Workshop at the 7th Symposium on Advances in Approximate Bayesian Inference (non-archival), 2025 1–20

Semantic-Level Confidence Calibration of Language Models via Temperature Scaling

Tom A. Lamb *
Desi Ivanova
Philip H.S. Torr
University of Oxford, Oxford, UK

Tim G.J. Rudner
New York University, NY US

Abstract

Calibration of language models is typically studied at the token level, with scalar temperature scaling serving as the primary approach for recalibrating models. Recent multi-sampling techniques allow us to elicit semantic uncertainty measures from language models. However, these techniques focus on summary statistics of the limited existing semantic confidence distributions rather than on how well-calibrated these distributions are, a crucial factor in ensuring that the resulting semantic likelihoods are both meaningful and reliable. In this paper, we investigate whether and how temperature scaling, which directly influences generative diversity and token-level calibration, affects semantic calibration. We address these question by investigating semantic-level calibration in both pre-trained and fine-tuned models. In particular, we introduce a framework for assessing semantic confidence that incorporates both existing and novel confidence measures, comparing them to a single-generation confidence measure. Furthermore, we investigate various temperature scaling methods and their effect on semantic calibration. Our experiments span both open-book and closed-book question answering datasets. Our empirical findings demonstrate that scalar temperature scaling, when appropriately applied, provides a simple, widely applicable, and effective method for improving semantic calibration in language models.

1. Introduction

Token-level calibration in LMs is well-studied (Achiam et al., 2023), but sentence-level semantic calibration, where a model's confidence reflects the correctness of its meaning, remains underexplored. For instance, both *Tchaikovsky* and *Pyotr Ilyich Tchaikovsky* correctly answer *Which Russian composer wrote the ballets "The Stone Flower" and "Romeo and Juliet"?*, whereas *Shostakovich* and *Sergei Prokofiev* do not. A semantically calibrated model should assign high confidence to correct meanings and low confidence to incorrect ones, expressing uncertainty at a semantic level rather than tied to a particular wording.

Temperature scaling (Guo et al., 2017) is a popular post-hoc technique, adopted for token-level recalibration—especially after RLHF fine-tuning where calibration can degrade (Ouyang et al., 2022; Achiam et al., 2023; Kadavath et al., 2022). Beyond calibration, temperature is used heuristically to control output diversity and for semantic uncertainty estimation: e.g., semantic entropy clusters multiple generations by meaning for uncertainty

* Corresponding Author: thomas.lamb@eng.ox.ac.uk

measures (Kuhn et al., 2023; Farquhar et al., 2024). However, there is no unified approach to deriving semantic confidence scores nor evaluating semantic-level calibration in NLG tasks.

Prior work shows LMs can self-assess via few-shot prompting in open-ended tasks (Kadavath et al., 2022), but this requires an extra assessment step and does not leverage initial-generation likelihoods or operate inherently at the semantic level. Moreover, calibration discussions for NLG beyond multiple-choice QA are limited.

These discussions raise the following key questions that we address in this paper: (i) How can we formally define and measure semantic calibration in NLG? (ii) How does temperature scaling interact within this framework? (iii) Can temperature scaling provide simple semantic-level recalibration akin to its token-level success? In answering these questions, we make the following contributions

1. **Semantic Confidence Framework:** A unified framework integrating existing and novel metrics to evaluate semantic calibration in pre-trained and fine-tuned LMs.
2. **Temperature Scaling Analysis:** A systematic study of scalar and adaptive temperature methods across calibration objectives.
3. **Empirical Evaluation:** Benchmarks on open-book (CoQA, SQuAD) and closed-book (TriviaQA, Natural Questions) QA vs. single-generation confidence methods.
4. **Ablations:** Demonstrating robustness of semantic calibration via temperature scaling to model size and sample count, enabling efficient recalibration.

Reflecting its token-level success, we show that **temperature scaling**, when properly applied, offers a simple and effective means to enhance semantic calibration in LMs.

2. Confidence metrics

We consider an autoregressive language model, p_ϕ, operating on a vocabulary $\mathcal{V}$. Given an input prompt $\boldsymbol{x} \in \mathcal{V}^l$ consisting of l tokens, the LM $p_\phi(\cdot \mid x)$ generates a sequence $\boldsymbol{y} = (y_1, \ldots, y_n) \in \mathcal{V}^n$ of n tokens as a response. The log-probability of the response $\boldsymbol{y}$ given the prompt $\boldsymbol{x}$ under the LM is:

$$\log p_\phi(\boldsymbol{y} \mid \boldsymbol{x}) = \sum_{i=1}^{n} \log p_\phi(y_i \mid \boldsymbol{y}_{<i}, \boldsymbol{x}),, \quad \text{where } \boldsymbol{y}_{<i} = (y_1, \ldots, y_{i-1}). \tag{1}$$

In what follows, we define five confidence measures: one *single generation confidence* (p^{SGC}), and four *semantic confidence* (SC) measures based on a consistency-based approach.

2.1. Single Generation Confidence (SGC)

We interpret the log-probability in Equation 1 as a single-generation confidence (SGC) score, where higher (less negative) values indicate greater confidence. Since log-probabilities depend on the sequence length $n = |\boldsymbol{y}|$ (Wu, 2016), we length-normalise to obtain the length-normalised log-likelihood (LN-LL):

$$\bar{\ell}(\boldsymbol{y} \mid \boldsymbol{x}) = \frac{1}{n} \sum_{i=1}^{n} \log p_\phi(y_i \mid \boldsymbol{y}_{<i}, \boldsymbol{x}). \tag{2}$$

Exponentiating this, equivalent to taking the geometric mean of token probabilities yields:

$$p_\phi^{\text{SGC}}(\boldsymbol{y} \mid \boldsymbol{x}) \coloneqq \exp\left(\bar{\ell}(\boldsymbol{y} \mid \boldsymbol{x})\right) = p_\phi(\boldsymbol{y} \mid \boldsymbol{x})^{\frac{1}{n}} \in [0, 1]. \tag{3}$$

2.2. Semantic Confidence (SC)

For a given input prompt $\boldsymbol{x} \in \mathcal{V}^l$, we generate m responses from a LM: $\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(m)} \sim p_\phi(\cdot \mid \boldsymbol{x})$. We then follow Kuhn et al. (2023) and cluster the responses based on which responses are semantically equivalent. This produces k semantic clusters, $C_1, \ldots, C_k$, where the number of clusters k depends on the input $\boldsymbol{x}$ and generations $\boldsymbol{y}^{(i)}$.

Empirical Semantic Confidence (E-SC). The simplest way to measure the confidence of a semantic cluster C_i would be to consider the empirical proportion of responses that belong to it. We define this to be the *Empirical Semantic Confidence* (E-SC):

$$p_\phi^{\text{E-SC}}(C_i \mid x) \coloneqq \frac{|C_i|}{\sum_{j=1}^{k} |C_j|} = \frac{|C_i|}{m}, \quad i = 1, \ldots, k. \tag{4}$$

We note that this is the same distribution used to compute the semantic entropy of black-box models in (Farquhar et al., 2024).

