Scientific Reasoning: Assessment of Multimodal Generative LLMs

Florian Dreyer (3667877) [*1] **Ekaterina Kolos (3688474)** [*1] **Daria Matiash (3668740)** [*1]

Abstract

This project assesses the capabilities of pre-trained multimodal LLMs to perform scientific reasoning tasks on multimodal Question Answering (QA) data. We further explore how Prefix Tuning and Low-Rank Adaptation (LoRA) can improve performance of smaller LLMs. We then attempt to distill knowledge to a smaller LLM using Prefix Tuning and LoRA.

1. Introduction

Scientific machine reasoning is the application of AI techniques to simulate scientific reasoning processes, such as data interpretation, hypothesis generation, and causal inference.

Recent developments in foundation models, such as o1/o3 from OpenAI (OpenAI, 2024) or R1 from DeepSeek (DeepSeek-AI et al., 2025), have shown remarkable performance in challenges that require contextual awareness and reasoning. However, these models are often resource consuming, which limits their scalability and accessibility for broader applications.

This leads us to our research questions:

RQ1 How can large multimodal LLMs deal with multimodal scientific reasoning?

RQ2 How do Prefix Tuning and LoRA affect the reasoning capabilities of smaller pre-trained models?

RQ3 How good is knowledge distillation with adapter methods compared to training on manually annotated data?

We first want to evaluate how six large front-tier LLMs perform on the SCIENCEQA dataset (Lu et al., 2022) using several metrics. Following that we evaluate how well Prefix Tuning (Li & Liang, 2021) and LoRA (Hu et al., 2021) perform for fine-tuning two smaller LLMs on this dataset. Last, we perform knowledge distillation using Prefix Tuning and LoRA and compare how the distilled models compare to the previously fine-tuned models.

[*]Equal contribution [1]University of Stuttgart. Correspondence to: Florian Dreyer <st182762@stud.uni-stuttgart.de>, Ekaterina Kolos <st186032@stud.uni-stuttgart.de>, Daria Matiash <st185745@stud.uni-stuttgart.de>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

2. Background and Related Work

Prefix Tuning Prefix Tuning, as introduced in (Li & Liang, 2021), is a parameter-efficient fine-tuning method that freezes the pre-trained model's parameters and trains a small, learnable "prefix" as illustrated in figure 1 that guides the model during task-specific inference. This allows the model to adapt to new tasks while preserving its general pre-trained knowledge.

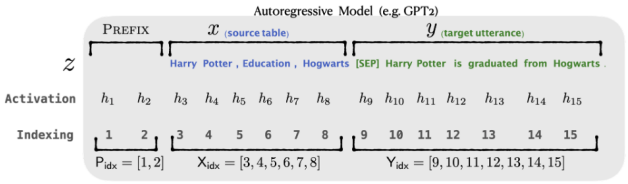

Figure 1. Intuition behind Prefix Tuning. Source: (Li & Liang, 2021)

By significantly reducing the number of trainable parameters, Prefix Tuning is especially advantageous in resource-constrained scenarios or when deploying models for multiple tasks. Studies have shown that it achieves performance comparable to full fine-tuning in many applications, such as natural language generation and classification.

LoRA Low-Rank Adaptation (LoRA) (Hu et al., 2021) is a parameter-efficient fine-tuning technique designed to adapt pre-trained models to downstream tasks with minimal computational overhead. LoRA introduces trainable low-rank matrices into the model's attention layers while keeping the original weights frozen. These matrices encode task-specific knowledge, allowing the model to be fine-tuned with a significantly smaller number of parameters compared to traditional fine-tuning.

LoRA is particularly effective in scenarios where memory efficiency is critical, as it avoids modifying or storing the full set of model weights. Studies have shown that LoRA achieves competitive performance on various NLP tasks, such as machine translation and text classification (Mao et al., 2025).

Knowledge distillation (KD) allows to obtain smaller models capable of successfully following the behavior of larger teacher models. This is particularly useful for privacy reasons (mobile AI apps) and when access to large models in the cloud is not guaranteed (e.g. in cars). The teacher model can be used to intelligently select examples on which the student model is trained (dataset distillation) (Yu et al., 2023), or provide negative samples to show the student what incorrect answers or reasoning paths it should avoid to improve task accuracy (Li et al., 2024). Training small models on a chain-of-thought reasoning path of a larger model was also shown to be a way to obtain a small student model replicating reasoning capabilities of teacher on downstream tasks (Magister et al., 2022). In our project, we follow a similar approach and perform *response-based knowledge distillation* (Gou et al., 2021), learning to mimic the output of the teacher.

