Direct Confidence Alignment: Aligning Verbalized Confidence with Internal Confidence In Large Language Models

Glenn Zhang, Treasure Mayowa, Jason Fan, Yicheng Fu,
Aaron Sandoval, Sean O'Brien, Kevin Zhu
Algoverse AI Research
kevin@algoverse.us

Abstract

Producing trustworthy and reliable Large Language Models (LLMs) has become increasingly important as their usage becomes more widespread. Calibration seeks to achieve this by improving the alignment between the model's confidence and the actual likelihood of its responses being correct or desirable. However, it has been observed that the internal confidence of a model, derived from token probabilities, is not well aligned with its verbalized confidence, leading to misleading results with different calibration methods. In this paper, we propose Direct Confidence Alignment (DCA), a method using Direct Preference Optimization to align an LLM's verbalized confidence with its internal confidence rather than ground-truth accuracy, enhancing model transparency and reliability by ensuring closer alignment between the two confidence measures. We evaluate DCA across multiple open-weight LLMs on a wide range of datasets. To further assess this alignment, we also introduce three new calibration error-based metrics. Our results show that DCA improves alignment metrics on certain model architectures, reducing inconsistencies in a model's confidence expression. However, we also show that it can be ineffective on others, highlighting the need for more model-aware approaches in the pursuit of more interpretable and trustworthy LLMs.

1 Introduction

LLMs have revolutionized natural language tasks, achieving impressive performance across various applications (Wei et al., 2022; Naveed et al., 2024) Despite their capabilities, there are still concerns about the calibrations of these models, that is, the alignment between the confidence they assign to their predictions and the actual accuracy of those predictions(Jiang et al., 2021). For example, in a well-calibrated model, predictions assigned a 70% confidence level should be correct approximately 70% of the time. These limitations are especially critical in high-risk applications such as decision support systems, healthcare settings (Peng et al., 2023), and legal consultations (Lai et al., 2024), where overconfidence in incorrect answers can lead to severe consequences. Examples include erroneous recommendations in decision support systems that can lead to significant financial operational losses, misdiagnoses in healthcare, and flawed legal advice that may affect case outcomes.

Existing model confidence estimation methods can be categorized into two types: Internal and Verbalized Confidence.

Internal Confidence (C_i) is most commonly quantified as the probability of predicting a particular output token semantically linked to an answer given a context. There have also been alternative approaches to estimating internal confidence, such as self-consistency-based approaches and ensemble methods (Geng et al., 2024; Portillo Wightman et al., 2023).

Verbalized Confidence (C_v) is defined as the LLM's expression of its confidence level as a certainty percentage in its output answer to a given prompt (Lin et al., 2022a).

Whilst existing literature predominantly focuses on accuracy-based calibration, which involves aligning models' predicted confidence with ground-truth accuracy, they do not cover the effects of calibrating verbalized confidence C_v to internal confidence C_i instead of against accuracy. Furthermore, internal confidence C_i derived from logits and verbalized confidence within LLMs are often misaligned with each other, leading to inconsistent confidence expressions, especially in unfamiliar questions where models can be verbally overconfident (Ni et al., 2024).

To address these, we propose Direct Confidence Alignment: a method that involves aligning verbalized confidence C_v with internal confidence C_i using Direct Preference Optimization (DPO) (Rafailov et al., 2024). While aligning C_v to C_i

may suggest that C_i is better calibrated than C_v, our method is not focused on accuracy-based calibration. We instead treat C_i as a reference signal of the model's internally expressed uncertainty, and argue that by aligning verbalized confidence with internal confidence, models can provide more transparent and consistent confidence reporting in their responses. We evaluate our approach on a range of datasets and alignment metrics. We make the following contributions:

1. We introduce a novel method of aligning verbalized confidence C_v with internal confidence C_i using DPO training, taking internal confidence as ground truth to improve the transparency and reliability of LLMs.

2. We show the effects and implications of DCA on various LLMs with a wide range of architectures across multiple datasets, highlighting its varied impact across models.

3. We introduce and evaluate our method on three new metrics based on calibration error ϵ, which in this paper refers to the model's internal confidence C_i subtracted from its verbalized confidence C_v for each response. Our proposed metrics in 4.3 provide a more detailed assessment of the magnitude and consistency of alignment between verbalized and internal confidence within LLMs.

2 Related Works

Confidence Calibration Calibration has been an area of extensive research in LLMs. Lin et al. (2022a); Park and Caragea (2022); Kadavath et al. (2022); Kuhn et al. (2022); Guo et al. (2017) show that a pre-trained LLM's calibration can improve with model size, fine-tuning, prompting, self-consistency, or post-hoc methods such as temperature scaling. Temperature scaling in LLM calibration applies a single scalar parameter to adjust model logits before softmax. Known for its simplicity and effectiveness in improving calibration while preserving accuracy, it outperforms techniques such as Platt scaling and isotonic regression across a range of NLP tasks (Guo et al., 2017; Desai and Durrett, 2020). Other approaches involve forms of self-consistency, however, Zhao et al. (2021) demonstrates that a model's confidence can be sensitive to prompting variation. To address this, (Wang et al., 2024; Portillo Wightman et al., 2023) generates an ensemble of prompts, using prompt agreement to generate a calibrated confidence. More recently, Tao et al. (2024) proposed a Confidence-Quality-Order-preserving alignment approach, which incentivizes the model to verbalize greater confidence for responses of higher quality, addressing the lack of a definite ground truth standard for confidence that aligns with response quality in other methods.

Verbalized Confidence As model logits are either inaccessible in black box LLMs or rendered inaccurate due to RLHF, recent work (Tian et al., 2023; Xiong et al., 2024) explores the calibration of verbalized confidence. For example, Tian et al. (2023) takes the mean of k verbalized confidence samples; however, it is sensitive to the prompting structure, making it difficult to generalize sequential reasoning and limited to short answers. To explore this, Xiong et al. (2024) asks the model to elicit verbal confidences using different temperatures and prompt strategies, including Chain-of-Thought, Multi-Step, and Top-K reasoning.

Unlike the above techniques for confidence calibration, our work seeks to align a model's verbalized confidence with its internal confidence, making no reference to ground-truth accuracy or response quality.

Confidence-Probability Alignment Kumar et al. (2024) introduces the concept of Confidence-Probability Alignment, a measurement of the correlation between a model's verbalized certainty and its internal confidence, quantified using answer token probabilities. They posit that Confidence-Probability Alignment is crucial for the reliability of a model's output. Our work expands on this study by aligning these two confidence measures using DPO.

Direct Preference Optimization Rafailov et al. (2024) demonstrates that Direct Preference Optimization (DPO) achieves comparable or superior performance to existing reinforcement learning from human feedback (RLHF) methods in various text generation tasks while being computationally efficient. Although DPO has been shown to successfully align LLMs with human preferences for sentiment control and dialogue quality, our work leverages it specifically to align a model's verbalized confidence (C_v) with its internal confidence (C_i). By using a preference dataset as the learning signal, distinguishing between preferred and non-preferred outputs as opposed to a reward function, DPO's pairwise format is ideally suited for confidence alignment.

