Assignment 2 for NLP@THU

> Prompt tuning, delta tuning, and finetuning.
>
> DDL: May 14, 2024

Dataset

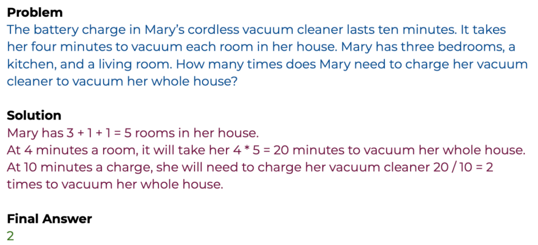
For this assignment, we will engage in training and testing MiniCPM across two pivotal tasks that have emerged as benchmarks in recent advancements within the field of NLP:

1. **GSM8K Dataset** [1]: This dataset is comprised of 8,000 mathematical word problems provided by OpenAI, designed to evaluate a model's numerical and math problem-solving capabilities.
2. **MMLU (Massive Multi-task Language Understanding) Dataset** [2]: Encompassing 57 varied categories of multiple-choice questions, this dataset serves to assess a model's breadth of world knowledge.

These datasets not only represent distinct types of tasks but also embody two prevalent testing methodologies for contemporary LLMs:

1. **Generative Task**: Within the context of the GSM8K dataset, the model is prompted with a question to which it must generate a free-form explanation or rationale, culminating in a final answer. The correctness of this answer is subsequently verified. In certain generative scenarios, we also examine how closely the model-generated text aligns with a reference or "golden" paragraph in terms of similarity.
2. **Classification Task**: The MMLU dataset, on the other hand, consists of questions each accompanied by four possible answers. The model's task is to select the most appropriate answer from the given options. (Also, you can use generative format in MMLU to combine techniques like CoT)

The accompanying examples from each dataset illustrate the nature of tasks MiniCPM is expected to perform. MiniCPM has demonstrated foundational capabilities on these tasks, setting a benchmark for subsequent evaluations.

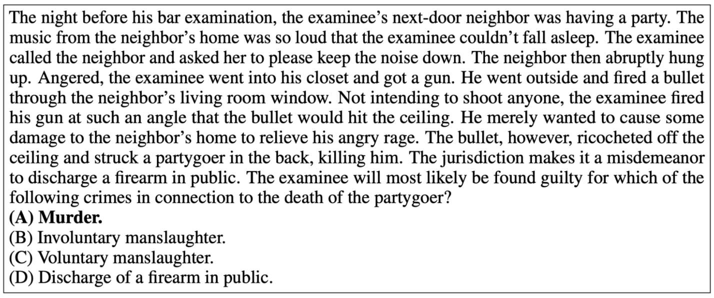


Problem
The battery charge in Mary's cordless vacuum cleaner lasts ten minutes. It takes her four minutes to vacuum each room in her house. Mary has three bedrooms, a kitchen, and a living room. How many times does Mary need to charge her vacuum cleaner to vacuum her whole house?

Solution
Mary has 3 + 1 + 1 = 5 rooms in her house.
At 4 minutes a room, it will take her 4 * 5 = 20 minutes to vacuum her whole house.
At 10 minutes a charge, she will need to charge her vacuum cleaner 20 / 10 = 2 times to vacuum her whole house.

Final Answer
2

GSM8K example



The night before his bar examination, the examinee's next-door neighbor was having a party. The music from the neighbor's home was so loud that the examinee couldn't fall asleep. The examinee called the neighbor and asked her to please keep the noise down. The neighbor then abruptly hung up. Angered, the examinee went into his closet and got a gun. He went outside and fired a bullet through the neighbor's living room window. Not intending to shoot anyone, the examinee fired his gun at such an angle that the bullet would hit the ceiling. He merely wanted to cause some damage to the neighbor's home to relieve his angry rage. The bullet, however, ricocheted off the ceiling and struck a partygoer in the back, killing him. The jurisdiction makes it a misdemeanor to discharge a firearm in public. The examinee will most likely be found guilty for which of the following crimes in connection to the death of the partygoer?
(A) Murder.
(B) Involuntary manslaughter.
(C) Voluntary manslaughter.
(D) Discharge of a firearm in public.

MMLU example

By refining the presentation and clarity of these assignment instructions, the goal is to ensure a more coherent and accessible understanding of the task requirements for students engaged in English NLP coursework. MiniCPM have shown the base performance on these two tasks.