Models We experiment with the following foundation models: `Pixtral-12b-2409`, `LLaVA-1.5-7b-hf`, `Gemini 1.5 Flash`, `Gemini 1.5 Flash 8B`, `GPT 4`, `GPT 4o mini`, `Qwen2-VL-2B-Instruct`, `paligemma-3b-pt-224`.

3. Dataset

This study is based on the SCIENCEQA dataset (Lu et al., 2022). The SCIENCEQA dataset is a benchmark for multimodal reasoning in science, consisting of more than 21,000 questions across topics like Natural science or Social science. Each question includes textual prompts, optionally visual aids (e.g., diagrams, charts), answer options, detailed explanations, and lecture context. The dataset spans various difficulty levels, enabling evaluation of models on both basic and advanced scientific reasoning, reaching from elementary to high school level questions. The dataset contains some notable irregularities. For some datapoints, the image data is missing (sometimes when it is required to solve the task, sometimes when the answer can be deduced from text alone). Apart from the image field, missing values can occur in lecture and solution fields (about 9% and 15% respectively). As about 50% of the datapoints are missing an image, we decide to process both text-only and text+image datapoints similarly with multimodal models, attaching the image to the prompt if it was present in the data. When the image was not available, we generated a blank empty image as a placeholder.

4. Methodology

In the following we present the methodology we used to examine the three research questions (RQ).

RQ1 We benchmark six front-tier LLMs using accuracy and the average of five text similarity metrics introduced in the metrics section 5. To investigate how additional information and the correct solution in the prompt influence the performance, we use four different prompt settings:

1. **question - choices - hint - image - task**
2. **question - choices - hint - image - task + lecture**
3. **question - choices - hint - image - task + lecture + solution**
4. **question - choices - hint - image - task + solution**

Each model is benchmarked on all four settings using the SCIENCEQA validation data. We use the validation split instead of the test split since we later use the results to choose a champion teacher model. This decision can't be made using the test data since it's utilized to evaluate knowledge distillation performance.

RQ2 For the comparison of Prefix Tuning and LoRA we fine-tune two smaller multimodal LLMs using the two techniques separately. As the label to train on we choose the solution to enable the adapters to learn more of the reasoning. We then compare the performance of the four fine-tuned models to the zero-shot performance of the two base models on SCIENCEQA test data.

RQ3 We further investigate the impact of knowledge distillation on the performance of adapter tuning. Would learning from the outputs of a champion teacher LLM drop the performance significantly and consistently, compared to learning from the original data? For this, we obtain outputs of the champion teacher model on the train partition of SCIENCEQA. We train 2 different adapters on 2 different students on this data and on the original train partition of SCIENCEQA, and compare the text similarity results in section 6.3.

We compare the performance of the eight resulting models on the SCIENCEQA test data.

5. Metrics

We evaluate model's performance with QA accuracy. The reasoning steps were evaluated with semantic similarity metrics, adopted from machine translation, such as BLEU, METEOR, ROUGE, and cosine similarity.

5.1. Multiple choice Evaluation

Owing to the simplicity of the test format, only accuracy score is computed, following original evaluation strategy by (Lu et al., 2022). To extract the answers from the output we prompt the models to output in JSON format with "answer" and "solution" as keys.

5.2. Answer Reasoning Evaluation

Due to the peculiarity of scientific texts and approaches to the evaluation of automatically generated texts, the following evaluation approaches were chosen.

BLEU BLEU-score (Papineni et al., 2002) can measure how closely the model's generated solutions aligns with the human-authored explanation, calculating modified *n*-gram precisions adjusted by brevity penalty. BLEU-1 measures if the model uses the right scientific terms or key words (e.g., *"photosynthesis"*, *"temperature"*). However, it ignores word order, so it can't evaluate reasoning or explanation quality. That is why BLEU-4 score is also computed to capture both vocabulary usage and phrase structures. By computing both BLEU-1 and BLEU-4, we balance term accuracy with linguistic structure, ensuring a more reliable evaluation of the model's explanatory capabilities in scientific reasoning.

