Self-Consistency Boosts Calibration for Math Reasoning

Ante Wang[1,2], **Linfeng Song**[3*], **Ye Tian**[3], **Baolin Peng**[3], **Lifeng Jin**[3], **Haitao Mi**[3],
Jinsong Su[1,2*] and **Dong Yu**[3]
[1]School of Informatics, Xiamen University, China
[2]Key Laboratory of Digital Protection and Intelligent Processing of Intangible Cultural Heritage of Fujian and Taiwan (Xiamen University), Ministry of Culture and Tourism, China
[3]Tencent AI Lab, Bellevue, WA
wangante@stu.xmu.edu.cn, lfsong@global.tencent.com, jssu@xmu.edu.cn

Abstract

Calibration, which establishes the correlation between accuracy and model confidence, is important for LLM development. We design three off-the-shelf calibration methods based on self-consistency (Wang et al., 2022) for math reasoning tasks. Evaluation on two popular benchmarks (GSM8K and MathQA) using strong open-source LLMs (Mistral and LLaMA2) shows that our methods better bridge model confidence and accuracy than existing methods based on *p(True)* (Kadavath et al., 2022) or *logit* (Guo et al., 2017).

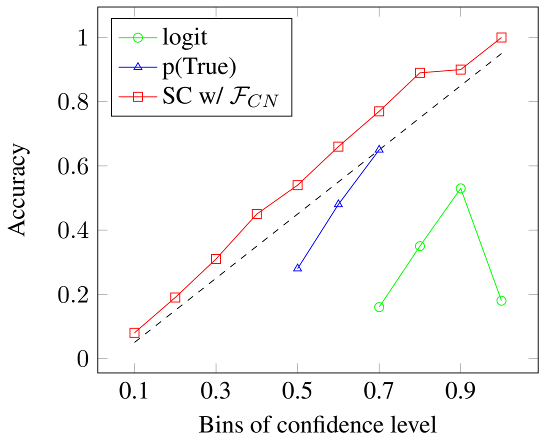

Figure 1: Comparison of several calibration methods on Mistral-7B, where *SC w/* $\mathcal{F}_{CN}$ is one of our methods based on self-consistency, which will be introduced in §3.

1 Introduction

Mathematical reasoning tasks (Cobbe et al., 2021; Hendrycks et al., 2021; Amini et al., 2019) involve mapping a question into a series of equations, which are then solved to obtain the final answer. Math reasoning has long been recognized challenging. Existing solutions propose to map input questions into equations via semantic parsing (Matsuzaki et al., 2017; Hopkins et al., 2017) or AST decoding (Li et al., 2019; Qin et al., 2021; Wu et al., 2021). Yet, the performance can degradate dramatically even with slight changes to the questions (Patel et al., 2021; Li et al., 2022).

Recently, large language models (LLM, Achiam et al. 2023; Touvron et al. 2023; Jiang et al. 2024) have shown great potential for solving many math reasoning tasks, even though they are not specifically trained on these tasks. For instance, with chain-of-thought prompting (Wei et al., 2022) and self-consistency (Wang et al., 2022), open-source LLMs, such as Mixtral 8×7B (Jiang et al., 2024), can reach an accuracy of around 80% on the GSM8K benchmark (Cobbe et al., 2021). On the other hand, conventional pretrained models (e.g., T5 (Raffel et al., 2020)) that are specifically fine-tuned on the GSM8K training set can only report accuracies around 10% to 20% (Shridhar et al., 2023; Magister et al., 2023).

However, LLMs lack adequate calibration out of the box – the probabilities of model predictions are often poorly aligned with the actual accuracy (Xiong et al., 2023; Chen et al., 2023). Calibration is important for LLM development, as a well-calibrated LLM can precisely *tell* how likely its responses are correct or not. With such information, LLM developers can take multiple options to handle low-confidence responses, such as letting the LLM refuse to answer or keep resampling until a confident response is produced.