Likelihood-based Semantic Confidence (L-SC). Assuming access to likelihoods, we define an alternative confidence measure by combining the SGC metric (Equation 2) with the E-SC measure (Equation 4). For each semantic cluster C_i, we compute its score by summing the length-normalised likelihoods of its responses:

$$s(C_i \mid \boldsymbol{x}) \coloneqq \sum_{\boldsymbol{y} \in C_i} p_\phi(\boldsymbol{y} \mid \boldsymbol{x})^{\frac{1}{|\boldsymbol{y}|}}, \quad i = 1, \ldots, k. \tag{5}$$

Normalising these scores yields the *Likelihood-based Semantic Confidence* (L-SC) distribution:

$$p_\phi^{\text{L-SC}}(C_i \mid \boldsymbol{x}) \coloneqq \frac{s(C_i \mid \boldsymbol{x})}{\sum_{j=1}^{k} s(C_j \mid \boldsymbol{x})}, \quad i = 1, \ldots, k. \tag{6}$$

Originally introduced by Farquhar et al. (2024), we use this distribution to measure calibration rather than to compute derived quantities like its entropy.

Mean Likelihood-based Semantic Confidence (ML-SC). Summing length normalised likelihoods may bias scores toward larger clusters, so we compute the mean likelihood for each cluster:

$$\bar{s}(C_i \mid \boldsymbol{x}) \coloneqq \frac{s(C_i \mid \boldsymbol{x})}{|C_i|}, \quad i = 1, \ldots, k. \tag{7}$$

Normalising these scores yields the *Mean Likelihood-based Semantic Confidence*ML-SC distribution:

$$p_\phi^{\text{ML-SC}}(C_i \mid \boldsymbol{x}) \coloneqq \frac{\bar{s}(C_i \mid \boldsymbol{x})}{\sum_{j=1}^{k} \bar{s}(C_j \mid \boldsymbol{x})}, \quad i = 1, \ldots, k. \tag{8}$$

Bayesian Semantic Confidence (B-SC). We introduce a Bayesian inspired semantic confidence measure that merges the E-SC and L-SC approaches. Specifically, we adopt the empirical distribution from Equation 4 as a prior over clusters:

$$\pi(C_j \mid \boldsymbol{x}) \coloneqq p^{\text{E-SC}}(C_j \mid \boldsymbol{x}), \quad \forall j = 1, \ldots, k. \tag{9}$$

Table 1: **Closed-book results.** Table showing the average ECE (↓) and AUROC (↑) scores for single-generation and semantic confidence measures across the three LMs using $m = 10$ generations on closed-book datasets, Natural Questions and TriviaQA. Individual model results are displayed in Figure 7 and Figure 8 in the Appendix.

		SGC		**E-SC**		**ML-SC**		**L-SC**		**B-SC**	
		ECE	AUROC	ECE	AUROC	ECE	AUROC	ECE	AUROC	ECE	AUROC
NQ	p_{PT}	0.343	0.727	0.140	0.707	0.106	0.698	0.156	0.699	0.127	0.707
	p_{SFT}	0.247	0.767	0.105	0.760	0.133	0.728	0.123	0.741	0.116	0.738
	p_{NLL}	0.207	**0.769**	**0.083**	**0.768**	0.123	0.726	0.106	**0.743**	0.098	**0.739**
	p_{SS}	**0.043**	0.757	0.092	0.767	**0.083**	0.693	**0.078**	0.734	**0.084**	0.691
	p_{ATS}	0.245	0.724	0.103	0.761	0.092	**0.728**	0.153	0.740	0.143	0.737
TriviaQA	p_{PT}	0.156	0.765	0.103	0.800	0.170	0.790	0.096	0.793	0.128	0.779
	p_{SFT}	0.120	0.840	0.052	0.850	0.128	0.839	0.060	0.842	**0.054**	**0.846**
	p_{NLL}	0.115	0.840	0.051	0.848	0.133	**0.843**	0.060	0.842	0.059	0.844
	p_{SS}	**0.097**	**0.842**	0.083	**0.861**	0.163	0.841	**0.040**	**0.844**	0.078	0.825
	p_{ATS}	0.177	0.783	**0.046**	0.849	**0.064**	0.833	0.105	0.836	0.119	0.836

We then define the joint length normalised likelihood for all generated responses, $\boldsymbol{y}^{(1):(m)} := (\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(m)})$, given a cluster C_i and input $\boldsymbol{x}$, as

$$\bar{p}_\phi\left(\boldsymbol{y}^{(1):(m)} \mid C_i, \boldsymbol{x}\right) = \prod_{\boldsymbol{y} \in C_i} p_\phi(\boldsymbol{y} \mid \boldsymbol{x})^{\frac{1}{|\boldsymbol{y}|}} = \prod_{\boldsymbol{y} \in C_i} p^{\text{SGC}}(\boldsymbol{y} \mid \boldsymbol{x}), \quad i = 1, \ldots, k. \tag{10}$$

This yields the posterior distribution over clusters:

$$p_\phi^{\text{B-SC}}(C_i \mid \boldsymbol{x}) = \frac{\bar{p}_\phi\left(\boldsymbol{y}^{(1):(m)} \mid C_i, \boldsymbol{x}\right) \pi(C_i \mid \boldsymbol{x})}{\sum_{j=1}^{k} \bar{p}_\phi\left(\boldsymbol{y}^{(1):(m)} \mid C_j, \boldsymbol{x}\right) \pi(C_j \mid \boldsymbol{x})}, \quad i = 1, \ldots, k. \tag{11}$$

We refer to this measure as *Bayesian Semantic Confidence* (B-SC).

Selecting a final response. From the m responses grouped into k clusters, we select the final output by identifying the most confident cluster C^* and choosing the response within it with the highest LN-LL: $\boldsymbol{y}^* =_{\boldsymbol{y} \in C^*} \bar{\ell}(\boldsymbol{y} \mid \boldsymbol{x})$..

3. Experiments

3.1. Experiment setup

Models and Datasets. We assess the semantic calibration of LMs and the effectives of temperature scaling for semantic recalibration using the semantic confidence measures from section 2. Our experiments use Llama-3.1-8B-Instruct (Dubey et al., 2024), Ministral-8B-Instruct-2410 (MistralAI, 2024), and Qwen-2.5-7B-Instruct (Team, 2024), evaluated on generative QA datasets. For closed-book QA, we use TriviaQA (Joshi et al., 2017) and Natural Questions (NQ, Kwiatkowski et al., 2019), and for open-book QA, we use CoQA (Reddy et al., 2019) and SQuAD (Rajpurkar, 2016).

Training procedure: Supervised fine-tuning and calibration. For each dataset, we first perform supervised fine-tuning (SFT) followed by calibration training on a separate

Table 2: **Open-book results.** Table showing the average ECE (↓) and AUROC (↑) scores for single-generation and semantic confidence measures across the three LMs using $m = 10$ generations on the open-book datasets, CoQA and SQuAD. Individual model results are displayed in Figure 9 and Figure 10 in the Appendix.

		SGC		**E-SC**		**ML-SC**		**L-SC**		**B-SC**	
		ECE	AUROC	ECE	AUROC	ECE	AUROC	ECE	AUROC	ECE	AUROC
CoQA	p_{PT}	0.141	0.721	0.156	0.720	0.262	0.705	0.145	0.707	0.191	0.712
	p_{SFT}	**0.048**	0.747	0.069	0.792	0.145	0.793	0.063	0.786	**0.084**	**0.789**
	p_{NLL}	0.050	**0.748**	0.063	0.800	0.145	0.792	**0.059**	0.789	0.087	0.785
	p_{SS}	0.106	0.735	0.103	**0.802**	0.200	**0.797**	0.069	**0.795**	0.120	0.717
	p_{ATS}	0.111	0.729	**0.062**	0.797	**0.080**	0.787	0.079	0.782	0.099	0.782
SQuAD	p_{PT}	0.086	**0.729**	**0.071**	0.594	0.088	0.598	0.072	0.588	0.082	0.628
	p_{SFT}	**0.036**	0.688	0.099	0.695	0.167	0.688	0.087	0.690	0.105	0.699
	p_{NLL}	0.038	0.687	0.097	0.692	0.167	0.689	0.089	0.681	0.107	0.701
	p_{SS}	0.077	0.670	0.135	**0.737**	0.230	**0.737**	0.115	**0.706**	0.153	**0.731**
	p_{ATS}	0.049	0.679	0.099	0.683	**0.078**	0.684	**0.055**	0.681	**0.052**	0.685

subset, reflecting deployment of small- to medium-sized models. All metrics are reported on a held-out test split.