ROUGE ROUGE is a set of metrics introduced in (Lin, 2004), again, to estimate the quality of a generated hypothesis compared to one or more golden references. ROUGE-1 and ROUGE-2, like BLEU-*n*, count overlaps of *n*-grams between the hypothesis and the reference, calculating not only the precision, but also the recall and F1-score. Following evaluation strategy in (Lu et al., 2022), we adopt ROUGE-L score (Lin, 2004), that calculates the longest common subsequence between the hypothesis and the reference. As opposed to ROUGE-1 and ROUGE-2, this score would assign a higher value to e.g. *"the theory which Einstein proposed"* than to *"Einstein proposed the theory"* for a reference like *"the theory that Einstein proposed"*. This score, however, would not capture synonyms by default, which is why we do not rely on this score alone. [1]

METEOR METEOR (Banerjee & Lavie, 2005) provides a more fine-grained and linguistically informed approach to evaluating text similarity, balancing precision and recall, and encouraging correct alignment of semantically meaningful components through fragmentation penalty. Later versions of METEOR introduce stemming, synonym matching, as well as fine-grained weights for individual languages for an even better agreement with human judgments.

Cosine similarity with Sentence Transformers `all-MiniLM-L6-v2` was utilized in Sentence Transformers library in our evaluation strategy. Unlike `SciBERT`, which is specialized for scientific text and could serve as a sentence embedder, `all-MiniLM-L6-v2` was pre-trained on diverse datasets, allowing it to better handle variations in phrasing and style found in the Science QA Dataset. Its broader linguistic adaptability ensures that evaluation considers both semantic accuracy and logical flow, which is crucial for assessing reasoning across different scientific domains.

Overall score To make the performance of the models to be easier for comparison, we computed an overall score of the textual similarity metrics. In order to obtain a numeric description of the model's performance for these text similarity metrics, we computed the average of them (having normalized cosine similarity to the range [0;1]).

6. Experiments

In the following we present the experiments to answer the three RQs.

6.1. Benchmarking large LLMs

As described in 4 we benchmarked six large LLMs on the validation part of the dataset. Our experiments resulted in the following results for accuracy:

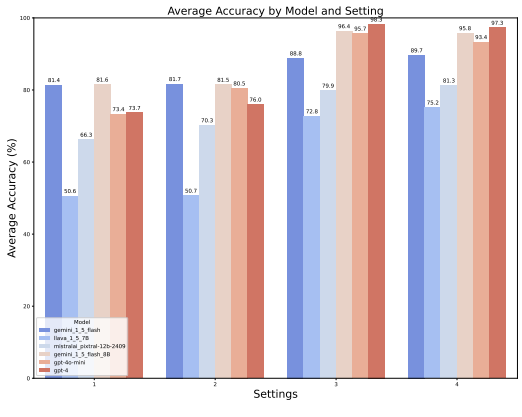

Figure 2. Accuracy scores in answer generation by LLMs. Benchmarking.

[1] We performed our evaluation following the scripts proposed by the authors of SCIENCEQA for consistent comparison. Upon more careful investigation done at the stage of writing this report, we found out that the package used for ROUGE was https://pypi.org/project/rouge/, which does not calculate the score consistently to the algorithm proposed by (Lin, 2004). A better implementation, that is more accurate to it, would be this one: https://pypi.org/project/rouge-score/.

For the overall score as introduced in the 5 the experiments resulted in the following scores:

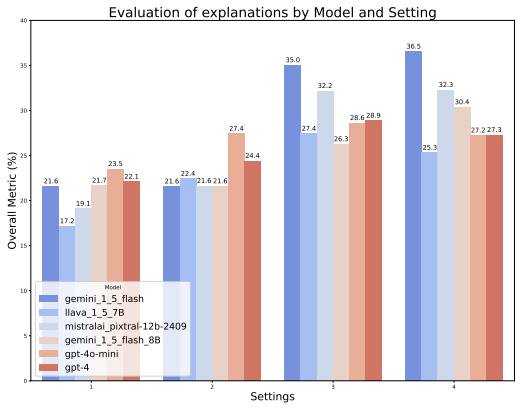

Figure 3. Overall scores in reasoning by LLMs. Benchmarking.