3 Methodology

We define DCA as a method to improve the alignment between verbalized confidence and internal confidence within LLMs using DPO, expanding on the study of (Kumar et al., 2024), which introduced this concept.

3.1 Verbalized Confidence Extraction

To extract the model's verbalized confidence C_v, we prompt it in the format of our prompt template in A.1. We then extract the C_v from its response by parsing the numerical value outputted after `Probability:` as shown in Figure 1. The observed error rate for extraction was <5% for all experiments across all models as some responses did not contain a valid C_v.

3.2 Internal Confidence Extraction

To extract the model's internal confidence C_i, we use the computed softmax probability of the answer token (e.g., A, B, C, D) in its output.

3.3 Preference Dataset Creation

To generate an entry in our preference dataset for DPO training, we first generate a sample with full-text completion via our base prompt in A.1 to obtain a formatted answer. We then extract C_i using our method in 3.2 and extract C_v from the model response. Using these values, we create two versions of the answer:

Original Response: Original response of the model
Modified Response: A copy of the original response where the model's C_v is overwritten with its C_i.

For each entry in our preference dataset, the modified response will be the chosen option, and the original response will be the rejected option. See Figure 1 for a visual summary of this process. This is done separately for all models and applied to their individual DPO training runs.

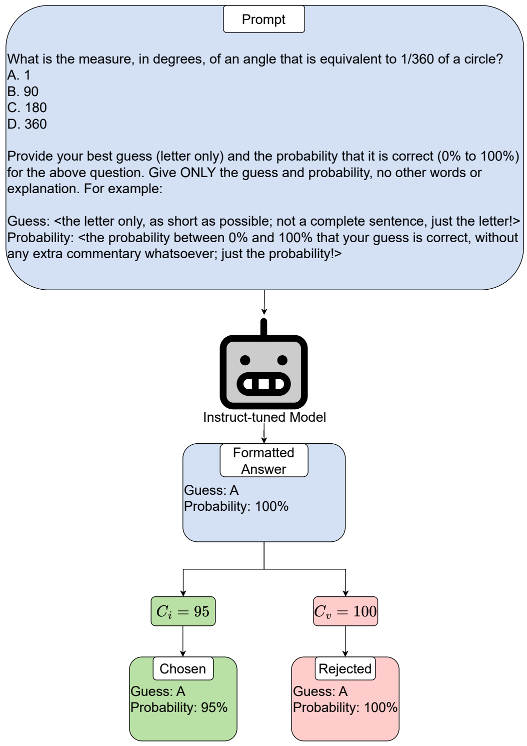


Figure 1: An overview of the entry generation process for our preference dataset. Sample question and response are from MMLU elementary mathematics and Gemma-2-9B-Instruct, respectively.

4 Experiment

4.1 Models

We use three open-weight instruct tuned LMs for our experimental setup, namely Meta's Llama-3.2-3B-Instruct (Team, 2024b); Google's Gemma-2-9B-Instruct (Team, 2024a); and Mistral AI's Mistral-7B-Instruct (Team, 2023).

4.2 Datasets

We use the following datasets for experimentation:

- *OpenBookQA* (Mihaylov et al., 2018) - A science multiple choice dataset modelled after open-book exams testing knowledge and applications of facts
- *TruthfulQA* (Lin et al., 2022b) - A dataset crafted to test LLMs' ability to truthfully answer questions. Scoring well reflects the model's ability to avoid generating false answers from imitating human text.
- *CosmosQA* (Huang et al., 2019) - A reading comprehension dataset based on common sense and reading between the lines for a diverse set of personal everyday narratives.
- *Massive Multitask Language Understanding* (MMLU) (Hendrycks et al., 2021) - An evaluation benchmark designed to test knowledge

Model	Method	OpenBookQA				TruthfulQA				CosmosQA				MMLU				Mean			
		$\rho\uparrow$	$\sigma_\epsilon\downarrow$	$\overline{\lvert\epsilon\rvert}\downarrow$	$\sigma_M\downarrow$	$\rho\uparrow$	$\sigma_\epsilon\downarrow$	$\overline{\lvert\epsilon\rvert}\downarrow$	$\sigma_M\downarrow$	$\rho\uparrow$	$\sigma_\epsilon\downarrow$	$\overline{\lvert\epsilon\rvert}\downarrow$	$\sigma_M\downarrow$	$\rho\uparrow$	$\sigma_\epsilon\downarrow$	$\overline{\lvert\epsilon\rvert}\downarrow$	$\sigma_M\downarrow$	$\rho\uparrow$	$\sigma_\epsilon\downarrow$	$\overline{\lvert\epsilon\rvert}\downarrow$	$\sigma_M\downarrow$
Mistral-7B-Instruct	Vanilla	**0.17**	25.06	**20.08**	1.12	**0.20**	30.64	**25.99**	1.07	**0.20**	**20.59**	**19.53**	**0.53**	**0.18**	26.24	**24.25**	0.67	**0.19**	25.63	**22.96**	0.85
	DCA	0.14	**20.77**	47.83	**0.93**	0.06	**24.47**	43.90	**0.86**	0.16	23.23	52.47	0.59	0.17	**23.23**	51.53	**0.59**	0.13	**22.93**	48.93	**0.74**
Gemma-2-9B-Instruct	Vanilla	0.32	19.43	9.86	0.87	0.41	17.21	10.74	0.60	0.30	14.88	9.39	0.39	0.33	16.36	9.64	0.43	0.34	16.97	9.91	0.57
	DCA	**0.39**	**16.83**	**5.06**	**0.76**	**0.51**	**12.71**	**5.06**	**0.46**	**0.38**	**9.97**	**4.00**	**0.25**	**0.39**	**13.64**	**6.00**	**0.35**	**0.42**	**13.79**	**5.03**	**0.46**
Llama-3.2-3B-Instruct	Vanilla	**0.31**	42.01	**37.55**	1.90	**0.17**	43.40	38.48	1.57	**0.46**	37.91	**38.69**	0.97	0.18	43.45	**39.95**	1.15	**0.28**	41.19	**38.67**	1.40
	DCA	0.30	**23.20**	46.00	**1.04**	0.15	**23.76**	**38.04**	**0.83**	0.24	**21.00**	50.47	**0.54**	**0.22**	**23.54**	43.62	**0.60**	0.23	**22.88**	44.03	**0.75**

Table 1: Alignment evaluation across OpenBookQA, TruthfulQA, CosmosQA, and MMLU. ↑ indicates higher is better, ↓ indicates lower is better. Best values per column are bolded. Mean values of each metric for each model are also shown for aggregation. All values of ρ are significant (p < 0.01).

gained from pretraining, containing 57 subjects and a wide range of difficulty levels.

For the preference dataset, we use samples from the "train" split of CosmosQA and an equal number of samples split evenly between subjects in the "test" split of MMLU.

For the evaluation dataset, we use all questions from the "test" split of OpenBookQA and the "validation" split of TruthfulQA's multiple choice subset for evaluation on out-of-distribution (OOD) datasets, as well as an equal sample of questions from the "validation" splits of MMLU and CosmosQA for evaluation on in-distribution (ID) datasets.

Further details about the preference and evaluation datasets can be found in A.2.

