)\right|,$$

where $\text{acc}(B_m) = \frac{1}{|B_m|} \sum_{i \in B_m} \mathbb{I}(\hat{y}_i = y_i)$ and $\text{conf}(B_m) = \frac{1}{|B_m|} \sum_{i \in B_m} \hat{p}_i$, with $\mathbb{I}(\cdot)$ being the

indicator function. This severe overconfidence phenomenon necessitates a paradigm shift toward principled uncertainty quantification.

3.3 Variational Inference

Bayesian neural networks mitigate overconfidence by placing prior distributions over weights to capture epistemic uncertainty (Blundell et al., 2015). Since computing the exact posterior $p(\mathbf{W} \mid \mathcal{D})$ is intractable, Variational Inference (VI) approximates it with a tractable distribution $q_{\boldsymbol{\phi}}(\mathbf{W})$ parameterized by variational parameters $\boldsymbol{\phi}$.

The optimal $\boldsymbol{\phi}$ is obtained by minimizing the Kullback-Leibler (KL) divergence to the true posterior, which is mathematically equivalent to minimizing the negative Evidence Lower Bound:

$$-\mathbb{E}_{q_{\boldsymbol{\phi}}}\left[\log p(\mathcal{D} \mid \mathbf{W})\right] + \mathrm{KL}\left(q_{\boldsymbol{\phi}}(\mathbf{W}) \,\|\, p(\mathbf{W})\right).$$

The first term corresponds to the expected negative log-likelihood, and the second term regularizes the deviation of $q_{\boldsymbol{\phi}}(\mathbf{W})$ from the prior $p(\mathbf{W})$. To enable end-to-end backpropagation through the expectation term, reparameterization tricks (Kingma and Welling, 2013) are often required. For instance, if $q_{\boldsymbol{\phi}}$ is Gaussian with $\boldsymbol{\phi} = (\boldsymbol{\mu}, \boldsymbol{\Sigma})$, a weight sample is drawn via $\mathbf{W} = \boldsymbol{\mu} + \boldsymbol{\Sigma}^{1/2}\boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is independent of $\boldsymbol{\phi}$. This framework lays the theoretical foundation for our method.

4 Method

We propose Data-Adaptive Lower-Rank Adaptation (DALorRA), a variational Bayesian sparse framework tailored for uncertainty-aware LLM fine-tuning. Unlike LoRA (Figure 1a) that relies on fixed dense updates, or recent Bayesian extensions, like BLoB (Figure 1b), which may suffer from high optimization overhead of Gaussian posteriors over the dense LoRA adapters, our approach represents a structural paradigm shift. Our core insight is that the standard LoRA update matrix $\Delta\mathbf{W} = \mathbf{BA}$ inherently consists of r rank-one components. This provides a natural and meaningful way to turn off unnecessary model capacity. Instead of modeling uncertainty over LLM weights or LoRA adapters, DALorRA leverages this insight to model uncertainty directly at the rank level (Figure 1c).

Specifically, given a linear layer with a frozen pre-trained weight $\mathbf{W}_0$ and maximum allowable rank r, we introduce a stochastic diagonal mask matrix $\mathbf{D}$ and define the weight update as

$$\Delta\mathbf{W} = \mathbf{BDA}, \tag{1}$$

where $\mathbf{B} \in \mathbb{R}^{d_{\text{out}} \times r}$ and $\mathbf{A} \in \mathbb{R}^{r \times d_{\text{in}}}$ are deterministic adapters, and $\mathbf{D} = \mathrm{diag}(\mathbf{z})$, $\mathbf{z} \in \{0,1\}^r$, is an r-dimensional random diagonal matrix. Each diagonal element $z_j \in \{0,1\}$ for $j = 1, \ldots, r$, dynamically activates or deactivates the j-th rank component.

We assign an independent and identical Bernoulli prior $p(z_j) = \mathrm{Bernoulli}(z_j; p_0)$ over the mask with a constant $p_0 \in (0,1)$. The exact posterior inference is intractable, and we approximate it with a factorized Bernoulli variational distribution parameterized by $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_r) \in \mathbb{R}^r$, which serves as the learnable variational parameters:

$$q_{\boldsymbol{\phi}}(\mathbf{z}) = \prod_{j=1}^{r} \mathrm{Bernoulli}(z_j; \sigma(\phi_j)),$$

where $\sigma(\cdot)$ is the sigmoid function. The deterministic adapters $\mathbf{A}$ and $\mathbf{B}$ and the variational parameters $\boldsymbol{\phi}$ are jointly learned by minimizing the negative Evidence Lower Bound:

$$\mathcal{L}(\mathbf{A}, \mathbf{B}, \boldsymbol{\phi}) = -\,\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z})}[\log p(\mathcal{D} \mid \mathbf{A}, \mathbf{B}, \mathbf{z})] + \mathrm{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}) \| p(\mathbf{z})). \tag{2}$$

The first term of Eq. (2) represents the expected negative log-likelihood with weight updates by Eq. (1). Since it lacks an analytical solution, we evaluate it via Monte Carlo approximation. To enable gradient-based optimization through the discrete Bernoulli mask, we employ the Gumbel-Sigmoid reparameterization (Jang et al., 2016). Specifically, a continuous relaxation of $\mathbf{z} \sim q_{\boldsymbol{\phi}}(\mathbf{z})$ is sampled via

$$\mathbf{z} = \sigma((\boldsymbol{\phi} + \boldsymbol{\epsilon})/\tau), \tag{3}$$

where $\boldsymbol{\epsilon} = \log \mathbf{u} - \log(1 - \mathbf{u})$, $\mathbf{u} \sim \mathrm{Uniform}(0,1)^r$, and $\tau > 0$ is the temperature parameter.

The second term in Eq. (2) is the KL divergence, which regularizes the approximate posterior towards the prior. Because both $p(\mathbf{z})$ and $q_{\boldsymbol{\phi}}(\mathbf{z})$ are Bernoulli distributions, it has the analytical form

$$\sum_{j=1}^{r} [\sigma(\phi_j) \log \frac{\sigma(\phi_j)}{p_0} + (1 - \sigma(\phi_j)) \log \frac{1 - \sigma(\phi_j)}{1 - p_0}].$$

Substituting the Gumbel-Sigmoid sampling and the analytical KL divergence into Eq. (2) yields the

Algorithm 1 DALorRA Fine-Tuning

Input: Dataset $\mathcal{D}$, frozen weights $\mathbf{W}_0$, prior Bernoulli probability p_0, Gumbel-sigmoid temperature τ, Monte Carlo sample size S

Output: $\mathbf{B}$, $\mathbf{A}$, and $\boldsymbol{\phi}$

1: Initialize $\mathbf{A}, \mathbf{B}$ using standard LoRA initialization and $\boldsymbol{\phi} \leftarrow \log(p_0/(1-p_0)) \cdot \mathbf{1}_r$
2: **while** not stopped **do**
3: Sample a mini-batch of $\mathcal{D}$.
4: Sample $\mathbf{z}_1, \ldots, \mathbf{z}_S$ by Eq. (3)
5: Compute the mini-batch objective by Eq. (4) and update $\mathbf{B}$, $\mathbf{A}$, and $\boldsymbol{\phi}$ by stochastic gradient descent
6: **end while**

ultimate differentiable objective function:

$$\begin{aligned}\mathcal{L}(\mathbf{A}, \mathbf{B}, \boldsymbol{\phi}) = &-\frac{1}{S}\sum_{s=1}^{S} \log p(\mathcal{D} \mid \mathbf{A}, \mathbf{B}, \mathbf{z}_s) \\ &+ \sum_{j=1}^{r} \Big[\sigma(\phi_j) \log \frac{\sigma(\phi_j)}{p_0} + \\ &(1-\sigma(\phi_j)) \log \frac{1-\sigma(\phi_j)}{1-p_0}\Big],\end{aligned} \tag{4}$$

where $\mathbf{z}_s$ for $s = 1, \ldots, S$ are independently sampled using Eq. (3).

