ADVANCING THE CALIBRATION-COMPUTE FRONTIER OF LLM CLASSIFIERS VIA ASYMMETRIC DUOS

Anonymous authors
Paper under double-blind review

ABSTRACT

The importance of reliable uncertainty estimation in large language models (LLMs) has become increasingly pronounced, yet modern models remain poorly calibrated and frequently overconfident. In this study, we address this by proposing an efficient method for LLM classifier calibration using the *Asymmetric Duos* framework. Our approach combines a strong base model with a smaller sidekick model via logit-space fusion, utilizing global scaling parameters optimized on a validation set. Across diverse benchmarks, this Asymmetric LLM Duo consistently achieves best or second-best performance on expected calibration error and negative log-likelihood, without degrading predictive accuracy. More importantly, these gains are realized at a fraction of the computational cost of more complex baselines, placing our method on the calibration-compute Pareto frontier. We further demonstrate that our method acts orthogonal to various established calibration-aware techniques, meaning it can be easily combined with them to consistently boost their performance. Finally, our analysis of the logit-space behavior illustrates how asymmetric fusion successfully mitigates base model overconfidence. Overall, our findings demonstrate that the benefits of an asymmetric fusion successfully transfer to the language domain, providing a simple, highly cost-efficient approach to improving uncertainty estimation in LLM classifiers.

1 INTRODUCTION

Large language models (LLMs) have demonstrated remarkable predictive performance across a wide range of tasks and domains (Brown et al., 2020; OpenAI, 2023; Touvron et al., 2023; Grattafiori et al., 2024; Minaee et al., 2024). These capabilities can be further increased for downstream tasks through fine-tuning methods (Hu et al., 2022; Yang et al., 2024a; Wang et al., 2024; Lialin et al., 2024; Wang et al., 2022). Despite the strong predictive performance of LLMs, these models often remain poorly calibrated and exhibit overconfidence (Guo et al., 2017; Xiong et al., 2024; Leng et al., 2025; Shorinwa et al., 2024), and fine-tuning procedures can further amplify this behavior (Jiang et al., 2020). LLMs being overconfident in their predictions can yield serious problems for critical decision-making domains (Kumar et al., 2023), and being overconfident can also undermine the reliability and restrict the use cases of LLMs in those domains (Li et al., 2025). Therefore, in real-world applications, these neural networks should also be able to indicate when they can be incorrect (Guo et al., 2017). This capability is known as uncertainty quantification or confidence calibration.

Overconfidence is not specific to LLMs. Modern neural networks are often poorly calibrated and exhibit overconfidence (Guo et al., 2017). One solution to this is the recently proposed *Asymmetric Duo* (Zhou et al., 2025). They showed that using a smaller sidekick neural network model together with a larger base model can improve the uncertainty estimates of those neural networks. Their results suggest that uncertainty estimation can be improved with only a modest computational overhead relative to the larger base model.

Since Zhou et al. (2025) exclusively focused on computer vision tasks and models, it remains unclear whether and how Asymmetric Duos can work for LLM classifiers. In this paper, we pursue this question for LLMs fine-tuned to multiple-choice question answering (MCQA). Specifically, we examine whether asymmetric fusion improves the calibration of LLMs in MCQA and whether it changes the predictive behavior of the base model in a systematic and cost-efficient way. This

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

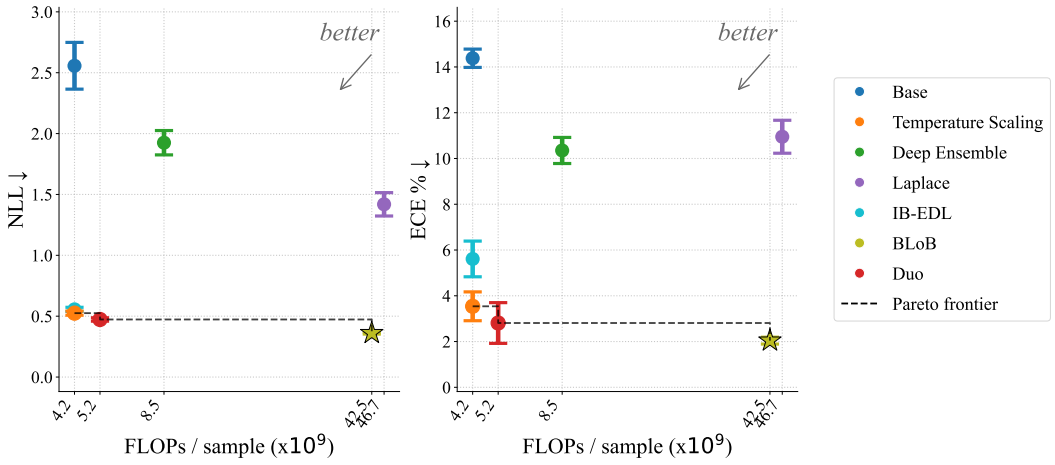

Figure 1: Pareto frontier of floating-point operations (FLOPs) vs. negative log-likelihood (NLL; left) and expected calibration error (ECE; right) for the CSQA dataset. For both metrics, Temperature Scaling, Duo, and BLoB lie on Pareto frontier. The method with the best NLL/ECE performance is marked with a star.

positions the paper at the intersection of LLM calibration, uncertainty-aware adaptation, and efficient multi-model uncertainty estimation. Our contributions are organized as follows:

1. **Cost-efficient calibration:** We show that Asymmetric Duos improve LLM calibration on the cost–performance frontier, matching or surpassing state-of-the-art baselines often at a fraction of the inference FLOPs across all benchmarks (cf. Figure 1).
2. **Compatibility with calibration-aware methods:** We show that Asymmetric Duos compose additively with existing calibration-aware approaches such as BLoB, Laplace-LoRA, and IB-EDL, consistently improving calibration performance when applied in composition with these methods.
3. **Logit-space mechanism:** Our mechanistic study sheds light on how asymmetric logit fusion reshapes the logit geometry of the base model, reducing overconfidence by smoothing extreme predictions while preserving decision boundaries, thereby improving calibration without degrading accuracy.

2 Related Work & Background

Calibration and uncertainty in neural networks and LLMs. The objective of confidence calibration is to match the confidence assigned by a model with the empirical probability of correctness. Although modern neural networks achieve strong predictive performance, this does not necessarily imply well-calibrated confidence estimates, and overconfidence remains a well-known issue (Guo et al., 2017; Jiang et al., 2020; Shorinwa et al., 2024). This has motivated substantial research on uncertainty estimation and post-hoc calibration, with temperature scaling being one of the most widely used and computationally simple approaches (Guo et al., 2017). Evidential methods instead aim to capture predictive uncertainty directly within the model (Sensoy et al., 2018).

Despite their strong performance, LLMs may produce unreliable uncertainty estimates, particularly after adaptation to downstream tasks (He et al., 2023; Li et al., 2025). Calibration is also affected by prompting, decoding strategies, and answer extraction, making it more complex than in classical neural networks (Zhang et al., 2024).

Multi-model uncertainty estimation and Asymmetric Duos. Ensemble-based approaches improve uncertainty estimation by combining predictions from multiple models, but standard deep ensembles are computationally expensive because they require training and evaluating several models

Algorithm 1 Asymmetric Duo training and inference

Require: $\mathcal{D}_{\text{train}}, \mathcal{D}_{\text{val}}, \mathcal{D}_{\text{test}}$, models $f_{\text{base}}, f_{\text{side}}$

Train $f_{\text{base}}, f_{\text{side}}$ on $\mathcal{D}_{\text{train}}$ ▷ *Independent fine-tuning*

for each $(\mathbf{x}_i, y_i) \in \mathcal{D}_{\text{val}}$ **do**

  $\mathbf{z}_{\text{base}}^{(i)} \leftarrow f_{\text{base}}(\mathbf{x}_i)$

  $\mathbf{z}_{\text{side}}^{(i)} \leftarrow f_{\text{side}}(\mathbf{x}_i)$ ▷ *Logit extraction*

end for

$(\alpha^*, \beta^*) = \arg\min_{\alpha,\beta \geq 0} \mathcal{L}(\alpha, \beta)$ ▷ *Weight optimization (Eqs. (1) and (2))*

for each $\mathbf{x} \in \mathcal{D}_{\text{test}}$ **do** ▷ *Inference*

  $\mathbf{z}_{\text{duo}}(\mathbf{x}) \leftarrow \alpha^* \mathbf{z}_{\text{base}}(\mathbf{x}) + \beta^* \mathbf{z}_{\text{side}}(\mathbf{x})$

  $\hat{y} \leftarrow \arg\max_{y \in \mathcal{Y}} \text{softmax}(\mathbf{z}_{\text{duo}}(\mathbf{x}))$

end for

for each prediction (Lakshminarayanan et al., 2017; Fort et al., 2019). This cost has motivated more efficient uncertainty-aware alternatives. In the LLM setting, recent work has focused on uncertainty estimation through LoRA ensembles and Bayesian parameter-efficient adaptation methods. For example, Laplace-LoRA applies a Laplace approximation to LoRA adapters for post-hoc Bayesian inference (Yang et al., 2024a; Daxberger et al., 2021), while BLoB performs Bayesian LoRA through variational inference during fine-tuning (Wang et al., 2024). Ensemble-based approaches for LoRA-adapted LLMs have also been explored (Balabanov & Linander, 2025; Wang et al., 2023).

Further background on MCQA as a testbed for uncertainty estimation and on the relationship between parameter-efficient fine-tuning and calibration-aware training is provided in Appendix A.

3 Asymmetric LLM Duos

Our work builds on the idea of Zhou et al. (2025) by studying whether asymmetric logit fusion transfers to LLM-based MCQA. In addition, we investigate whether the Asymmetric Duos setup can further improve performance when combined with existing calibration-aware methods such as BLoB and Laplace-LoRA.

3.1 Problem formulation

Let $\mathbf{x} \in \mathcal{X}$ denote the input and $y \in \mathcal{Y}$ the corresponding label, where $|\mathcal{Y}| = K$. We consider two predictive models: a base model f_{base} and a sidekick model f_{side}. Each model produces a logit vector over the label space, $\mathbf{z}_{\text{base}}(\mathbf{x}) \in \mathbb{R}^K$ and $\mathbf{z}_{\text{side}}(\mathbf{x}) \in \mathbb{R}^K$, which yield class probability vectors $p_{\text{base}}(y \mid \mathbf{x}) = \text{softmax}(\mathbf{z}_{\text{base}}(\mathbf{x}))$ and $p_{\text{side}}(y \mid \mathbf{x}) = \text{softmax}(\mathbf{z}_{\text{side}}(\mathbf{x}))$. Our goal is to construct a combined predictor that improves the quality of uncertainty estimates while maintaining predictive performance.

Following Zhou et al. (2025), we define the Asymmetric Duo predictor via a linear combination in logit space, $\mathbf{z}_{\text{duo}}(\mathbf{x}) = \alpha\, \mathbf{z}_{\text{base}}(\mathbf{x}) + \beta\, \mathbf{z}_{\text{side}}(\mathbf{x})$. These combined logits parameterize a categorical distribution over answer choices, with class probabilities given by $p_{\text{duo}}(y \mid \mathbf{x}) = \text{softmax}(\mathbf{z}_{\text{duo}}(\mathbf{x}))$. The final prediction is obtained as $\hat{y}(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} p_{\text{duo}}(y \mid \mathbf{x})$. The weights α and β are global parameters shared across all inputs. We deliberately restrict the fusion to global, input-independent weights in order to avoid introducing additional model complexity and training overhead.

3.2 Learning the fusion weights

Let $\mathcal{D}_{\text{val}} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$ denote the validation set. For each validation example, we precompute and store the aligned logit vectors $\mathbf{z}_{\text{base}}^{(i)}$ and $\mathbf{z}_{\text{side}}^{(i)}$. We then estimate the fusion scales α and β by solving the optimization problem

$$(\alpha^*, \beta^*) = \arg\min_{\alpha,\beta \geq 0} \mathcal{L}(\alpha, \beta), \tag{1}$$

where the validation negative log-likelihood (NLL) is defined as

$$\mathcal{L}(\alpha, \beta) = -\frac{1}{N} \sum_{i=1}^{N} \log \left[\mathrm{softmax} \left(\alpha \mathbf{z}_{\text{base}}^{(i)} + \beta \mathbf{z}_{\text{side}}^{(i)} \right) \right]_{y_i} . \tag{2}$$

In our implementation, this is a two-dimensional post-hoc optimization problem solved with L-BFGS-B, a quasi-Newton method for bound-constrained optimization (Zhu et al., 1997). The fusion scales are not constrained to sum to one. Instead, we only enforce non-negativity through lower bounds on α and β. Thus, α and β should be interpreted as effective logit scales rather than normalized mixture weights.

The logits of both the base and sidekick models are matched by example index before optimization, ensuring that the two models are combined on the same inputs. Since this procedure uses only saved validation predictions, the Duo can be constructed after independent fine-tuning without any joint training of the two models; Algorithm 1 gives the complete training and inference pipeline.

3.3 Interpretation

The learned weights α and β determine the relative contributions of the base and sidekick models across the dataset. Since the optimization is driven by validation NLL, the resulting combination reflects which model provides more informative or better-calibrated logits. If the sidekick model provides little useful information, the optimization assigns it a small weight, effectively recovering the base model. Conversely, when the sidekick model captures complementary patterns, its contribution increases and improves the Duo predictor. In our experiments, we empirically validate this behavior in Table 1, where we observe consistent improvements in NLL and expected calibration error (ECE) across all datasets.

Implicit per-model temperature scaling. Beyond inter-model weighting, the unconstrained non-negative parameterization of (α, β) has a direct calibration interpretation: each scale acts as an inverse temperature on its own model's logits. Concretely, $\alpha\, \mathbf{z}_{\text{base}}(\mathbf{x}) = \mathbf{z}_{\text{base}}(\mathbf{x})/T_{\text{base}}$ with $T_{\text{base}} = 1/\alpha$, and analogously for the sidekick model. The validation-NLL minimization in Eq. (1) therefore performs two operations in a single two-dimensional fit: (i) per-model temperature scaling (Guo et al., 2017), and (ii) inter-model relative weighting. This is in contrast with formulations that fix $\alpha + \beta = 1$, which pin the joint logit magnitude and decouple it from any temperature effect. By keeping (α, β) unconstrained on the non-negative orthant, Duo includes both calibration steps without an extra optimization stage. Consequently, a Duo built from temperature-scaled base and sidekick logits, $\mathbf{z}_{\text{duo}}(\mathbf{x}) = \alpha\, \mathbf{z}_{\text{base}}(\mathbf{x})/T_{\text{base}} + \beta\, \mathbf{z}_{\text{side}}(\mathbf{x})/T_{\text{side}}$, can always be reparameterized as $\tilde{\alpha}\, \mathbf{z}_{\text{base}}(\mathbf{x}) + \tilde{\beta}\, \mathbf{z}_{\text{side}}(\mathbf{x})$ with $\tilde{\alpha} = \alpha/T_{\text{base}}$ and $\tilde{\beta} = \beta/T_{\text{side}}$, effectively recovering the same predictor as the plain Duo.