4.3 Metrics

We use **Spearman's Rank Correlation Coefficient** ρ (Spearman.,1904) to directly evaluate the effectiveness of our method on improving Confidence-Probability Alignment (Kumar et al., 2024). However, ρ only measures the strength of a monotonic correlation and does not reference the perfect calibration line of $y = x$. Hence, we introduce and use three metrics based on calibration error $\epsilon = C_v - C_i$ below:

Standard Deviation of Calibration Error σ_ϵ measures the deviation of individual ϵ values from its mean value, quantifying the variability in ϵ.

Mean Absolute Calibration Error $\overline{|\epsilon|}$ measures the average magnitude of ϵ, showing how much C_v and C_i deviate from each other on average.

Standard Error of Calibration Error σ_M estimates the uncertainty in the mean ϵ, indicating how much the average alignment between C_v and C_i would vary when evaluated on different samples of questions within the same distribution.

These additional metrics are used as they can intrinsically reference the perfect calibration line of $y = x$ as a global extremum and isolate the overall bias within the C_v of the models.

5 Results and Analysis

5.1 Confidence Alignment

Table 1 presents our results for all models across all datasets. Gemma-2-9B-Instruct showed the strongest and most consistent improvements in metrics after DCA, demonstrating superior alignment across all datasets. Most notably, it demonstrated the largest improvements in ρ and $\overline{|\epsilon|}$ of all models on TruthfulQA. However, we observed that Gemma-2-9B-Instruct's initial verbalized and internal confidence distributions were already heavily skewed towards the 90-100% range. This raises a possibility that DCA may have been more successful as a very clear majority of "chosen" confidence values were within this range, thus the training process may have reinforced this existing bias. Consequently, the observed improvements in confidence alignment may partially be due to a collapse towards high confidence values.

In contrast, mixed results were observed for Llama-3.2-3B-Instruct and Mistral-7B-Instruct across all datasets. For example, Llama-3.2-3B-Instruct demonstrates an increase in ρ from 0.18 to 0.22 for MMLU however ρ fell from 0.46 to 0.24 on CosmosQA. Mistral-7B-Instruct demonstrates a large increase in $\overline{|\epsilon|}$ from 19.53 to 52.47 for CosmosQA and a large reduction in ρ from 0.20 to 0.06 on TruthfulQA. These findings indicate that DCA is ineffective for these models on certain tasks.

Gemma-2-9B-Instruct's consistent performance on OOD datasets suggest that DCA was effective at generalising its stronger alignment between C_v and C_i to unseen questions.

Model	OpenBookQA		TruthfulQA		CosmosQA		MMLU	
	Vanilla	DCA	Vanilla	DCA	Vanilla	DCA	Vanilla	DCA
Mistral-7B-Instruct	**59.00%**	58.23%	**32.84%**	20.98%	**60.48%**	54.02%	**55.91%**	48.85%
Gemma-2-9B-Instruct	86.06%	**86.21%**	59.68%	**60.85%**	79.63%	**80.01%**	**72.41%**	72.05%
Llama-3.2-3B-Instruct	47.14%	**64.00%**	29.71%	**37.75%**	66.43%	**73.55%**	39.92%	**49.77%**

Table 2: Comparison of accuracy across our datasets for models before and after DCA. Higher accuracy between Vanilla and DCA versions of each model are in bold.

σ_ϵ and σ_M improved across most models and datasets, suggesting that DCA lowered the variance in calibration error for all models, especially for Llama-3.2-3B-Instruct (see Figure 13 for an example).

However, a low σ_ϵ is only useful if $\overline{|\epsilon|}$ is also low, which would indicate consistent and strong alignment between verbalized and internal confidence as shown by Gemma-2-9B-Instruct (see Figure 12 for an example).

The similarity between results on ID datasets and OOD datasets across models also suggests that the effectiveness of DCA may be more model-dependent than task-dependent, relying more on the model architecture and how different models process confidence elicitation in QA tasks.

5.2 Model Accuracy

Despite our method not being designed to explicitly improve the accuracy of model responses, we also evaluate the downstream effects of DCA on model accuracy. Table 2 shows that DCA can have mixed impacts on model accuracy. While accuracy remained stable on Gemma-2-9B-Instruct, Mistral-7B-Instruct demonstrated lower accuracies after DCA, especially on TruthfulQA. Interestingly, accuracy increased for Llama-3.2-3B-Instruct across all datasets.

6 Conclusion

In this paper we present Direct Confidence Alignment: a method of using DPO to improve the alignment between verbalized and internal confidence in LLMs. Our results show that DCA can be effective at improving this alignment as demonstrated by Gemma-2-9B-Instruct, but also highlight the pressing need for improvements, such as expanding the method to be compatible with a wider range of model architectures, and exploring more strategies to improve this alignment.

Limitations

Access to logits Our method is limited to models with access to internal logits to extract model internal confidence. This makes it inapplicable to state-of-the-art (SOTA) closed-source models.

Reliance on well-calibrated token probabilities Our method will be most useful if the internal confidence of the model is better calibrated against accuracy than its verbalized confidence, and thus may require other ground-truth-based calibration techniques to be used in conjunction for best results.

Impacts of DCA on model accuracy Our method focuses on aligning LLMs' verbalized and internal confidence expressions in their answers rather than directly improving the correctness of those answers. Consequently, the entries in the preference dataset include some incorrect answer choices. We acknowledge this as a potential source of degradation in model accuracy and leave strategies to mitigate this limitation to future work.

References

Mohammad Gheshlaghi Azar, Mark Rowland, Bilal Piot, Daniel Guo, Daniele Calandriello, Michal Valko, and Rémi Munos. 2023. A General Theoretical Paradigm to Understand Learning from Human Preferences. *arXiv preprint*. ArXiv: 2310.12036.

Michael Han Daniel Han and Unsloth team. 2023. Unsloth.

Shrey Desai and Greg Durrett. 2020. Calibration of pre-trained transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 295–302, Online. Association for Computational Linguistics.

Jiahui Geng, Fengyu Cai, Yuxia Wang, Heinz Koeppl, Preslav Nakov, and Iryna Gurevych. 2024. A survey of confidence estimation and calibration in 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 6577–6595, Mexico City, Mexico. Association for Computational Linguistics.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pages 1321–1330. JMLR.org. Event-place: Sydney, NSW, Australia.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring Massive Multitask Language Understanding. *arXiv preprint*. ArXiv: 2009.03300.

Lifu Huang, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Cosmos QA: Machine Reading Comprehension with Contextual Commonsense Reasoning. *arXiv preprint*. ArXiv: 1909.00277.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. 2021. 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.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. 2022. Language Models (Mostly) Know What They Know. *arXiv preprint*. ArXiv: 2207.05221.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2022. Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation.

Abhishek Kumar, Robert Morabito, Sanzhar Umbet, Jad Kabbara, and Ali Emami. 2024. Confidence under the hood: An investigation into the confidence-probability alignment in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 315–334, Bangkok, Thailand. Association for Computational Linguistics.

Jinqi Lai, Wensheng Gan, Jiayang Wu, Zhenlian Qi, and Philip S. Yu. 2024. Large language models in law: A survey. *AI Open*, 5:181–196.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022a. Teaching Models to Express Their Uncertainty in Words. *Transactions on Machine Learning Research*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022b. TruthfulQA: Measuring How Models Mimic Human Falsehoods. *arXiv preprint*. ArXiv: 2109.07958.