For the individual textual similarity metrics we got the following results:

model	s.	bl-1	bl-4	r.	m.	cos.	overall
GEMINI	1	0.04	0.01	0.28	0.14	0.80	21.59
	2	0.04	0.01	0.28	0.14	0.80	21.58
	3	0.12	**0.12**	**0.52**	**0.30**	**0.84**	**35.03**
	4	0.14	**0.14**	**0.53**	**0.32**	**0.85**	**36.55**
LLAVA	1	0.03	0.01	0.24	0.09	0.74	17.16
	2	0.08	**0.05**	**0.33**	0.06	0.80	22.43
	3	0.06	0.06	0.41	0.20	0.82	27.43
	4	0.05	0.05	0.41	0.13	0.81	25.32
PIXTRAL	1	0.05	0.01	0.23	0.11	0.78	19.11
	2	0.06	0.02	0.27	0.14	0.79	21.64
	3	0.12	0.11	0.46	0.27	0.83	32.17
	4	0.11	0.11	0.47	0.27	0.83	32.29
GEMINI 8B	1	0.05	0.01	0.28	0.14	**0.81**	21.68
	2	0.05	0.01	0.27	0.14	0.81	21.59
	3	0.06	0.04	0.37	0.19	0.82	26.26
	4	0.08	0.08	0.44	0.24	0.84	30.37
GPT-4O -MINI	1	**0.14**	**0.03**	0.28	**0.17**	0.78	**23.51**
	2	**0.17**	0.04	0.31	**0.21**	**0.82**	**27.43**
	3	**0.18**	0.05	0.33	0.22	0.82	28.61
	4	**0.17**	0.04	0.31	0.21	0.82	27.24
GPT-4	1	0.11	0.02	**0.30**	0.16	0.76	22.09
	2	0.13	0.04	0.32	0.18	0.78	24.41
	3	0.16	0.06	0.35	0.23	0.82	28.88
	4	0.14	0.04	0.33	0.21	0.82	27.29

Table 1. Metrics on scientific reasoning by LLMs. *s.* - setting, *bl-1* - BLEU-1, *bl-4* - BLEU-4, *r.* - ROUGE-L, *m.* - METEOR, *cos.* - cosine similarity, *overall* - overall metric in %.

Accuracy The highest performance was observed in settings that included solutions, indicating that the models were generally capable of extracting relevant information effectively. We noticed that setting 3 including the lecture did yield worse results for most models compared to the lecture-free setting 4. `GPT-4` and `Gemini-1.5-flash-8B` consistently received highest ratings in scenarios in which lecture and/or solution information was included. The `Gemini` family of models and `Pixtral-12b-2409` model showed better robustness in many scenarios and came in second overall. Notably, `Gemini` models achieved an average accuracy advantage of 8% over `GPT-family` models in the "pure" task context (without lecture or solution information). The `LLaVA-1.5-7b-hf`, on the other hand, demonstrated the lowest average accuracy scores, particularly in environments without lecture and solution data. For setting 1, which was considered as the most important for further experiments, `Gemini` models performed with highest accuracy scores.

BLEU-1 Across settings, the highest BLEU-1 scores were mostly observed in settings 3 and 4 (those incorporating solution or lecture+solution information), indicating that models performed better with richer contextual data. The `GPT-4o-mini` model demonstrated superior performance, surpassing `GPT-4`, in achieving the highest results. Lower scores in settings 1 and 2 suggest that models, particularly `LLaVA-1.5-7b-hf`, may have had trouble in correctly aligning their outputs with reference words when given less auxiliary information.

BLEU-4 `GPT-4` models had higher scores in comparison to other models, which represents a better capability of information extraction. `Gemini`, at the same time, gained the highest score in the setting without solutions, which indicates the ability of extracting knowledge from massive textual data.

METEOR METEOR scores reveal how well the models capture fluency, grammar, and word-level alignment with the references. Similar to cosine similarity, `GPT-4o-mini` leads with the highest METEOR scores. Notably, `Pixtral-12b-2409` achieves relatively high METEOR scores in settings 3 and 4, highlighting its ability to produce fluent outputs. When concrete tasks without any additional helpful material are given to the models as an input, `GPT-4o-mini` model shows a capability of generating concise, fluent answers with correct relevant terminology and explanations.

ROUGE The highest ROUGE scores were observed in setting 3, where both lectures and solutions were available, highlighting the importance of comprehensive input for producing informative responses. Setting 1 showed lower

ROUGE scores across almost all models, reflecting limited informativeness when models were provided with minimal context. GPT family of models had highest ROUGE-scores on settings without solutions. Outputs of GPT-4 without knowledge of the correct answer had higher ROUGE score than GPT-4o-mini given solutions in the input.