For SFT, we apply parameter-efficient fine-tuning (PEFT) using LoRA (Hu et al., 2021), selecting the model with the best accuracy on a held-out SFT-validation set for calibration. Full details on dataset splits, training, and hyperparameters are provided in section C.

Calibration and Discrimination Metrics. We assess calibration and discriminative ability using Expected Calibration Error (ECE) and AUROC (see Appendix D.1) respectively. For short-form QA tasks, we view correctness as binary, defined as $c = \mathbf{1}(\hat{\boldsymbol{y}} \sim \boldsymbol{y} \mid \boldsymbol{x})$, where $\sim$ denotes semantic equivalence given $\boldsymbol{x}$, and $\hat{\boldsymbol{y}} \sim p_\phi(\cdot \mid \boldsymbol{x}; \tau)$ is the final response. As described in section 2, the final response to be assessed is selected as the one with the highest LN-LL within the most confident cluster C_*. We report calibration metrics both for the LN-LL of this response and for its corresponding confidence cluster.

Calibration methods and baselines. Our empirical investigation is focused on the role of temperature scaling in semantic calibration. We compare a range of temperature calibration methods and baselines using the aforementioned metrics on held-out test sets. The calibration methods and models we consider are:

- **No calibration baselines:** *the original pre-trained LM*, p_{PT}, and the SFT model p_{SFT}.
- **Scalar temperature optimisation methods:** with the *NLL loss* (Eq. 13) and the *SS loss* (Eq. 14) (Xie et al., 2024). We denote results pertaining to these methods by p_{CE} and p_{SS}, respectively.
- **Adaptive temperature scaling**: optimising a temperature prediction head as in subsection B.2, using the *SS loss* (Eq. 14), $p_{\text{ATS-SS}}$ (Xie et al., 2024).

3.2. Semantic calibration on closed-book datasets (TriviaQA and NQ)

Table 1 summarises the average ECE and AUROC scores for single-generation and SC measures across three LMs using $m = 10$ generations on the closed-book datasets, Natural Questions and TriviaQA. Note that single-generation confidence reflects a single output's

score, while SC measures capture a broader confidence in the overall response meaning; these metrics are thus fundamentally different and not directly comparable.

The pre-trained models p_{PT} exhibit high ECE values, indicating poor calibration, whereas p_{SFT} shows improvement. Moreover, all temperature scaling methods consistently lower ECE values, demonstrating enhanced calibration in both cases. Notably, the SS loss method generally yields the best overall calibration performance, outperforming the more complex and expressive $p_{\text{ATS-SS}}$ method although we do see greater improvements of the $p_{\text{ATS-SS}}$ method on TriviaQA for E-SC and ML-SC methods. **In conclusion, temperature scaling improves semantic calibration as well as single-generation confidence calibration.**

Regarding AUROC, which measures discrimination ability, temperature scaling methods generally achieve higher scores than both p_{PT} and p_{SFT} (except for B-SC on TriviaQA), **further supporting the benefits of temperature optimisation for both calibration and discrimination**.

3.3. Semantic calibration on open-book datasets (CoQA and SQuAD)

Table 2 shows the average ECE and AUROC scores for single generation and semantic confidence measures using $m = 10$ generations on the open book datasets CoQA and SQuAD. For SC measures (except E-SC on SQuAD and B-SC on CoQA), temperature scaling reduces ECE relative to p_{PT} and p_{SFT}; in contrast, SGC shows no such improvement on either dataset. Although the optimal method for ECE improvement is less clear cut than for the closed-book datasets, the ATS method generally yields better calibration as evidenced by, for example, by the lowest ECE scores of 0.080 on ML-SC for CoQA and 0.078 on ML-SC for SQuAD, suggesting that longer context aids in training the adaptive temperature head. Regarding AUROC, the SS loss method consistently achieves the highest scores, which in context of these open-book datasets results, indicates enhanced discriminability at the expense of calibration. **Overall, our findings indicate that temperature scaling, particularly via ATS, generally improves SC but tends to hurt single-generation confidence measures on the open-book datasets that we experiment on.**

4. Conclusion

In this work, we investigated both existing and novel extensions of semantic confidence measures, focusing on their calibration at a semantic level. Furthermore, we explored how temperature scaling, a simple token-level recalibration technique that affects sample diversity and hence multi-sample confidence measures, can influence semantic calibration within our framework. Our results indicate that pre-trained models tend to be poorly calibrated at a semantic level, whereas applying temperature scaling to fine-tuned models generally yields improvements over both pre-trained and fine-tuned variants. This shows that simple token-level calibration techniques can be extended to improve calibration at the more meaningful semantic level for NLG tasks. We hope that these findings will motivate future research into the development of more advanced methods for improving semantic calibration using token-level operations.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Maximilian Bachmann. Rapidfuzz: A fast fuzzy string matching library in c++ and python, 2021. URL https://github.com/maxbachmann/RapidFuzz.

Neil Band, Xuechen Li, Tengyu Ma, and Tatsunori Hashimoto. Linguistic calibration of language models. *arXiv preprint arXiv:2404.00474*, 2024.

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

Andrew P Bradley. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145–1159, 1997.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 15(3):1–45, 2024.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630, 2024.

Peter A Flach. Classifier calibration. In *Encyclopedia of machine learning and data mining*. Springer US, 2016.

Markus Freitag and Yaser Al-Onaizan. Beam search strategies for neural machine translation. *arXiv preprint arXiv:1702.01806*, 2017.

Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477):359–378, 2007.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR, 2017.

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. *arXiv preprint arXiv:2106.09685*, 2021.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*, 2017.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *International Conference on Learning Representations*, 2023.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.

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

Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. Generating with confidence: Uncertainty quantification for black-box large language models. *arXiv preprint arXiv:2305.19187*, 2023.

Jeremiah Liu, Zi Lin, Shreyas Padhy, Dustin Tran, Tania Bedrax Weiss, and Balaji Lakshminarayanan. Simple and principled uncertainty estimation with deterministic deep learning via distance awareness. *Advances in neural information processing systems*, 33: 7498–7512, 2020.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

MistralAI. Introducing ministrel: Our new lightweight model, October 2024. URL https://mistral.ai/news/ministraux/. Accessed: 2025-01-17.

Jishnu Mukhoti, Andreas Kirsch, Joost van Amersfoort, Philip HS Torr, and Yarin Gal. Deep deterministic uncertainty: A new simple baseline. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24384–24394, 2023.

Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29 (1), 2015.

Alexander Nikitin, Jannik Kossen, Yarin Gal, and Pekka Marttinen. Kernel language entropy: Fine-grained uncertainty quantification for llms from semantic similarities. *arXiv preprint arXiv:2405.20003*, 2024.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, pages 27730–27744, 2022.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

P Rajpurkar. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*, 2016.

Siva Reddy, Danqi Chen, and Christopher D Manning. Coqa: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7: 249–266, 2019.

Qwen Team. Qwen2.5: Advancing open-source language models, September 2024. URL https://qwenlm.github.io/blog/qwen2.5/. Accessed: 2025-01-17.