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. *arXiv preprint*. ArXiv: 1809.02789.

Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. 2024. A Comprehensive Overview of Large Language Models. *arXiv preprint*. ArXiv: 2307.06435.

Shiyu Ni, Keping Bi, Lulu Yu, and Jiafeng Guo. 2024. Are Large Language Models More Honest in Their Probabilistic or Verbalized Confidence? *arXiv preprint*. ArXiv: 2408.09773.

Seo Yeon Park and Cornelia Caragea. 2022. On the calibration of pre-trained language models using mixup guided by area under the margin and saliency. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5364–5374, Dublin, Ireland. Association for Computational Linguistics.

Cheng Peng, Xi Yang, Aokun Chen, Kaleb E. Smith, Nima PourNejatian, Anthony B. Costa, Cheryl Martin, Mona G. Flores, Ying Zhang, Tanja Magoc, Gloria Lipori, Duane A. Mitchell, Naykky S. Ospina, Mustafa M. Ahmed, William R. Hogan, Elizabeth A. Shenkman, Yi Guo, Jiang Bian, and Yonghui Wu. 2023. A study of generative large language model for medical research and healthcare. *npj Digital Medicine*, 6(1):210.

Gwenyth Portillo Wightman, Alexandra Delucia, and Mark Dredze. 2023. Strength in numbers: Estimating confidence of large language models by prompt agreement. In *Proceedings of the 3rd Workshop on Trustworthy Natural Language Processing (TrustNLP 2023)*, pages 326–362, Toronto, Canada. Association for Computational Linguistics.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2024. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. *arXiv preprint*. ArXiv: 2305.18290.

Shuchang Tao, Liuyi Yao, Hanxing Ding, Yuexiang Xie, Qi Cao, Fei Sun, Jinyang Gao, Huawei Shen, and Bolin Ding. 2024. When to trust LLMs: Aligning confidence with response quality. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 5984–5996, Bangkok, Thailand. Association for Computational Linguistics.

Gemma Team. 2024a. Gemma 2: Improving Open Language Models at a Practical Size. *arXiv preprint*. ArXiv: 2408.00118.

Llama 3 Team. 2024b. The Llama 3 Herd of Models. *arXiv preprint*. ArXiv: 2407.21783.

Mistral Team. 2023. Mistral 7B. *arXiv preprint*. ArXiv: 2310.06825.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher Manning. 2023. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5433–5442, Singapore. Association for Computational Linguistics.

Ante Wang, Linfeng Song, Ye Tian, Baolin Peng, Lifeng Jin, Haitao Mi, Jinsong Su, and Dong Yu. 2024. Self-Consistency Boosts Calibration for Math Reasoning. *arXiv preprint*. ArXiv: 2403.09849.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent Abilities of Large Language Models. *arXiv preprint*. ArXiv: 2206.07682.

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. *arXiv preprint*. ArXiv: 2306.13063.

Tony Z. Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate Before Use: Improving Few-Shot Performance of Language Models. *arXiv preprint*. ArXiv: 2102.09690.

A Appendix

A.1 Prompt Template

{Question} *{Options}* *Provide your best guess (letter only) and the probability that it is correct (0% to 100%) for the above question. Give ONLY the guess and probability, no other words or explanation. For example:* *Guess: <the letter only, as short as possible; not a complete sentence, just the letter!>* *Probability: <the probability between 0% and 100% that your guess is correct, without any extra commentary whatsoever; just the probability!>*

We use a slightly modified version of (Tian et al., 2023)'s Verb. 1S top-1 prompt as our prompt template. We match this prompt across all of our experiments and training processes to ensure consistent responses and output formats during training and post-training evaluation.

A.2 Dataset Details

For the preference dataset, "train" splits were used where possible. However, MMLU's "auxiliary_train" split did not contain subject labels, and hence the "test" split was used to ensure an equal sample of questions from each subject. The final number of instances for the preference dataset was 9348.

For the evaluation dataset, "test" splits were also used where possible. However, TruthfulQA's multiple choice subset only contained only 1 "validation" split, and no test split was available. CosmosQA's test split did not contain answer labels due to it using a leaderboard evaluation system, thus, the "validation" split was used instead. The final number of instances for the evaluation dataset was 4379.

A.3 DCA Training

For DPO training, we use the Unsloth library (Daniel Han and team, 2023) for improved training speeds and efficient memory usage. We loaded LoRA adapters onto our Instruct models using the configurations in Table 3 before training. Training was run on RTX 4000 Ada GPUs, and we used the `ipo` loss function (Azar et al., 2023) to avoid overfitting on the preference dataset. The complete training parameters can be found in Table 4.

A.4 Supplementary Figures

Figures 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, and 13 below show supplementary figures for the results of Mistral-7B-Instruct, Gemma-2-9B-Instruct, and Llama-3.2-3B-Instruct along with their DCA-trained counterparts on OpenbookQA, TruthfulQA, CosmosQA, and MMLU respectively. As seen in the figures, for each model the observed visual trends were broadly consistent across all datasets, also suggesting that the effects of DCA are more model-dependent than task-dependent.

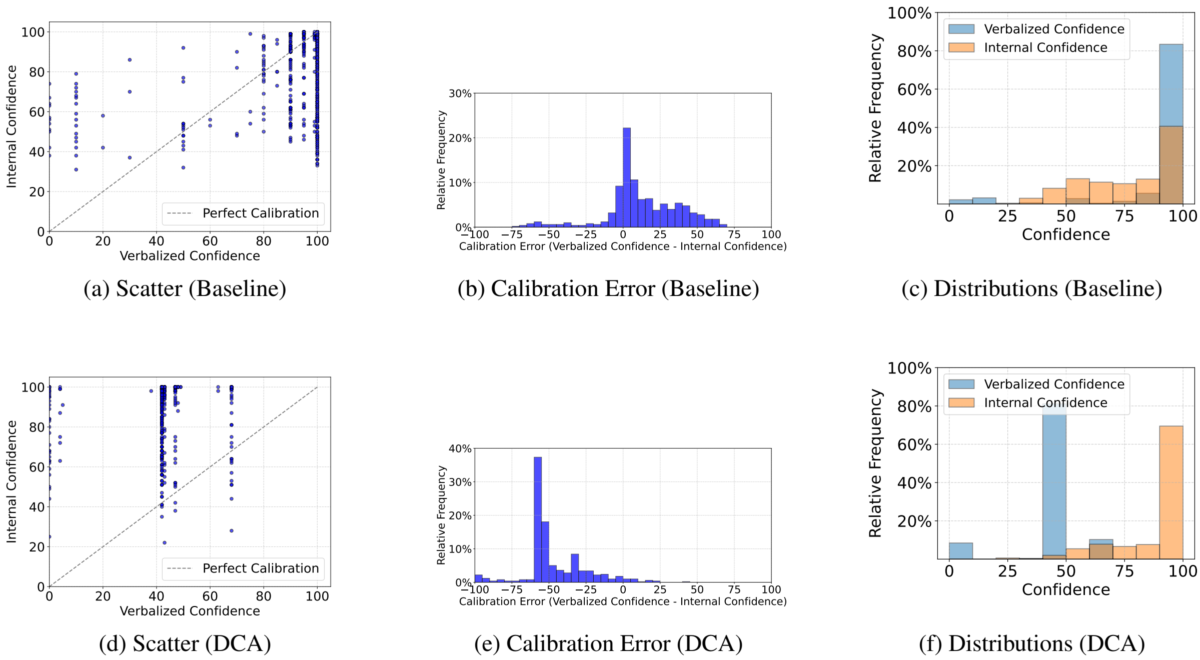
In particular, Mistral-7B-Instruct and Llama-3.2-3B-Instruct demonstrate consistent verbalized underconfidence after DCA across all datasets, with Mistral-7B-Instruct responding with verbalized confidence values between 40-50% for the majority of questions during evaluation, and Llama-3.2-3B-Instruct's verbalized confidence distribution shifting towards 0-50%. Interestingly, the internal confidence distributions of Mistral-7B-Instruct tended to skew more heavily towards higher confidence values after DCA, with an increased number of internal confidence values in the 75-100% range. In addition, the internal confidence distributions of Llama-3.2-3B-Instruct tended to change from favoring confidence values between 25-50% to more skewed towards values of 50-100% after DCA. Unlike the other models, Gemma-2-9B-Instruct's internal and verbalized confidence distributions did not change significantly both before and after DCA.

