MoDEM: Mixture of Domain Expert Models

Toby Simonds Kemal Kurniawan Jey Han Lau

The University of Melbourne
tsimonds@student.unimelb.edu.au
{kurniawan.k, laujh}@unimelb.edu.au

Abstract

We propose a novel approach to enhancing the performance and efficiency of large language models (LLMs) by combining domain prompt routing with domain-specialized models. We introduce a system that utilizes a BERT-based router to direct incoming prompts to the most appropriate domain expert model. These expert models are specifically tuned for domains such as health, mathematics and science. Our research demonstrates that this approach can significantly outperform general-purpose models of comparable size, leading to a superior performance-to-cost ratio across various benchmarks. The implications of this study suggest a potential shift in LLM development and deployment. Rather than focusing solely on creating increasingly large, general-purpose models, the future of AI may lie in developing ecosystems of smaller, highly specialized models coupled with sophisticated routing systems. This approach could lead to more efficient resource utilization, reduced computational costs, and superior overall performance.

1 Introduction

Domain-specific models have demonstrated encouraging performance across various fields, often surpassing state-of-the-art general models in their respective domains. In mathematics, models like Qwen 2 72B Math (Yang et al., 2024) and DeepSeek Math (Shao et al., 2024) have shown superior performance, while in code generation, specialized models such as Code Llama and CodeMistral exhibit significant improvements over comparable general-purpose models (AI, 2024). Also, Zhao et al. (2024) found that models with fewer than 8 billion parameters, when fine-tuned for specific tasks, can rival or even outperform larger models like GPT-4 in certain domains.

Despite the promise of domain-specific AI models, a significant gap exists in integrating these specialized models into a comprehensive and versatile framework. The AI community faces a crucial challenge: how to harness the power of domain-specific models across diverse tasks without sacrificing the versatility of general-purpose models.

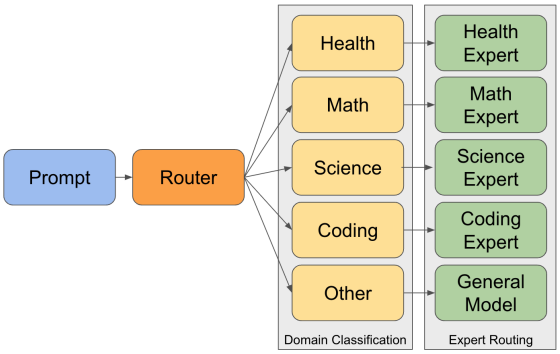

Figure 1: MoDEM architecture diagram

We propose MoDEM (Mixture of Domain Expert Models) to address this. At its core, MoDEM consists of two main components: a router and a collection of domain-specific expert models (Figure 1). The router is designed to classify incoming prompts or queries, determining which domain they best fit into. Once classified, the prompt is then directed to the expert model specialized in that particular domain. This approach allows us to harness the superior performance of domain-specific models while maintaining the ability to handle a wide range of tasks. By leveraging smaller specialized models, we achieve state-of-the-art results in various domains without the computational overhead of larger general-purpose models. This approach dramatically lowers inference costs, as only the relevant expert model is activated for each query. The result is a highly efficient system that delivers strong performance while minimizing resource utilization.

MoDEM key advantage lies in its ability to train and integrate models separately, offering significant benefits in development efficiency and system

capabilities. This approach allows for independent optimization of domain experts, facilitates parallel development, and enables easy integration of new models. The modular design ultimately allows for customization across various industries and applications.

To summarise, our main contributions are:

- We propose an architecture for creating a lightweight router system that effectively directs prompts to domain-specific expert models.
- We demonstrate that domain-based routing to specialized experts can produce state-of-the-art results with significant inference cost reduction.

2 Related Work

Mixture of Experts (MoE) is a machine learning technique that combines multiple specialized models or "experts" to solve complex tasks. In the context of language models, MoE approaches have been explored to enhance both performance and efficiency. There are primarily two categories of MoE implementations in current research:

2.1 Integrated MoE Architectures

Sparse Mixture of Experts (MoE) transformers is first introduced by Shazeer et al. (2017) and further developed in models such as GShard (Lepikhin et al., 2020) and Switch Transformers (Fedus et al., 2022), which integrate expert modules within a single model architecture. These methods use a gating mechanism to dynamically route tokens or layers to different expert sub-networks during training and inference, significantly improving model efficiency by activating only a subset of experts. However, these approaches encounter challenges such as training instability, architectural complexity, and load balancing issues (Li et al., 2024).