Dennis Ulmer, Martin Gubri, Hwaran Lee, Sangdoo Yun, and Seong Joon Oh. Calibrating large language models using their generations only. *arXiv preprint arXiv:2403.05973*, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In Marilyn Walker, Heng Ji, and Amanda Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1101. URL https://aclanthology.org/N18-1101/.

Yonghui Wu. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016.

Johnathan Xie, Annie S Chen, Yoonho Lee, Eric Mitchell, and Chelsea Finn. Calibrating language models with adaptive temperature scaling. *arXiv preprint arXiv:2409.19817*, 2024.

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. *arXiv preprint arXiv:2306.13063*, 2023.

Adam X Yang, Maxime Robeyns, Xi Wang, and Laurence Aitchison. Bayesian low-rank adaptation for large language models. *arXiv preprint arXiv:2308.13111*, 2023.

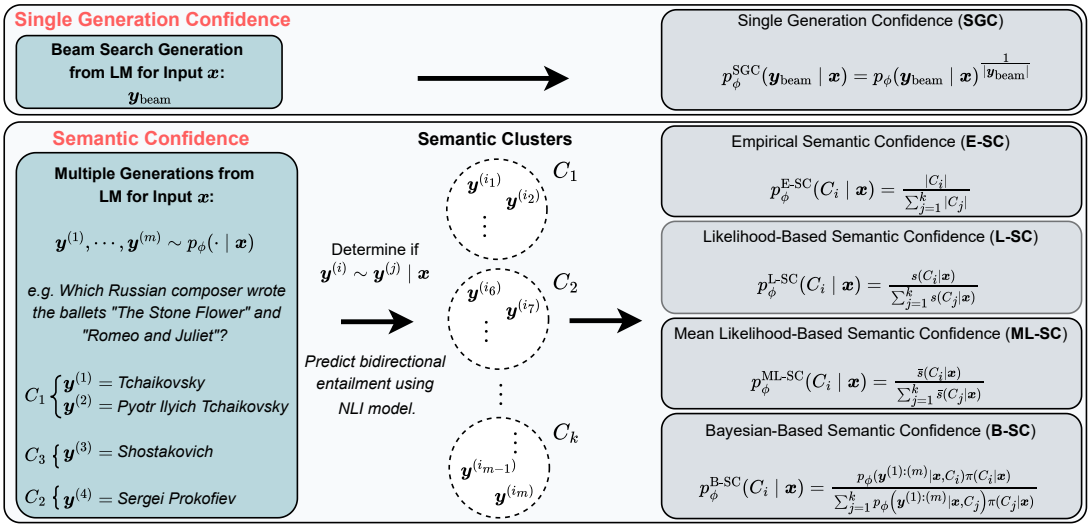

Figure 1: **Generative Confidence Framework**. This framework integrates single-generation confidence (SGC) and semantic confidence (SC) methods for LMs. SGC (top box) is the length-normalised likelihood of a beam-searched response for prompt $\boldsymbol{x}$. SC (bottom box) computes a semantic confidence score by sampling multiple responses from $p_\phi(\cdot \mid \boldsymbol{x})$ and using an NLI model to assess bidirectional entailment, determining whether responses $\boldsymbol{y}^i$ and $\boldsymbol{y}^j$ are semantically equivalent ($\boldsymbol{y}^i \sim \boldsymbol{y}^j$). Here, $s(C \mid \boldsymbol{x})$ denotes the sum, and $\bar{s}(C \mid \boldsymbol{x})$ the average, of length-normalised log-likelihoods within cluster C.

Appendix A. Background and Related Work

Confidence and calibration for LMs. Confidence in large language models (LLMs) typically relies on model likelihoods, derived from model outputs, post-processing steps, or additional modules (Ulmer et al., 2024). Calibration, defined as the alignment between model confidence and correctness, has traditionally been examined primarily at the token level. While pre-trained models such as `GPT-4` exhibit strong token-level calibration (Achiam et al., 2023), this calibration often degrades following fine-tuning (Achiam et al., 2023; Kadavath et al., 2022). To address this, temperature scaling, originally introduced for recalibration by Guo et al. (2017), is widely applied in LLMs to adjust confidence and control response diversity (Chang et al., 2024). Recent approaches have gone beyond scalar temperatures, introducing input-dependent temperature parameters trained per decoding step (Xie et al., 2024) to enable more granular calibration adjustments. Furthermore, Xie et al. (2024) employ a selective smoothing loss to increase uncertainty specifically on incorrectly predicted tokens.

However, token-level calibration alone is insufficient for open-ended generative tasks. Such tasks require uncertainty to be assessed at the *semantic or meaning level*, accommodating semantically equivalent correct responses. Band et al. (Band et al., 2024) introduce linguistic calibration (LC) for long-form generation, calibrating outputs to yield downstream user forecasts. Nevertheless, this method does not directly ensure that the model's self-reported confidence reflects response correctness semantically. In contrast, our work introduces a

formal framework explicitly designed to elicit and evaluate semantic confidence scores and their calibration.

Semantic measures of uncertainty for LMs. Traditional uncertainty quantification methods such as Bayesian techniques (Blundell et al., 2015), latent-space metrics (Mukhoti et al., 2023; Liu et al., 2020), predictive entropy, and ensembles (Lakshminarayanan et al., 2017)—primarily target classification tasks, often proving challenging to scale for modern large language models (LLMs) (Yang et al., 2023). These methods typically quantify uncertainty at token or prediction levels, limiting their applicability to open-ended generative tasks requiring semantic-level uncertainty evaluation due to multiple valid outputs (Kuhn et al., 2023).

Recently, semantic uncertainty approaches have emerged, utilizing multi-sampling and consistency-based strategies (Kuhn et al., 2023; Lin et al., 2023; Nikitin et al., 2024). These methods cluster semantically similar outputs to measure uncertainty at the meaning level. Notably, semantic entropy (SE) (Kuhn et al., 2023) employs natural language inference (Williams et al., 2018) to group outputs and compute uncertainty, while graph-based (Lin et al., 2023) and kernel-based (Nikitin et al., 2024) extensions further refine this approach.

However, current semantic uncertainty methods do not directly measure whether a model's reported confidence aligns with the semantic correctness of its outputs. Similarly, self-reported confidence approaches (Kadavath et al., 2022; Xiong et al., 2023) usually operate post-hoc and lack a direct semantic connection to initial predictions. Thus, there's a clear need for methods explicitly linking semantic confidence to the correctness of model-generated responses, a gap this work aims to address.

Appendix B. Confidence Calibration

A model is well-calibrated when its confidence matches its empirical accuracy (Flach, 2016; Ulmer et al., 2024). Calibration techniques adjust output probabilities for over- or underconfident predictions. We focus on temperature scaling (Guo et al., 2017) for its simplicity, popularity, and dual role in controlling response diversity and token-level calibration.

B.1. Scalar Temperature Scaling (STS)

Given logits $\boldsymbol{z}_t \in \mathbb{R}^{|\mathcal{V}|}$ at decoding step t from an LM $p_\phi(\cdot \mid \boldsymbol{x})$, the output probabilities are computed as $p_\phi(y_t \mid \boldsymbol{x}, \boldsymbol{y}_{<t}; \tau) = \sigma\left(\boldsymbol{z}_t/\tau\right)$, where $\sigma : \mathbb{R}^{|\mathcal{V}|} \to \Delta^{|\mathcal{V}|-1}$ is the softmax function and $\tau > 0$ is the scalar temperature parameter.