Hyperparameter	Value	Notes
r (LoRA rank)	16	Low-rank dimension for adapter updates
target_modules	"q_proj", "k_proj", "v_proj", "o_proj", "gate_proj", "up_proj", "down_proj"	Only these weight matrices receive LoRA updates
lora_alpha	16	Scales the low-rank updates
lora_dropout	0.0	No dropout on LoRA adapters
bias	"none"	Do not update any bias parameters in LoRA
use_gradient _checkpointing	"unsloth"	Unsloth's gradient-checkpointing strategy
random_state	3407	Seed for LoRA weight initialization and any randomness
use_rslora	False	Standard LoRA (RSLORA disabled)
loftq_config	None	No custom quantization configuration

Table 3: LoRA / PEFT Hyperparameters

Training Parameter	Value
logging_steps	10
loss_type	ipo
bf16	True
save_steps	100
per_device_train_batch_size	2
gradient_accumulation_steps	32
learning_rate (default)	1e-06
weight_decay (default)	0.0
num_train_epochs (default)	3
optimizer (default)	AdamW (β_1=0.9, β_2=0.999)
lr_scheduler_type (default)	constant (no warmup)
seed	3407

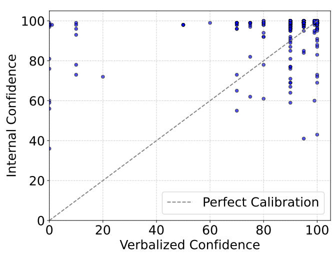
Table 4: DPO Fine-Tuning Hyperparameters



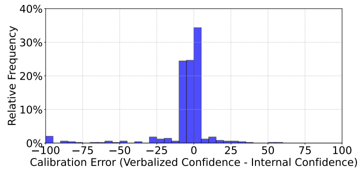
(a) Scatter (Baseline)
(b) Calibration Error (Baseline)
(c) Distributions (Baseline)

(d) Scatter (DCA)
(e) Calibration Error (DCA)
(f) Distributions (DCA)

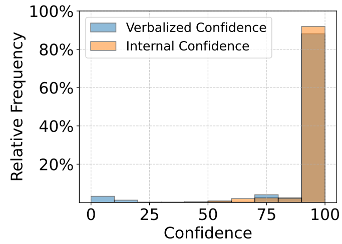
Figure 2: Comparison of baseline vs. DCA-trained Mistral-7B-Instruct on OpenbookQA. Top row: Verbalized vs. internal confidence scatter plot, calibration error histogram, and confidence score distributions for the baseline model. Bottom row: Same visualizations for the DCA-trained model.



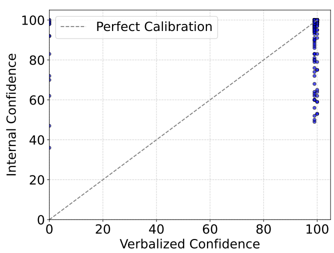
(a) Scatter (Baseline)



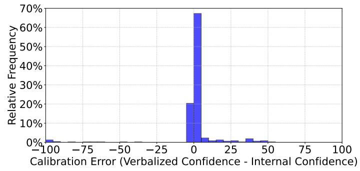
(b) Calibration Error (Baseline)



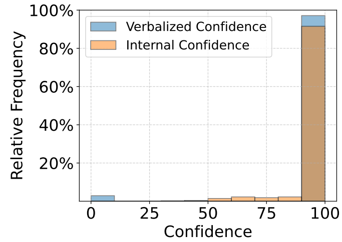
(c) Distributions (Baseline)



(d) Scatter (DCA)

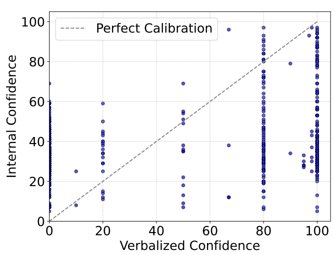


(e) Calibration Error (DCA)

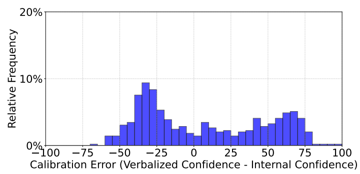


(f) Distributions (DCA)

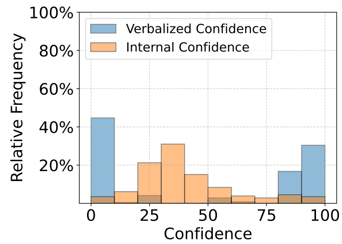
Figure 3: Comparison of baseline vs. DCA-trained Gemma-2-9B-Instruct on OpenbookQA. Top row: Verbalized vs. internal confidence scatter plot, calibration error histogram, and confidence score distributions for the baseline model. Bottom row: Same visualizations for the DCA-trained model.