In this work, we propose calibration methods based on self-consistency (Wang et al., 2022) for math reasoning tasks. Self-consistency performs clustering over multiple LLM samples before picking one from the largest cluster as the response to each input query. Here we consider several ways to estimate model confidence using the clustering results: *cluster size* that estimates how many samples agree with the selected one, *cluster number* that measures to what extent samples disagree with

*Corresponding Authors

Findings of the Association for Computational Linguistics: EMNLP 2024, pages 6023–6029

each other, and *pairwise comparison* that captures relative differences between pairs of clusters.

We conduct experiments using strong open-source LLMs: Mistral (Jiang et al., 2023, 2024) and LLaMA2 (Touvron et al., 2023) series models with / without being aligned with instructions. Results on GSM8K (Cobbe et al., 2021) and MathQA (Amini et al., 2019) show that all our methods better calibrate these models than exiting popular methods, such as *p(True)* (Kadavath et al., 2022) and *logit* (Guo et al., 2017) over the whole reasoning path or target answer span only.

2 Preview: Self-Consistency with CoT Prompting

For math reasoning, there are usually multiple trajectories to reach the final solution. To replicate this process, Wang et al. (2022) initially sample various reasoning paths $r_1, ..., r_N$ from the LLM given input x with Chain-of-Thought (CoT) prompting.[1] Then, the answers $a_1, ..., a_N$ are extracted from the paths, and the most consistent answer (the one win by majority vote among the answers) is selected as the final answer $\mathbf{a}$:

$$\begin{aligned} \mathbf{a} &= \max_{\hat{a}} \sum_{i=1}^{N} \mathbb{1}(a_i = \hat{a}), \\ r_i, a_i &\sim \texttt{LLM}_\theta(x), \end{aligned} \tag{1}$$

where r_i, a_i denote the i-th sampled reasoning path and its corresponding answer, respectively.

3 Calibration using Self-Consistency

After performing self-consistency on input x using $\texttt{LLM}_\theta$, we obtain a set of clusters $\mathcal{C} = \{c_1, ..., c_{|\mathcal{C}|}\}$ with each cluster c_i comprising n_i sampled responses with the same answers. We design the following strategies, tailored to the characteristics of these clusters, to estimate the confidence of $\texttt{LLM}_\theta$.

Cluster Number We initially consider the *Cluster Number* $|\mathcal{C}|$. This is motivated by the finding of previous work (Wang et al., 2022; Xiong et al., 2023): LLMs tend to generate consistent answers when they are confident about their predictions, and thus the cluster number (number of distinct answers) tends to be small. We further divide the cluster number by the sample size N to normalize the score into the range of $[0, 1]$, before reversing it by "$1 - x$":

$$\mathcal{F}_{CN}(x, \theta) = 1 - \frac{|\mathcal{C}|}{N}. \tag{2}$$

Cluster Size In a similar vein, we adopt the *Cluster Size*: the number of samples (e.g., n_i) within a specific cluster (e.g., c_i). Again, we compute its proportion relative to the total sample size to normalize the score into the range $[0, 1]$:

$$\mathcal{F}_{CS}(x, \theta) = \frac{n_i}{N}. \tag{3}$$

In contrast to the cluster number, the cluster size is more universally applicable across diverse prompts, as the cluster number can easily become ineffective when the output space of an LLM is restricted, such as when options for a question are provided.

Pairwise Comparison The *Cluster Number* and *Cluster Size* primarily consider the number of distinct answers and the number of sampled paths within a single cluster, respectively. They both overlook the information by comparing different clusters. For example, they may fail to consider the situation when the sizes of the top-ranked clusters are close. Consequently, we introduce the *Pairwise Comparison* method, which computes the winning rate of the chosen cluster (c_i) against each of the remaining clusters:

$$\mathcal{F}_{PC}(x, \theta) = \prod_{j \neq i}^{|\mathcal{C}|} \frac{n_i}{n_i + n_j}, \tag{4}$$

where $\frac{n_i}{n_i + n_j}$ represents the winning rate of selected cluster c_i against another cluster c_j by comparing the respective cluster sizes.