Gains beyond temperature scaling. To separate what a single temperature achieves from what the Sidekick adds, we evaluate each base method (plain LoRA, LA, and IB-EDL) in three forms: its original logits (Base), Base TS with one learned scale, and Duo with two learned scales. Base TS $-$ Base then measures temperature scaling alone, and Duo $-$ Base TS the additional contribution of the sidekick (Table 4). For plain LoRA and LA, temperature scaling provides most of the NLL improvement (e.g., -1.579 on ARC-C for plain LoRA), and Duo adds a further 0.003–0.054. For IB-EDL, whose evidential objective already calibrates the temperature, TS changes NLL by at most 0.014, yet Duo still reduces it by up to 0.064 (CSQA). Since Duo improves on Base TS in all 18 method–dataset combinations, the Sidekick contributes information that no rescaling of Base can provide.

4 Experiments

We now study our adaptation of the Asymmetric Duo for LLM MCQA.

Setup. Unless stated otherwise, we use Qwen2-7B as Base (B) and Qwen2-1.5B as Sidekick (S) (Yang et al., 2024b), both fine-tuned with LoRA (Hu et al., 2022), on six MCQA benchmarks:

Table 1: Calibration performance of baseline and Duo methods fine-tuned on Qwen2 7B ("B" for Base) and Qwen2 1.5B ("S" for Sidekick). Arrows (↑ or ↓) indicate the direction of better performance for each metric. The best results are highlighted in **bold**, and the second-best results are underlined. Additionally, ± denotes the standard deviation over 5 runs with 5 different seeds. The Duo method has either the best or the second-best performance across most of the datasets and metrics.

Metric	Method	ARC-C	ARC-E	CSQA	OBQA	RACE	SciQ
NLL ↓	S	$4.234_{\pm 0.046}$	$1.968_{\pm 0.100}$	$3.493_{\pm 0.182}$	$1.497_{\pm 0.076}$	$0.590_{\pm 0.021}$	$0.546_{\pm 0.022}$
	B	$2.075_{\pm 0.041}$	$0.655_{\pm 0.065}$	$2.557_{\pm 0.192}$	$0.990_{\pm 0.080}$	$0.340_{\pm 0.009}$	$0.333_{\pm 0.018}$
	B TS	$0.496_{\pm 0.011}$	$0.213_{\pm 0.015}$	$0.525_{\pm 0.017}$	$0.357_{\pm 0.012}$	$0.268_{\pm 0.005}$	$0.183_{\pm 0.008}$
	B DE	$1.682_{\pm 0.069}$	$0.538_{\pm 0.029}$	$1.925_{\pm 0.100}$	$0.750_{\pm 0.054}$	$0.319_{\pm 0.006}$	$0.295_{\pm 0.007}$
	B LA	$1.212_{\pm 0.052}$	$0.401_{\pm 0.029}$	$1.419_{\pm 0.096}$	$0.602_{\pm 0.068}$	$0.308_{\pm 0.006}$	$0.259_{\pm 0.016}$
	B IB-EDL	$0.547_{\pm 0.018}$	$0.220_{\pm 0.021}$	$0.556_{\pm 0.017}$	$0.368_{\pm 0.016}$	$0.267_{\pm 0.003}$	$0.205_{\pm 0.005}$
	B BLoB	$1.347_{\pm 0.101}$	$\mathbf{0.183_{\pm 0.039}}$	$\mathbf{0.361_{\pm 0.008}}$	$\mathbf{0.240_{\pm 0.016}}$	$\mathbf{0.247_{\pm 0.009}}$	$\mathbf{0.136_{\pm 0.002}}$
	Duo	$\mathbf{0.469_{\pm 0.009}}$	$0.188_{\pm 0.014}$	$0.473_{\pm 0.014}$	$0.302_{\pm 0.007}$	$0.262_{\pm 0.006}$	$0.167_{\pm 0.006}$
Acc ↑	S	$71.54_{\pm 0.90}$	$85.51_{\pm 0.61}$	$74.63_{\pm 0.27}$	$80.32_{\pm 0.52}$	$84.29_{\pm 0.42}$	$90.04_{\pm 0.15}$
	B	$86.59_{\pm 0.52}$	$95.08_{\pm 0.40}$	$85.34_{\pm 0.41}$	$90.24_{\pm 0.17}$	$91.89_{\pm 0.11}$	$94.96_{\pm 0.21}$
	B TS	$86.59_{\pm 0.52}$	$95.08_{\pm 0.40}$	$85.34_{\pm 0.41}$	$90.24_{\pm 0.17}$	$91.89_{\pm 0.11}$	$94.96_{\pm 0.21}$
	B DE	$\mathbf{87.15_{\pm 0.43}}$	$95.09_{\pm 0.14}$	$85.81_{\pm 0.35}$	$90.20_{\pm 0.31}$	$\mathbf{91.94_{\pm 0.23}}$	$95.12_{\pm 0.25}$
	B LA	$86.66_{\pm 0.48}$	$95.11_{\pm 0.32}$	$85.32_{\pm 0.42}$	$90.04_{\pm 0.14}$	$91.80_{\pm 0.15}$	$94.84_{\pm 0.23}$
	B IB-EDL	$86.28_{\pm 0.60}$	$\mathbf{95.23_{\pm 0.17}}$	$85.39_{\pm 0.52}$	$90.84_{\pm 0.38}$	$91.85_{\pm 0.26}$	$95.02_{\pm 0.22}$
	B BLoB	$86.31_{\pm 0.67}$	$94.89_{\pm 0.38}$	$\mathbf{86.55_{\pm 0.61}}$	$\mathbf{91.84_{\pm 0.83}}$	$91.57_{\pm 0.59}$	$\mathbf{95.42_{\pm 0.08}}$
	Duo	$86.81_{\pm 0.49}$	$95.14_{\pm 0.40}$	$85.73_{\pm 0.42}$	$90.60_{\pm 0.14}$	$91.82_{\pm 0.14}$	$95.24_{\pm 0.26}$
ECE ↓	S	$27.73_{\pm 0.96}$	$14.16_{\pm 0.59}$	$24.61_{\pm 0.36}$	$18.57_{\pm 0.65}$	$10.11_{\pm 0.47}$	$8.69_{\pm 0.14}$
	B	$13.05_{\pm 0.52}$	$4.74_{\pm 0.42}$	$14.38_{\pm 0.40}$	$9.55_{\pm 0.32}$	$5.63_{\pm 0.15}$	$4.75_{\pm 0.13}$
	B TS	$4.35_{\pm 0.84}$	$1.53_{\pm 0.21}$	$3.54_{\pm 0.63}$	$4.13_{\pm 1.07}$	$2.73_{\pm 0.30}$	$3.07_{\pm 0.53}$
	B DE	$10.11_{\pm 0.93}$	$3.76_{\pm 0.13}$	$10.35_{\pm 0.57}$	$7.14_{\pm 0.22}$	$5.17_{\pm 0.25}$	$4.12_{\pm 0.07}$
	B LA	$9.79_{\pm 0.39}$	$3.55_{\pm 0.31}$	$10.95_{\pm 0.72}$	$7.03_{\pm 0.42}$	$4.77_{\pm 0.23}$	$3.81_{\pm 0.25}$
	B IB-EDL	$4.88_{\pm 1.04}$	$3.90_{\pm 1.01}$	$5.61_{\pm 0.78}$	$9.37_{\pm 1.20}$	$3.84_{\pm 0.55}$	$2.89_{\pm 0.65}$
	B BLoB	$12.27_{\pm 0.67}$	$5.48_{\pm 2.90}$	$\mathbf{2.04_{\pm 0.15}}$	$\mathbf{2.71_{\pm 0.75}}$	$\mathbf{2.17_{\pm 0.31}}$	$\mathbf{1.55_{\pm 0.38}}$
	Duo	$\mathbf{3.90_{\pm 1.44}}$	$\mathbf{1.30_{\pm 0.15}}$	$2.81_{\pm 0.89}$	$3.09_{\pm 0.53}$	$2.78_{\pm 0.37}$	$1.94_{\pm 0.42}$

ARC-C and ARC-E (Clark et al., 2018), OpenBookQA (OBQA) (Mihaylov et al., 2018), CommonsenseQA (CSQA) (Talmor et al., 2019), SciQ (Welbl et al., 2017), and RACE (Lai et al., 2017). We compare against the fine-tuned Base (B), its temperature-scaled version (B TS) (Guo et al., 2017), a two-member deep ensemble (B DE) (Lakshminarayanan et al., 2017), Laplace-LoRA (B LA) (Yang et al., 2024a), IB-EDL (B IB-EDL) (Li et al., 2025), and BLoB (B BLoB) (Wang et al., 2024). An effective uncertainty-aware method should both enhance model calibration and maintain predictive accuracy comparable to standard MAP training (Li et al., 2025). Therefore, we use accuracy (Acc), ECE, and NLL as mean ± SD over five seeds, and measure inference cost in FLOPs, summing Base and Sidekick for Duo. Further details are given in Appendix E.

4.1 Cost-efficient uncertainty quantification

As shown in Table 1, Duo is highly competitive across all metrics and datasets compared to baselines. Duo achieves the best NLL on the ARC-C dataset, providing an almost 78% decrease on average compared to Base ($2.075 \rightarrow 0.469$). For the other five datasets, Duo achieves the second best performance, with only small gaps relative to the best performing method; for example, on ARC-E, B BLoB obtains an average NLL of 0.183, while Duo achieves 0.188 and their confidence intervals overlap. Calibration gains also hold for ECE. Duo provides a reduction of approximately $60 - 80\%$ compared to Base. Additionally, for five of the six datasets, Duo is either the best or second-best-performing method, with RACE being the only dataset on which it does not achieve this ranking. However, its standard deviation band overlaps with that of the second-best-performing method, which is B TS. Despite consistently improving the calibration, Duo also preserves the predictive accuracy of Base. Across all datasets, Duo either improves accuracy or maintains it at a comparable level. From these perspectives, it can be considered as an effective method of predicting uncertainty.

Other model pairings. To test whether these results depend on the Qwen2-7B/1.5B pairing, we repeat the experiment with three further pairings, comparing against B, B TS, and B DE (Tables 5–7); LA, IB-EDL, and BLoB are omitted because each requires method-specific training or posterior

fitting for every model and dataset. (i) With a smaller Sidekick, Qwen2-0.5B, Duo still attains the lowest NLL and ECE on all six datasets; its NLL gain over B TS is about 30–50% of that obtained with the 1.5B Sidekick, so a larger Sidekick helps but is not required. (ii) With a second model family, Llama-3.1-8B Base and Llama-3.2-1B Sidekick, Duo attains the lowest NLL on all six datasets and the lowest ECE on five (within one SD of B TS on ARC-C), even though the Sidekick is far weaker than Base (e.g., 52.4% vs. 80.3% accuracy on ARC-C). (iii) In a cross-family pairing, Llama-3.1-8B Base and Qwen2-1.5B Sidekick, Duo improves over B TS in both NLL and ECE on every dataset, with the largest NLL gains of all pairings (7–20%), and outperforms the Llama–Llama Duo in NLL and accuracy on all six datasets; as this Sidekick is also larger (1.5B vs. 1B), the comparison does not separate model family from Sidekick size. Overall, Duo improves on B TS in NLL in all 24 dataset–pairing combinations and in ECE in 22, changes accuracy by at most 0.4 points, and is far better calibrated than the twice-as-expensive B DE in every setting. Its gains are therefore not specific to one model pair or family.

Finally, we evaluated the sensitivity of Duo to the amount of validation data used to fit its coefficients. Refitting (α, β) on random 20% and 40% subsets of the validation set (ten draws each) leaves the coefficients and test NLL nearly unchanged; at 20%, test NLL is within 0.011 of its full-data value on five datasets (Tables 8 and 9). The exception is ARC-C, where 20% amounts to about 60 examples, and the variance grows. ECE follows the same overall trend but is noisier. These results suggest that a few hundred labeled examples suffice.

FLOP-Performance tradeoff. The main competitiveness of Duo comes from its computational efficiency. Unlike other uncertainty-aware methods that require costly ensembles, repeated stochastic forward passes, or additional Bayesian approximation procedures, Duo achieves similar, and sometimes better, calibration improvements with substantially lower computational overhead. At the same time, it preserves the predictive accuracy of the Base model, making it a practical and scalable alternative for an uncertainty-aware calibration method. On CSQA (Figure 1), B BLoB achieves the best NLL and ECE, but Duo also lies on the Pareto frontier at roughly one eighth of its cost; it lies on or next to the frontier on the other datasets as well (Appendix B.1).

Taken together, these results demonstrate that Asymmetric Duo is flexible across different model configurations and represents a promising approach to improve calibration with only a modest increase in computational cost in this domain.

4.2 Compatibility with calibration-aware methods

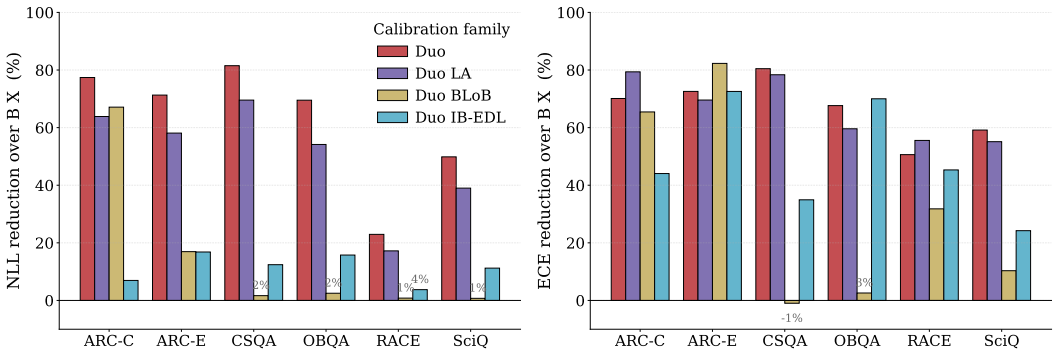

Figure 2: Relative reduction (%) of NLL (left) and ECE (right) achieved by Duo X over its base predictor B X across the six MCQA benchmarks.