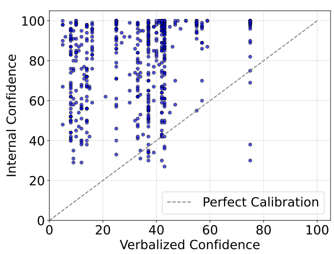
(a) Scatter (Baseline)



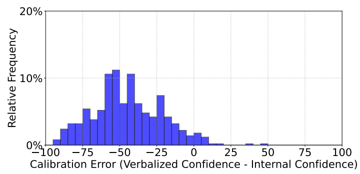
(b) Calibration Error (Baseline)



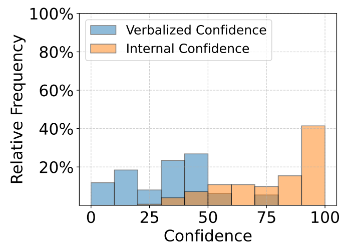
(c) Distributions (Baseline)



(d) Scatter (DCA)

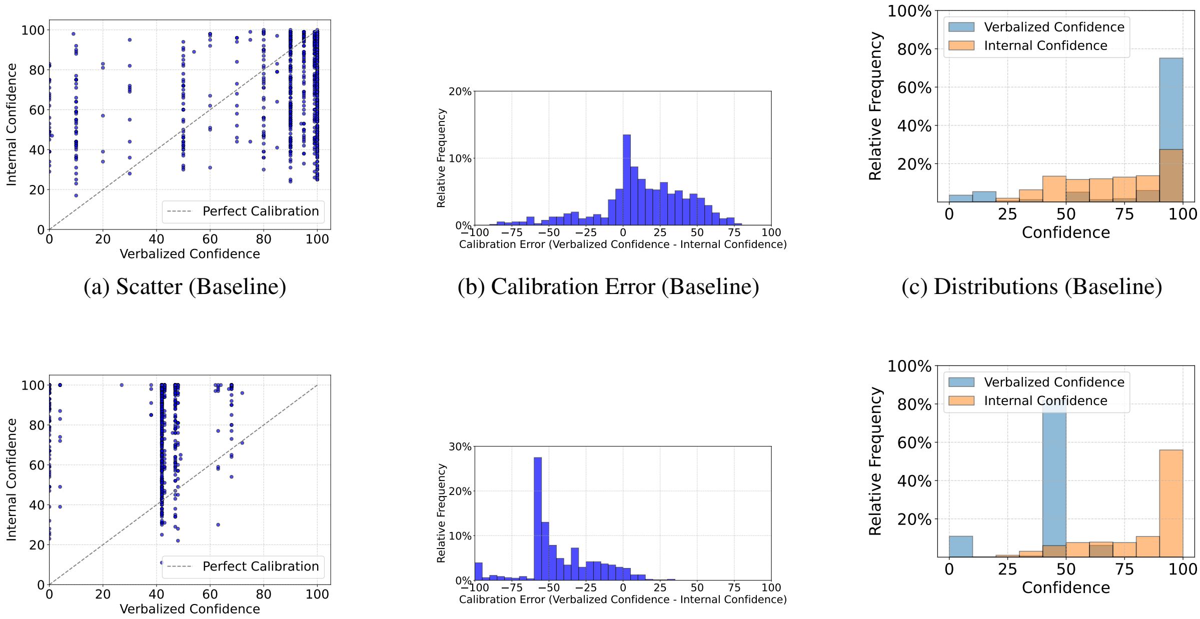


(e) Calibration Error (DCA)



(f) Distributions (DCA)

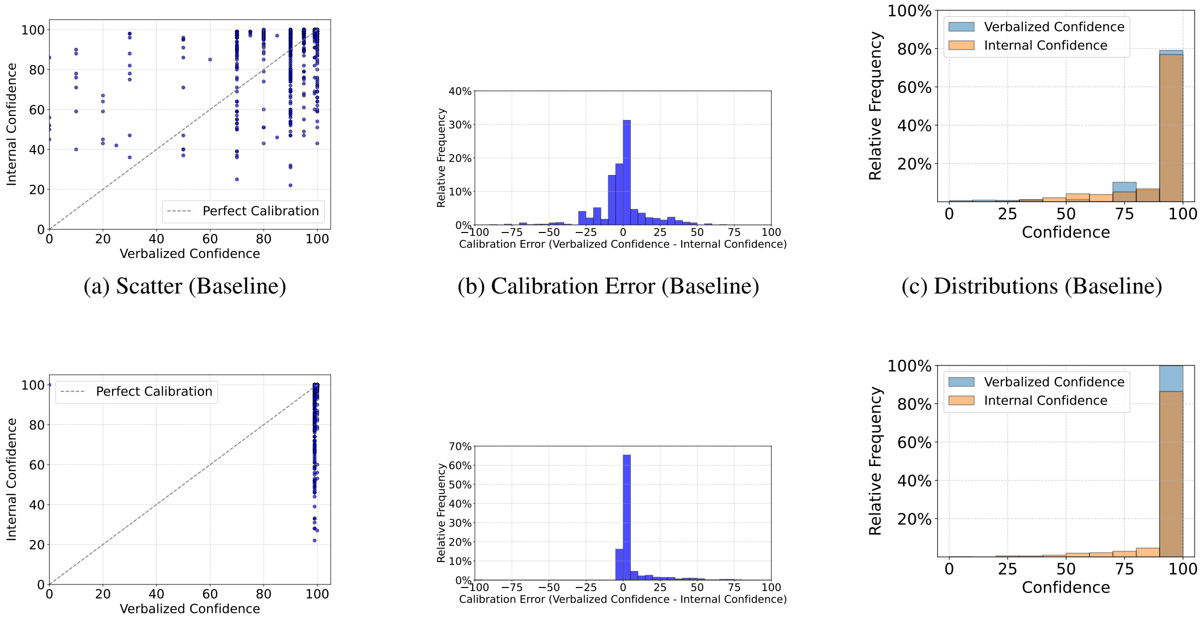
Figure 4: Comparison of baseline vs. DCA-trained Llama-3.2-3B-Instruct on OpenbookQA. Top row: Verbalized vs. internal confidence scatter plot, calibration error histogram, and confidence score distributions for the baseline model. Bottom row: Same visualizations for the DCA-trained model.



(a) Scatter (Baseline)
(b) Calibration Error (Baseline)
(c) Distributions (Baseline)

(d) Scatter (DCA)
(e) Calibration Error (DCA)
(f) Distributions (DCA)

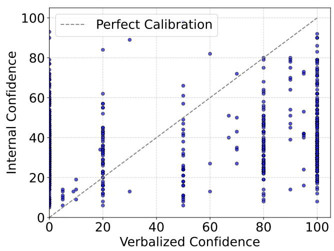
Figure 5: Comparison of baseline vs. DCA-trained Mistral-7B-Instruct on TruthfulQA. Top row: Verbalized vs. internal confidence scatter plot, calibration error histogram, and confidence score distributions for the baseline model. Bottom row: Same visualizations for the DCA-trained model.



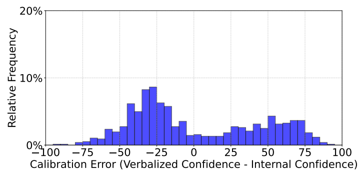
(a) Scatter (Baseline)
(b) Calibration Error (Baseline)
(c) Distributions (Baseline)

(d) Scatter (DCA)
(e) Calibration Error (DCA)
(f) Distributions (DCA)

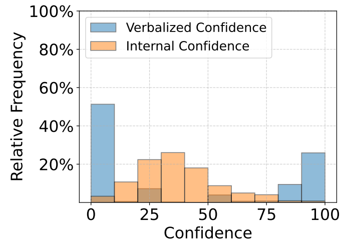
Figure 6: Comparison of baseline vs. DCA-trained Gemma-2-9B-Instruct on TruthfulQA. Top row: Verbalized vs. internal confidence scatter plot, calibration error histogram, and confidence score distributions for the baseline model. Bottom row: Same visualizations for the DCA-trained model.