4 Experiments

4.1 Setup

Datasets We conduct experiments on two popular math reasoning benchmarks of different type of questions, GSM8K (Cobbe et al., 2021) and MathQA (Amini et al., 2019). Particularly, GSM8K comprises 1,319 linguistically diverse grade school math word problems for testing. On the other hand, MathQA offers 2,985 *multiple-choice* math word problems for evaluation.

Evaluation Metrics We adopt Brier Score and Expected Calibration Error (ECE) as evaluating

[1] Here we follow common practice to adopt demonstrations with rationales for pretrained only models (e.g., Mistral-7B) and use "*Let's think step by step*" (Kojima et al., 2022) for instruction-tuned models (e.g., Mistral-7B-Inst).

		Mistral-7B		Mistral-7B-Inst		Mixtral-8×7B		Mixtral-8×7B-Inst	
		ECE ↓	Brier ↓	ECE ↓	Brier ↓	ECE ↓	Brier ↓	ECE ↓	Brier ↓
GSM8K	logit w/ Path	0.394	0.399	0.414	0.414	0.178	0.265	0.233	0.252
	logit w/ Answer	0.505	0.488	0.467	0.458	0.307	0.312	0.236	0.238
	p(True)	0.127	0.267	0.406	0.407	**0.070**	0.201	0.195	0.198
	Self-Consistency								
	w/ $\mathcal{F}_{CN}$	**0.092**	0.186	0.125	0.182	0.136	0.157	**0.075**	0.092
	w/ $\mathcal{F}_{CS}$	0.148	0.185	0.163	0.180	0.173	0.156	0.085	0.086
	w/ $\mathcal{F}_{PC}$	0.248	0.229	0.253	0.226	0.238	0.194	0.110	0.096
	w/ ALL	0.101	**0.172**	**0.123**	**0.162**	0.135	**0.141**	0.078	**0.083**
MathQA	logit w/ Path	0.500	0.499	0.539	0.510	0.333	0.380	0.364	0.373
	logit w/ Answer	0.356	0.362	0.291	0.319	0.266	0.290	0.220	0.281
	p(True)	0.350	0.309	0.271	0.317	0.228	0.253	0.273	0.272
	Self-Consistency								
	w/ $\mathcal{F}_{CN}$	0.331	0.336	0.374	0.359	0.143	0.236	0.128	0.215
	w/ $\mathcal{F}_{CS}$	0.091	0.225	0.114	0.227	**0.080**	**0.190**	**0.035**	**0.171**
	w/ $\mathcal{F}_{PC}$	**0.052**	**0.220**	**0.065**	**0.219**	0.143	0.203	0.054	0.174
	w/ ALL	0.144	0.238	0.172	0.242	0.089	**0.190**	0.072	0.177

Table 1: Main test results on GSM8K and MathQA when using Mistral family models. Specifically, ∗-Inst indicates instruction-tuned models.

metrics following common practice (Geng et al., 2023).

Given instances $(x_1, y_1), ..., (x_{\bar{N}}, y_{\bar{N}})$ and their corresponding LLM predictions $\hat{y}_1, ..., \hat{y}_{\bar{N}}$, ECE is computed by first binning the predictions into $M = 10$ intervals based on their LLM confidence levels (e.g., $p(\hat{y}_i)$). For each bin (e.g. B_m), it then calculates the accuracy (acc(B_m)) and the average confidence (conf(B_m)):

$$\begin{aligned} \text{acc}(B_m) &= \frac{1}{|B_m|} \sum_{i \in B_m} \mathbb{1}(y_i = \hat{y}_i), \\ \text{conf}(B_m) &= \frac{1}{|B_m|} \sum_{i \in B_m} p(\hat{y}_i), \end{aligned} \tag{5}$$

where $|B_m|$ is the number of samples in bin B_m. Finally, the difference between accuracy and confidence is averaged across all bins to obtain the ECE score:

$$\text{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{\bar{N}} |\text{acc}(B_m) - \text{conf}(B_m)| \tag{6}$$