We next ask whether the Duo procedure composes additively with calibration-aware baseline methods. For each calibration family $X \in \{\text{LA}, \text{BLoB}, \text{IB-EDL}\}$ we form a Duo whose base and sidekick models are calibrated with X and fit (α, β) on the validation split, at $\approx 1.22\times$ the inference FLOPs of B X. Figure 2 reports the relative NLL and ECE reduction of Duo X over B X. Full numerical results, including standard deviations and accuracy, are deferred to Table 13 in Appendix D.1. We also exclude a Duo formed from temperature-scaled base and sidekick models, as it reduces to the

Table 2: Ablation on the Duo fusion weights. Trivial Duo (Duo (0.5)) with $\alpha = \beta = 0.5$ (logit averaging) skips the Duo optimization stage. The best method per metric–dataset combination is marked in **bold**. All metrics are mean ± SD over 5 repetitions.

Metric	Method	ARC-C	ARC-E	CSQA	OBQA	RACE	SciQ
NLL ↓	Duo (0.5)	$1.442_{\pm 0.052}$	$0.459_{\pm 0.042}$	$1.489_{\pm 0.103}$	$0.523_{\pm 0.029}$	$0.315_{\pm 0.008}$	$0.236_{\pm 0.012}$
	Duo	$\mathbf{0.469}_{\pm 0.009}$	$\mathbf{0.188}_{\pm 0.014}$	$\mathbf{0.473}_{\pm 0.014}$	$\mathbf{0.302}_{\pm 0.007}$	$\mathbf{0.262}_{\pm 0.006}$	$\mathbf{0.167}_{\pm 0.006}$
Acc ↑	Duo (0.5)	$84.32_{\pm 0.84}$	$93.48_{\pm 0.48}$	$85.11_{\pm 0.54}$	$90.36_{\pm 0.36}$	$91.03_{\pm 0.17}$	$95.14_{\pm 0.27}$
	Duo	$\mathbf{86.81}_{\pm 0.49}$	$\mathbf{95.14}_{\pm 0.40}$	$\mathbf{85.73}_{\pm 0.42}$	$\mathbf{90.60}_{\pm 0.14}$	$\mathbf{91.82}_{\pm 0.14}$	$\mathbf{95.24}_{\pm 0.26}$
ECE ↓	Duo (0.5)	$11.64_{\pm 0.67}$	$4.30_{\pm 0.36}$	$12.63_{\pm 0.36}$	$6.71_{\pm 0.27}$	$4.46_{\pm 0.21}$	$3.43_{\pm 0.43}$
	Duo	$\mathbf{3.90}_{\pm 1.44}$	$\mathbf{1.30}_{\pm 0.15}$	$\mathbf{2.81}_{\pm 0.89}$	$\mathbf{3.09}_{\pm 0.53}$	$\mathbf{2.78}_{\pm 0.37}$	$\mathbf{1.94}_{\pm 0.42}$

standard Duo by reparameterization (cf. Section 3.3). The three families behave differently, and the improvements separate into two regimes set by how well-calibrated the underlying base already is.

Laplace-LoRA only mildly recalibrates the Base, so B LA remains overconfident. Duo LA over B LA produces large reductions in NLL and ECE with non-overlapping SD bands in Table 13, mirroring the Duo-over-Base improvements quantified in Section 4.1. In both cases, the optimizer converges to $\alpha, \beta \ll 1$ (Table 14), so Duo is performing global temperature scaling in addition to combining the two predictors.

On the other hand, BLoB and IB-EDL already calibrate the base model substantially before any fusion, in several cells matching or beating plain Duo, so the room left for Duo to improve is smaller and many cells of Table 13 now have margins within one SD. The optimizer does not compress these base models; it converges to $\alpha \approx 1$ for both families and instead uses β to decide how much to incorporate the Sidekick. For Duo BLoB, β collapses to (near) zero on RACE and SciQ, so Duo BLoB reduces to B BLoB on those datasets (Figure 2). For Duo IB-EDL, β stays above zero, so the sidekick still contributes, and the relative gains remain consistent but smaller than those for LA.

4.3 Ablation: Trivial Duos

The Duo optimization stage is essential. Table 2 replaces the learned (α, β) with the setting, $\alpha = \beta = 0.5$ (logit averaging). Unlike the optimized Duo formulation, this variant relies on fixed fusion coefficients and therefore cannot benefit from either global temperature scaling or asymmetric model weighting.

As a result, NLL and ECE degrade markedly on every benchmark: compared to the optimized Duo, trivial averaging increases NLL by 1.2–3.2× and ECE by 1.6–4.5×. Because a common scale cannot change the argmax, it also inherits the accuracy of unweighted fusion, which falls below Base on four of six datasets; only the validation-fitted weights down-weight the weaker Sidekick.

5 Mechanistic logit-space analysis of Asymmetric Duos

In this section, we analyze how Duo improves the calibration of LLMs by examining changes in their logit-level behavior. In particular, we compare the logit distributions of the base and sidekick models and Duo. For clarity, we focus on the CSQA dataset using a single seed, while results for the remaining datasets are provided in Figures 10–19 in Appendix B.2 and Table 15 in Appendix D.3.

Let $\mathcal{D}_{\text{test}} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$ denote the test set, where $\mathbf{x}_i$ is the input and $y_i \in \{1, \ldots, K\}$ is the correct class. For each input $\mathbf{x}_i$, the model outputs logits $\mathbf{z}_i = (z_{i,1}, z_{i,2}, \ldots, z_{i,K}) \in \mathbb{R}^K$, where $z_{i,k}$ denotes the logit assigned to class k. The predicted class is $\hat{y}_i = \arg\max_{k \in \{1,\ldots,K\}} p_{i,k}$, and the corresponding predictive probabilities are obtained using the softmax function $p_{i,k} = \text{softmax}(\mathbf{z}_i)$.

We consider the following three quantities in our analysis:

- The logit margin, $z_{i,y_i} - \max_{k \neq y_i} z_{i,k}$, which measures the gap between the correct-class logit and the highest incorrect-class logit. Positive values indicate that the correct class receives the highest logit, and negative values indicate that at least one incorrect class receives a higher logit;
- confidence, $p_{i,\hat{y}_i} = \max_k p_{i,k}$, which is the probability assigned to the predicted class;

- the correct-class probability, p_{i,y_i}, which is the probability assigned to the correct class.

5.1 Duo softens Base's overconfidence by compressing logit margins

The confidence CDF in Figure 3 (right) shows that the fine-tuned Base places nearly all probability mass on a single option, which explains its poor NLL and ECE despite high accuracy (Table 1). Figure 3 illustrates how Duo mitigates this overconfidence at the logit level. Duo compresses the logit margin toward zero (left; median $22.1 \rightarrow 3.8$) and yields a more spread-out distribution of confidence and correct-class probability (right). This behavior follows from the unconstrained fusion mechanism: since the weights α and β are optimized without enforcing a simplex constraint, the resulting logits can be rescaled relative to the original models, leading to a softer predictive distribution. Sections 5.2 and 5.4 isolate the part contributed by the Sidekick.

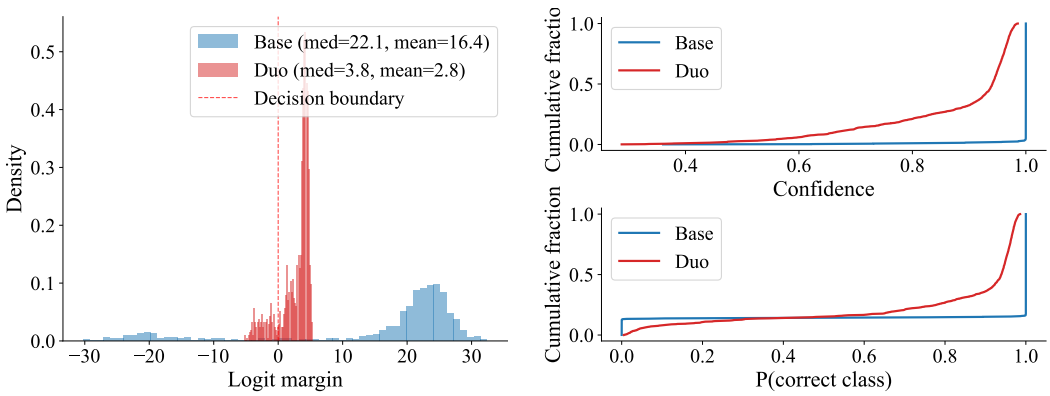

Figure 3: Logit margin (left) and empirical cumulative distribution functions (CDFs) of probability and confidence (right) for Duo and Base on the CSQA dataset.

5.2 Disentangling scaling and asymmetric weighting

Duo's coefficients control both global logit scaling and the relative contributions of the Base and Sidekick. To separate these effects, we introduce the nested ablation in Table 11. Base TS isolates global scaling, symmetric learned fusion adds the Sidekick under an equal-weight constraint, and Duo relaxes this constraint by learning separate coefficients. All variants are fitted on validation NLL using the same optimization procedure. Duo oracle is fitted directly on the test set and serves only as a lower-bound reference.

As shown in Table 12, Duo achieves the lowest NLL among the validation-fitted methods and outperforms symmetric learned fusion across all datasets. Thus, asymmetric weighting provides an additional benefit beyond global scaling and symmetric fusion. Duo also remains within the bounds of Duo oracle, indicating that its coefficients generalize well to the test data.

5.3 Calibration gains come from the wrong predictions of Base

Table 3 splits NLL and ECE by whether Base classifies a test example correctly. A clear asymmetry can be observed. Compressing the margin slightly lowers p_{i,y_i} on examples that Base already classifies correctly (NLL $0.003 \rightarrow 0.121$). The major improvement comes from misclassified samples: under Base, the strongest competing logit dominates by a wide margin, resulting in a significantly small p_{i,y_i} and producing a large NLL contribution (17.705) that can outweigh correct samples. Duo's compressed margins increase p_{i,y_i} on these samples, collapsing the per-sample NLL to 2.552, roughly a seven-fold

Table 3: NLL and ECE of Base and Duo, computed on the CSQA test samples that Base classifies correctly/incorrectly.

Metric	Method	Base Correct	Base Wrong
NLL ↓	Base	**0.003**	17.705
	Duo	0.121	**2.552**
ECE ↓	Base	**0.28**	98.46
	Duo	9.18	**65.59**

reduction. At the same time, Duo recovers the argmax of the Base in most cases, so accuracy (Table 1) remains largely unchanged or slightly improves. Because NLL is unbounded above, the per-sample improvement on incorrect predictions far exceeds the per-sample degradation on correct ones, and the mean drops sharply. The same Base-Correct/Base-Wrong asymmetry appears in ECE ($0.28 \rightarrow 9.18$ versus $98.46 \rightarrow 65.59$), and the mechanism is the same redistribution of probability mass: Base concentrates nearly all of its mass in the top-confidence bin, where the gap between $p_{i,\hat{y}_i}$ and empirical bin accuracy is widest. Duo spreads this mass into intermediate-confidence bins whose accuracy more closely matches their stated confidence, shrinking the calibration gap precisely where it dominates the metric.

5.4 Confidence adjustment versus prediction changes

To understand whether Duo's gains arise from changing predictions or adjusting confidence, we partition the test set according to whether Duo changes the Base prediction. Changed predictions are further classified as wrong-to-right or right-to-wrong. We compare Duo with Base TS to exclude improvements attributable solely to temperature scaling. Since temperature scaling preserves the predicted class, the partition is identical for Base and Base TS.

As shown in Table 10, Duo changes the predicted class on only 0.5%–2.1% of test examples. Nevertheless, examples with unchanged predictions account for 89.1%–103.6% of the total NLL improvement. The share exceeds 100% on RACE because the changed-prediction subset contributes negatively. Wrong-to-right flips generally outnumber right-to-wrong flips, except on RACE, explaining the small and dataset-dependent changes in accuracy. Overall, Duo's calibration gains arise primarily from adjusting confidence while retaining the Base prediction. Because ECE depends on confidence bins, it does not admit the same additive decomposition as NLL.

6 Discussion

In this study, we propose and study an adaptation of the Asymmetric Duos framework for uncertainty estimation in LLM classifiers as a simple yet cost-efficient alternative to existing methods. Across six MCQA benchmarks, the Duo framework consistently achieves the best or second-best calibration performance, ranking among the top methods in uncertainty quantification metrics such as ECE and NLL, while maintaining or slightly improving the base model's accuracy. Moreover, the Asymmetric Duo framework achieves these improvements at a fraction of the computational cost of stronger baselines such as BLoB, which often achieves better results but requires substantially higher compute. In contrast, the Duo method lies close to the Pareto frontier, offering a favorable trade-off between calibration quality and efficiency.

Beyond standalone performance, we also show that the Duo framework composes additively with other calibration-aware methods. It provides consistent improvements when applied on top of approaches such as Laplace-LoRA, IB-EDL, and BLoB. This suggests that asymmetric fusion of two distinct models captures complementary information that existing single-model techniques do not exploit.

Overall, these results position Asymmetric Duos as a simple yet very effective and broadly applicable approach to improving uncertainty estimation in LLM classifiers. By combining strong performance, low computational overhead, and compatibility with existing calibration techniques, the method offers a practical alternative to more complex and resource-intensive approaches.

Limitations and future work. Although our results demonstrate the effectiveness of the Asymmetric Duo framework, several limitations remain. First, we focus on MCQA as a controlled evaluation setting. Extending the analysis to tasks such as open-ended generation or sentiment classification could provide further insights, although defining calibration for open-ended sequences remains challenging. Second, we do not evaluate current state-of-the-art large-scale models. Such experiments could clarify the scalability of the approach but would require substantial computational resources and access to closed-source models. Finally, Duo requires a separate, smaller Sidekick model, which introduces additional memory overhead. Future work could address this limitation by using early exits from the Base model's layers to perform the role of the Sidekick.

AI USE STATEMENT

In this work, we used generative AI tools to check grammar and improve the clarity of the content, adjust the presentation of visualizations, and assist with code debugging. We did not use these tools to formulate the research questions or hypotheses, design the methodology or experiments, generate data, or interpret the results. All AI-assisted text was reviewed and revised by the authors, and all suggested code changes were manually inspected and tested. The scientific findings and conclusions were determined by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work is a computational study based on publicly available datasets and pretrained models. It does not involve human participants or the collection of personal or sensitive data. We do not identify any specific ethical concerns beyond the general risks associated with large language models.

REPRODUCIBILITY STATEMENT

To support reproducibility, we release the complete codebase at https://anonymous.4open.science/r/asymmetric-llm-duos-DC28, with a README describing how to set up the environment, run the experiments, and reproduce all reported results. All datasets and pretrained models are publicly available; models, datasets, hyperparameters, and training details are listed in Appendix E. Every method is evaluated on every dataset with five random seeds.