Temperature scaling adjusts token confidence and modulates diversity: lower temperatures yield more deterministic outputs (with $\tau \to 0$ equivalent to greedy decoding) while higher temperatures promote diversity. However, under deterministic methods such as beam search (Freitag and Al-Onaizan, 2017), temperature scaling does not affect candidate ranking and thus the final output.

B.2. Adaptive Temperature Scaling (ATS)

ATS (Xie et al., 2024) replaces the global scalar $\tau \in \mathbb{R}_{>0}$ with token-specific temperatures via a learned prediction head. Given input $\boldsymbol{x}$ and final hidden representations $\boldsymbol{h} \in \mathbb{R}^{d_{\text{model}} \times n}$,

ATS applies a transformation $\psi : \mathbb{R}^{d_{\text{model}} \times n} \to \mathbb{R}^n$, implemented as a single-layer transformer block (Vaswani et al., 2017), to produce a scalar temperature for each token position:

$$\boldsymbol{\tau}^{-1} = \exp(\psi(\boldsymbol{h})), \quad p_\phi(y_t \mid \boldsymbol{x}, \boldsymbol{y}_{<t}; \boldsymbol{\tau}) = \sigma\left(\boldsymbol{z}_t / \boldsymbol{\tau}_t\right), \tag{12}$$

with all operations on $\boldsymbol{\tau} \in \mathbb{R}^n$ performed element-wise.

B.3. Calibration Loss Functions

Negative Log-Likelihood (NLL). Calibration is often achieved by optimizing the negative log-likelihood, equivalent to the standard cross-entropy loss with one-hot targets and a proper scoring rule (Gneiting and Raftery, 2007). The NLL loss is defined as:

$$\ell_{\text{NLL}}(p_\phi(\cdot \mid \boldsymbol{x}, \boldsymbol{y}_{<t}; \tau), y_t) = -\log p_\phi(y_t \mid \boldsymbol{x}, \boldsymbol{y}_{<t}; \tau). \tag{13}$$

Selective Smoothing (SS). Introduced by Xie et al. (2024), the selective smoothing loss minimizes the NLL for correct token predictions while maximizing the entropy for incorrect ones:

$$\begin{aligned} &\ell_{\text{SS}}(p_\phi(\cdot \mid \boldsymbol{x}, \boldsymbol{y}_{<t}; \tau), y_t) \\ &= -(1-\alpha) \log p_\phi(y_t \mid \boldsymbol{x}, \boldsymbol{y}_{<t}; \tau) \mathbf{1}(\hat{y}_t = y_t) - \frac{\alpha}{|\mathcal{V}|} \sum_{y \in \mathcal{V}} \log p_\phi(y \mid \boldsymbol{x}, \boldsymbol{y}_{<t}; \tau) \mathbf{1}(\hat{y}_t \neq y_t), \end{aligned} \tag{14}$$

where $\mathbf{1}(\cdot)$ is the indicator function, $\hat{y}_t = _{y \in \mathcal{V}}\, p_\phi(y \mid \boldsymbol{x}, \boldsymbol{y}_{<t}; \tau)$ is the model's top prediction, and $\alpha \in [0, 1]$ controls the balance between the two terms. We employ this loss to optimise the temperature head in the ATS method described in subsection B.2.

Appendix C. Training and Optimisation Settings

C.1. Dataset splits

Recall that our training procedure consists of two stages: Supervised fine-tuning stage (SFT) and post-SFT calibration via temperature optimisation. Accordingly, we split original data sets into 7 splits, 6 of which are used during training and one is a held-out test set.

- *Supervised fine-tuning stage.* We have SFT-training, SFT-early-stopping SFT-validation, used for SFT training and selection of LMs. We fine-tune models on the SFT-training set, applying early stopping based on the accuracy on the SFT-early-stopping set. The final SFT model is chosen as the model with the highest accuracy on the SFT-validation. This model then proceeds to the calibration stage.

- *Calibration stage:* In this stage, we have calibration-training, calibration-early-stopping and calibration-validation data splits. Using the best-performing SFT model, we conduct calibration training on the calibration-training split. We use early stopping based on the calibration loss on the calibration-validation split. We then choose the hyperparameter settings that optimises each metric for each confidence measure independently, and then proceed to use this hyperparameter settings to report the final metric score for the particular confidence score on the test set.

- *Final Evaluation* We use the test split for evaluating and reporting final calibration and discriminative metrics (ECE, AUROC score) of all methods based on the hyperparameter settings chosen during the calibration-stage.

For further clarity, Figure 2 illustrates visually the training procedures and dataset splits used within our SFT and calibration training pipeline described above.

To ensure fair comparison across datasets, we restrict each split size to be comparable for each dataset. Table 3 table gives each dataset's specific split sizes.

Table 3: **Dataset sizes.** Dataset split sizes for SFT, calibration, and final evaluation stages across datasets.

Stage	Split	TriviaQA	Natural Questions	CoQA	SQuAD
SFT	Training	41639	43200	42441	41781
	Early stopping	1156	1200	1178	1160
	Validation	3471	3600	3538	3483
Calib.	Training	12337	12800	12575	12380
	Early-Stopping	771	800	785	773
	Validation	2314	2400	2359	2322
Final Eval.	Test	4000	3610	4000	4000

C.2. Hyperparameter Settings for SFT and Calibration

Below we list the hyperparameter settings swept over for both SFT and temperature calibration. Note that for both SFT and temperature calibration across all methods, we use the AdamW optimiser (Loshchilov and Hutter, 2017) with a cosine-annealing learning rate scheduler with a linear warm-up consisting of 10% of the first epoch of optimisation (Radford et al., 2018). We additionally perform early stopping based on a held out early stopping set for a particular training stage (SFT vs calibration), with a patience of 4 epochs. For SFT we early stop based on early-stopping accuracy, whereas for each calibration method, we early stop based on the early stopping loss value.

SFT Training. For the SFT training stage, we use PEFT using LoRA (Hu et al., 2021) . We sweep over the following set of hyperparamaters:

- Learning rates: $[10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}]$.
- Weight decay: $[0.0, 0.01]$.
- Maximum number of training epochs: [16].
- LoRA α : [64]
- LoRA $r = [32]$.

Scalar Temperature Optimisation. For each scalar temperature loss used for temperature calibration, we sweep over the following set of hyperparamaters:

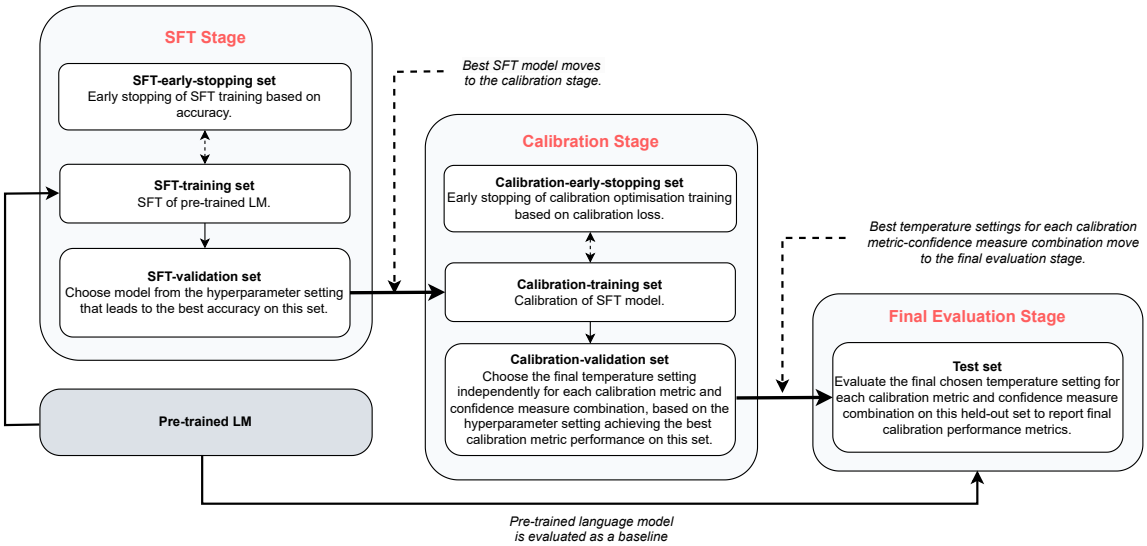

Figure 2: **SFT-calibration training and evaluation pipeline.** Figure illustrating the dataset splits and how they are used within our SFT and calibration training pipelines described in subsection C.1.