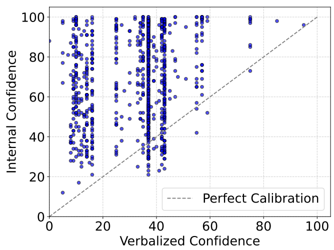
(a) Scatter (Baseline)



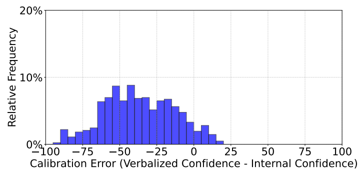
(b) Calibration Error (Baseline)



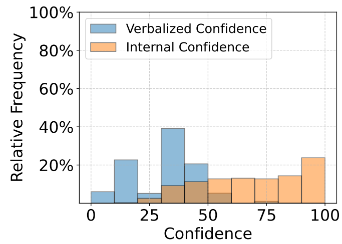
(c) Distributions (Baseline)



(d) Scatter (DCA)

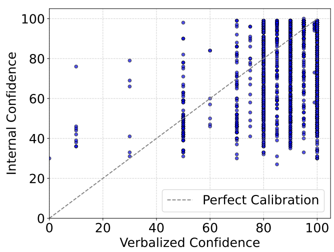


(e) Calibration Error (DCA)

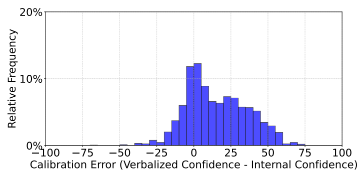


(f) Distributions (DCA)

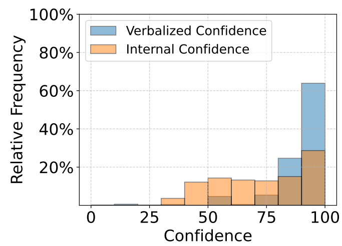
Figure 7: Comparison of baseline vs. DCA-trained Llama-3.2-3B-Instruct on TruthfulQA. Top row: Verbalized vs. internal confidence scatter plot, calibration error histogram, and confidence score distributions for the baseline model. Bottom row: Same visualizations for the DCA-trained model.



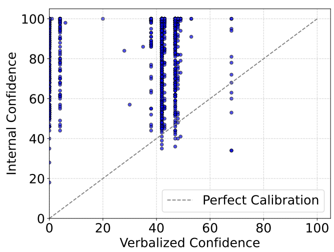
(a) Scatter (Baseline)



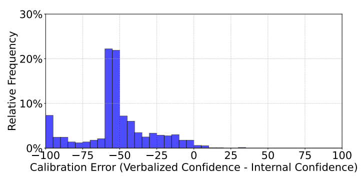
(b) Calibration Error (Baseline)



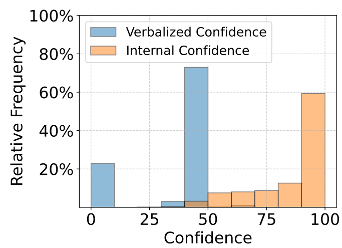
(c) Distributions (Baseline)



(d) Scatter (DCA)

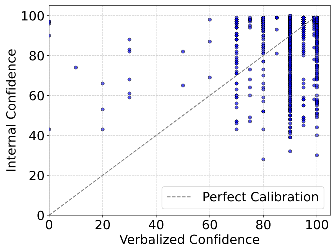


(e) Calibration Error (DCA)

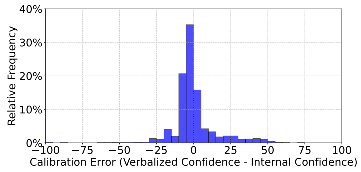


(f) Distributions (DCA)

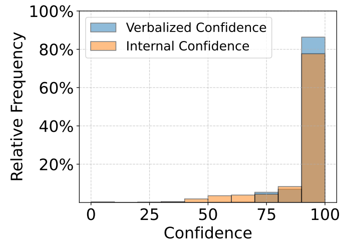
Figure 8: Comparison of baseline vs. DCA-trained Mistral-7B-Instruct on CosmosQA. Top row: Verbalized vs. internal confidence scatter plot, calibration error histogram, and confidence score distributions for the baseline model. Bottom row: Same visualizations for the DCA-trained model.



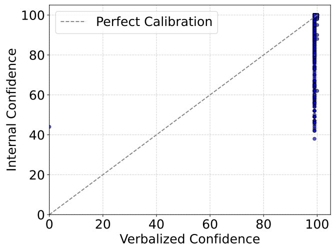
(a) Scatter (Baseline)



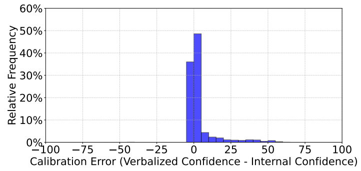
(b) Calibration Error (Baseline)



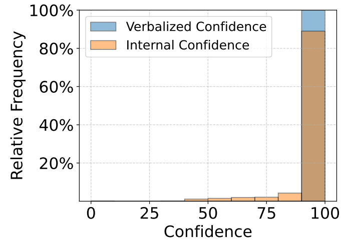
(c) Distributions (Baseline)



(d) Scatter (DCA)

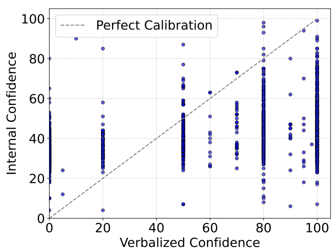


(e) Calibration Error (DCA)

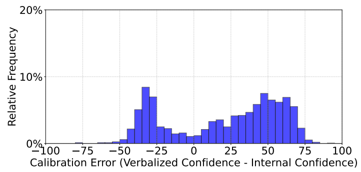


(f) Distributions (DCA)

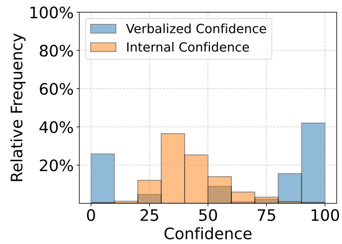
Figure 9: Comparison of baseline vs. DCA-trained Gemma-2-9B-Instruct on CosmosQA. Top row: Verbalized vs. internal confidence scatter plot, calibration error histogram, and confidence score distributions for the baseline model. Bottom row: Same visualizations for the DCA-trained model.