REFERENCES

Oleksandr Balabanov and Hampus Linander. Uncertainty quantification in fine-tuned llms using lora ensembles. In *ICLR Workshop: Quantify Uncertainty and Hallucination in Foundation Models: The Next Frontier in Reliable AI*, 2025.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, 2020.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv:1803.05457v1*, 2018.

Erik Daxberger, Agustinus Kristiadi, Alexander Immer, Runa Eschenhagen, Matthias Bauer, and Philipp Hennig. Laplace redux – effortless bayesian deep learning. In *Advances in Neural Information Processing Systems*, 2021.

Ning Ding, Yujia Qin, Guang Yang, et al. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*, 5(3):220–235, 2023.

Stanislav Fort, Huiyi Hu, and Balaji Lakshminarayanan. Deep ensembles: A loss landscape perspective. *ArXiv*, abs/1912.02757, 2019.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Aman Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, et al. The llama 3 herd of models. *ArXiv*, abs/2407.21783, 2024.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. *ArXiv*, abs/1706.04599, 2017.

Guande He, Jianfei Chen, and Jun Zhu. Preserving pre-trained features helps calibrate fine-tuned language models. In *The Eleventh International Conference on Learning Representations*, 2023.

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. In *International Conference on Learning Representations*, 2022.

Zhengbao Jiang, J. Araki, Haibo Ding, and Graham Neubig. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977, 2020.

Bhawesh Kumar, Cha-Chen Lu, Gauri Gupta, Anil Palepu, David R. Bellamy, Ramesh Raskar, and Andrew L. Beam. Conformal prediction with large language models for multi-choice question answering. *ArXiv*, abs/2305.18404, 2023.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. RACE: Large-scale ReAding comprehension dataset from examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 785–794. Association for Computational Linguistics, September 2017.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, 2017.

Jixuan Leng, Chengsong Huang, Banghua Zhu, and Jiaxin Huang. Taming overconfidence in llms: Reward calibration in rlhf. In *The Thirteenth International Conference on Learning Representations*, 2025.

Yawei Li, David Rügamer, Bernd Bischl, and Mina Rezaei. Calibrating LLMs with information-theoretic evidential deep learning. In *The Thirteenth International Conference on Learning Representations*, 2025.

Vladislav Lialin, Vijeta Deshpande, Xiaowei Yao, and Anna Rumshisky. Scaling down to scale up: A guide to parameter-efficient fine-tuning. *ArXiv*, abs/2303.15647, 2024.

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, Benjamin Bossan, and Marian Tietz. PEFT: State-of-the-art parameter-efficient fine-tuning methods. https://github.com/huggingface/peft, 2022.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 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*, pp. 2381–2391. Association for Computational Linguistics, October-November 2018.

Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. Large language models: A survey. *ArXiv*, abs/2402.06196, 2024.

OpenAI. Gpt-4 technical report. *ArXiv*, abs/2303.08774, 2023.

Pouya Pezeshkpour and Estevam Hruschka. Large language models sensitivity to the order of options in multiple-choice questions. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 2006–2017. Association for Computational Linguistics, June 2024.

Mario Sanz-Guerrero, Minh Duc Bui, and Katharina von der Wense. Mind the gap: A closer look at tokenization for multiple-choice question answering with LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 19573–19583. Association for Computational Linguistics, November 2025.

Murat Sensoy, Lance Kaplan, and Melih Kandemir. Evidential deep learning to quantify classification uncertainty. In *Advances in Neural Information Processing Systems*, 2018.

Ola Shorinwa, Zhiting Mei, Justin Lidard, Allen Z. Ren, and Anirudha Majumdar. A survey on uncertainty quantification of large language models: Taxonomy, open research challenges, and future directions. *ArXiv*, abs/2412.05563, 2024.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. CommonsenseQA: A question answering challenge targeting commonsense knowledge. 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)*, pp. 4149–4158. Association for Computational Linguistics, June 2019.

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

Xi Wang, Laurence Aitchison, and Maja Rudolph. Lora ensembles for large language model fine-tuning. *ArXiv*, abs/2310.00035, 2023.

Xunzhi Wang, Zhuowei Zhang, Gaonan Chen, Qiongyu Li, Bitong Luo, Zhixin Han, Haotian Wang, Zhiyu Li, Hang Gao, and Mengting Hu. UBench: Benchmarking uncertainty in large language models with multiple choice questions. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 8076–8107. Association for Computational Linguistics, July 2025.

Yaqing Wang, Subhabrata Mukherjee, Xiaodong Liu, Jing Gao, Ahmed Awadallah, and Jianfeng Gao. List: Lite prompted self-training makes parameter-efficient few-shot learners. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pp. 2262–2281, 2022.

Yibin Wang, Haizhou Shi, Ligong Han, Dimitris Metaxas, and Hao Wang. Blob: Bayesian low-rank adaptation by backpropagation for large language models. In *Advances in Neural Information Processing Systems*, 2024.

Sheng-Lun Wei, Cheng-Kuang Wu, Hen-Hsen Huang, and Hsin-Hsi Chen. Unveiling selection biases: Exploring order and token sensitivity in large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 5598–5621. Association for Computational Linguistics, August 2024.

Johannes Welbl, Nelson F. Liu, and Matt Gardner. Crowdsourcing multiple choice science questions. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pp. 94–106. Association for Computational Linguistics, September 2017.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45. Association for Computational Linguistics, October 2020.

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

Adam X. Yang, Maxime Robeyns, Xi Wang, and Laurence Aitchison. Bayesian low-rank adaptation for large language models. In *The Twelfth International Conference on Learning Representations*, 2024a.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Ke-Yang Chen, Kexin Yang, Mei Li, Min Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yunyang Wan, Yunfei Chu, Zeyu Cui, Zhenru Zhang, and Zhi-Wei Fan. Qwen2 technical report. *ArXiv*, abs/2407.10671, 2024b.

Mozhi Zhang, Mianqiu Huang, Rundong Shi, Linsen Guo, Chong Peng, Peng Yan, Yaqian Zhou, and Xipeng Qiu. Calibrating the confidence of large language models by eliciting fidelity. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 2959–2979. Association for Computational Linguistics, November 2024.

Tim G. Zhou, Evan Shelhamer, and Geoff Pleiss. Asymmetric duos: Sidekicks improve uncertainty. In *Advances in Neural Information Processing Systems*, 2025.

Ciyou Zhu, Richard H Byrd, Peihuang Lu, and Jorge Nocedal. Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on Mathematical Software*, December 1997.

A Additional Related Work and Background

Multiple-choice question answering as a testbed for LLM uncertainty. MCQA is a prominent use case of LLMs for evaluating uncertainty since it reduces free-form generation to a finite decision space. This enables a meaningful comparison of confidence scores across different models and methods (Wang et al., 2025; Kumar et al., 2023). At the same time, MCQA results can be affected by answer-option ordering, token selection biases, and the tokenization used for next-token answer extraction (Pezeshkpour & Hruschka, 2024; Wei et al., 2024; Sanz-Guerrero et al., 2025). Despite these limitations, MCQA provides a suitable framework for controlled comparisons between models and calibration methods.

Fine-tuning, parameter-efficient adaptation, and calibration-aware training. Fine-tuning is widely used to adapt LLMs to downstream tasks, while parameter-efficient methods such as Low-Rank Adaptation (LoRA) reduce the computational cost of this process (Hu et al., 2022; Ding et al., 2023). However, adaptation and calibration serve different purposes: fine-tuning specializes a model for a task, whereas calibration improves the reliability of its confidence estimates. Fine-tuned language models can remain overconfident, motivating calibration-aware training methods in addition to standard adaptation (He et al., 2023). For example, Li et al. (2025) propose IB-EDL as a calibration-aware training method for fine-tuned LLMs.

B ADDITIONAL FIGURES

All figures in this appendix use the main configuration (Qwen2-7B Base, Qwen2-1.5B Sidekick).

B.1 PARETO FRONTIERS

The per-dataset Pareto frontiers in Figures 4–8 confirm that the cost-calibration story discussed for CSQA in the main text is not dataset-specific. Across all six benchmarks, Duo sits in the lower-left "elbow" alongside B TS, achieving comparable NLL and ECE at $\approx 1.22\times$ the FLOPs of the base model and at $\approx 1/8$ the FLOPs of B BLoB. The one partial exception is RACE, where B BLoB and B TS edge Duo on ECE by a small margin (within Duo's standard-deviation band), so Duo is no longer strictly Pareto-dominant there but still lies on the low-cost segment of the frontier.

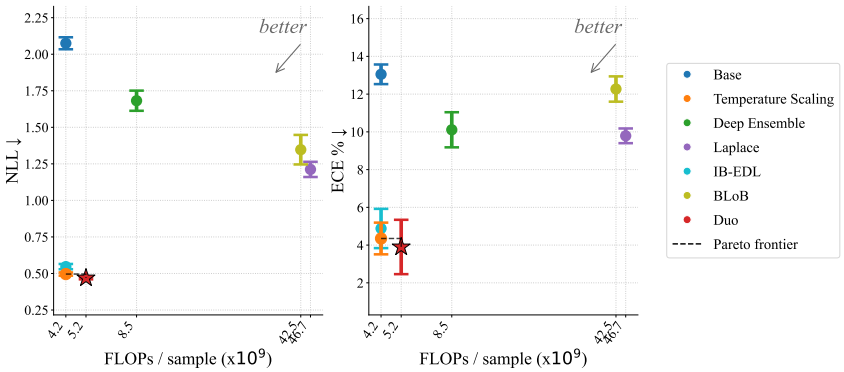

Figure 4: Pareto frontier of FLOPs vs. NLL (left) and ECE (right) for ARC-C dataset. For both metrics, Temperature Scaling and Duo lie on Pareto frontier. The method with the best NLL/ECE performance is marked with a star.

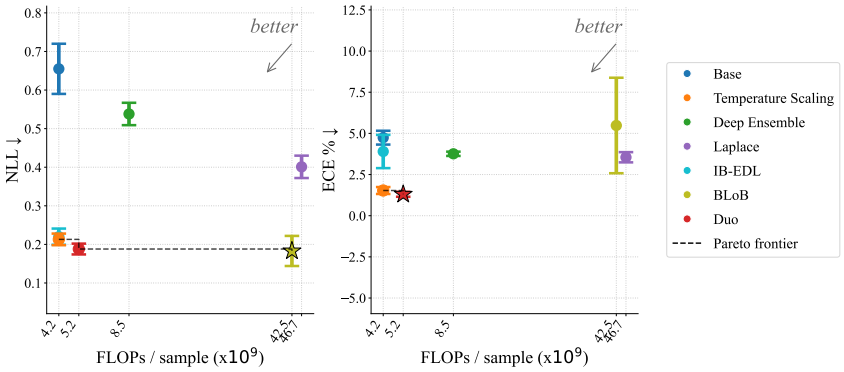

Figure 5: Pareto frontier of FLOPs vs. NLL (left) and ECE (right) for the ARC-E dataset. For NLL, Temperature Scaling, Duo, and BLoB lie on the Pareto frontier; for ECE, Temperature Scaling and Duo lie on the Pareto frontier. The method with the best NLL/ECE performance is marked with a star.

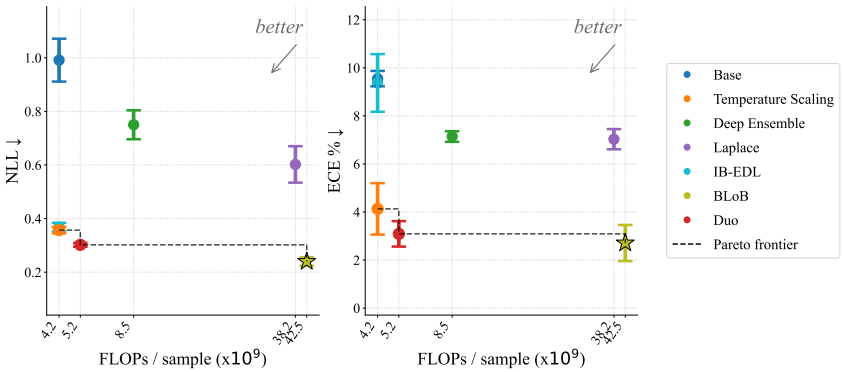

Figure 6: Pareto frontier of FLOPs vs. NLL (left) and ECE (right) for OBQA dataset. For both metrics, Temperature Scaling, Duo, and BLoB lie on Pareto frontier. The method with the best NLL/ECE performance is marked with a star.

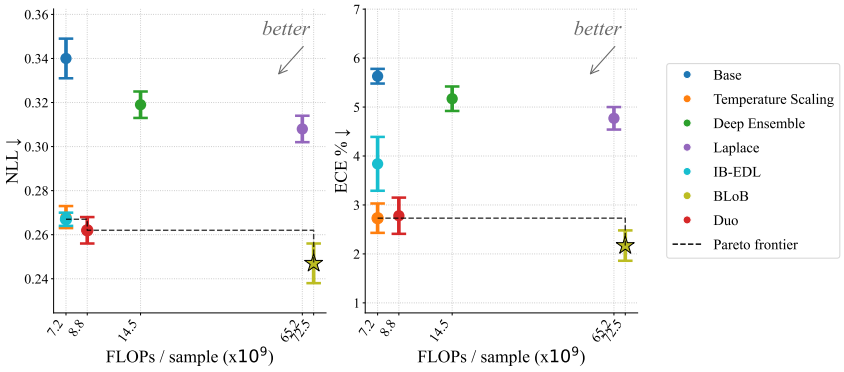

Figure 7: Pareto frontier of FLOPs vs. NLL (left) and ECE (right) for RACE dataset. For NLL, IB-EDL, Duo, and BLoB lie on the Pareto frontier; for ECE, Temperature Scaling and Duo lie on the Pareto frontier. The method with the best NLL/ECE performance is marked with a star.

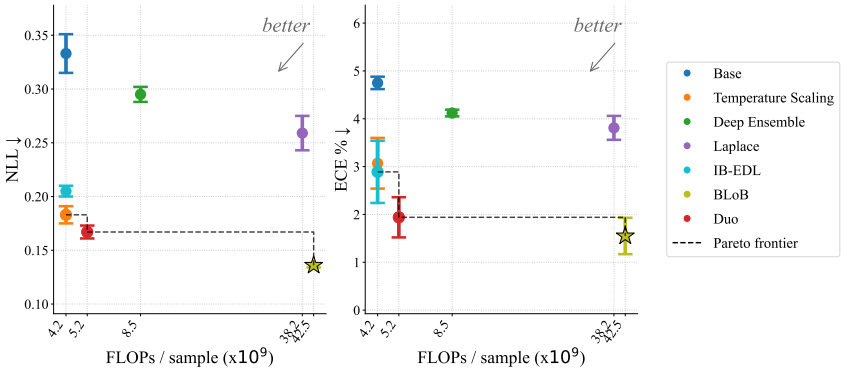

Figure 8: Pareto frontier of FLOPs vs. NLL (left) and ECE (right) for SciQ dataset. For NLL, Temperature Scaling, Duo, and BLoB lie on the Pareto frontier; for ECE, IB-EDL, Duo, and BLoB lie on the Pareto frontier. The method with the best NLL/ECE performance is marked with a star.