To ensure stable optimization and prevent KL divergence explosion, we initialize $q_{\boldsymbol{\phi}}(\boldsymbol{z})$ equal to $p_0(\boldsymbol{z})$ by setting $\boldsymbol{\phi} = \log(p_0/(1-p_0)) \cdot \mathbf{1}_r$. We summarize DALorRA fine-tuning in Algorithm 1.

At inference time, the predictive distribution is obtained by averaging over M mask matrices sampled from the learned variational posterior. Specifically, the estimated predictive probability $\hat{p}(y \mid \mathbf{x}, \mathbf{W}_0, \mathbf{B}, \mathbf{A})$ is

$$\frac{1}{M}\sum_{m=1}^{M} p(y \mid \mathbf{x}, \mathbf{W}_0, \mathbf{B}, \mathbf{A}, \mathbf{D}_m),$$

where $\mathbf{D}_m = \mathrm{diag}(\mathbf{z}_m)$ and $\mathbf{z}_m \overset{iid}{\sim} q_{\boldsymbol{\phi}}(\mathbf{z})$. Empirically, setting $M = 10$ (Wang et al., 2024; Shi et al., 2026; Rahmati et al., 2026) achieves competitive performance with acceptable inference overhead.

Intuitively, the mechanism that allows DALorRA to excel stems from combining the theoretical rigor of Bayesian neural networks with the empirical soundness of deep ensembles, entirely within a parameter-efficient regime. The DALorRA weight update in Eq. (1) essentially is

$$\Delta\mathbf{W} = \sum_{j=1}^{r} z_j \mathbf{b}_j \mathbf{a}_j, \tag{5}$$

where $\mathbf{b}_j$ is the j-th column of $\mathbf{B}$, and $\mathbf{a}_j$ is the j-th row of $\mathbf{A}$. This effectively trains an ensemble of sub-networks with stochastic weights z_j whose distributions are to be learned. During training, DALorRA acts as a parameter-efficient Bayesian neural network, capturing epistemic uncertainty and penalizing unnecessary complexity through learning the mask posterior distribution. At inference time, averaging predictions over multiple sampled masks mirrors the behavior of a deep ensemble, aggregating outputs across diverse rank-one LoRAs with data-adaptive weights. By pruning superfluous capacity (via some z_j equal to 0), DALorRA marries the principled uncertainty quantification of Bayesian methods with the robust generalization of ensembles, mitigating overconfidence while retaining the LoRA-like efficiency.

5 Experiments

In this section, we conduct comprehensive experiments to evaluate the effectiveness of our proposed DALorRA. We first introduce experiment settings and then report the consistently outstanding performance of DALorRA on LLM calibration without sacrificing reasoning accuracy. We also provide analyses to show the importance of learning the rank mask posterior and the necessity of sufficient model capacity combined with rank sparsity, and illustrate the posterior mask probabilities that vary across transformer layers, projection modules, and data. More implementation details and experiment results are deferred to Appendices A and B.

5.1 Experiment Settings

LLM fine-tuning. We fine-tune Llama-3.1-8B (Grattafiori et al., 2024) and Llama2-7B (Touvron et al., 2023) using the PEFT library (Mangrulkar et al., 2022) on common-sense reasoning benchmarks, including Winogrande-Small (WG-S) (Sakaguchi et al., 2021), Winogrande-Medium (WG-M) (Sakaguchi et al., 2021), ARC-Challenge (ARC-C) (Clark et al., 2018), ARC-Easy (ARC-E) (Clark et al., 2018), OpenBookQA (OBQA) (Mihaylov et al., 2018), and BoolQ (Clark et al., 2019). For out-of-distribution evaluation, we fine-tune models on OBQA (Mihaylov et al., 2018) and evaluate on ARC-C (Clark et al., 2018), ARC-E (Clark

Table 1: Performance comparison. All methods use Llama3.1-8B pre-trained weights. Accuracy (ACC) and Expected Calibration Error (ECE) are reported in percentages, and Negative Log-Likelihood (NLL) is reported on its original scale. "↑" and "↓" indicate that higher and lower values are preferred, respectively. **Boldface** and underlined numbers denote the best and the second-best performance, respectively.

	Method	In-Distribution						Out-of-Distribution (OBQA→X)			
								Small Shift		*Large Shift*	
		WG-S	ARC-C	ARC-E	WG-M	OBQA	BoolQ	ARC-C	ARC-E	Chem	Phy
ACC (↑)	MCD	78.03±0.61	81.64±1.79	91.37±0.38	83.18±0.84	87.20±1.02	89.93±0.16	81.42±1.38	**87.27±0.84**	47.92±2.25	46.53±0.49
	ENS	**78.82±0.52**	**82.55±0.42**	91.84±0.36	**83.99±0.74**	87.37±0.67	**90.50±0.14**	79.62±0.57	86.56±0.60	49.65±3.22	44.44±1.96
	LA	76.05±0.92	79.95±0.42	90.73±0.08	82.83±0.85	87.90±0.20	89.36±0.52	81.08±1.20	87.21±1.20	48.26±3.93	46.18±1.30
	MLE	77.87±0.54	81.08±0.48	91.67±0.36	82.30±0.53	87.90±0.87	89.58±0.26	81.48±2.41	86.83±0.87	45.83±0.85	42.36±1.77
	MAP	76.90±0.97	81.08±2.48	91.61±0.44	82.59±0.28	85.73±0.19	90.09±0.28	79.98±0.87	86.58±0.79	43.40±4.98	38.54±3.40
	BLoB	77.34±0.25	80.86±1.24	90.83±0.68	81.64±0.62	87.66±0.37	88.69±1.26	78.68±0.24	86.63±0.18	43.75±2.65	46.96±3.31
	TFB	74.65±1.36	80.18±1.37	**91.90±0.30**	82.04±0.24	88.20±0.20	88.84±0.21	81.12±1.32	86.81±0.49	42.65±5.12	46.58±1.98
	C-LoRA	77.26±0.12	81.70±1.17	90.79±0.51	81.62±0.56	86.93±1.62	87.77±0.64	**81.60±0.35**	85.48±0.55	45.64±3.76	40.38±3.76
	DALorRA	77.43±1.24	81.74±0.23	90.97±0.68	82.68±0.54	**88.24±0.42**	89.43±0.24	**81.60±0.20**	86.56±0.20	**50.00±1.08**	**47.22±0.60**
ECE (↓)	MCD	16.13±0.54	13.69±1.11	6.73±0.71	13.05±0.99	9.76±0.71	7.95±0.17	13.63±1.18	9.27±0.60	30.91±3.57	33.08±1.40
	ENS	14.72±0.17	13.45±1.19	6.59±0.45	11.17±0.92	8.17±0.86	7.35±0.55	11.37±1.82	7.21±1.13	18.92±6.03	26.80±3.23
	LA	**4.18±0.11**	9.26±3.08	5.27±0.51	3.50±0.78	8.93±0.34	1.93±0.22	7.83±1.49	7.80±1.99	**14.49±0.57**	13.17±2.14
	MLE	17.02±0.46	16.35±0.68	7.00±0.53	13.83±0.65	9.77±0.81	8.69±0.21	14.45±2.19	10.78±0.50	32.46±2.60	38.41±4.44
	MAP	18.71±0.74	15.77±1.60	6.62±0.64	14.26±0.92	12.19±0.55	8.40±0.25	16.46±0.44	11.36±0.58	34.79±3.76	38.50±2.18
	BLoB	8.84±0.36	5.87±1.12	4.24±0.68	3.42±0.42	3.35±0.82	2.46±0.35	7.02±0.46	5.12±0.88	14.79±0.66	**12.34±3.68**
	TFB	8.23±0.68	6.19±0.86	3.00±0.92	3.59±0.64	4.51±0.33	3.80±0.61	7.12±1.22	4.85±0.36	15.32±2.12	16.32±4.37
	C-LoRA	16.84±1.88	10.75±0.81	4.95±1.43	9.97±1.02	6.50±1.52	4.36±0.90	7.06±0.83	5.63±0.94	26.48±2.19	30.28±4.12
	DALorRA	7.81±1.08	**5.60±0.64**	**2.88±0.37**	**3.25±0.39**	**3.12±0.49**	**1.82±0.22**	**6.23±0.79**	**3.81±1.09**	14.58±1.65	15.46±3.08
NLL (↓)	MCD	0.83±0.01	0.99±0.10	0.45±0.06	0.64±0.03	0.62±0.08	0.49±0.01	1.03±0.02	0.61±0.03	1.91±0.18	2.02±0.15
	ENS	0.75±0.02	0.80±0.11	0.38±0.03	0.55±0.02	0.45±0.05	0.42±0.05	0.72±0.07	0.44±0.03	1.40±0.18	1.50±0.13
	LA	**0.56±0.00**	1.18±0.02	1.04±0.01	0.51±0.00	0.94±0.00	0.43±0.00	1.17±0.01	1.11±0.00	1.27±0.01	1.28±0.00
	MLE	0.88±0.04	1.20±0.11	0.46±0.04	0.68±0.01	0.61±0.06	0.52±0.01	1.07±0.06	0.72±0.06	1.91±0.16	2.25±0.21
	MAP	0.99±0.07	1.12±0.23	0.46±0.03	0.74±0.07	0.79±0.02	0.52±0.01	1.19±0.04	0.83±0.06	1.97±0.13	2.32±0.10
	BLoB	0.58±0.01	0.59±0.02	0.30±0.01	0.47±0.01	0.38±0.01	**0.27±0.01**	0.61±0.03	0.46±0.01	**1.23±0.06**	1.28±0.22
	TFB	0.59±0.01	0.62±0.04	**0.25±0.01**	**0.43±0.01**	**0.36±0.02**	0.29±0.02	0.60±0.04	0.44±0.02	1.34±0.08	1.26±0.12
	C-LoRA	0.99±0.11	0.68±0.03	0.33±0.06	0.57±0.06	0.40±0.04	0.30±0.01	0.55±0.03	0.48±0.03	1.66±0.16	1.88±0.09
	DALorRA	0.58±0.01	**0.56±0.02**	0.28±0.01	0.51±0.01	**0.36±0.02**	**0.27±0.01**	**0.54±0.07**	**0.40±0.01**	1.25±0.02	**1.25±0.06**