As another popular metric, Brier score is similar to ECE but conducted at the instance level:

$$\text{Brier} = \frac{1}{\bar{N}} \sum_{i=1}^{\bar{N}} (p(\hat{y}_i) - \mathbb{1}(y_i = \hat{y}_i))^2. \tag{7}$$

Both metrics range from 0 to 1 with lower values indicating better calibration. We take Brier score as the *main* metric, as it is more robust to unbalanced distribution across bins (e.g. instances concentrate to one or two bins).

Settings We conduct experiments on LLaMA2 and Mistral-family models and investigate both pretrained or instruction-tuned variations. We use nucleus sampling to obtain $N = 16$ samples by default for each instance and use temperatures of 0.6 / 1.0 for all pretrained / instruction-tuned models.

Baselines We take the three representative baselines below for comparison:

- *logit w/ Path*: It averages the probabilities of the tokens from the whole path to estimate the confidence of each prediction.
- *logit w/ Answer*: It is similar to *logit w/ Path* but only consider the tokens from the predicted answer span.
- *p(True)*: It asks the LLM itself to classify its prediction as *True* or *False*. Then, it takes the predicted probability of *True* as its confidence. We follow Kadavath et al. (2022) to construct 8-shot demonstrations for prompting pretrained models but directly use instruction for instruction-tuned models.

4.2 Results and Analysis

Main Results Table 1 presents the main results obtained from both benchmarks using Mistral-family models. *p(True)* performs best among the baselines, echoing the findings of Kadavath et al. (2022). However, due to its reliance on prompt design and in-context examples to aid the LLM

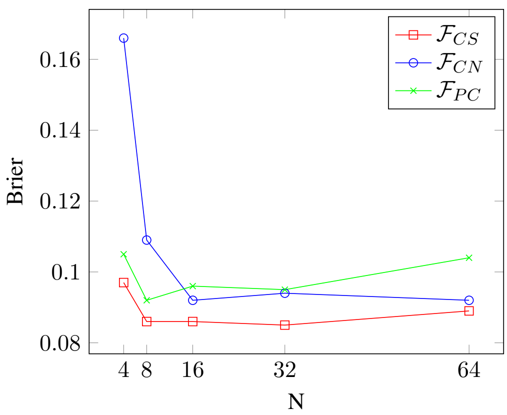

Figure 2: Calibration results on GSM8K when using Mixtral-8×7B-Inst with different N.

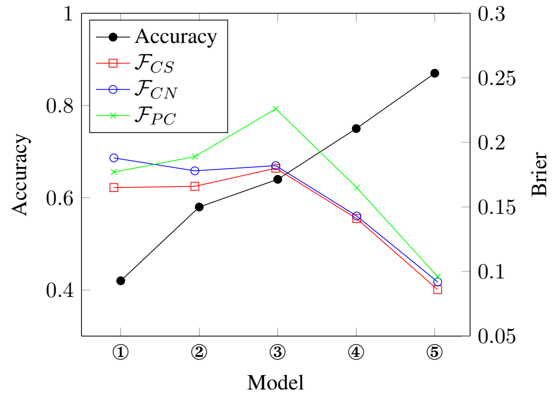

Figure 3: Performance and calibration results on GSM8K using different models below sorted by their performance: ① LLaMA2-7B-Chat, ② LLaMA2-13B-Chat, ③ Mistral-7B-Inst, ④ LLaMA2-70B-Chat, ⑤ Mixtral-8×7B-Inst.

to classify its predictions, it can be challenging to construct effective demonstrations or instructions.

In general, self-consistency-based methods surpass baselines in most cases regarding Brier and ECE, validating the efficacy of employing self-consistency features for estimating model confidence. We also note that baselines can occasionally yield impressive ECE scores (*p(True)* on GSM8K with Mixtral-8×7B). However, we observe that this is attributed to the concentration of most samples in just a few bins (e.g., Figure 1), leading to unreliable measurements. Nevertheless, our approaches still exhibit strong performance in terms of ECE scores across various settings.