B.2 LOGIT DISTRIBUTIONS

The per-dataset logit/spread/margin/CDF plots in Figures 10–19 show the same logit-space mechanism we identified for CSQA in Section 5 on every other benchmark. Base and sidekick models concentrate correct-class logits at large positive values with a wide spread, and the Duo predictor systematically pulls the logit margin toward zero. We read this as evidence that asymmetric logit fusion mitigates overconfidence through a single, dataset-agnostic mechanism. The considered quantities are the correct-class logit, z_{i,y_i}, which denotes the logit assigned to the correct class; the logit spread, $\max_k z_{i,k} - \min_k z_{i,k}$, which measures the range of the output logits; as well as the logit margin, confidence, and correct-class probability described in Section 5.

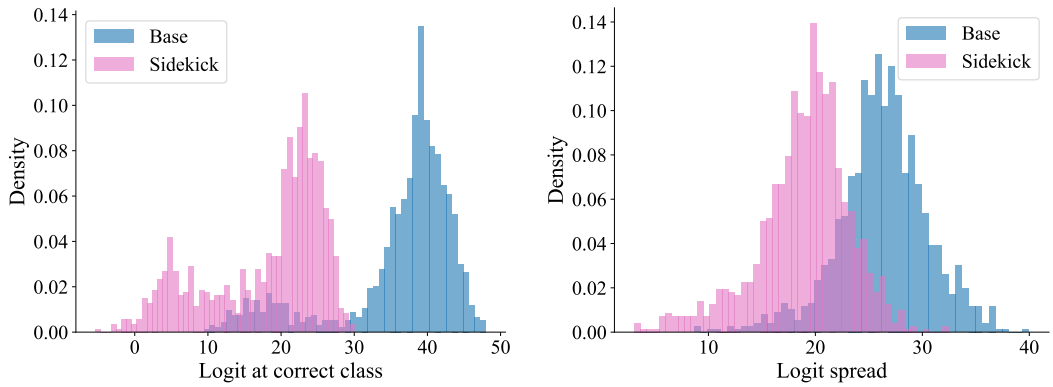

Figure 9: Logit (left) and logit spread (right) distributions of base and sidekick models on the CSQA dataset.

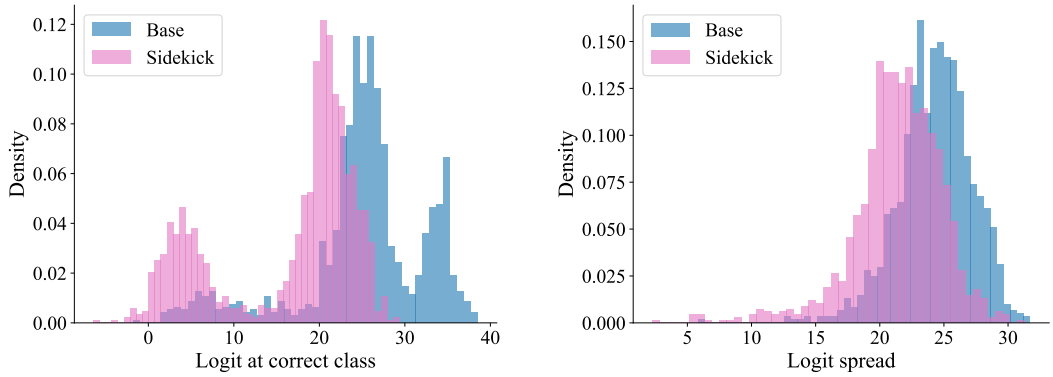

Figure 10: Logit (left) and logit spread (right) distributions of base and sidekick models for ARC-C dataset.

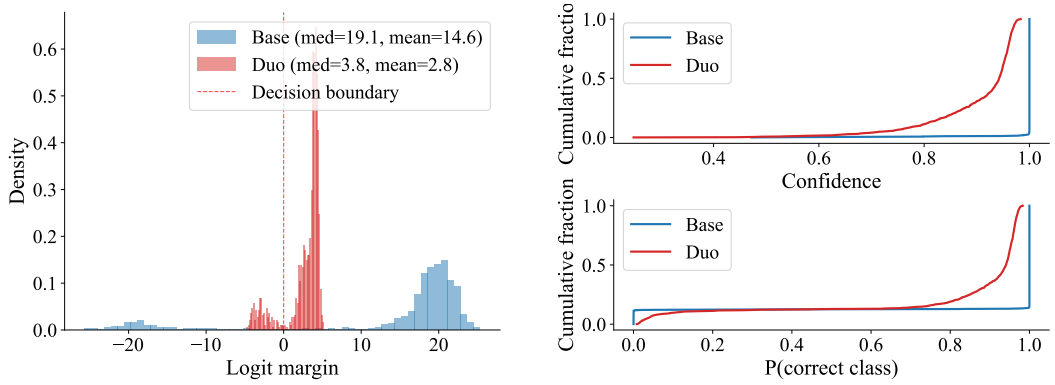

Figure 11: Logit margin (left) and empirical cumulative distribution functions (CDF) of probability and confidence (right) for Duo and Base for ARC-C dataset.

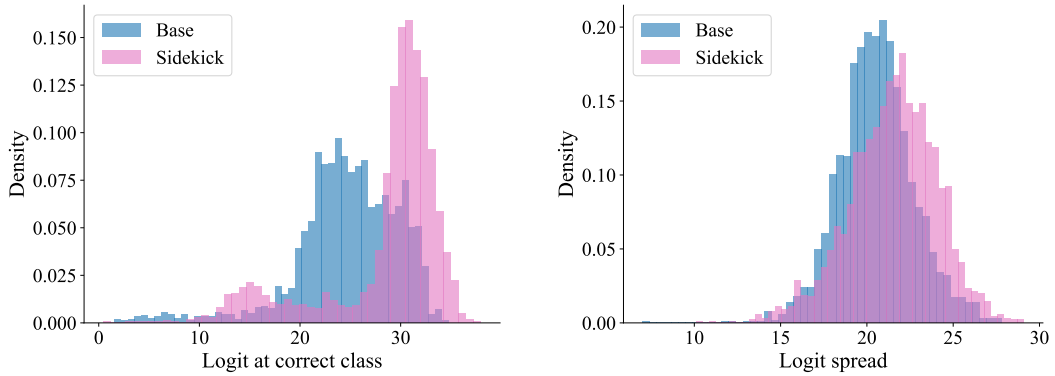

Figure 12: Logit (left) and logit spread (right) distributions of base and sidekick models for ARC-E dataset.

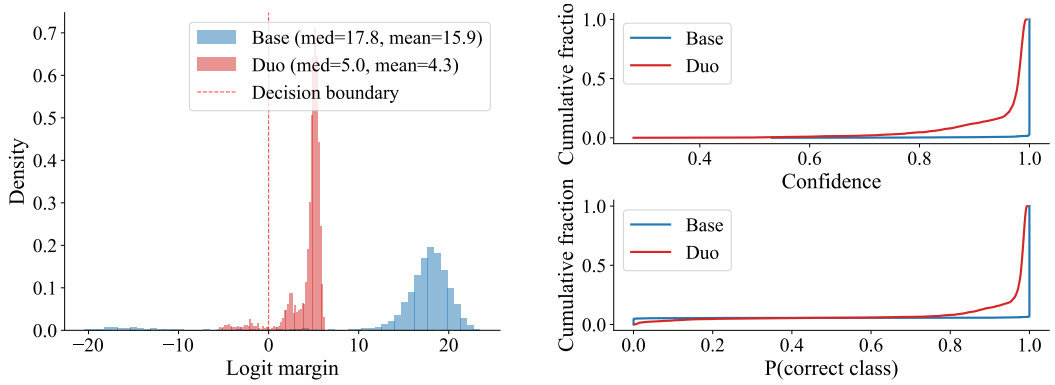

Figure 13: Logit margin (left) and empirical cumulative distribution functions (CDF) of probability and confidence (right) for Duo and Base for ARC-E dataset.

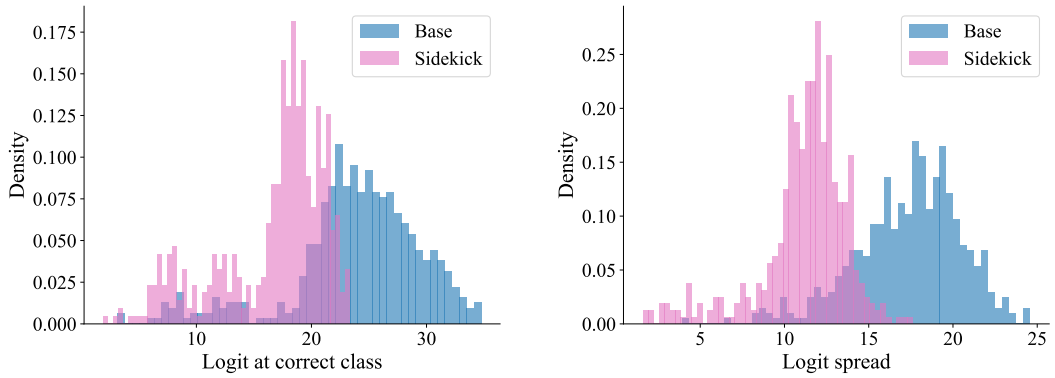

Figure 14: Logit (left) and logit spread (right) distributions of base and sidekick models for OBQA dataset.

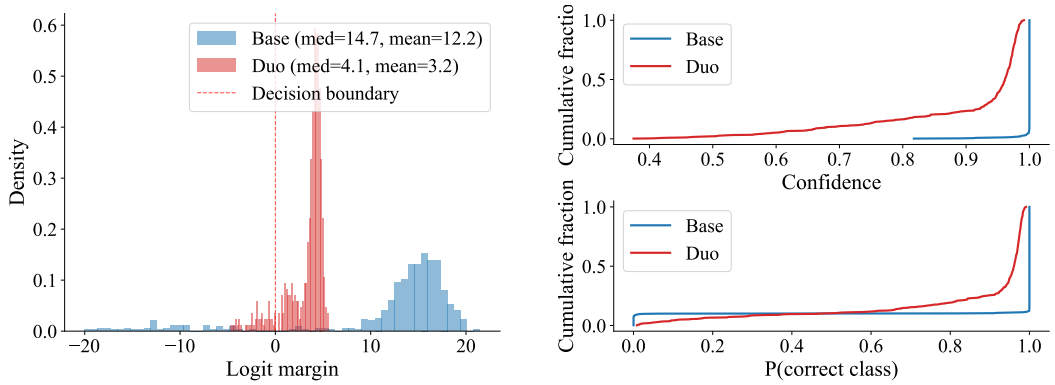

Figure 15: Logit margin (left) and empirical cumulative distribution functions (CDF) of probability and confidence (right) for Duo and Base for OBQA dataset.

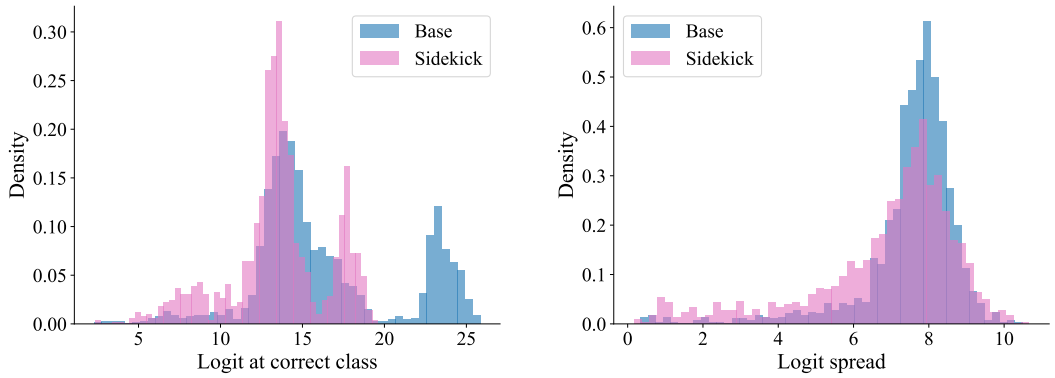

Figure 16: Logit (left) and logit spread (right) distributions of base and sidekick models for RACE dataset.

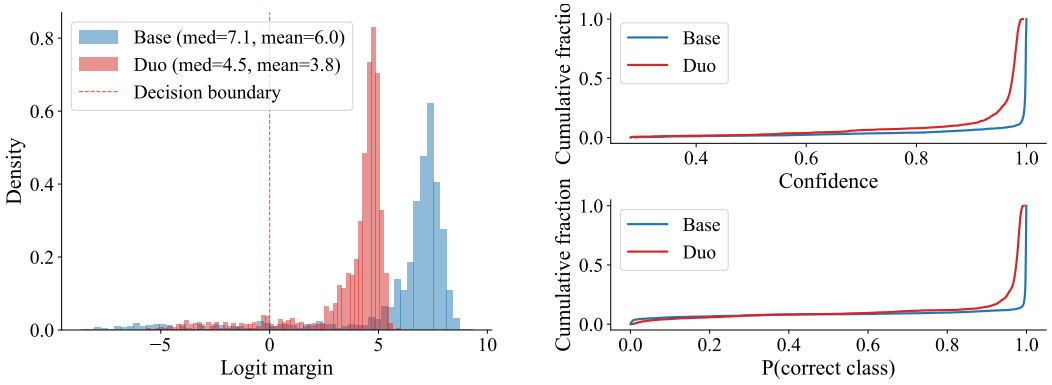

Figure 17: Logit margin (left) and empirical cumulative distribution functions (CDF) of probability and confidence (right) for Duo and Base for RACE dataset.