et al., 2018), and college-level chemistry (Chem) and physics (Phy) (Hendrycks et al., 2020).

Evaluation metrics. We report accuracy (ACC), expected calibration error (ECE) (Naeini et al., 2015), and negative log-likelihood (NLL). ACC measures predictive performance. ECE and NLL evaluate calibration and probabilistic reliability. Higher ACC and lower ECE and NLL are preferred.

Baseline methods. We compare DALorRA with representative deterministic and uncertainty-aware baselines for LLM fine-tuning. The deterministic baselines include Maximum Likelihood Estimation (MLE) implemented with standard LoRA (Hu et al., 2022), and Maximum A Posteriori estimation (MAP), which serves as a point-estimate baseline with weight-decay regularization. For uncertainty-aware baselines, we include Monte-Carlo Dropout (MCD) (Gal and Ghahramani, 2016), Deep Ensemble (ENS) (Lakshminarayanan et al., 2017; Balabanov and Linander, 2024; Wang et al., 2023), Laplace-LoRA (LA) (Yang et al., 2024), Bayesian LoRA by Backpropagation (BLoB) (Wang et al., 2024), Training-Free Bayesianization (TFB) (Shi et al., 2026), and Contextual Low-Rank Adaptation (C-LoRA) (Rahmati et al., 2026). We use $M = 10$ Monte Carlo samples for inference (Shi et al., 2026) in all sampling-based methods, including BLoB, TFB, C-LoRA, and DALorRA.

5.2 Main Results

Calibration performance. Table 1 reports the in- and out-of-distribution performance of different LoRA-based uncertainty estimation methods on Llama3.1-8B. Overall, Table 1 demonstrates DALorRA's exceptional uncertainty calibration and reliability without sacrificing reasoning accuracy across both in-distribution and out-of-distribution (OOD) evaluations. On average, DALorRA achieves the strongest performance in both the reasoning (ACC) and calibration (ECE and NLL) metrics across the 10 evaluation settings. Specifically, when evaluated by ECE and NLL, DALorRA consistently outperforms the baselines by securing the best or second-best performance

Table 2: Efficiency Comparison among LoRA, BLoB, C-LoRA, and DALorRA. The experiments are conducted on a single NVIDIA A40 GPU and based on Llama-3.1-8B with rank $r = 8$, batch size 4, and 2,000 training iterations. The subscripts indicate the relative cost compared to LoRA, with red or green denoting worse or better efficiency.

Method	Trainable/Extra Params	WG-S		ARC-E		OBQA	
		Train/Eval (s)	Mem. (MB)	Train/Eval (s)	Mem. (MB)	Train/Eval (s)	Mem. (MB)
LoRA	4,466,688/0	1,237.07/104.19	7,690.08	1,799.69/66.76	11,296.05	1,524.87/50.99	9,410.75
BLoB	$6{,}596{,}608_{1.48\times}/+2{,}129{,}920$	$1{,}316.11_{1.06\times}/1{,}058.12_{10.16\times}$	$8{,}214.50_{1.07\times}$	$1{,}901.34_{1.06\times}/683.49_{10.24\times}$	$12{,}874.76_{1.14\times}$	$1{,}610.29_{1.06\times}/520.53_{10.21\times}$	$10{,}453.27_{1.11\times}$
C-LoRA	$5{,}044{,}928_{1.13\times}/+578{,}240$	$4{,}640.89_{3.75\times}/714.55_{6.86\times}$	$11{,}245.92_{1.46\times}$	$4{,}911.09_{2.73\times}/366.43_{5.49\times}$	$15{,}718.29_{1.39\times}$	$4{,}745.12_{3.11\times}/340.10_{6.67\times}$	$13{,}410.96_{1.43\times}$
DALorRA (Ours)	$4{,}467{,}208_{1.0001\times}/+520$	$894.31_{0.72\times}/809.60_{7.77\times}$	$9{,}133.99_{1.19\times}$	$1{,}078.27_{0.60\times}/414.98_{6.22\times}$	$13{,}670.46_{1.21\times}$	$990.34_{0.65\times}/346.64_{6.80\times}$	$11{,}336.79_{1.20\times}$

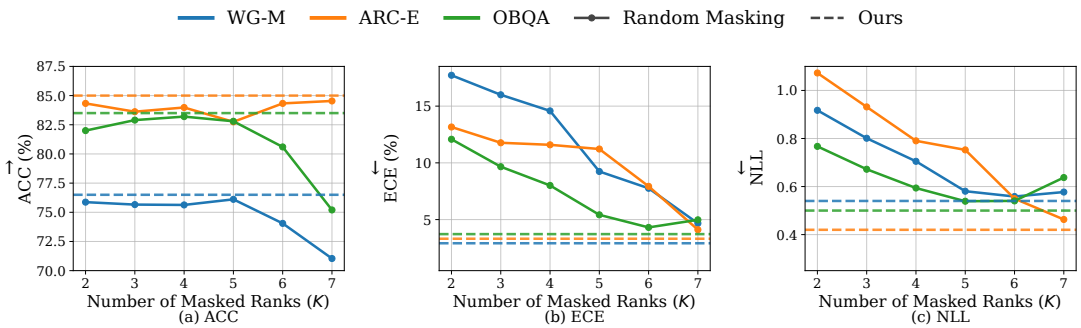

Figure 2: Learning mask posterior (DALorRA) versus random masking. Solid lines denote randomly dropping out K rank dimensions during training, and dashed lines denote DALorRA.

on 9 out of the 10 settings, respectively. Moreover, it achieves this without compromising reasoning capabilities (ACC). In particular, DALorRA maintains highly competitive predictive performance compared to the baseline approaches, achieving the best or second-best ACC on 5 out of the 10 settings. The results suggest that DALorRA's uncertainty over rank remarkably mitigates overconfidence.