Cosine similarity Cosine similarity highlights the semantic alignment between the model outputs and the correct answers. Gemini-1.5-flash consistently achieved the highest similarity scores, particularly in settings 3 and 4, indicating strong alignment with the reference answers. GPT-4 variants maintain steady scores across all settings, demonstrating robustness, while Pixtral-12b-2409 and LLaVA-1.5-7b-hf show more variability, suggesting sensitivity to specific settings. Overall, cosine similarity shows that Gemini-1.5-flash excels in semantic understanding across the dataset.

Accuracy and Overall score The overall results indicate that the GPT and Gemini model families demonstrate exceptional ability in extracting accurate answers. Notably, the Gemini models exhibited superior performance on datasets lacking relevant lecture information in most of the metrics. The second-best performance on "pure" (without solutions and/or lectures) datasets was achieved by GPT-4o-mini, with only a slight difference in overall score.

Summary Overall, the findings demonstrate how important enriched contextual data, such as lectures and solutions, in enhancing model performance across all metrics. The GPT-4o-mini model outperformed GPT-4 in multiple instances, indicating the efficacy of smaller, more targeted designs, even if GPT-4 continuously received excellent accuracy and scores for reasoning. While LLaVA-1.5-7b-hf model failed without extra input, highlighting its dependence on extensive contextual knowledge to function well, the Gemini family of models demonstrated remarkable robustness and semantic alignment, particularly in enriched situations. Lecture information doesn't really affect models' performance, it can even decrease the ratio of correct answers in some cases.

6.2. Prefix Tuning and LoRA

For training the Prefix Tuning adapters we used 20 epochs for Prefix Tuning as derived from (Li & Liang, 2021) and limited by computing resources. For training the LoRA adapter we used 10 epochs as the loss flattened after more epochs in first experiments. After training Qwen and texttt PaliGemma using both Prefix Tuning and LoRA we benchmarked the resulting models on the SCIENCEQA test data. We were not able to measure the accuracy because all four models were not able to output valid answers. For the overall score of the text similarity the experiments resulted in the following scores:

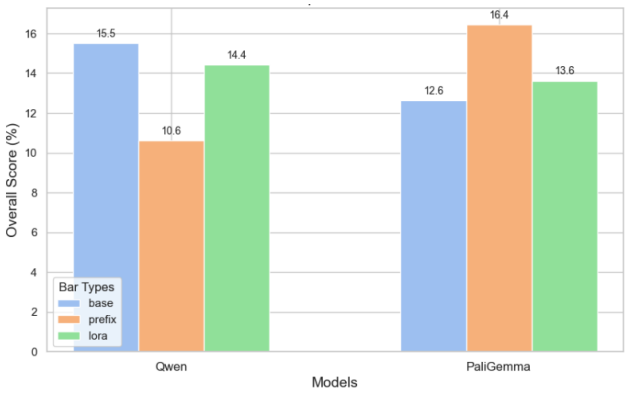

Figure 4. Overall score in reasoning by LLMs in base, after Prefix-tuning, and LoRA Adapter-tuning.

Overall we cannot see a consistent trend, for Qwen, the base model outperformed both adapters while for PaliGemma Prefix Tuning resulted in the best overall score, while the base model performed the worst. It was noticed that the more epochs of efficient parameter tuning was done, the shorter the text became that was generated by both of the models.

6.3. Knowledge Distillation

For training we use the same number of epochs as for the previous experiments. For evaluating the perfomance of Knowledge Distillation compared to normal fine-tuning using the adapter methods we trained both Qwen and Paligemma using Prefix Tuning and LoRA separately. All eight trained models were not able to output valid answers so we will not compare the accuracy. For the overall score of the text similarity the experiments resulted in the following scores:

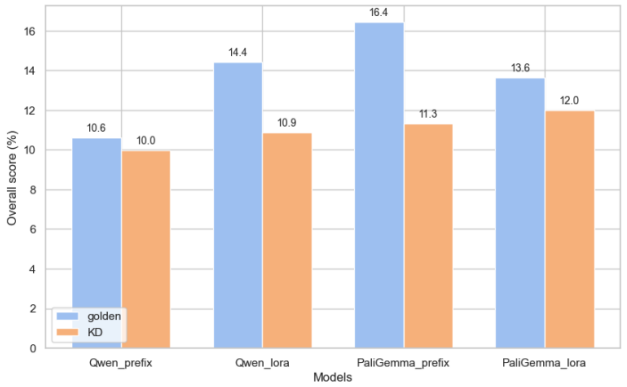

Figure 5. Overall score in reasoning by LLMs.