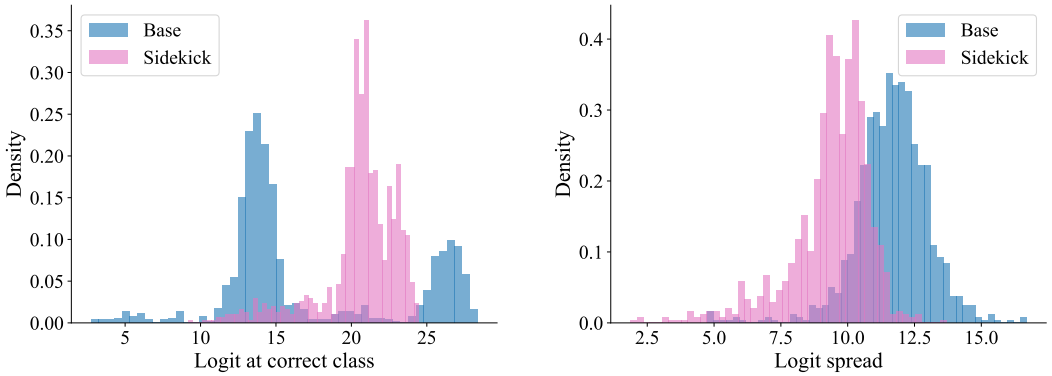

Figure 18: Logit (left) and logit spread (right) distributions of base and sidekick models for SciQ dataset.

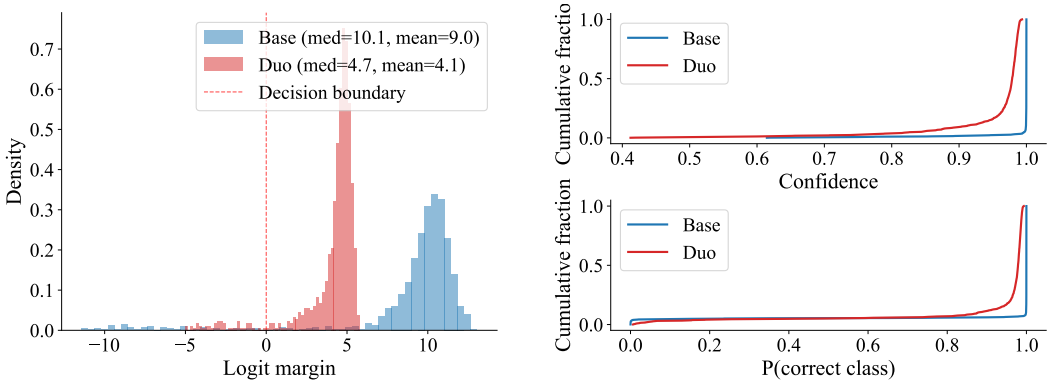

Figure 19: Logit margin (left) and empirical cumulative distribution functions (CDF) of probability and confidence (right) for Duo and Base for SciQ dataset.

C SUPPLEMENTARY RESULTS

In this section, we provide the results of additional experiments to support our findings in the main text. Their interpretations are made in the respective sections of the paper.

Table 4: Changes in test NLL produced by temperature scaling and the additional changes produced by Duo. Negative values indicate reductions in NLL and therefore improved performance.

Comparison	Base method	ARC-C	ARC-E	CSQA	OBQA	RACE	SciQ
Base TS − Base	Plain LoRA	−1.579	−0.442	−2.032	−0.634	−0.072	−0.150
	Laplace-LoRA	−0.725	−0.210	−0.941	−0.275	−0.051	−0.091
	IB-EDL	−0.003	−0.006	−0.005	−0.014	−0.005	−0.001
Duo − Base TS	Plain LoRA	−0.027	−0.025	−0.052	−0.054	−0.006	−0.016
	Laplace-LoRA	−0.023	−0.020	−0.038	−0.040	−0.003	−0.009
	IB-EDL	−0.035	−0.030	−0.064	−0.044	−0.005	−0.022

Table 5: Calibration performance of baseline and Duo methods fine-tuned on Qwen2 7B ("B" for Base) and Qwen2 0.5B ("S" for Sidekick). Accuracy and ECE are reported as percentages. The best results among the shown methods are highlighted in **bold**, and the second-best results are underlined. Additionally, ± denotes the standard deviation across five runs with different seeds.

Metric	Method	ARC-C	ARC-E	CSQA	OBQA	RACE	SciQ
NLL ↓	B	$2.075_{\pm 0.041}$	$0.655_{\pm 0.065}$	$2.557_{\pm 0.192}$	$0.991_{\pm 0.080}$	$0.340_{\pm 0.009}$	$0.333_{\pm 0.018}$
	B TS	$0.496_{\pm 0.011}$	$0.213_{\pm 0.015}$	$0.525_{\pm 0.017}$	$0.357_{\pm 0.012}$	$0.268_{\pm 0.005}$	$0.183_{\pm 0.008}$
	Duo	$\mathbf{0.487_{\pm 0.011}}$	$\mathbf{0.201_{\pm 0.012}}$	$\mathbf{0.502_{\pm 0.014}}$	$\mathbf{0.341_{\pm 0.008}}$	$\mathbf{0.265_{\pm 0.004}}$	$\mathbf{0.175_{\pm 0.006}}$
	B DE	$1.682_{\pm 0.069}$	$0.538_{\pm 0.029}$	$1.925_{\pm 0.101}$	$0.750_{\pm 0.054}$	$0.319_{\pm 0.006}$	$0.295_{\pm 0.007}$
Acc ↑	B	$86.59_{\pm 0.52}$	$95.08_{\pm 0.40}$	$85.34_{\pm 0.41}$	$\mathbf{90.24_{\pm 0.17}}$	$91.89_{\pm 0.11}$	$94.96_{\pm 0.21}$
	B TS	$86.59_{\pm 0.52}$	$95.08_{\pm 0.40}$	$85.34_{\pm 0.41}$	$\mathbf{90.24_{\pm 0.17}}$	$91.89_{\pm 0.11}$	$94.96_{\pm 0.21}$
	Duo	$86.86_{\pm 0.48}$	$\mathbf{95.13_{\pm 0.40}}$	$85.41_{\pm 0.42}$	$90.20_{\pm 0.24}$	$91.89_{\pm 0.22}$	$95.06_{\pm 0.25}$
	B DE	$\mathbf{87.15_{\pm 0.43}}$	$95.09_{\pm 0.14}$	$\mathbf{85.81_{\pm 0.35}}$	$90.20_{\pm 0.32}$	$\mathbf{91.94_{\pm 0.23}}$	$\mathbf{95.12_{\pm 0.25}}$
ECE ↓	B	$13.05_{\pm 0.51}$	$4.74_{\pm 0.42}$	$14.38_{\pm 0.41}$	$9.55_{\pm 0.32}$	$5.63_{\pm 0.15}$	$4.75_{\pm 0.13}$
	B TS	$4.35_{\pm 0.84}$	$1.53_{\pm 0.21}$	$3.54_{\pm 0.63}$	$4.13_{\pm 1.07}$	$2.73_{\pm 0.30}$	$3.07_{\pm 0.53}$
	Duo	$\mathbf{3.17_{\pm 0.67}}$	$\mathbf{1.22_{\pm 0.19}}$	$\mathbf{2.91_{\pm 0.36}}$	$\mathbf{3.65_{\pm 1.45}}$	$\mathbf{2.64_{\pm 0.24}}$	$\mathbf{2.80_{\pm 0.24}}$
	B DE	$10.11_{\pm 0.93}$	$3.76_{\pm 0.13}$	$10.35_{\pm 0.57}$	$7.14_{\pm 0.22}$	$5.17_{\pm 0.25}$	$4.12_{\pm 0.07}$

Table 6: Calibration performance of baseline and Duo methods fine-tuned on Llama-3.1-8B ("B" for Base) and Llama-3.2-1B ("S" for Sidekick). Arrows (↑ or ↓) indicate the direction of better performance. The best results are highlighted in **bold**, and the second-best results are underlined. Additionally, ± denotes the standard deviation over five runs with five different seeds.

Metric	Method	ARC-C	ARC-E	CSQA	OBQA	RACE	SciQ
NLL ↓	B	$3.212_{\pm 0.058}$	$1.132_{\pm 0.079}$	$2.635_{\pm 0.091}$	$1.326_{\pm 0.080}$	$0.441_{\pm 0.009}$	$0.395_{\pm 0.032}$
	B TS	$0.721_{\pm 0.013}$	$0.330_{\pm 0.016}$	$0.650_{\pm 0.008}$	$0.466_{\pm 0.020}$	$0.339_{\pm 0.004}$	$0.204_{\pm 0.009}$
	Duo	$\mathbf{0.701_{\pm 0.011}}$	$\mathbf{0.309_{\pm 0.013}}$	$\mathbf{0.608_{\pm 0.013}}$	$\mathbf{0.437_{\pm 0.018}}$	$\mathbf{0.333_{\pm 0.003}}$	$\mathbf{0.185_{\pm 0.009}}$
	S	$6.631_{\pm 0.381}$	$3.330_{\pm 0.115}$	$2.912_{\pm 0.161}$	$2.152_{\pm 0.151}$	$0.835_{\pm 0.016}$	$0.633_{\pm 0.018}$
	B DE	$2.639_{\pm 0.028}$	$0.876_{\pm 0.067}$	$2.005_{\pm 0.054}$	$0.991_{\pm 0.042}$	$0.399_{\pm 0.007}$	$0.351_{\pm 0.016}$
Acc ↑	B	$80.27_{\pm 0.71}$	$92.34_{\pm 0.42}$	$82.28_{\pm 0.36}$	$87.16_{\pm 0.67}$	$89.69_{\pm 0.28}$	$94.44_{\pm 0.17}$
	B TS	$80.27_{\pm 0.71}$	$92.34_{\pm 0.42}$	$82.28_{\pm 0.36}$	$87.16_{\pm 0.67}$	$89.69_{\pm 0.28}$	$94.44_{\pm 0.17}$
	Duo	$80.44_{\pm 0.66}$	$92.37_{\pm 0.38}$	$82.38_{\pm 0.59}$	$86.84_{\pm 0.82}$	$89.71_{\pm 0.38}$	$94.40_{\pm 0.17}$
	S	$52.44_{\pm 0.80}$	$74.82_{\pm 0.66}$	$69.78_{\pm 1.06}$	$69.68_{\pm 1.51}$	$75.82_{\pm 0.47}$	$87.58_{\pm 0.58}$
	B DE	$\mathbf{80.85_{\pm 0.48}}$	$\mathbf{92.95_{\pm 0.40}}$	$\mathbf{82.80_{\pm 0.34}}$	$\mathbf{87.20_{\pm 0.32}}$	$\mathbf{90.10_{\pm 0.10}}$	$\mathbf{94.54_{\pm 0.32}}$
ECE ↓	B	$19.57_{\pm 0.66}$	$7.52_{\pm 0.46}$	$17.54_{\pm 0.31}$	$12.86_{\pm 0.73}$	$7.36_{\pm 0.31}$	$5.19_{\pm 0.25}$
	B TS	$\mathbf{5.39_{\pm 1.06}}$	$2.41_{\pm 0.47}$	$4.64_{\pm 1.25}$	$4.87_{\pm 1.09}$	$4.55_{\pm 0.33}$	$3.32_{\pm 0.68}$
	Duo	$5.43_{\pm 0.29}$	$\mathbf{2.05_{\pm 0.38}}$	$\mathbf{3.57_{\pm 0.96}}$	$\mathbf{4.42_{\pm 0.73}}$	$\mathbf{2.87_{\pm 0.40}}$	$\mathbf{1.75_{\pm 0.25}}$
	S	$46.44_{\pm 0.85}$	$24.57_{\pm 0.66}$	$28.70_{\pm 1.02}$	$28.24_{\pm 1.59}$	$14.03_{\pm 0.47}$	$10.43_{\pm 0.57}$
	B DE	$15.05_{\pm 0.58}$	$5.40_{\pm 0.26}$	$12.49_{\pm 0.43}$	$8.93_{\pm 0.24}$	$6.04_{\pm 0.22}$	$4.64_{\pm 0.21}$

Table 7: Calibration performance of baseline and Duo methods fine-tuned on Llama-3.1-8B ("B" for Base) and Qwen2-1.5B ("S" for Sidekick). Arrows (↑ or ↓) indicate the direction of better performance. The best results are highlighted in **bold**, and the second-best results are underlined. Additionally, ± denotes the standard deviation over five runs with five different seeds.

Metric	Method	ARC-C	ARC-E	CSQA	OBQA	RACE	SciQ
NLL ↓	B	$3.212_{\pm 0.058}$	$1.132_{\pm 0.079}$	$2.635_{\pm 0.091}$	$1.326_{\pm 0.080}$	$0.441_{\pm 0.009}$	$0.395_{\pm 0.032}$
	B TS	$0.721_{\pm 0.013}$	$0.330_{\pm 0.016}$	$0.650_{\pm 0.008}$	$0.466_{\pm 0.020}$	$0.339_{\pm 0.004}$	$0.204_{\pm 0.009}$
	Duo	$\mathbf{0.633_{\pm 0.011}}$	$\mathbf{0.264_{\pm 0.010}}$	$\mathbf{0.576_{\pm 0.006}}$	$\mathbf{0.385_{\pm 0.014}}$	$\mathbf{0.314_{\pm 0.004}}$	$\mathbf{0.179_{\pm 0.006}}$
	S	$4.234_{\pm 0.046}$	$1.968_{\pm 0.100}$	$3.493_{\pm 0.182}$	$1.497_{\pm 0.076}$	$0.590_{\pm 0.021}$	$0.546_{\pm 0.022}$
	B DE	$2.639_{\pm 0.028}$	$0.876_{\pm 0.067}$	$2.005_{\pm 0.054}$	$0.991_{\pm 0.042}$	$0.399_{\pm 0.007}$	$0.351_{\pm 0.016}$
Acc ↑	B	$80.27_{\pm 0.71}$	$92.34_{\pm 0.42}$	$82.28_{\pm 0.36}$	$87.16_{\pm 0.67}$	$89.69_{\pm 0.28}$	$94.44_{\pm 0.17}$
	B TS	$80.27_{\pm 0.71}$	$92.34_{\pm 0.42}$	$82.28_{\pm 0.36}$	$87.16_{\pm 0.67}$	$89.69_{\pm 0.28}$	$94.44_{\pm 0.17}$
	Duo	$80.63_{\pm 0.40}$	$92.69_{\pm 0.35}$	$82.47_{\pm 0.21}$	$\mathbf{87.28_{\pm 0.69}}$	$89.79_{\pm 0.36}$	$\mathbf{94.60_{\pm 0.37}}$
	S	$71.54_{\pm 0.90}$	$85.51_{\pm 0.61}$	$74.63_{\pm 0.27}$	$80.32_{\pm 0.52}$	$84.29_{\pm 0.42}$	$90.04_{\pm 0.15}$
	B DE	$\mathbf{80.85_{\pm 0.48}}$	$\mathbf{92.95_{\pm 0.40}}$	$\mathbf{82.80_{\pm 0.34}}$	$87.20_{\pm 0.32}$	$\mathbf{90.10_{\pm 0.10}}$	$94.54_{\pm 0.32}$
ECE ↓	B	$19.57_{\pm 0.66}$	$7.52_{\pm 0.46}$	$17.54_{\pm 0.31}$	$12.86_{\pm 0.73}$	$7.36_{\pm 0.31}$	$5.19_{\pm 0.25}$
	B TS	$5.39_{\pm 1.06}$	$2.41_{\pm 0.47}$	$4.64_{\pm 1.25}$	$4.87_{\pm 1.09}$	$4.55_{\pm 0.33}$	$3.32_{\pm 0.68}$
	Duo	$\mathbf{5.27_{\pm 0.99}}$	$\mathbf{1.28_{\pm 0.16}}$	$\mathbf{3.00_{\pm 0.54}}$	$\mathbf{4.15_{\pm 0.81}}$	$\mathbf{3.19_{\pm 0.30}}$	$\mathbf{2.11_{\pm 0.33}}$
	S	$27.73_{\pm 0.96}$	$14.16_{\pm 0.59}$	$24.61_{\pm 0.36}$	$18.57_{\pm 0.65}$	$10.11_{\pm 0.47}$	$8.69_{\pm 0.14}$
	B DE	$15.05_{\pm 0.58}$	$5.40_{\pm 0.26}$	$12.49_{\pm 0.43}$	$8.93_{\pm 0.24}$	$6.04_{\pm 0.22}$	$4.64_{\pm 0.21}$

Table 8: Learned Duo coefficients at different validation subset sizes. The full-data results are highlighted in gray as a reference.