On in-distribution datasets, DALorRA achieves the best ECE on five benchmarks and top-tier NLL, offering consistently better uncertainty estimates. The reliability extends to OOD evaluations, where DALorRA secures the best ECE and NLL under small shifts, and the highest accuracy alongside highly competitive NLL under large shifts (Chem and Phy). Ultimately, these results confirm that learning a structured posterior over LoRA rank effectively captures uncertainty without sacrificing reasoning capabilities, in spite of distribution shifts.

Efficiency and resource consumption. We compare the computing efficiency of the sampling-based uncertainty-aware fine-tuning methods (BLoB, C-LoRA, and DALorRA) with LoRA. Table 2 shows DALorRA's high efficiency. It adds only 520 extra trainable parameters, which is negligible and much fewer than BLoB and C-LoRA. DALorRA takes the least training time, even faster than LoRA due to its sparse configuration; it uses only 0.72×, 0.60×, and 0.65× of LoRA's training time on WG-S, ARC-E, and OBQA, respectively. DALorRA remains faster than BLoB and comparable to C-LoRA in evaluation (inference) time with moderate memory overhead. In addition, DALorRA's overall (training plus evaluation) time is comparable to LoRA's on the three datasets.

5.3 Ablation and Additional Analyses

Necessity of mask posterior learning. To demonstrate the critical need for learning the mask posterior, we compare DALorRA against a random masking strategy that similarly induces rank uncertainty. Specifically, we fix $r = 8$ and randomly zero out K (up to 7) diagonal entries of $\mathbf{D}$ in every mini-batch iteration of stochastic gradient descent, acting as a rank-level dropout. The inference is the same as in DALorRA, where $M = 10$ Monte Carlo samples of $\mathbf{D}$ are aggregated for prediction.

The results are illustrated in Figure 2. Random masking, even with an optimal K, still underperforms DALorRA, and K remarkably affects its performance and depends on downstream tasks. For instance, aggressively masking rank dimensions severely degrades accuracy on WG-M and OBQA due to capacity starvation but slightly improves accuracy on ARC-E. Moreover, on WG-M and OBQA, better ECE and NLL are achieved at a smaller K, which, however, severely compromises ACC, whereas all three metrics are op-

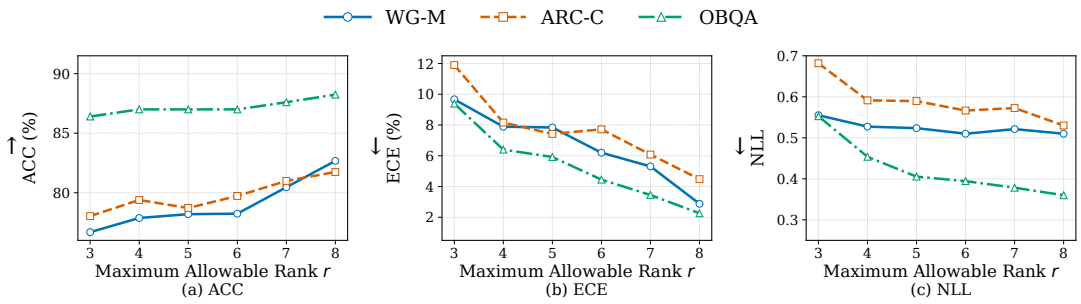

Figure 3: Impact of maximum allowable rank r. A large r generally improves both accuracy and calibration.

timized at a larger K on ARC-E. This inconsistent accuracy-calibration trade-off highlights the necessity of data-adaptive rank uncertainty. By learning the mask posterior and performing posterior sampling for inference, DALorRA achieves consistently strong performance without manually specifying masking intensity.

Maximum allowable rank. We show the important role of the maximum allowable rank r. We vary r between 3 and 8 for different levels of intrinsic model capacities and compare DALorRA performances. As shown in Figure 3, increasing r generally improves accuracy and reduces ECE and NLL. This indicates that better performance stems from combining sufficient capacity with rank sparsity. In addition, the performance trends vary across data. For example, in Figure 3c, DALorRA's NLL decreases with r faster on ARC-C and OBQA than on WG-M, which also necessitates learning the mask posterior from the data.

Posterior mask probabilities. Figure 4 illustrates the learned posterior mask probabilities for the transformer Query and Value projection matrices. We observe non-uniform patterns across both transformer layers and rank indices, indicating that DALorRA's rank sparsity depends on the layer, projection module, and data. This qualitative evidence suggests that the proposed masking mechanism learns data-adaptive rank allocation instead of imposing a uniform rank structure.

Additional results. We provide additional experiment results in Appendix B, including using an alternative foundation model, Llama2-7B, performance on combined datasets, impacts of the Bernoulli prior, and efficiency comparison on other data.

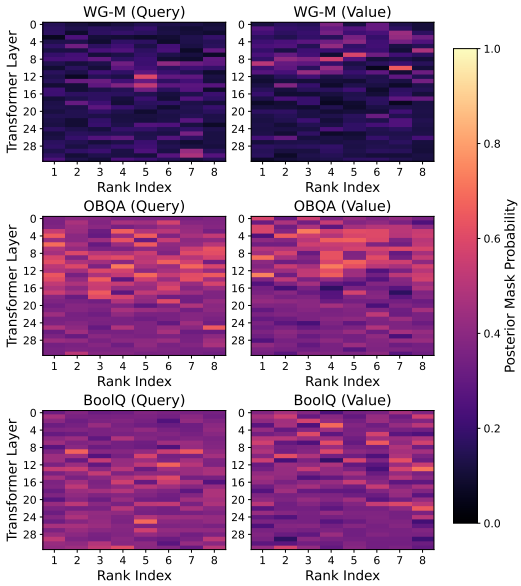

Figure 4: Posterior Bernoulli probabilities of DALorRA masks $\mathbf{D}$ on WG-M, OBQA, and BoolQ.

6 Conclusion

We propose DALorRA as a novel parameter-efficient fine-tuning framework for LLM uncertainty quantification. By systematically learning a variational posterior over the LoRA rank components, DALorRA prunes superfluous capacity while effectively capturing epistemic uncertainty. Our comprehensive evaluations across diverse tasks demonstrate that DALorRA significantly improves LLM calibration without sacrificing fundamental reasoning accuracy. By introducing negligible parameter overhead and training costs compared to standard LoRA, our approach offers a scalable solution. Bridging the principled uncertainty estimation of Bayesian methods with the robust generalization of deep ensembles, DALorRA provides a practical and reliable pathway for deploying trustworthy LLMs in complex, real-world environments.