Model	Average	English Average	Chinese Average	C-Eval	CMMLU	MMLU	HumanEval	MBPP	GSM8K	MATH	BBH	ARC-E	ARC-C	HellaSwag
Llama2-7B	35.40	36.21	31.77	32.42	31.11	44.32	12.2	27.17	13.57	1.8	33.23	75.25	42.75	75.62*
Qwen-7B	49.46	47.19	59.66	58.96	60.35	57.65	17.07	42.15	41.24	5.34	37.75	83.42	64.76	75.32*
Deepseek-7B	39.96	39.15	43.64	42.82	44.45	47.82	20.12	41.45	15.85	1.53	33.38	74.58*	42.15*	75.45*
Mistral-7B	48.97	49.96	44.54	46.12	42.96	62.69	27.44	45.2	33.13	5.0	41.06	83.92	70.73	80.43*
Llama2-13B	41.48	42.44	37.19	37.32	37.06	54.71	17.07	32.55	21.15	2.25	37.92	78.87*	58.19	79.23*
MPT-30B	38.17	39.82	30.72	29.34	32.09	46.56	21.95	35.36	10.31	1.56	38.22	78.66*	46.08*	79.72*
Falcon-40B	43.62	44.21	40.93	40.29	41.57	53.53	24.39	36.53	22.44	1.92	36.24	81.94*	57.68	83.26*
MiniCPM-2B	52.33	52.6	51.1	51.13	51.07	**53.46**	50.00	47.31	**53.83**	10.24	36.87	85.44	68.00	68.25

Common techniques

To enhance the performance on specific tasks, three prevalent techniques are commonly utilized:

1. **Prompt Tuning** [3]: This technique involves designing tailored prompts that effectively guide the model in addressing a particular task. These prompts can encapsulate methods, prior knowledge relevant to the task, or encourage the model to articulate its reasoning process step-by-step. Additionally, providing in-context examples can instruct the model on the expected approach to the task.
2. **Fine-tuning**: When a substantial corpus of task-specific examples is available, fine-tuning allows the model to assimilate new knowledge and improve its performance. This process involves adjusting the model's parameters based on the examples provided.
3. **Delta-tuning** [4]: In scenarios where computational resources are limited, making it impractical to fine-tune all parameters of a model, delta tuning offers a viable alternative. This method focuses on modifying a small fraction of the model's parameters or introducing a modest number of new parameters. Although delta tuning generally yields inferior results compared to fine-tuning when abundant training examples are present, it requires significantly less computational power—often just 1% (or even as little as 0.1%) of what would otherwise be needed.

Your Task (20 points in total)

Your task in this assignment involves evaluating MiniCPM's performance and implementing strategies to enhance it.

1. Apply the model on one of the datasets. Evaluating both datasets could get additional points.
2. Document both the baseline performance and the improved performance achieved through one of the aforementioned techniques. Employing multiple techniques may lead to extra points.
3. Analyze the experiment and share insights gleaned from the assignment.

The scoring for this assignment is as follows:

- **Baseline Performance (5 Points)**: Your baseline results should align closely with those reported in the MiniCPM official repository. A deviation of up to 5% is acceptable.
- **Enhanced Performance (10-12 Points)**: The points awarded for improved performance vary based on the technique used, reflecting the differing levels of difficulty:
    - Prompt Tuning (using Chain-of-Thought only): 10 points
    - Fine-tuning: 11 points
    - Delta tuning: 12 points
    - Implementing more than one technique guarantees at least 12 points.
- **Interesting Experiment (3 Points)**: Conduct one or two innovative experiments of your choice and discuss your findings. Additional experiments may earn extra points.
- **Answer the following question(2point)**: What do you think that is most important for this task?

Insight examples

To foster creativity and guide the design of your experiments within this assignment, consider exploring the following open-ended questions. These inquiries are meant to challenge conventional approaches and inspire innovative experimentation with MiniCPM:

1. **Calibration** [5,6]: Investigate whether the model can accurately express its uncertainty. For instance, if the model assigns a 90% probability to option "A," does this reflect an actual accuracy rate of approximately 90%? How does the model perform when its confidence level is around 40%?
2. **Honesty** [7]: Assess the model's ability to acknowledge its limitations by expressing "I don't know" in scenarios where it struggles with the task. If the model were to limit its attempts to only 50% of the tasks where it feels most confident, would we observe a 50% performance improvement?
3. **Sycophantic Behavior** [8]: Explore the model's reaction to suggestions that might influence its decision-making, such as being told "I think maybe C is wrong/correct." How does this affect the model's choice?
4. **Scaling** [9]: In the context of model training, consider the possibility of enhancing performance by expanding the dataset (e.g., incorporating Metamath [10] data for the GSM8K task). Does access to more training data correlate with improved model performance?
5. **CoreSet** [11]: Some tasks may prove too challenging for smaller LLMs. Investigate whether using a subset of the training data can achieve similar, or even superior, results. This approach focuses on identifying the core instances that are most informative for the model's learning process.
6. **Consistency** [12]: Examine the effect of having the LLM answer a question multiple times (using a temperature setting) and then deciding based on a majority vote. Could this method lead to enhanced performance, and if so, why?
7. **Automatic Prompt Search** [13]: Previous research has shown that certain prompts, such as "Let us think step by step," encourage the model to engage in a chain-of-thought process, thereby improving performance. DeepMind researchers discovered that the prompt "Let us take a deep breath and answer" yielded even better results. Is it possible to automate the search for an optimal prompt that significantly boosts performance?
8. **In-Context Examples**: If providing relevant examples is crucial, consider whether including the most similar training instance in the context when addressing a new query can enhance the model's accuracy.
9. **External Tools** [14]: Evaluate the impact of equipping the LLM with external tools, such as a calculator or

Python interpreter for the GSM8K task, or access to a search engine for the MMLU task. How do these additions affect the model's performance?