Coefficient	Val. subset	ARC-C	ARC-E	CSQA	OBQA	RACE	SciQ
α	20%	$0.175_{\pm 0.043}$	$0.230_{\pm 0.039}$	$0.122_{\pm 0.014}$	$0.228_{\pm 0.042}$	$0.582_{\pm 0.069}$	$0.343_{\pm 0.053}$
	40%	$0.162_{\pm 0.020}$	$0.224_{\pm 0.029}$	$0.119_{\pm 0.010}$	$0.225_{\pm 0.031}$	$0.572_{\pm 0.047}$	$0.327_{\pm 0.023}$
	100%	$0.162_{\pm 0.006}$	$0.220_{\pm 0.012}$	$0.120_{\pm 0.008}$	$0.219_{\pm 0.022}$	$0.568_{\pm 0.022}$	$0.330_{\pm 0.013}$
β	20%	$0.055_{\pm 0.039}$	$0.064_{\pm 0.031}$	$0.070_{\pm 0.015}$	$0.118_{\pm 0.050}$	$0.080_{\pm 0.068}$	$0.126_{\pm 0.067}$
	40%	$0.043_{\pm 0.020}$	$0.071_{\pm 0.027}$	$0.072_{\pm 0.011}$	$0.127_{\pm 0.032}$	$0.091_{\pm 0.042}$	$0.135_{\pm 0.033}$
	100%	$0.047_{\pm 0.010}$	$0.072_{\pm 0.012}$	$0.071_{\pm 0.005}$	$0.126_{\pm 0.017}$	$0.083_{\pm 0.017}$	$0.135_{\pm 0.020}$

Table 9: Test NLL obtained by fitting Duo on different validation subset sizes. The full-data results are highlighted in gray.

Val. subset	ARC-C	ARC-E	CSQA	OBQA	RACE	SciQ
20%	$0.498_{\pm 0.067}$	$0.196_{\pm 0.015}$	$0.477_{\pm 0.014}$	$0.313_{\pm 0.014}$	$0.266_{\pm 0.007}$	$0.171_{\pm 0.008}$
40%	$0.475_{\pm 0.016}$	$0.192_{\pm 0.015}$	$0.475_{\pm 0.013}$	$0.308_{\pm 0.012}$	$0.263_{\pm 0.006}$	$0.168_{\pm 0.006}$
100%	$0.469_{\pm 0.009}$	$0.188_{\pm 0.014}$	$0.473_{\pm 0.014}$	$0.302_{\pm 0.008}$	$0.262_{\pm 0.006}$	$0.167_{\pm 0.006}$

Table 10: Test-set fractions and contributions to the total NLL improvement of Duo over Base TS.

Measure	Subset	ARC-C	ARC-E	CSQA	OBQA	RACE	SciQ
Test-set fraction	Prediction unchanged	0.9915	0.9950	0.9789	0.9896	0.9943	0.9934
	Prediction changed	0.0085	0.0050	0.0211	0.0104	0.0057	0.0066
	Changed: wrong to right	0.0044	0.0024	0.0118	0.0052	0.0019	0.0046
	Changed: right to wrong	0.0022	0.0018	0.0079	0.0036	0.0026	0.0018
Share of NLL gain	Prediction unchanged	0.9704	0.9781	0.9257	0.9645	1.0360	0.8909
	Prediction changed	0.0296	0.0219	0.0743	0.0355	−0.0360	0.1091

Table 11: Variants used to separate global scaling from asymmetric weighting.

Variant	Parameters	Fusion rule
Base	0	$\mathbf{z}_b$
Base TS	1	$\alpha\mathbf{z}_b$
Symmetric learned	1	$\gamma(\mathbf{z}_b + \mathbf{z}_s)$
Duo	2	$\alpha\mathbf{z}_b + \beta\mathbf{z}_s$
Duo oracle	2	$\alpha\mathbf{z}_b + \beta\mathbf{z}_s$, fitted on test

Table 12: Test NLL for the ablation separating global scaling from asymmetric weighting. Bold and underlined values indicate the best and second-best validation-fitted methods, respectively. Duo oracle is fitted on the test set and shown only as a lower-bound reference. The symbol ± denotes the standard deviation over five runs.

Variant	ARC-C	ARC-E	CSQA	OBQA	RACE	SciQ
Base	$2.075_{\pm 0.041}$	$0.655_{\pm 0.065}$	$2.557_{\pm 0.192}$	$0.990_{\pm 0.080}$	$0.340_{\pm 0.009}$	$0.333_{\pm 0.018}$
Base TS	$\underline{0.496}_{\pm 0.010}$	$\underline{0.213}_{\pm 0.015}$	$0.525_{\pm 0.017}$	$0.357_{\pm 0.012}$	$\underline{0.268}_{\pm 0.005}$	$0.183_{\pm 0.008}$
Symmetric learned	$0.513_{\pm 0.011}$	$0.216_{\pm 0.012}$	$\underline{0.486}_{\pm 0.010}$	$\underline{0.309}_{\pm 0.008}$	$0.281_{\pm 0.006}$	$\underline{0.173}_{\pm 0.005}$
Duo	$\mathbf{0.469}_{\pm \mathbf{0.009}}$	$\mathbf{0.188}_{\pm \mathbf{0.014}}$	$\mathbf{0.473}_{\pm \mathbf{0.014}}$	$\mathbf{0.302}_{\pm \mathbf{0.007}}$	$\mathbf{0.262}_{\pm \mathbf{0.006}}$	$\mathbf{0.167}_{\pm \mathbf{0.006}}$
Duo oracle (lower bound)	$0.465_{\pm 0.007}$	$0.186_{\pm 0.013}$	$0.473_{\pm 0.013}$	$0.300_{\pm 0.007}$	$0.260_{\pm 0.004}$	$0.165_{\pm 0.006}$

D FULL NUMERICAL RESULTS

All tables in this appendix report the main configuration (Qwen2-7B Base, Qwen2-1.5B Sidekick); results for the other model pairings are in Appendix C.

D.1 COMPATIBILITY WITH CALIBRATION-AWARE METHODS

Table 13 reports the per-(metric, dataset) numbers that underlie Figure 2 of the main text. Within each B X vs Duo X pair, Duo X achieves the better NLL on every dataset across all three calibration families. On ECE, Duo X improves over B X in 17 of 18 cells, with the single exception of CSQA-BLoB where the two methods are essentially tied (B BLoB 2.04 vs Duo BLoB 2.06, within standard deviation). Accuracy is preserved or improved in 16 of 18 cells, with two minor exceptions where Base wins by a small margin (OBQA-BLoB and RACE-LA).

D.2 LEARNED DUO FUSION WEIGHTS

Table 14 reports the fusion weights (α, β) for every Duo variant in this paper, aggregated as mean $\pm$ standard deviation over 5 repetitions. Two regimes emerge clearly. For plain Duo and Duo LA the optimizer settles on $\alpha, \beta \ll 1$, consistent with both base and sidekick logits being sharply overconfident: the optimizer compresses them in addition to combining them, in line with the trivial-Duo ablation of Section 4.3. For Duo BLoB and Duo IB-EDL the calibration-aware base has already softened the answer-token logits, so the optimizer leaves them near unit scale ($\alpha \approx 1$) and tunes β to decide how much to incorporate the sidekick. On RACE and SciQ the BLoB sidekick is so saturated that β collapses to the optimizer's lower bound (10^{-6}), and Duo BLoB reduces to B BLoB on those datasets.

D.3 MECHANISTIC LOGIT-SPACE ANALYSIS OF ASYMMETRIC DUOS

Table 15 reports the Base-Correct vs Base-Wrong decomposition of Section 5 for the remaining five benchmarks. The same asymmetry holds for every dataset: Duo trades a small confidence cost on Base's correct subset for a substantial reduction on Base's wrong subset.

E DETAILED IMPLEMENTATION

A reproducible codebase for all experiments of this work is provided at https://anonymous.4open.science/r/asymmetric-llm-duos-DC28.

Models. We fine-tune Qwen2-7B, Qwen2-1.5B, and Qwen2-0.5B (Yang et al., 2024b) and, from a second model family, Llama-3.1-8B and Llama-3.2-1B (Grattafiori et al., 2024), all with LoRA (Hu et al., 2022) via PEFT (Mangrulkar et al., 2022) and Transformers (Wolf et al., 2020). All models are loaded in `bfloat16`. Neither family provides a dedicated padding token, so we reuse the end-of-sequence token for padding (`<|endoftext|>` for Qwen2, `<|end_of_text|>` for Llama 3).

Datasets. In the experiments, six MCQA datasets are used, which are ARC-C and ARC-E (Clark et al., 2018), OpenBookQA (OBQA) (Mihaylov et al., 2018), CommonsenseQA (CSQA) (Talmor et al., 2019), SciQ (Welbl et al., 2017), and RACE (Lai et al., 2017). These datasets are chosen to align with experimental settings in prior work, facilitating consistent and comparable evaluation across different methods (Wang et al., 2024; Hu et al., 2022; Li et al., 2025). The target space is defined as tokens corresponding to different pre-determined options like A, B, C, D, and E.

Baselines. Our method is compared to six different baselines: LoRA fine-tuned MAP predictions (**B**), a deep ensemble with two members (**B DE**) (Lakshminarayanan et al., 2017; Fort et al., 2019), Temperature Scaling (**B TS**) (Guo et al., 2017), Laplace-LoRA (**B LA**) (Yang et al., 2024a), IB-EDL (**B IB-EDL**) (Li et al., 2025), and BLoB (**B BLoB**) (Wang et al., 2024). In order to test whether applying the Duo method to existing uncertainty-aware approaches can further lead to a performance boost, we additionally applied the Duo procedure to the outputs of the following baselines: Laplace-LoRA (**Duo LA**), IB-EDL (**Duo IB-EDL**), and BLoB (**Duo BLoB**).

Table 13: Compatibility of the Duo procedure with calibration-aware baseline methods. Each block compares the calibration-aware base predictor B X against its Duo counterpart Duo X for $X \in \{\text{LA}, \text{BLoB}, \text{IB-EDL}\}$. Within each pair, the better entry per (metric, dataset) is in **bold**; ties are bolded on both rows. All metrics are mean $\pm$ SD over 5 repetitions. Plain-LoRA Duo and a Duo built from temperature-scaled Base and Sidekick are omitted because both reduce to the standard Duo of Table 1 (see remark in Section 4.2).

Metric	Method	ARC-C	ARC-E	CSQA	OBQA	RACE	SciQ
NLL ↓	B LA	$1.212_{\pm 0.052}$	$0.401_{\pm 0.029}$	$1.419_{\pm 0.096}$	$0.602_{\pm 0.068}$	$0.308_{\pm 0.006}$	$0.259_{\pm 0.016}$
	Duo LA	$\mathbf{0.438}_{\pm 0.006}$	$\mathbf{0.168}_{\pm 0.009}$	$\mathbf{0.432}_{\pm 0.010}$	$\mathbf{0.276}_{\pm 0.019}$	$\mathbf{0.255}_{\pm 0.005}$	$\mathbf{0.158}_{\pm 0.006}$
	B BLoB	$1.347_{\pm 0.101}$	$0.183_{\pm 0.039}$	$0.361_{\pm 0.008}$	$0.240_{\pm 0.016}$	$0.247_{\pm 0.009}$	$0.136_{\pm 0.002}$
	Duo BLoB	$\mathbf{0.443}_{\pm 0.014}$	$\mathbf{0.152}_{\pm 0.012}$	$\mathbf{0.355}_{\pm 0.007}$	$\mathbf{0.234}_{\pm 0.017}$	$\mathbf{0.245}_{\pm 0.009}$	$\mathbf{0.135}_{\pm 0.003}$
	B IB-EDL	$0.547_{\pm 0.018}$	$0.220_{\pm 0.021}$	$0.556_{\pm 0.017}$	$0.368_{\pm 0.016}$	$0.267_{\pm 0.003}$	$0.205_{\pm 0.005}$
	Duo IB-EDL	$\mathbf{0.509}_{\pm 0.015}$	$\mathbf{0.183}_{\pm 0.018}$	$\mathbf{0.487}_{\pm 0.018}$	$\mathbf{0.310}_{\pm 0.016}$	$\mathbf{0.257}_{\pm 0.003}$	$\mathbf{0.182}_{\pm 0.005}$
Acc ↑	B LA	$86.66_{\pm 0.48}$	$95.11_{\pm 0.32}$	$85.32_{\pm 0.42}$	$90.04_{\pm 0.14}$	$\mathbf{91.80}_{\pm 0.15}$	$94.84_{\pm 0.23}$
	Duo LA	$\mathbf{87.12}_{\pm 0.32}$	$\mathbf{95.23}_{\pm 0.40}$	$\mathbf{85.81}_{\pm 0.67}$	$\mathbf{90.64}_{\pm 0.17}$	$91.75_{\pm 0.49}$	$\mathbf{95.22}_{\pm 0.33}$
	B BLoB	$86.31_{\pm 0.67}$	$94.89_{\pm 0.38}$	$86.55_{\pm 0.61}$	$\mathbf{91.84}_{\pm 0.83}$	$\mathbf{91.57}_{\pm 0.59}$	$95.42_{\pm 0.08}$
	Duo BLoB	$\mathbf{86.55}_{\pm 0.30}$	$\mathbf{94.97}_{\pm 0.50}$	$\mathbf{86.58}_{\pm 0.62}$	$91.36_{\pm 0.83}$	$\mathbf{91.57}_{\pm 0.59}$	$\mathbf{95.44}_{\pm 0.11}$
	B IB-EDL	$86.28_{\pm 0.60}$	$95.23_{\pm 0.17}$	$85.39_{\pm 0.52}$	$90.84_{\pm 0.38}$	$91.85_{\pm 0.26}$	$95.02_{\pm 0.22}$
	Duo IB-EDL	$\mathbf{86.55}_{\pm 0.67}$	$\mathbf{95.45}_{\pm 0.24}$	$\mathbf{85.57}_{\pm 0.76}$	$\mathbf{90.92}_{\pm 0.59}$	$\mathbf{91.88}_{\pm 0.26}$	$\mathbf{95.24}_{\pm 0.32}$
ECE ↓	B LA	$9.79_{\pm 0.39}$	$3.55_{\pm 0.31}$	$10.95_{\pm 0.72}$	$7.03_{\pm 0.42}$	$4.77_{\pm 0.23}$	$3.81_{\pm 0.25}$
	Duo LA	$\mathbf{2.02}_{\pm 0.41}$	$\mathbf{1.08}_{\pm 0.43}$	$\mathbf{2.37}_{\pm 0.76}$	$\mathbf{2.84}_{\pm 0.37}$	$\mathbf{2.12}_{\pm 0.25}$	$\mathbf{1.71}_{\pm 0.23}$
	B BLoB	$12.27_{\pm 0.67}$	$5.48_{\pm 2.90}$	$\mathbf{2.04}_{\pm 0.15}$	$2.71_{\pm 0.75}$	$2.17_{\pm 0.31}$	$1.55_{\pm 0.38}$
	Duo BLoB	$\mathbf{4.24}_{\pm 0.47}$	$\mathbf{0.97}_{\pm 0.21}$	$2.06_{\pm 0.30}$	$\mathbf{2.64}_{\pm 0.52}$	$\mathbf{1.48}_{\pm 0.35}$	$\mathbf{1.39}_{\pm 0.24}$
	B IB-EDL	$4.88_{\pm 1.04}$	$3.90_{\pm 1.01}$	$5.61_{\pm 0.78}$	$9.37_{\pm 1.20}$	$3.84_{\pm 0.55}$	$2.89_{\pm 0.65}$
	Duo IB-EDL	$\mathbf{2.73}_{\pm 1.03}$	$\mathbf{1.07}_{\pm 0.14}$	$\mathbf{3.65}_{\pm 0.52}$	$\mathbf{2.81}_{\pm 0.68}$	$\mathbf{2.10}_{\pm 0.64}$	$\mathbf{2.19}_{\pm 0.41}$