Limitations

While DALorRA presents a highly parameter-efficient approach to LLM calibration, several limitations highlight promising directions for future research. First, our empirical evaluation is currently limited to classification-style reasoning tasks with fixed answer spaces; future work can extend this framework to open-ended generation scenarios where hallucinations and uncertainty are more complex to characterize. Second, the reliance on multiple Bernoulli mask samples for inference introduces extra overhead, which poses challenges for latency-sensitive applications and necessitates the development of more efficient inference approximations. Third, because DALorRA models uncertainty solely at the LoRA rank level rather than over full adapter weights, it may miss fine-grained, weight-level uncertainty. To address this, future work could seamlessly integrate our data-adaptive rank masking with more expressive, weight-level Bayesianization approaches. Finally, the current variational posterior assumes an independent Bernoulli factorization that ignores potential dependencies across ranks, layers, and projection modules. Developing structured variational posteriors capable of capturing these complex architectural dependencies remains an important area for future investigation.

Ethical Considerations

We anticipate no ethical concerns related to this work, as the study exclusively uses publicly available data with sensitive information redacted. We used AI writing assistants only for language polishing, formatting suggestions, and auxiliary writing support. All scientific claims, experimental results, analyses, and final manuscript content were verified and approved by the authors.

References

Yavuz Bakman, Sungmin Kang, Zhiqi Huang, Duygu Nur Yaldiz, Catarina G Belém, Chenyang Zhu, Anoop Kumar, Alfy Samuel, Salman Avestimehr, Daben Liu, and 1 others. 2025. Uncertainty as feature gaps: Epistemic uncertainty quantification of llms in contextual question-answering. *arXiv preprint arXiv:2510.02671*.

Oleksandr Balabanov and Hampus Linander. 2024. Uncertainty quantification in fine-tuned llms using lora ensembles. *arXiv preprint arXiv:2402.12264*.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. 2015. Weight uncertainty in neural network. In *International conference on machine learning*, pages 1613–1622. PMLR.

Dylan Bouchard, Mohit Singh Chauhan, David Skarbrevik, Ho-Kyeong Ra, Viren Bajaj, and Zeya Ahmad. 2026. Uqlm: A python package for uncertainty quantification in large language models. *Journal of Machine Learning Research*, 27(13):1–10.

Jeonghwan Cheon and Se-Bum Paik. 2026. Brain-inspired warm-up training with random noise for uncertainty calibration. *Nature Machine Intelligence*, pages 1–12.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. Boolq: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 conference of the north American chapter of the association for computational linguistics: Human language technologies, volume 1 (long and short papers)*, pages 2924–2936.

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

Guanzhi Deng, Bo Li, Ronghao Chen, Huacan Wang, Lijie Wen, and Linqi Song. 2026. Dr-lora: Dynamic rank lora for mixture-of-experts adaptation. *arXiv preprint arXiv:2601.04823*.

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR.

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

Jiancheng Gu, Jiabin Yuan, Jiyuan Cai, Xianfa Zhou, and Lili Fan. 2025. La-lora: Parameter-efficient fine-tuning with layer-wise adaptive low-rank adaptation. *Neural Networks*, page 108095.

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

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *Iclr*, 1(2):3.

Weizhong Huang, Yuxin Zhang, Xiawu Zheng, Yang Liu, Jing Lin, Yiwu Yao, and Rongrong Ji. 2025. Dynamic low-rank sparse adaptation for large language models. *arXiv preprint arXiv:2502.14816*.

Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.

Mingjian Jiang, Yangjun Ruan, Sicong Huang, Saifei Liao, Silviu Pitis, Roger Baker Grosse, and Jimmy Ba. 2023. Calibrating language models via augmented prompt ensembles. In *ICML Workshop on Challenges in Deployable Generative AI*.

Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *arXiv preprint arXiv:2302.09664*.

Vishnuprasadh Kumaravelu, Sunil Gupta, and PK Srijith. 2026. Post-optimization adaptive rank allocation for lora. *arXiv preprint arXiv:2604.27796*.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30.

Yawei Li, David Rügamer, Bernd Bischl, and Mina Rezaei. 2025. Calibrating llms with information-theoretic evidential deep learning. *arXiv preprint arXiv:2502.06351*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Teaching models to express their uncertainty in words. *arXiv preprint arXiv:2205.14334*.

Zequan Liu, Jiawen Lyn, Wei Zhu, Xing Tian, and Yvette Graham. 2024. Alora: Allocating low-rank adaptation for fine-tuning large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 622–641.

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. 2022. Peft: State-of-the-art parameter-efficient fine-tuning methods.

Patryk Marszałek, Klaudia Bałazy, Jacek Tabor, and Tomasz Kuśmierczyk. 2025. Minimal ranks, maximum confidence: parameter-efficient uncertainty quantification for lora. In *Findings of the Association for Computational Linguistics: EMNLP 2025*. Association for Computational Linguistics.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 2381–2391.

Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. 2015. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29.

Lakshmana Sri Harsha Nemani, PK Srijith, and Tomasz Kuśmierczyk. 2025. Efficient uncertainty in llms through evidential knowledge distillation. *arXiv preprint arXiv:2507.18366*.

Ruijia Niu, Dongxia Wu, Rose Yu, and Yi-An Ma. 2024. Functional-level uncertainty quantification for calibrated fine-tuning on llms. *arXiv preprint arXiv:2410.06431*.

Amir Hossein Rahmati, Sanket Jantre, Weifeng Zhang, Yucheng Wang, Byung-Jun Yoon, Nathan Urban, and Xiaoning Qian. 2026. C-lora: Contextual low-rank adaptation for uncertainty estimation in large language models. *Advances in Neural Information Processing Systems*, 38:67459–67485.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106.

Artem Shelmanov, Maxim Panov, Roman Vashurin, Artem Vazhentsev, Ekaterina Fadeeva, and Timothy Baldwin. 2025. Uncertainty quantification for large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 5: Tutorial Abstracts)*, pages 3–4.

Haizhou Shi, Yibin Wang, Ligong Han, Huan Zhang, and Hao Wang. 2026. Training-free bayesianization for low-rank adapters of large language models. *Advances in Neural Information Processing Systems*, 38:41663–41700.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Xi Wang, Laurence Aitchison, and Maja Rudolph. 2023. Lora ensembles for large language model fine-tuning. *arXiv preprint arXiv:2310.00035*.

Yibin Wang, Haizhou Shi, Ligong Han, Dimitris Metaxas, and Hao Wang. 2024. Blob: Bayesian low-rank adaptation by backpropagation for large language models. *Advances in neural information processing systems*, 37:67758–67794.

Haotian Xiang, Bingcong Li, and Qin Lu. 2026. Scalable variational bayesian fine-tuning of llms via orthogonalized low-rank adapters. *arXiv preprint arXiv:2604.03388*.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2024. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. In *International*

Conference on Learning Representations, volume 2024, pages 23650–23678.

Adam Yang, Maxime Robeyns, Xi Wang, and Laurence Aitchison. 2024. Bayesian low-rank adaptation for large language models. In *international conference on learning representations*, volume 2024, pages 1812–1842.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. 2023. Adalora: Adaptive budget allocation for parameter-efficient fine-tuning. *arXiv preprint arXiv:2303.10512*.

Yuhua Zhou, Changhai Zhou, Shiyang Zhang, Fei Yang, Yi Zhang, and Aimin Pan. 2026. Lara: Layer-wise rank allocation for efficient fine-tuning of pruned large language models. *Information Processing & Management*, 63(3):104538.