Among the self-consistency-based methods, $\mathcal{F}_{CN}$ yields better ECE results on GSM8K, while $\mathcal{F}_{CS}$ achieves the highest Brier score. Conversely, for MathQA, $\mathcal{F}_{CN}$ performs significantly worse than the other two. This is because MathQA is a *multi-choice* task, and thus the cluster number of LLM answers is strictly limited by the provided choices. In conclusion, $\mathcal{F}_{CS}$ demonstrates greater generality across diverse settings, but $\mathcal{F}_{CN}$ and $\mathcal{F}_{PC}$ do offer improved estimation in certain cases.

Finally, we also explore averaging the proposed three metrics as a unified one, denoted as "ALL". Results show that it does not typically yield the best results. However, it serves as a robust choice, demonstrating competitive performance.

Influence of Sample Size N Previous research (Wang et al., 2022) has demonstrated that the sample size N can significantly affect the accuracy of self-consistency. When N increases, the model performance initially continues to improve before stabilizing once N reaches a sufficient level. Therefore, we take Mixtral-8×7B-Inst as a case study to examine the impact of N on calibration.

As illustrated in Figure 2, the Brier scores for all our methods initially decline and then remain constant as N grows. For $\mathcal{F}_{CS}$ and $\mathcal{F}_{PC}$, $N = 8$ is adequate for accurate estimation. In contrast, $\mathcal{F}_{CN}$ requires a larger N, indicating that the cluster number is more susceptible to the randomness of sampling.

Correlation between Performance and Calibration We finally explore the associations between model performance (Accuracy) and calibration. Figure 3 showcases the results on instruction-tuned LLaMA2 and Mistral series models, arranged in ascending order based on their performance. We generally observe a positively correlated trend between calibration (lower the better) and performance (higher the better) among the studied models. This observation indicates that more powerful models also exhibit enhanced calibration, echoing the findings of Kadavath et al. (2022). This phenomenon can be attributed to the fact that when a tested LLM is stronger, it is capable of generating more reasonable and consistent responses, leading to improved calibration.

5 Conclusion

In this paper, we extend the widely-used inference strategy, self-consistency, to the field of calibration. Specifically, we develop three off-the-shelf calibration methods based on self-consistency for math reasoning tasks. Compared to conventional methods (*p(True)* and *logit*), our approaches yield significantly improved ECE and Brier scores on popular GSM8K and MathQA datasets. Future research directions include designing more effective calibration methods, leveraging richer features and employing more strategies (e.g., *temperature scaling* (Guo et al., 2017)) to enhance calibration performance. Our ultimate goal is to construct reliable and honest LLMs with the help of accurate confidence estimation.

Limitations

Our methods are founded on the principle of self-consistency, which relies on sampling multiple times for prediction. This approach, however, needs additional cost for inference, which may not be efficient and eco-friendly. Besides, our current work is limited to mathematical problems and does not explore other types of tasks, such as question-answering. Although it is crucial to extend our methods to encompass other tasks, this is non-trivial due to the inherent difficulty in dividing certain tasks' model predictions into distinct clusters.

Ethics Statement

We focus on ethical AI research and strive to achieve a balance between technological advancements and our ethical responsibilities. This work studies calibration, which aims to enhance the reliability of LLMs. Besides, we conduct experiments only on publicly available datasets, upholding privacy and anonymity rules.

References

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

Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. Mathqa: Towards interpretable math word problem solving with operation-based formalisms. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2357–2367.

Yangyi Chen, Lifan Yuan, Ganqu Cui, Zhiyuan Liu, and Heng Ji. 2023. A close look into the calibration of pre-trained language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1343–1367.