Table 14: Learned Duo fusion weights (α, β) on the validation logits of base and sidekick, mean $\pm$ SD over 5 repetitions. Values reported as 0.000 for β are at the optimizer's lower bound (10^{-6}).

Method	Weight	ARC-C	ARC-E	CSQA	OBQA	RACE	SciQ
Duo	α	$0.162_{\pm 0.006}$	$0.220_{\pm 0.012}$	$0.120_{\pm 0.008}$	$0.219_{\pm 0.022}$	$0.568_{\pm 0.022}$	$0.330_{\pm 0.013}$
	β	$0.046_{\pm 0.010}$	$0.072_{\pm 0.012}$	$0.071_{\pm 0.005}$	$0.126_{\pm 0.017}$	$0.083_{\pm 0.017}$	$0.135_{\pm 0.020}$
Duo LA	α	$0.227_{\pm 0.012}$	$0.289_{\pm 0.015}$	$0.167_{\pm 0.008}$	$0.281_{\pm 0.015}$	$0.626_{\pm 0.027}$	$0.399_{\pm 0.014}$
	β	$0.068_{\pm 0.012}$	$0.080_{\pm 0.015}$	$0.100_{\pm 0.005}$	$0.155_{\pm 0.029}$	$0.056_{\pm 0.022}$	$0.111_{\pm 0.023}$
Duo BLoB	α	$0.228_{\pm 0.013}$	$1.540_{\pm 0.118}$	$0.922_{\pm 0.041}$	$1.022_{\pm 0.097}$	$1.116_{\pm 0.010}$	$1.031_{\pm 0.032}$
	β	$0.049_{\pm 0.014}$	$0.142_{\pm 0.099}$	$0.261_{\pm 0.028}$	$0.145_{\pm 0.057}$	$0.000_{\pm 0.000}$	$0.001_{\pm 0.003}$
Duo IB-EDL	α	$0.960_{\pm 0.053}$	$0.936_{\pm 0.095}$	$0.865_{\pm 0.030}$	$0.981_{\pm 0.045}$	$1.062_{\pm 0.022}$	$0.820_{\pm 0.023}$
	β	$0.411_{\pm 0.054}$	$0.430_{\pm 0.081}$	$0.484_{\pm 0.023}$	$0.428_{\pm 0.066}$	$0.135_{\pm 0.023}$	$0.320_{\pm 0.022}$

Measuring the computational cost. We use FLOPs to measure inference cost, as they provide a hardware-independent metric for comparing computational complexity across different methods. FLOPs are estimated from multiply-accumulate operations (MACs), where one MAC corresponds to two FLOPs. For the Duo framework, FLOPs are computed as the sum of the base and sidekick model inference costs. This allows for a fair comparison against baselines such as single models and deep ensembles of varying sizes. All estimates are obtained from standard model-level complexity calculations based on the model architecture and input sequence length.

LoRA hyperparameters. We adapt both models with LoRA (Hu et al., 2022) via the PEFT library (Mangrulkar et al., 2022) and Transformers (Wolf et al., 2020). LoRA adapters are applied to the four attention projections of every transformer block, `q_proj`, `k_proj`, `v_proj`, and `o_proj`. We use rank $r = 8$, scaling $\alpha_{\text{LoRA}} = 16$, dropout 0.1, and `bias = "lora_only"`, with `task_type = CAUSAL_LM`. The same configuration is shared across all six datasets and all five models (Qwen2-0.5B/1.5B/7B, Llama-3.2-1B, Llama-3.1-8B).

Table 15: NLL and ECE of Base and Duo, computed separately on test set samples of benchmark datasets that Base classifies correctly/incorrectly. Bold indicates the better value between Base and Duo within each subset.

Metric	Dataset	Method	Base Correct	Base Wrong
NLL ↓	ARC-C	Base	**0.001**	16.426
		Duo	0.090	**2.996**
	ARC-E	Base	**0.001**	13.313
		Duo	0.041	**2.935**
	OBQA	Base	**0.001**	11.055
		Duo	0.087	**2.224**
	RACE	Base	**0.016**	4.041
		Duo	0.055	**2.638**
	SciQ	Base	**0.003**	6.625
		Duo	0.037	**2.702**
ECE ↓	ARC-C	Base	**0.08**	97.53
		Duo	8.13	**78.04**
	ARC-E	Base	**0.06**	98.40
		Duo	3.47	**76.51**
	OBQA	Base	**0.11**	99.50
		Duo	6.88	**61.39**
	RACE	Base	**1.30**	82.71
		Duo	4.66	**72.57**
	SciQ	Base	**0.25**	95.65
		Duo	3.45	**74.08**

Training details. For all datasets, we use the original train/validation/test splits provided by their respective authors, accessed via the Hugging Face datasets library (Wolf et al., 2020); the sole exception is CSQA, which lacks a dedicated validation split, so we reserve 10% of its training data as the validation set. All MAP and IB-EDL runs use AdamW (HuggingFace `Trainer` defaults) with learning rate 5×10^{-5}, a cosine schedule with 20 warm-up steps, and `bfloat16` mixed precision. Per-device train and eval batch sizes are both 4. We train for $10\,080$ optimizer steps on every dataset except CSQA, which uses $30\,000$ steps to compensate for its smaller training split. The maximum tokenized prompt length is 300 tokens for ARC-C, ARC-E, CSQA, OBQA, and SciQ, and 512 tokens for RACE (whose passages are longer). For MAP we use weight decay 0; for IB-EDL we follow common evidential-training practice and add weight decay 10^{-3} together with gradient clipping at maximum global norm 20.0. We adopt the MCQA prompt templates and the next-token answer-classification setup of the IB-EDL reference implementation (Li et al., 2025) unchanged, so that differences between methods are attributable to the calibration procedure rather than to prompt design. Each method is run with 5 seeds and we report mean $\pm$ standard deviation across seeds. Validation and test answer-token logits at the final input position are saved once per run and reused by every post-hoc method below.

Computational details. All experiments were run on a DGX A100 cluster, using a single 40 GB GPU per run; we did not use multi-GPU or distributed training. Fine-tuning a base model on one dataset and evaluating on its test set takes roughly 30–40 minutes, with the exception of CSQA, which uses a larger number of training steps and takes around 90 minutes. For each dataset, we trained 5 base models and 5 sidekick models with different random seeds and report the mean and standard deviation across these runs. Computing the Duo fusion weights on top of the fine-tuned models takes only seconds.

Asymmetric Duo fusion. The fusion scales (α, β) are estimated on the validation logits with L-BFGS-B, initialized at $(\alpha, \beta) = (1.0, 1.0)$ and bounded below by 10^{-6} on each coordinate (no upper bound, no sum constraint). The loss is the validation NLL of $\mathrm{softmax}(\alpha z_{\text{base}} + \beta z_{\text{side}})$, evaluated in float64 with the standard log-sum-exp stabilization. The same optimizer settings are used for

every Duo variant in the paper, including the calibration-aware Duos (Duo LA, Duo BLoB, Duo IB-EDL), where α and β are fit on the validation logits of the corresponding calibration-aware Base and Sidekick. Logits from the two models are matched by example index before fitting.

Temperature scaling. A single scalar temperature $T > 0$ per model is fit on the validation NLL via L-BFGS-B in log-space (initialized at $T = 1$) with the analytical gradient $\partial\,\mathrm{NLL}/\partial \log T = (1/T)\,\mathbb{E}_i[\,\ell_{i,y_i} - \mathbb{E}_{p_i}[\ell_i]\,]$, then applied to the test logits (Guo et al., 2017). We optimize on float64 logits to avoid the gradient collapsing to zero under float32 precision.

Laplace-LoRA. We use the official `bayesian-lora` package (Yang et al., 2024a). Kronecker factors are computed on the training loader with at most $n_{\mathrm{KFAC}} = 10$ low-rank components per factor and a low-rank threshold of 100. The prior variance σ^2 is initialized at 0.1 and tuned by maximizing the model evidence (Yang et al., 2024a). Predictive variances are produced via the Jacobian-mean approximation. We did not perform additional hyperparameter tuning beyond the package defaults.

IB-EDL implementation details. We follow the IB-EDL formulation and implementation of Li et al. (2025). The pre-evidence head is the language-model head of the LLM evaluated at the last input position and restricted to the answer-token columns; positivity is enforced with a softplus and a Dirichlet over the K answer choices is constructed by adding 1 to the evidence vector. The variational information bottleneck draws $K_{\mathrm{IB}} = 20$ Gaussian noise samples per training example, keeps $\sigma_{\mathrm{mult}} = 0$ at evaluation (i.e. no post-hoc evidence shrinkage), and uses the sum-of-squares Bayes-risk loss without label smoothing. The KL regularization weight is ramped up linearly starting at 10% of training; we set the final regularization weight to 0 since the IB term itself already regularizes the evidence. The information-bottleneck coefficient β is the only IB-EDL hyperparameter we tune—per (model, dataset) pair—using the small grid in Table 16 in Appendix E.1.

BLoB. BLoB (Wang et al., 2024) is run with the authors' official `bayesian-peft` codebase using its default settings, including 10 Monte-Carlo samples at evaluation. We average the per-sample probabilities to form the Bayesian predictive and convert back to log-probabilities so that the saved BLoB logits live in the same log-probability space as the other methods.

Deep ensemble baseline (B DE). For each dataset we form 5 random ensembles of size 2 from the 5 available training seeds. Predictions are averaged in probability space, and metrics are then computed on each sampled pair and averaged over pairs.

Calibration metrics. ECE is computed with 15 equal-width confidence bins. NLL is the mean $-\log p_y$ over the test set, with a 10^{-12} floor on probabilities for numerical stability.

E.1 Selection of the IB-EDL β

The IB-EDL objective is governed by an information-bottleneck coefficient β that weights the KL regularizer relative to the evidential likelihood (Li et al., 2025). The optimal β does not transfer across model scales or datasets and must be selected from a small grid per (model, dataset) pair. For each dataset, we evaluated 4–5 candidate values spanning roughly two orders of magnitude (Table 16), and independently for Base and Sidekick, we picked the candidate that minimized the hold-out ECE on seed 1; the selected entries are bold-faced. The chosen β values were then frozen across all 5 repetitions used to produce the IB-EDL rows of Tables 1 and 13.

Table 16: Hold-out ECE (seed 1) of the base and sidekick models across the candidate β values evaluated for each dataset. The bold entry in each (dataset, model) column is the value used in the main experiments.

Dataset	β	Base ECE	Sidekick ECE
ARC-C	9×10^{-5}	**0.0439**	0.1118
	1.5×10^{-4}	0.0467	0.0986
	5×10^{-4}	0.1207	**0.0538**
	1.0×10^{-3}	0.1633	0.0898
ARC-E	2×10^{-6}	**0.0424**	0.0548
	2×10^{-5}	0.0684	**0.0508**
	9×10^{-5}	0.1138	0.0693
	1.0×10^{-4}	0.1065	0.0796
CSQA	5×10^{-5}	0.0577	0.0638
	9×10^{-5}	**0.0501**	**0.0417**
	1.5×10^{-4}	0.0686	0.0505
	5×10^{-4}	0.1224	0.0968
	1.0×10^{-3}	0.1515	0.1283
OBQA	6×10^{-5}	0.0812	**0.0560**
	9×10^{-5}	**0.0811**	0.1150
	1.5×10^{-4}	0.0880	0.0710
	5×10^{-4}	0.1357	0.1122
	1.0×10^{-3}	0.1606	0.1283
RACE	1×10^{-6}	0.0425	0.0477
	2×10^{-6}	**0.0405**	0.0476
	7×10^{-6}	0.0526	**0.0461**
	9×10^{-5}	0.0976	0.0791
SciQ	5×10^{-7}	0.0300	**0.0386**
	1×10^{-6}	**0.0179**	0.0493
	2×10^{-6}	0.0400	0.0476
	9×10^{-5}	0.1082	0.0924