A Implementation Details

A.1 Dataset Details

We provide dataset statistics in Table 3. AAO denotes the merged dataset of ARC-E, ARC-C, and OBQA, while WB denotes the merged dataset of Winogrande-Medium and BoolQ. The full Combined dataset merges all six commonsense reasoning benchmarks and uses the unified label set $\{\mathrm{A}, \mathrm{B}, \mathrm{C}, \mathrm{D}, \mathrm{E}, \mathrm{True}, \mathrm{False}\}$. Table 4 summarizes the prompt templates used for common-sense reasoning tasks.

A.2 Evaluation Metrics for Uncertainty Estimation

We evaluate predictive performance and uncertainty quality using Accuracy (ACC), Negative Log-Likelihood (NLL), and Expected Calibration Error (ECE) (Naeini et al., 2015). Given a test set $\mathcal{D}_{\text{test}} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, the predicted label is defined as $\hat{y}_i = \arg\max_y p_\theta(y \mid \mathbf{x}_i)$. Accuracy measures the fraction of correctly predicted examples:

$$\mathrm{ACC} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}(\hat{y}_i = y_i). \tag{6}$$

NLL evaluates the quality of the full predictive distribution by penalizing low probability assigned to the ground-truth label:

$$\mathrm{NLL} = -\frac{1}{N} \sum_{i=1}^{N} \log p_\theta(y_i \mid \mathbf{x}_i). \tag{7}$$

Compared with accuracy, NLL is more sensitive to over-confident wrong predictions, since assigning high confidence to an incorrect label increases the loss substantially.

ECE measures how well model confidence matches empirical correctness. Following standard practice, we partition predictions into N_{bin} confidence bins $\{B_b\}_{b=1}^{N_{\text{bin}}}$ and compute the weighted average gap between accuracy and confidence:

$$\mathrm{ECE} = \sum_{b=1}^{N_{\text{bin}}} \frac{|B_b|}{N} \left|\mathrm{acc}(B_b) - \mathrm{conf}(B_b)\right|. \tag{8}$$

Here, the accuracy and confidence of bin B_b are computed as

$$\mathrm{acc}(B_b) = \frac{1}{|B_b|} \sum_{i \in B_b} \mathbf{1}(\hat{y}_i = y_i), \tag{9}$$

$$\mathrm{conf}(B_b) = \frac{1}{|B_b|} \sum_{i \in B_b} \max_y p_\theta(y \mid \mathbf{x}_i). \tag{10}$$

In our experiments, we use $N_{\text{bin}} = 15$ bins for ECE computation. Higher ACC indicates better predictive performance, while lower NLL and ECE indicate better uncertainty estimation and calibration.

A.3 Bayesianization and Training

Shared Configuration. We use the same data splits, prompt templates, and evaluation protocol for all methods. Unless otherwise specified, all LoRA-based models are fine-tuned with AdamW using a batch size of 4, a linear learning-rate decay schedule, and a warmup ratio of 0.06. The maximum sequence length is set to 300 tokens. For standard LoRA adapters, we set the LoRA rank to $r = 8$ and the LoRA scaling factor to $\alpha = 16$. For sampling-based uncertainty estimation methods, we use $M = 10$ Monte Carlo samples during inference unless otherwise stated. All reported results are averaged over three random seeds when repeated runs are available.

Baseline Configuration. We compare DALorRA with representative deterministic and uncertainty-aware LoRA baselines. MLE and MAP are used as deterministic PEFT baselines, where MLE follows standard LoRA fine-tuning and MAP introduces prior-based regularization. MCD and Deep Ensemble are included as sampling- and ensemble-based baselines: MCD averages multiple predictions with dropout enabled at inference time, while Deep Ensemble averages independently fine-tuned LoRA models. LA is implemented as a post-hoc Bayesian method that applies a Laplace approximation to trained LoRA parameters.

For variational Bayesian baselines, we include BLoB, TFB, and C-LoRA. BLoB Bayesianizes the LoRA adapter by placing a Gaussian variational posterior of $\mathbf{A}$ matrix while keeping $\mathbf{B}$ deterministic (Wang et al., 2024). TFB is applied as a training-free Bayesianization method that converts trained LoRA adapters into Bayesian ones without additional gradient-based fine-tuning (Shi et al., 2026). C-LoRA follows its original contextual low-rank adaptation setting, where a lightweight contextual module is used to model input-dependent uncertainty in the low-rank space (Rahmati et al., 2026). All baselines are implemented following the protocols established in TFB (Shi et al., 2026).

Table 3: Dataset statistics.

	WG-S	ARC-C	ARC-E	WG-M	OBQA	BoolQ	AAO	WB	Chem	Phy	Combined
Size of Label Space	2	5	5	2	4	2	5	4	4	4	7
Size of Training Set	640	1,119	2,251	2,258	4,957	9,427	8,327	11,685	–	–	20,652
Size of Test Set	1,267	299	570	1,267	500	3,270	1,369	4,537	100	102	7,173

Table 4: Prompt templates for common sense reasoning tasks.

Task	Prompt
Winogrande (WG-S/WG-M)	Select one of the choices that answers the following question: {question} Choices: A. {option1}. B. {option2}. Answer:
ARC (ARC-C/ARC-E), OpenBookQA (OBQA), MMLU	Select one of the choices that answers the following question: {question} Choices: A. {choice1}. B. {choice2}. C. {choice3}. D. {choice4}. Answer:
BoolQ	Answer the question with only True or False: {question} Context: {passage}.

DALorRA Configuration. For DALorRA, we set the maximum allowable rank $r = 8$, scaling factor $\alpha = 16$, and dropout rate 0. All models are trained for 5,000 gradient steps with AdamW, learning rate 1×10^{-4}, batch size 4, warmup ratio 0.06, and maximum sequence length 300. For variational rank masking, we use a separate mask-logit learning rate of 1×10^{-2}, and set the prior Bernoulli probability. At inference time, we use $M = 10$ Bernoulli mask samples for Bayesian model averaging. All results are evaluated under the same protocol as the baseline methods.

B Additional Experimental Results

B.1 Performance on Llama2-7B

Table 5 reports additional results on Llama2-7B across six common-sense reasoning datasets. Compared with deterministic and ensemble-based baselines, DALorRA maintains competitive accuracy while achieving strong calibration performance. In particular, DALorRA obtains the best ECE on five out of six datasets and the second-best ECE on ARC-E, showing consistent improvements in confidence calibration. For NLL, DALorRA also achieves the best or competitive performance on most datasets, especially on ARC-C, WG-M, and OBQA. These results further confirm that posterior rank masking can improve uncertainty estimation without causing severe degradation in predictive accuracy.

B.2 Results on Combined Datasets

Figure 5 reports the results on combined datasets, where related benchmarks are merged according to their answer-space compatibility. Compared with the single-dataset results in Table 5, calibration metrics are generally improved on the merged datasets. In particular, ECE and NLL tend to decrease after merging related tasks, suggesting that increasing the effective training size may help alleviate overconfidence during LoRA fine-tuning.

Among the compared methods, our variants maintain competitive accuracy while achieving strong calibration performance. On the AAO and WB splits, our method achieves the lowest ECE, indicating that rank-level Bayesianization remains effective when related tasks are jointly trained. On the fully combined dataset, our amortized variant also obtains competitive ACC and NLL compared with BLoB, C-LoRA, and TFB variants.

These observations should not be interpreted as a strict causal conclusion, but they are consistent with prior findings that calibration can be affected by the effective training data scale and the amount of data available relative to model capacity (Wang et al., 2024; Cheon and Paik, 2026). Overall, the combined-dataset results suggest that DALorRA provides a favorable trade-off between predictive performance and calibration, while also indicating that data scale is an important factor in uncertainty estimation for LoRA-adapted LLMs.