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

Jiahui Geng, Fengyu Cai, Yuxia Wang, Heinz Koeppl, Preslav Nakov, and Iryna Gurevych. 2023. A survey of language model confidence estimation and calibration. *arXiv preprint arXiv:2311.08298*.

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

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Mark Hopkins, Cristian Petrescu-Prahova, Roie Levin, Ronan Le Bras, Alvaro Herrasti, and Vidur Joshi. 2017. Beyond sentential semantic parsing: Tackling the math sat with a cascade of tree transducers. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 795–804.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611.

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

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In

The Eleventh International Conference on Learning Representations.

Jierui Li, Lei Wang, Jipeng Zhang, Yan Wang, Bing Tian Dai, and Dongxiang Zhang. 2019. Modeling intra-relation in math word problems with different functional multi-head attentions. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 6162–6167.

Zhongli Li, Wenxuan Zhang, Chao Yan, Qingyu Zhou, Chao Li, Hongzhi Liu, and Yunbo Cao. 2022. Seeking patterns, not just memorizing procedures: Contrastive learning for solving math word problems. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2486–2496.

Lucie Charlotte Magister, Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. 2023. Teaching small language models to reason. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1773–1781.

Takuya Matsuzaki, Takumi Ito, Hidenao Iwane, Hirokazu Anai, and Noriko H Arai. 2017. Semantic parsing of pre-university math problems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2131–2141.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are nlp models really able to solve simple math word problems? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2080–2094.

Jinghui Qin, Xiaodan Liang, Yining Hong, Jianheng Tang, and Liang Lin. 2021. Neural-symbolic solver for math word problems with auxiliary tasks. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5870–5881.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Kumar Shridhar, Alessandro Stolfo, and Mrinmaya Sachan. 2023. Distilling reasoning capabilities into smaller language models. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 7059–7073.

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

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Qinzhuo Wu, Qi Zhang, Zhongyu Wei, and Xuan-Jing Huang. 2021. Math word problem solving with explicit numerical values. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5859–5869.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, YIFEI LI, Jie Fu, Junxian He, and Bryan Hooi. 2023. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. In *The Twelfth International Conference on Learning Representations*.

A Experiments on TriviaQA

Although this work mainly focuses on math problems, we also verify the effectiveness of our approach on TriviaQA (Joshi et al., 2017), a popular question-answering benchmark. Different from math problems, the outputs of LLMs for this task sometimes lack clear answers for clustering. For instance, Mistral-7B-Inst provides a lengthy response, *"The musical Phantom of the Opera premiered in the US on 10th December 1993..."*, instead of an answer span *"Phantom of the Opera"*, to the question *"Which Lloyd Webber musical premiered in the US on 10th December 1993?"*.

To address this issue, we follow Kuhn et al. to employ natural language inference (NLI) to identify responses with the same answers[2].

Results are shown in Table 2. Our approach significantly surpasses the baselines, particularly when utilizing instruction-tuned models, further demonstrating the effectiveness of our method.

[2]We adopt a publicly accessible model at https://huggingface.co/MoritzLaurer/DeBERTa-v3-large-mnli-fever-anli-ling-wanli.

	Mistral-7B		Mistral-7B-Inst		Mixtral-8×7B		Mixtral-8×7B-Inst	
	ECE ↓	Brier ↓	ECE ↓	Brier ↓	ECE ↓	Brier ↓	ECE ↓	Brier ↓
logit	0.119	0.156	0.385	0.377	0.069	0.116	0.256	0.298
p(True)	0.077	0.162	0.326	0.318	0.099	0.131	0.314	0.316
Self-Consistency								
w/ $\mathcal{F}_{CN}$	0.044	0.133	**0.054**	**0.173**	0.077	0.119	0.068	**0.115**
w/ $\mathcal{F}_{CS}$	**0.043**	**0.132**	0.072	0.186	**0.039**	**0.107**	**0.039**	0.116
w/ $\mathcal{F}_{PC}$	0.071	0.137	0.109	0.186	0.102	0.129	0.121	0.126

Table 2: Test results on TriviaQA.