- Learning rates: $[10^{-4}, 10^{-3}, 10^{-2}]$.
- Initial temperature value, τ: $[1.0]$.
- Loss weight α for the SS loss: $[0.25, 0.5, 0.75]$.
- Maximum number of training epochs: $[32]$.

Temperature head optimisation. For temperature head optimisation, ATS-SS, during calibration, we sweep over the following set of hyperparamaters:

- Learning rates: $[10^{-6}, 10^{-5}, 10^{-5}]$ - we use smaller learning to maintain stability during optimisation.
- Weight decay: $[0.0, 0.01]$
- Loss weight α for the SS loss: $[0.25, 0.5, 0.75]$.
- Gradient clipping with max gradient norm: $[1.0]$.
- Maximum number of training epochs: $[32]$
- Temperature head architecture: We a single transformer block for the temperature head with the same architecture as for that of the $8B$ parameter Llama-3 model (Dubey et al., 2024), with the internal model dimension adapted to the base model architecture for which the temperature head is being optimised for.

Appendix D. Evaluation

D.1. Evaluation metrics

Expected Calibration Error (ECE) ECE quantifies the misalignment between predicted confidence and actual correctness:

$$\mathbb{E}_{\boldsymbol{x}\sim\mathbb{P}(\cdot)}\left[|\mathbb{P}(c=1 \mid p_\phi(\hat{\boldsymbol{y}} \mid \boldsymbol{x})=p)-p|\right]. \tag{15}$$

Following Naeini et al. (2015), ECE is estimated by binning predictions into M intervals and computing the weighted average of the absolute accuracy-confidence difference:

$$\mathrm{ECE}=\sum_{m=1}^{M}\frac{|B_m|}{n}|\mathrm{acc}(B_m)-\mathrm{conf}(B_m)|. \tag{16}$$

AUROC The Area Under the Receiver Operating Characteristic Curve (AUROC) measures how well confidence scores distinguish between correct and incorrect responses. The ROC curve is constructed by varying a confidence threshold λ and plotting the true positive rate (TPR) against the false positive rate (FPR) at each threshold (Bradley, 1997).

Given a dataset of N examples with inputs $\boldsymbol{x}_i$, model-generated responses $\hat{\boldsymbol{y}}_i$, correctness indicators $c_i=\mathbf{1}(\hat{\boldsymbol{y}}_i\sim\boldsymbol{y}_i)$, and model confidence scores $p_\phi(\hat{\boldsymbol{y}}_i \mid \boldsymbol{x}_i)$, we define:

$$\mathrm{TPR}(\lambda)=\frac{\sum_{i:c_i=1}\mathbf{1}(p_\phi(\hat{\boldsymbol{y}}_i \mid \boldsymbol{x}_i)\geq\lambda)}{\sum_{i:c_i=1}1},\quad \mathrm{FPR}(\lambda)=\frac{\sum_{i:c_i=0}\mathbf{1}(p_\phi(\hat{\boldsymbol{y}}_i \mid \boldsymbol{x}_i)\geq\lambda)}{\sum_{i:c_i=0}1}. \tag{17}$$

AUROC is then computed as:

$$\mathrm{AUROC}=\int_0^1 \mathrm{TPR}(\lambda)\,d\mathrm{FPR}(\lambda), \tag{18}$$

which represents the probability that a randomly chosen correct response receives a higher confidence score than a randomly chosen incorrect response.

A higher AUROC indicates better uncertainty quantification, with AUROC = 0.5 corresponding to a random or uninformative confidence metric.

D.2. Evaluation of Accuracy

To assess accuracy in generative QA tasks, we implement a series of checks designed to balance correctness assessment with computational efficiency:

- **Initial text cleaning:** We first preprocess the model's response by discarding any extraneous text beyond its first direct answer to the posed question.
- **Direct answer matching:** If one of the reference answers appears in the model's response, we consider the response correct.
- **Fuzzy matching:** If the reference answer is not directly present, we apply fuzzy matching (Bachmann, 2021), leveraging string-distance metrics to check for semantically similar generations. If the fuzzy-match ratio exceeds 90, we classify the response as correct.

- **SQuAD F1 evaluation:** If the response remains unverified, we compute the SQuAD-F1 metric. If the F1 score is above 50.0, we deem the model's response correct.

An alternative approach to accuracy evaluation could involve using an additional model such as Llama-3.1 (Dubey et al., 2024) or GPT-4 (Achiam et al., 2023) as a judge to determine whether a response is equivalent to a reference answer. However, this introduces additional computational, time, and cost constraints. Our current methodology aligns with prior work (Kuhn et al., 2023; Farquhar et al., 2024) in the literature that uses token level matching criteria and performs effectively in practice for assessing model correctness. Unlike (Farquhar et al., 2024), we find that replying on more accuracy checks than just using the SQuAD-F1 is necessary to mitigate false negative judgments of correctness.

Appendix E. Results

E.1. Model Accuracies on Test Set

Model	TriviaQA	Natural Questions	CoQA	SQuAD
LLaMA-8B-Instruct	70.6	39.8	69.2	91.2
LLaMA-8B-Instruct-SFT	74.0	54.2	77.7	96.2
Mistral-8B-Instruct	66.0	32.4	70.3	90.5
Mistral-8B-Instruct-SFT	71.0	46.0	78.3	95.7
Qwen-7B-Instruct	59.7	30.4	68.7	91.4
Qwen-7B-Instruct-SFT	62.0	41.7	77.8	95.8

Table 4: **Pre-trained and FT Model Test Set Accuracies (%).** Test set accuracy (%) on NQ, TriviaQA and CoQA datasets for pre-trained and fine-tuned (without any further calibration) models using beam decoding.

E.2. Semantic Calibration and model size

We evaluate the impact of model size on semantic calibration using Qwen-2.5-Instruct models with 1.5B, 3B, and 7B parameters, each generating 10 samples per test prompt. As shown in Figure 3, SGC for the pre-trained models exhibits minimal changes in ECE or AUROC from 1.5B to 7B, whereas SC measures for the pre-trained worsen in calibration as model size increases. Meanwhile, temperature scaling methods consistently improve calibration over both pre-trained and SFT models, with ECE for these methods remaining largely stable, and with AUROC steadily improving in discriminability as model size increases. Overall, these results highlight three key points: (1) **SGC metrics remain relatively unaffected by model size**, (2) **larger pre-trained models can exhibit poorer semantic calibration**, and (3) **temperature scaling enhances discriminability and mitigates calibration degradation of pre-trained models, and improves calibration for SFT models as model size increases**.