Overall we can see a clear outperformance of Knowledge Distillation by the fine-tuned models. We can also notice that Knowledge Distillation performs quite similar for all four models while the fine-tuning approach has higher variance in performance. We noticed that the LoRA approach outperforms Prefix Tuning for Knowledge Distillation for both models. This is not the case for the fine-tuned models.

7. Error Analysis

Output format We noticed that even front-tier SOTA models could sometimes deviate from the expected output format; this error type was much more common in smaller models, despite attempts of controlled decoding. Furthermore, a fine-tuned version of `Paligemma` would consistently produce outputs with the *answer* only, not giving any *solution*. For example, for an input asking to output JSON with solution: "*Question: Which of the following organisms is the primary consumer in this food web? ⟨...⟩ Choices: ['copepod', 'black crappie', 'bacteria'] ⟨...⟩ Instruction: Please output the answer in JSON style with an answer and a solution field*" the finetuned `Paligemma` would only output "*The answer is A.*". Qwen models would sometimes produce Chinese utterances instead of expected English generations.

Models would sometimes generate long sequences going beyond a stop token, degenerating into completely irrelevant text or even code. The `Pixtral` model could finish answering the given task, and then generate a new $\langle s \rangle [INST]$ token, after which a new hallucinated task would follow, which the model would later answer. Finally, standardizing the answer, which, given 2-5 choices could be a number, a letter, or a string, proved rather difficult, with some 500 answers left unmapped in each experiment.

Sparse outputs Adapter training surprisingly made the outputs completely degenerate into quasi-empty strings, containing only spaces, functional words, numbers, or rare irrelevant or foreign tokens. We speculate that this results from an overly strict feedback that the models got every time they generated plausible, yet not exactly the same correct explanations.

Output quality Finally, the large LLMs from which the teacher model was selected could not generate perfectly adequate outputs either, being overly vague in explanations or, on the opposite, bringing up irrelevant or hallucinated details. For example, for a task question: "*What's the difference between weather and climate?*" one of the generations was: "*Climate is the pattern of weather in a certain place. It got down to 3°C in Athens, Greece, last night!*", which included irrelevant hallucinated information about the weather in Athens on a specific date.

8. Conclusion

In conclusion, while big LLMs demonstrate strong capabilities in multimodal scientific question answering and benefit from extracting information from available solutions, their performance in reasoning from lectures often falls short or even declines. On the other hand, small foundation models like `Qwen2-VL-2B-Instruct` and `paligemma-3b-pt-224` show limited effectiveness in scientific reasoning tasks, both in zero-shot settings and when fine-tuned with adapter-based methods. Furthermore, inconsistencies in evaluation metrics (zero-shot, Prefix Tuning, LoRA) across these models highlight challenges in establishing reliable performance benchmarks. The poor performance of adapter tuning may be attributed to the current loss function design, suggesting the need for refinement in optimization strategies. Lastly, knowledge distillation underperforms when compared to directly training on a curated, high-quality dataset, emphasizing the importance of data quality in achieving robust model performance.
It was noticed that all four models had more similar performance with knowledge distillation than with fine-tuning. Another oberservation is that LoRA outperforms Prefix Tuning on both models for knowledge distillation. [2]

9. Future work

While we observed that learning from teachers' outputs leads to lower performance compared to learning from human-curated solutions, the precise impact remains to be measured, since we were not able to derive accuracy for the student models. If the teacher's performance is estimated at 80% of human performance, would models trained on the teacher's outputs achieve 80% of the performance of those trained on golden data, or could the student's ability to learn from noisy data mitigate this effect? One could also run experiments with different Knowledge Distillation techniques like the ones introduced in Background (2) or (Gou et al., 2021).

Another direction for future work could be trying out other adapter architectures, starting from larger student models (which would require more computational resources) but probably provide better results, and learning with a less strict loss function.
We could additionally define a multi-head setup for fine-tuning, with part of the model being responsible for explanation generation and another one for answer prediction.

[2] All our work is available on our GitHub repository: https://github.com/katja-kolos/foundation_models

References

Banerjee, S. and Lavie, A. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pp. 65–72, 2005.