B.3 Effect of the Prior Bernoulli Probability

We study the impact of the prior Bernoulli probability, p_0, on the performance of DALorRA. As illustrated in Figure 6, the performance is robust when p_0 varies within a reasonable range. The small fluctuation in performance with p_0 is theoret-

Table 5: Performance of different methods applied to LoRA on Llama2-7B pre-trained weights, where Accuracy (ACC) and Expected Calibration Error (ECE) are reported in percentages. The evaluation is done across six common-sense reasoning tasks with a shared hyper-parameter setting after 5,000 gradient steps. We use $M = 10$ samples during inference in all sampling-based methods, including BLoB (Wang et al., 2024), TFB (Shi et al., 2026), and C-LoRA (Rahmati et al., 2026). "↑" and "↓" indicate that higher and lower values are preferred, respectively. **Boldface** and underlining denote the best and the second-best performance, respectively.

Metric	Method	Data					
		WG-S	ARC-C	ARC-E	WG-M	OBQA	BoolQ
ACC (↑)	MAP	**69.37**$_{\pm 1.04}$	67.67$_{\pm 1.18}$	85.20$_{\pm 0.63}$	74.57$_{\pm 0.73}$	81.60$_{\pm 0.40}$	87.68$_{\pm 0.02}$
	MCD	69.06$_{\pm 1.40}$	66.66$_{\pm 2.30}$	85.49$_{\pm 0.74}$	75.89$_{\pm 0.48}$	81.46$_{\pm 0.92}$	87.67$_{\pm 0.08}$
	Deep Ensemble	68.98$_{\pm 0.97}$	**68.57**$_{\pm 2.11}$	**86.24**$_{\pm 1.26}$	**77.39**$_{\pm 1.08}$	**82.20**$_{\pm 0.91}$	**88.07**$_{\pm 0.17}$
	LA	68.18$_{\pm 1.04}$	64.17$_{\pm 0.97}$	85.30$_{\pm 0.97}$	74.15$_{\pm 0.40}$	77.53$_{\pm 0.80}$	86.45$_{\pm 0.35}$
	BLoB	66.55$_{\pm 0.61}$	66.66$_{\pm 2.25}$	84.56$_{\pm 0.20}$	73.38$_{\pm 0.29}$	81.44$_{\pm 0.53}$	86.63$_{\pm 0.50}$
	TFB	66.84$_{\pm 1.52}$	67.62$_{\pm 1.12}$	84.52$_{\pm 0.62}$	73.13$_{\pm 2.38}$	81.10$_{\pm 0.61}$	86.36$_{\pm 0.26}$
	C-LoRA	66.21$_{\pm 1.24}$	67.79$_{\pm 1.27}$	84.38$_{\pm 0.67}$	70.48$_{\pm 1.71}$	78.26$_{\pm 2.61}$	84.64$_{\pm 0.81}$
	DALorRA	66.61$_{\pm 0.87}$	67.57$_{\pm 0.96}$	85.04$_{\pm 0.52}$	73.73$_{\pm 0.75}$	81.80$_{\pm 0.43}$	86.44$_{\pm 0.28}$
ECE (↓)	MAP	29.76$_{\pm 1.08}$	30.60$_{\pm 1.26}$	13.49$_{\pm 0.63}$	23.01$_{\pm 0.44}$	15.30$_{\pm 0.11}$	5.93$_{\pm 0.36}$
	MCD	28.49$_{\pm 1.60}$	29.60$_{\pm 2.77}$	12.69$_{\pm 0.60}$	20.73$_{\pm 0.38}$	14.34$_{\pm 1.11}$	5.13$_{\pm 0.25}$
	Deep Ensemble	28.72$_{\pm 1.46}$	27.75$_{\pm 1.86}$	11.87$_{\pm 0.16}$	18.67$_{\pm 0.29}$	13.98$_{\pm 1.12}$	5.24$_{\pm 0.27}$
	LA	11.41$_{\pm 0.17}$	30.54$_{\pm 0.70}$	45.85$_{\pm 2.08}$	10.80$_{\pm 0.38}$	35.65$_{\pm 1.14}$	18.22$_{\pm 0.41}$
	BLoB	11.23$_{\pm 1.45}$	10.77$_{\pm 1.91}$	4.29$_{\pm 1.08}$	4.52$_{\pm 0.91}$	3.82$_{\pm 0.96}$	1.46$_{\pm 0.36}$
	TFB	9.36$_{\pm 1.02}$	7.37$_{\pm 0.21}$	**3.03**$_{\pm 0.43}$	4.07$_{\pm 1.65}$	5.94$_{\pm 0.46}$	5.37$_{\pm 0.44}$
	C-LoRA	7.86$_{\pm 3.99}$	8.83$_{\pm 1.20}$	4.27$_{\pm 1.24}$	3.71$_{\pm 1.30}$	4.00$_{\pm 0.84}$	1.62$_{\pm 0.44}$
	DALorRA	**7.84**$_{\pm 0.92}$	**7.04**$_{\pm 0.66}$	3.31$_{\pm 0.58}$	**2.91**$_{\pm 0.41}$	**3.72**$_{\pm 0.52}$	**1.14**$_{\pm 0.29}$
NLL (↓)	MAP	2.86$_{\pm 0.23}$	3.07$_{\pm 0.09}$	1.13$_{\pm 0.10}$	1.26$_{\pm 0.12}$	1.04$_{\pm 0.02}$	0.34$_{\pm 0.00}$
	MCD	2.50$_{\pm 0.12}$	2.81$_{\pm 0.25}$	1.13$_{\pm 0.04}$	1.16$_{\pm 0.03}$	1.01$_{\pm 0.07}$	0.32$_{\pm 0.00}$
	Deep Ensemble	2.44$_{\pm 0.23}$	2.20$_{\pm 0.03}$	0.91$_{\pm 0.05}$	1.04$_{\pm 0.09}$	0.87$_{\pm 0.03}$	0.32$_{\pm 0.00}$
	LA	**0.62**$_{\pm 0.00}$	1.17$_{\pm 0.01}$	0.97$_{\pm 0.05}$	0.56$_{\pm 0.00}$	0.98$_{\pm 0.01}$	0.45$_{\pm 0.00}$
	BLoB	0.66$_{\pm 0.01}$	0.88$_{\pm 0.03}$	0.44$_{\pm 0.00}$	0.54$_{\pm 0.00}$	0.51$_{\pm 0.01}$	**0.31**$_{\pm 0.01}$
	TFB	0.62$_{\pm 0.03}$	0.86$_{\pm 0.01}$	**0.42**$_{\pm 0.03}$	0.56$_{\pm 0.03}$	0.50$_{\pm 0.01}$	0.34$_{\pm 0.00}$
	C-LoRA	0.63$_{\pm 0.02}$	0.88$_{\pm 0.00}$	0.48$_{\pm 0.02}$	0.57$_{\pm 0.03}$	0.59$_{\pm 0.05}$	0.35$_{\pm 0.02}$
	DALorRA	0.62$_{\pm 0.02}$	**0.83**$_{\pm 0.02}$	0.45$_{\pm 0.01}$	**0.54**$_{\pm 0.01}$	**0.49**$_{\pm 0.02}$	0.32$_{\pm 0.00}$