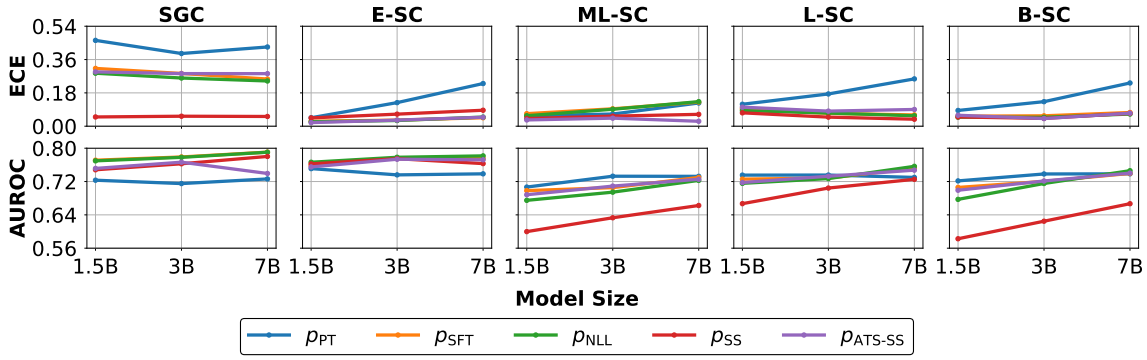

Figure 3: **SC measures over varying model size.** ECE and AUROC for both SGC and SC confidence measures over 1.5B, 3B and 7B models selected from the Qwen-2.5-Instruct family of models across SC measures.

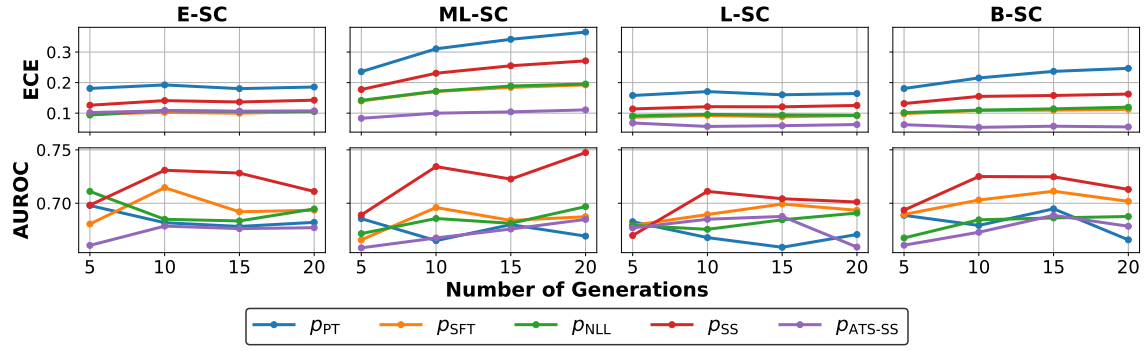

Figure 4: **SC measures over varying numbers of generations.** ECE and AUROC metrics for the Llama-3.1-8B-Instruct model on the SQuAD dataset over varying number of sample generations per input prompt.

E.3. Semantic calibration and number of generations

We evaluate the effect of the number of sample generations m on calibration and discriminability of SC measures. Figure Figure 4 reports ECE and AUROC for Llama-3.1-8B-Instruct on SQuAD for $m \in \{5, 10, 15, 20\}$. ECE remains largely stable across pre-trained, SFT, and temperature-calibrated models, except on ML-SC where we observe a mild degradation, particularly for the pre-trained model. In contrast, AUROC shows more sensitivity to m, with most methods peaking around $m = 10$ or $m = 15$, followed by a decline. In conclusion, **(1) Increasing the number of generations has negligible impact on ECE**, and **(2) AUROC benefits from moderate m (10–15 generations), but degrades beyond this.**

E.4. Distribution and reliability plots showing influence of temperature calibration on SC measures

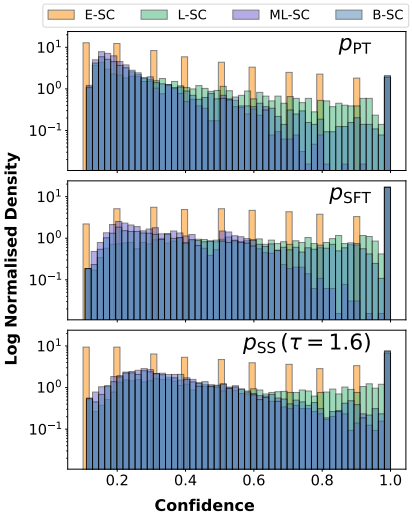

Figure 5: **SC Distributions.** Log-density of SC measures for the p_{PT}, p_{SFT}, and p_{SS} Mistral models.

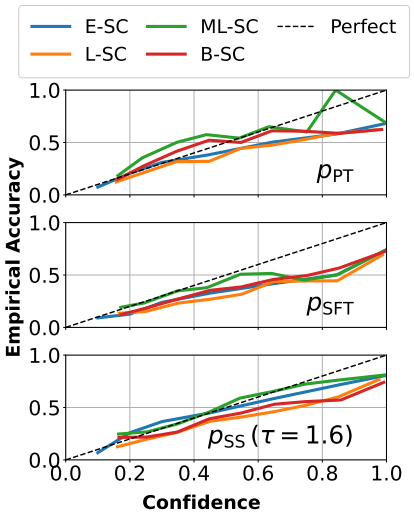

Figure 6: **Reliability Plots.** Reliability plots of SC measures for the p_{PT}, p_{SFT}, and p_{SS} Mistral models.

E.5. Final Temperature values attained via Temperature Calibration

	TriviaQA	Natural Questions	CoQA	SQuAD
p_{SS}	1.41 ± 0.171	1.6 ± 0.216	1.49 ± 0.162	1.49 ± 0.143
p_{CE}	1.02 ± 0.0153	1.09 ± 0.113	1.01 ± 0.0165	1.02 ± 0.0307

Table 5: **Average temperatures achieved through calibration training.** Average temperature paramters across all hyperparameter settings swept over for scalar temperature calibration. Results show mean ± sample standard deviation for each scalar temperature optimisation method and dataset.

E.6. Fine-Grained Results

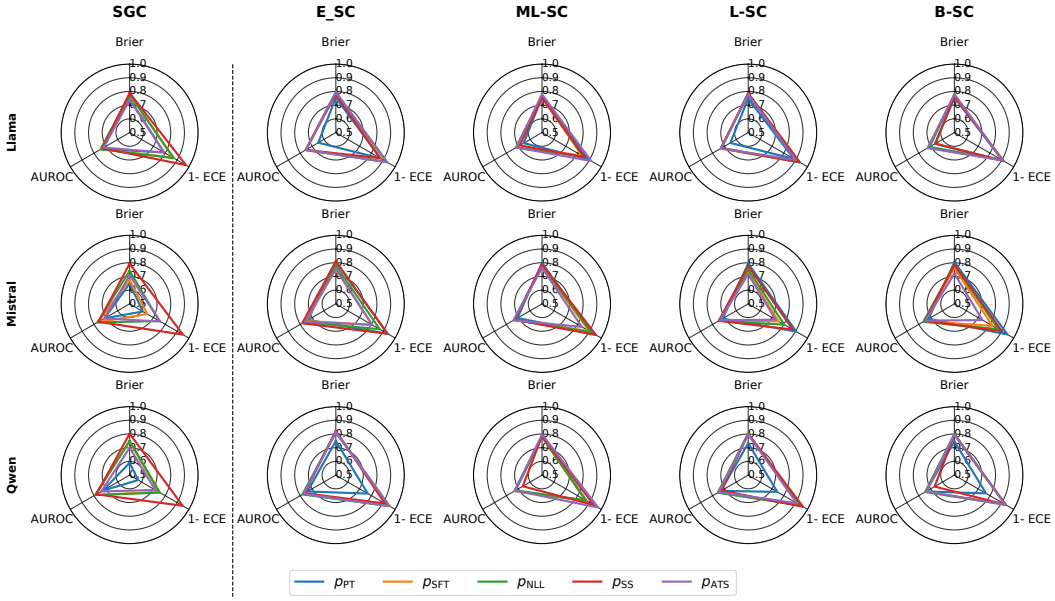

Figure 7: **Avg Calibration Metrics Using 10 Generations on NQ (↑).** Figure showing the average of different calibration metrics on the NQ dataset using different confidence measures (columns) for three different models (rows) across three repeats.