DeepSeek-AI, Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., Zhang, X., Yu, X., Wu, Y., Wu, Z. F., Gou, Z., Shao, Z., Li, Z., Gao, Z., Liu, A., Xue, B., Wang, B., Wu, B., Feng, B., Lu, C., Zhao, C., Deng, C., Zhang, C., Ruan, C., Dai, D., Chen, D., Ji, D., Li, E., Lin, F., Dai, F., Luo, F., Hao, G., Chen, G., Li, G., Zhang, H., Bao, H., Xu, H., Wang, H., Ding, H., Xin, H., Gao, H., Qu, H., Li, H., Guo, J., Li, J., Wang, J., Chen, J., Yuan, J., Qiu, J., Li, J., Cai, J. L., Ni, J., Liang, J., Chen, J., Dong, K., Hu, K., Gao, K., Guan, K., Huang, K., Yu, K., Wang, L., Zhang, L., Zhao, L., Wang, L., Zhang, L., Xu, L., Xia, L., Zhang, M., Zhang, M., Tang, M., Li, M., Wang, M., Li, M., Tian, N., Huang, P., Zhang, P., Wang, Q., Chen, Q., Du, Q., Ge, R., Zhang, R., Pan, R., Wang, R., Chen, R. J., Jin, R. L., Chen, R., Lu, S., Zhou, S., Chen, S., Ye, S., Wang, S., Yu, S., Zhou, S., Pan, S., Li, S. S., Zhou, S., Wu, S., Ye, S., Yun, T., Pei, T., Sun, T., Wang, T., Zeng, W., Zhao, W., Liu, W., Liang, W., Gao, W., Yu, W., Zhang, W., Xiao, W. L., An, W., Liu, X., Wang, X., Chen, X., Nie, X., Cheng, X., Liu, X., Xie, X., Liu, X., Yang, X., Li, X., Su, X., Lin, X., Li, X. Q., Jin, X., Shen, X., Chen, X., Sun, X., Wang, X., Song, X., Zhou, X., Wang, X., Shan, X., Li, Y. K., Wang, Y. Q., Wei, Y. X., Zhang, Y., Xu, Y., Li, Y., Zhao, Y., Sun, Y., Wang, Y., Yu, Y., Zhang, Y., Shi, Y., Xiong, Y., He, Y., Piao, Y., Wang, Y., Tan, Y., Ma, Y., Liu, Y., Guo, Y., Ou, Y., Wang, Y., Gong, Y., Zou, Y., He, Y., Xiong, Y., Luo, Y., You, Y., Liu, Y., Zhou, Y., Zhu, Y. X., Xu, Y., Huang, Y., Li, Y., Zheng, Y., Zhu, Y., Ma, Y., Tang, Y., Zha, Y., Yan, Y., Ren, Z. Z., Ren, Z., Sha, Z., Fu, Z., Xu, Z., Xie, Z., Zhang, Z., Hao, Z., Ma, Z., Yan, Z., Wu, Z., Gu, Z., Zhu, Z., Liu, Z., Li, Z., Xie, Z., Song, Z., Pan, Z., Huang, Z., Xu, Z., Zhang, Z., and Zhang, Z. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. 2025. URL https://arxiv.org/abs/2501.12948.

Gou, J., Yu, B., Maybank, S. J., and Tao, D. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819, 2021.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Li, X. L. and Liang, P. Prefix-tuning: Optimizing continuous prompts for generation. 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)*, pp. 4582–4597, 2021.

Li, Y., Yuan, P., Feng, S., Pan, B., Sun, B., Wang, X., Wang, H., and Li, K. Turning dust into gold: Distilling complex reasoning capabilities from llms by leveraging negative data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 18591–18599, 2024.

Lin, C.-Y. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pp. 74–81, 2004.

Lu, P., Mishra, S., Xia, T., Qiu, L., Chang, K.-W., Zhu, S.-C., Tafjord, O., Clark, P., and Kalyan, A. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521, 2022.

Magister, L. C., Mallinson, J., Adamek, J., Malmi, E., and Severyn, A. Teaching small language models to reason. *arXiv preprint arXiv:2212.08410*, 2022.

Mao, Y., Ge, Y., Fan, Y., Xu, W., Mi, Y., Hu, Z., and Gao, Y. A survey on lora of large language models. *Frontiers of Computer Science*, 19(7):197605, 2025.

OpenAI. Openai o1 system card, 2024. URL https://openai.com/index/openai-o1-system-card/.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pp. 311–318, 2002.

Yu, R., Liu, S., and Wang, X. Dataset distillation: A comprehensive review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.