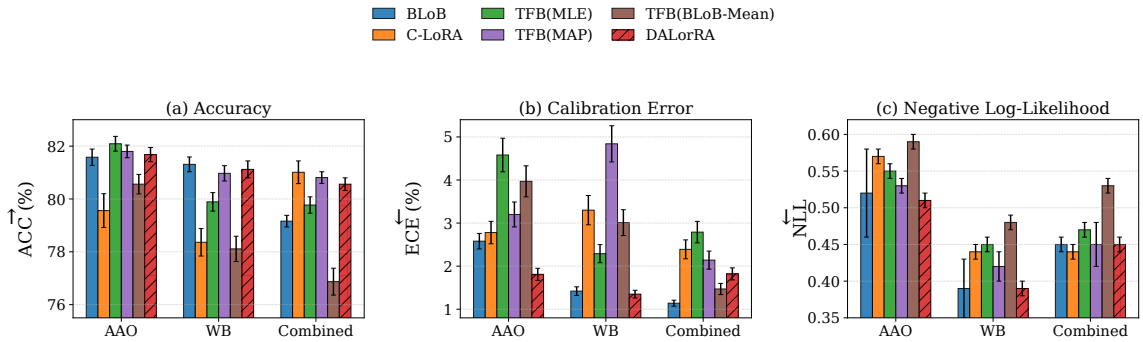

Figure 5: Performance comparison on combined datasets. We use $M = 5$ samples during inference in all sampling-based methods, including prior BLoB (Wang et al., 2024), TFB (Shi et al., 2026), and C-LoRA (Rahmati et al., 2026). AAO combines ARC-E, ARC-C, and OBQA; WB combines WG-M and BoolQ; and Combined merges all six benchmarks. DALorRA achieves strong calibration on AAO and WB while maintaining competitive accuracy across merged settings.

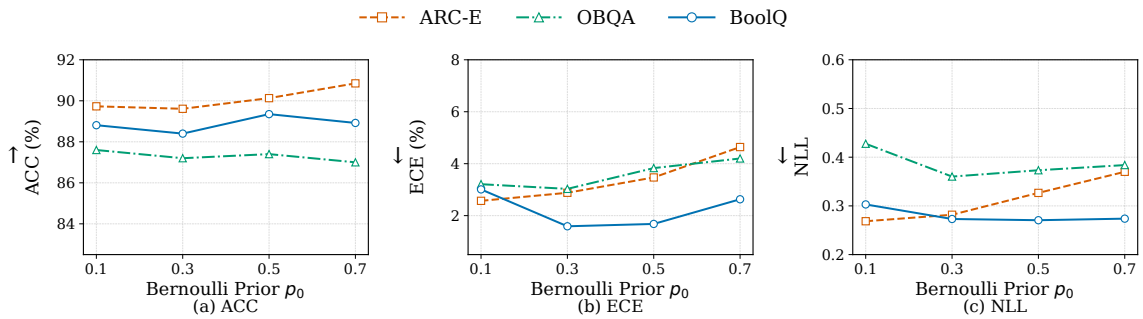

Figure 6: Sensitivity analysis of the prior Bernoulli probability p_0

Table 6: Efficiency Comparison among LoRA, BLoB, C-LoRA, and DALorRA. The experiments are conducted on a single NVIDIA A40 GPU and based on Llama-3.1-8B with rank $r = 8$, batch size 4, and 2,000 training iterations. The subscripts indicate the relative cost compared to LoRA,with red or green denoting worse or better efficiency.

Method	Trainable/Extra Params	ARC-C		WG-M		BoolQ	
		Train/Eval (s)	Mem. (MB)	Train/Eval (s)	Mem. (MB)	Train/Eval (s)	Mem. (MB)
LoRA	$4{,}466{,}688/0$	$1{,}977.63_{1.00\times}/38.28_{1.00\times}$	$11{,}462.94_{1.00\times}$	$1{,}266.59_{1.00\times}/105.79_{1.00\times}$	$7{,}755.07_{1.00\times}$	$3{,}530.78_{1.00\times}/756.36_{1.00\times}$	$14{,}279.04_{1.00\times}$
BLoB	$6{,}596{,}608_{1.48\times}/+2{,}129{,}920$	$2{,}034.89_{1.03\times}/383.57_{10.02\times}$	$13{,}092.71_{1.14\times}$	$1{,}316.69_{1.04\times}/1{,}046.98_{9.90\times}$	$8{,}306.96_{1.07\times}$	$3{,}600.76_{1.02\times}/7{,}581.51_{10.02\times}$	$16{,}708.24_{1.17\times}$
C-LoRA	$5{,}044{,}928_{1.13\times}/+578{,}240$	$5{,}015.91_{2.54\times}/194.37_{5.08\times}$	$16{,}283.28_{1.42\times}$	$4{,}676.12_{3.69\times}/738.95_{6.99\times}$	$11{,}338.00_{1.46\times}$	$5{,}717.40_{1.62\times}/2{,}684.16_{3.55\times}$	$20{,}601.88_{1.44\times}$
DALorRA (Ours)	$4{,}467{,}208_{1.0001\times}/+520$	$1{,}125.13_{0.57\times}/223.63_{5.84\times}$	$13{,}880.68_{1.21\times}$	$895.07_{0.71\times}/810.41_{7.66\times}$	$9{,}216.75_{1.19\times}$	$1{,}640.12_{0.46\times}/3{,}321.28_{4.39\times}$	$17{,}398.39_{1.22\times}$

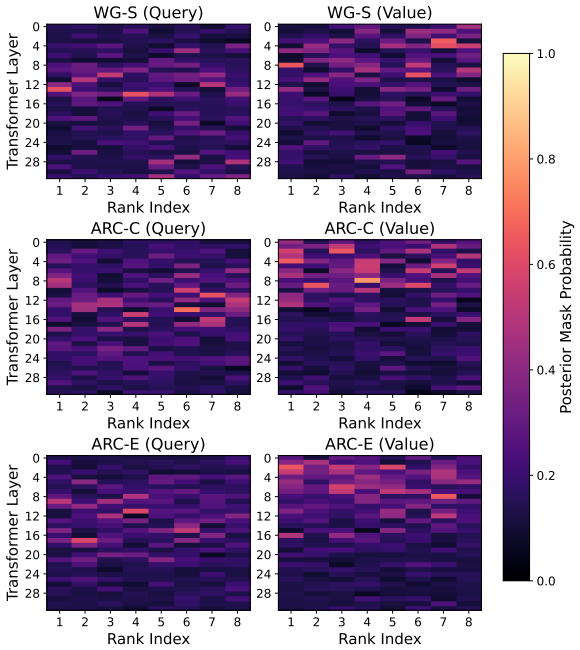

Figure 7: Posterior Bernoulli probabilities of DALorRA masks $\mathbf{D}$ on WG-S, ARC-C, and ARC-E.

ically expected, as our objective function strictly optimizes the exact evidence lower bound without introducing a hyperparameter to weight the Kullback-Leibler divergence. In contrast, variational inference approaches, such as C-LoRA, frequently rely on this hyperparameter—a practice functionally equivalent to manually selecting the prior distribution parameters. By adhering to the exact ELBO, DALorRA preserves theoretical rigor while avoiding hyperparameter tuning.

B.4 Efficiency and resource consumption

We compare the efficiency of the sampling-based uncertainty-aware fine-tuning methods (BLoB, C-LoRA, and DALorRA) with regular LoRA. As shown in Table 6, DALorRA adds only 520 extra trainable parameters, which is negligible compared with BLoB and C-LoRA. DALorRA takes the least training time, even faster than LoRA due to its sparse configuration; it uses only $0.57\times$, $0.71\times$, and $0.46\times$ training time of LoRA on ARC-C, WG-M, and BoolQ, respectively. DALorRA remains faster than BLoB and comparable to C-LoRA in evaluation (inference) time with moderate memory overhead. In addition, DALorRA's overall (training and evaluation) time is comparable to LoRA's on the three datasets. This shows that DALorRA achieves great calibration with minimal computational cost and favorable overall efficiency.