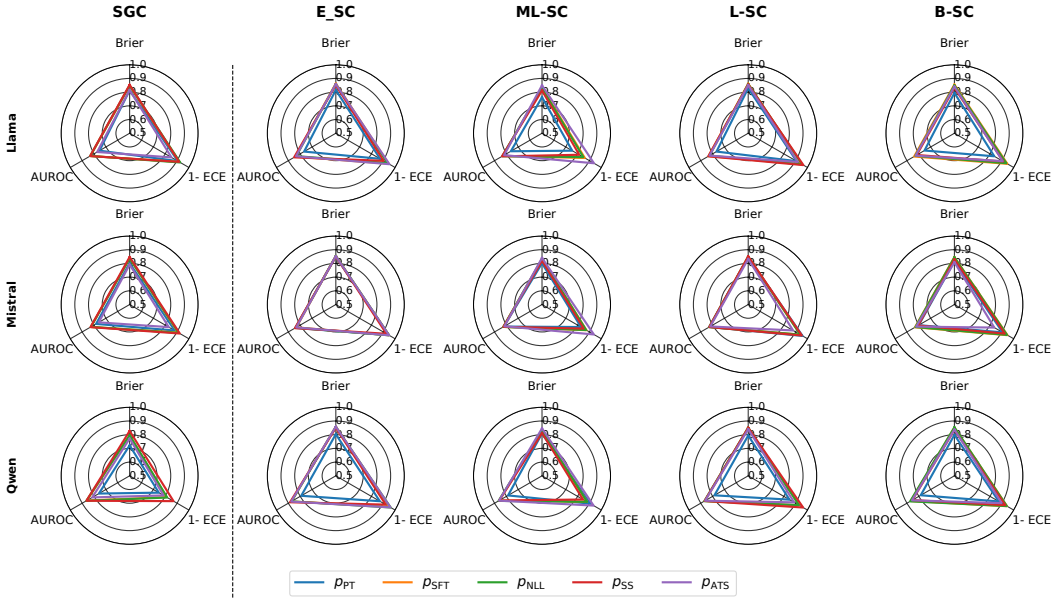

Figure 8: **Avg Calibration Metrics Using 10 Generations on TriviaQA (↑).** Figure showing the average of different calibration metrics on the TriviaQA dataset using different confidence measures (columns) for three different models (rows) across three repeats.

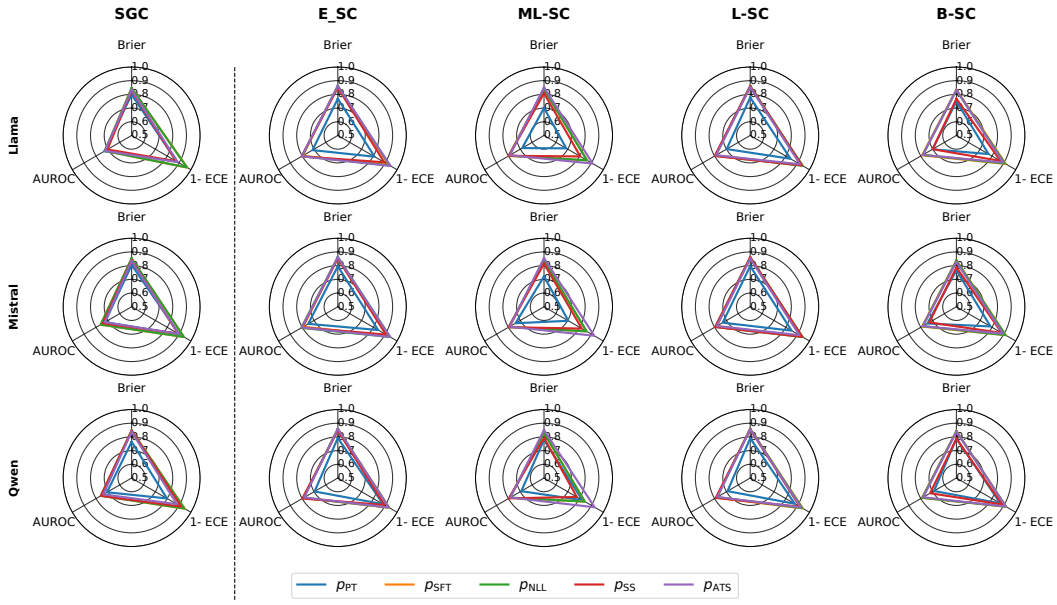

Figure 9: **Avg Calibration Metrics Using 10 Generations on CoQA (↑).** Figure showing the average of different calibration metrics on the CoQA dataset using different confidence measures (columns) for three different models (rows) across three repeats.

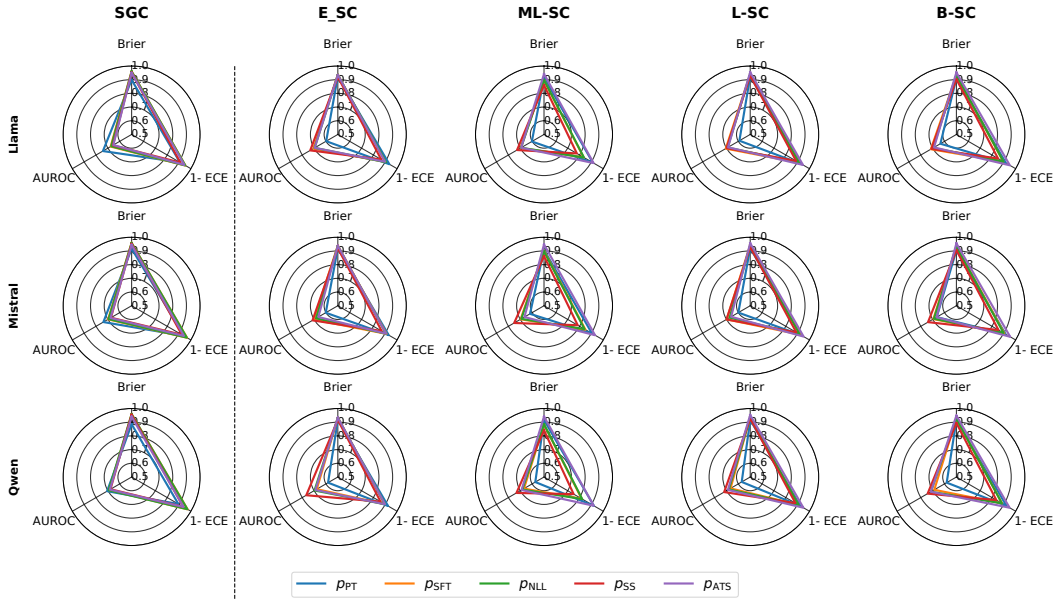

Figure 10: **Avg Calibration Metrics Using 10 Generations on SQuAD (↑).** Figure showing the average of different calibration metrics on the SQuAD dataset using different confidence measures (columns) for three different models (rows) across three repeats.