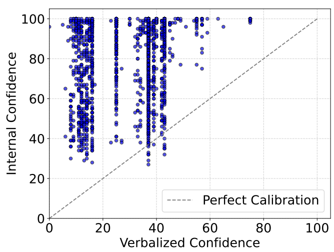
(a) Scatter (Baseline)



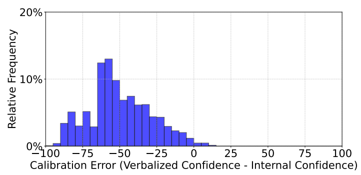
(b) Calibration Error (Baseline)



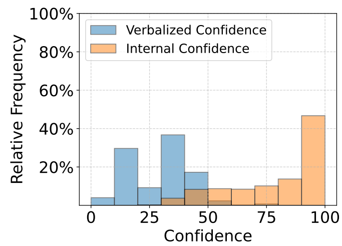
(c) Distributions (Baseline)



(d) Scatter (DCA)

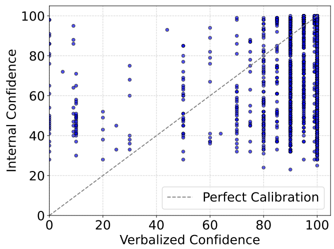


(e) Calibration Error (DCA)

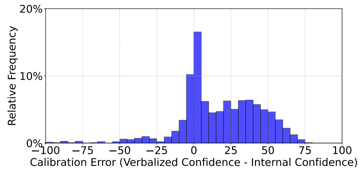


(f) Distributions (DCA)

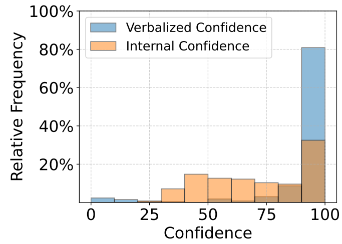
Figure 10: Comparison of baseline vs. DCA-trained Llama-3.2-3B-Instruct on CosmosQA. Top row: Verbalized vs. internal confidence scatter plot, calibration error histogram, and confidence score distributions for the baseline model. Bottom row: Same visualizations for the DCA-trained model.



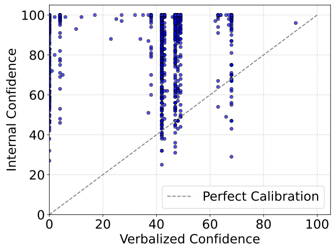
(a) Scatter (Baseline)



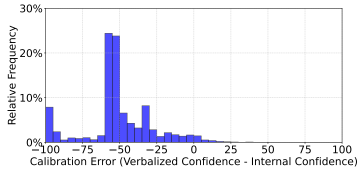
(b) Calibration Error (Baseline)



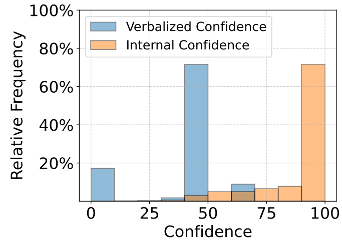
(c) Distributions (Baseline)



(d) Scatter (DCA)

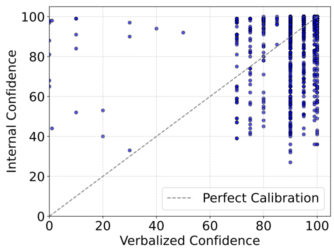


(e) Calibration Error (DCA)

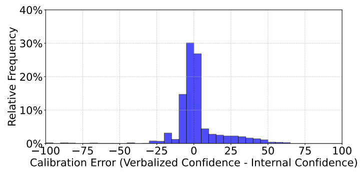


(f) Distributions (DCA)

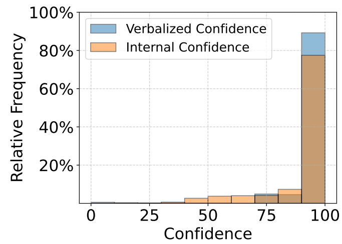
Figure 11: Comparison of baseline vs. DCA-trained Mistral-7B-Instruct on MMLU. Top row: Verbalized vs. internal confidence scatter plot, calibration error histogram, and confidence score distributions for the baseline model. Bottom row: Same visualizations for the DCA-trained model.



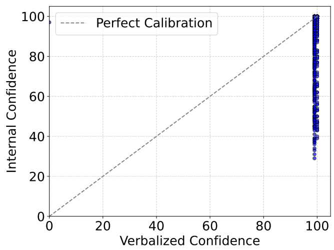
(a) Scatter (Baseline)



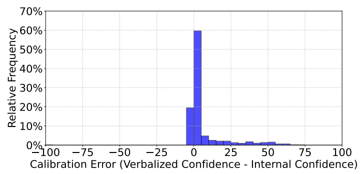
(b) Calibration Error (Baseline)



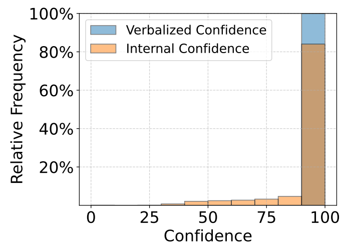
(c) Distributions (Baseline)



(d) Scatter (DCA)

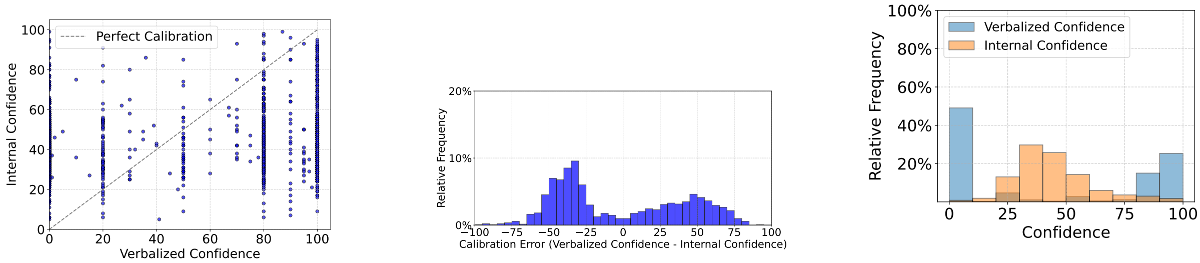


(e) Calibration Error (DCA)



(f) Distributions (DCA)

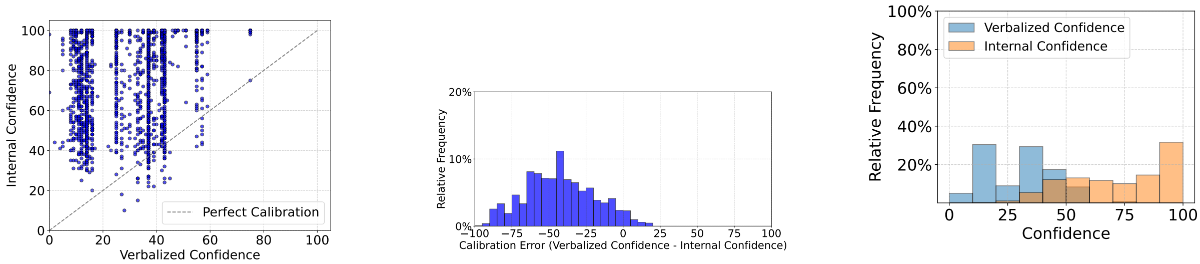
Figure 12: Comparison of baseline vs. DCA-trained Gemma-2-9B-Instruct on MMLU. Top row: Verbalized vs. internal confidence scatter plot, calibration error histogram, and confidence score distributions for the baseline model. Bottom row: Same visualizations for the DCA-trained model.



(a) Scatter (Baseline)

(b) Calibration Error (Baseline)

(c) Distributions (Baseline)



(d) Scatter (DCA)

(e) Calibration Error (DCA)

(f) Distributions (DCA)

Figure 13: Comparison of baseline vs. DCA-trained Llama-3.2-3B-Instruct on MMLU. Top row: Verbalized vs. internal confidence scatter plot, calibration error histogram, and confidence score distributions for the baseline model. Bottom row: Same visualizations for the DCA-trained model.