10. **Your Own Insights**: ...

Note

- **You should submit the code and the report, and rename it into your student id**. Don't submit the model training checkpoint. Report can be either Chinese or English:
    - MMLU: https://huggingface.co/datasets/cais/mmlu
    - GSM8K: https://huggingface.co/datasets/gsm8k
- The maximum achievable score for this assignment is 20 points, with a potential for up to 22 points for exceptional work.
- If computational resources are a constraint, you may opt to test a subset of the dataset.

Reference

[1] Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., & Schulman, J. (2021). Training Verifiers to Solve Math Word Problems. *ArXiv, abs/2110.14168*.
[2] Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D.X., & Steinhardt, J. (2020). Measuring Massive Multitask Language Understanding. *ArXiv, abs/2009.03300*.
[3] Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., & Neubig, G. (2021). Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. *ACM Computing Surveys, 55*, 1 - 35.
[4] Ding, N., Qin, Y., Yang, G., Wei, F., Yang, Z., Su, Y., Hu, S., Chen, Y., Chan, C., Chen, W., Yi, J., Zhao, W., Wang, X., Liu, Z., Zheng, H., Chen, J., Liu, Y., Tang, J., Li, J., & Sun, M. (2023). Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence, 5*, 220-235.
[5] Lin, S.C., Hilton, J., & Evans, O. (2022). Teaching Models to Express Their Uncertainty in Words. *Trans. Mach. Learn. Res., 2022*.
[6] Kadavath, S., Conerly, T., Askell, A., Henighan, T., Drain, D., Perez, E., Schiefer, N., Dodds, Z., DasSarma, N., Tran-Johnson, E., Johnston, S., El-Showk, S., Jones, A., Elhage, N., Hume, T., Chen, A., Bai, Y., Bowman, S., Fort, S., Ganguli, D., Hernandez, D., Jacobson, J., Kernion, J., Kravec, S., Lovitt, L., Ndousse, K., Olsson, C., Ringer, S., Amodei, D., Brown, T.B., Clark, J., Joseph, N., Mann, B., McCandlish, S., Olah, C., & Kaplan, J. (2022). Language Models (Mostly) Know What They Know. *ArXiv, abs/2207.05221*.
[7] Yang, Y., Chern, E., Qiu, X., Neubig, G., & Liu, P. (2023). Alignment for Honesty. *ArXiv, abs/2312.07000*.
[8] Sharma, M., Tong, M., Korbak, T., Duvenaud, D.K., Askell, A., Bowman, S.R., Cheng, N., Durmus, E., Hatfield-Dodds, Z., Johnston, S., Kravec, S., Maxwell, T., McCandlish, S., Ndousse, K., Rausch, O., Schiefer, N., Yan, D., Zhang, M., & Perez, E. (2023). Towards Understanding Sycophancy in Language Models. *ArXiv, abs/2310.13548*.
[9] Yuan, Z., Yuan, H., Li, C., Dong, G., Tan, C., & Zhou, C. (2023). Scaling Relationship on Learning Mathematical Reasoning with Large Language Models. *ArXiv, abs/2308.01825*.
[10] Yu, L.L., Jiang, W., Shi, H., Yu, J., Liu, Z., Zhang, Y., Kwok, J.T., Li, Z., Weller, A., & Liu, W. (2023). MetaMath:

Bootstrap Your Own Mathematical Questions for Large Language Models. *ArXiv, abs/2309.12284*.
[11] Sener, O., & Savarese, S. (2017). Active Learning for Convolutional Neural Networks: A Core-Set Approach. *arXiv: Machine Learning*.
[12] Wang, X., Wei, J., Schuurmans, D., Le, Q., Chi, E.H., & Zhou, D. (2022). Self-Consistency Improves Chain of Thought Reasoning in Language Models. *ArXiv, abs/2203.11171*.
[13] Ye, Q., Axmed, M., Pryzant, R., & Khani, F. (2023). Prompt Engineering a Prompt Engineer. *ArXiv, abs/2311.05661*.
[14] Chen, W., Ma, X., Wang, X., & Cohen, W.W. (2022). Program of Thoughts Prompting: Disentangling Computation from Reasoning for Numerical Reasoning Tasks. *ArXiv, abs/2211.12588*.