2.2 Multi-Model Routing Systems

Recent research has explored systems that leverage multiple distinct language models rather than subnetworks within a single architecture. For example, HuggingGPT (Shen et al., 2023) breaks tasks into subtasks and routes them to different specialized models. Another approach, RouteLLM (Ong et al., 2024), aims to optimize the cost-performance trade-off by selecting between two pre-trained models for different tasks. MoDEM is different to HuggingGPT and RouteLLM in that our approach routes questions into *domains* such as mathematics or health; this is a contrast to HuggingGPT where it routes based on tasks (e.g. OCR) or RouteLLM which attempts to directly predict different models performances in order to attempt to route to the best model.

3 Methodology

3.1 Benchmarks

We use the following evaluation benchmarks to measure the performance of MoDEM: MMLU, MMLU Pro, HumanEval, College Math, Math, GSM8k, and Olympiad Bench. These benchmarks were chosen to provide a balanced distribution of domain-specific and general tasks, ensuring a comprehensive evaluation across diverse areas of expertise. Benchmark sizes below refer to test set size

MMLU (Hendrycks et al., 2021b) (Massive Multitask Language Understanding) is a general-purpose benchmark consisting on 14k questions designed to test a model's proficiency across 57 subjects, including STEM, humanities, social sciences, and more. The questions are in multiple-choice format, covering a broad range of domains to evaluate the model's versatility.

MMLU Pro (Wang et al., 2024) is an extension of MMLU containing 12k questions that focuses on more advanced topics and professional-level knowledgeg. It uses multiple-choice questions similar to MMLU, but with more specialized and higher-level content.

GPQA (Rein et al., 2023) is designed to evaluate models on advanced topics and professional-level knowledge across a wide array of science domains. It contains 448 questions

HumanEval (Chen et al., 2021b) assesses code generation capabilities by providing programming problems that the model must solve. It's 134 questions focuses on domain-specific knowledge within the programming domain, using open-ended coding tasks that require the model to generate functioning code.

College Math (Liu et al., 2024) evaluates a model's understanding of undergraduate-level mathematics on open ended problems, covering topics such as calculus, linear algebra, and probability.

MATH (Liu et al., 2024) is a more general benchmark containing 1.2k questions that covers a wide range of math topics at varying levels, in-

cluding elementary arithmetic, algebra, and more complex problem-solving tasks.

GSM8k (Cobbe et al., 2021a) (Grade School Math 8k) is a benchmark containing 1.3k questions that evaluates mathematical reasoning skills on open ended problems, specifically targeting grade-school level word problems.

Olympiad Bench (He et al., 2024) includes 2.3k challenging open ended math and science problems typically found in international Olympiad competitions.

Of these benchmarks, MMLU, MMLU Pro and GPQA rely on multiple-choice questions (MCQ) to evaluate the model's proficiency across various domains, including general knowledge and professional-level topics. In contrast, HumanEval, College Math, Math, GSM8k, and Olympiad Bench focus on open-ended questions.

3.2 Router

We now describe the router, a key component used for directing incoming queries to the most appropriate domain-specific expert model.

3.2.1 Router Architecture

We used Microsoft DeBERTa-v3-large (He et al., 2023), a 304 million parameter model, and fine-tuned it for our specific routing task. The model was fine-tuned to predict the domain of the input prompt (e.g., Math). We chose DeBERTa-v3-large due to its successful application in classification tasks. With our largest expert models containing up to 73B parameters, the router represents only about 0.42% of the largest expert's size. This ratio ensures that we're not spending disproportionate computational resources on routing.

3.2.2 Domain Selection

The domains selected for our study were the following: Math, Health, Science, Coding and Other. Other represented domains outside of the selected domains. These domains were chosen based on the availability of high-quality specialized models that consistently outperform general-purpose models. They also represent a diverse range of tasks and have significant real-world applications, ensuring that the routing system demonstrates versatility across various areas.

3.2.3 Training Data

For the router, we curated a set of diverse and comprehensive training data covering multiple domains; full list of datasets for each domain is given in Table 1. Our focus was on selecting datasets that capture a broad range of tasks, and complexities within each domain to ensure thorough representation and variety. This approach ensures that our router is exposed to a variety of query formulations and problem types, enhancing its ability to accurately classify and route a broad range of real-world queries. We also use data from the benchmarks, specifically Math, GPQA, GSM8k and HumanEval (Section 3.1), but only from their training partition. Note that we do not use any data from MMLU or MMLU Pro.

To ensure balanced representation across different domains, we implemented a data pruning protocol. A maximum threshold of 30,000 instances per dataset in each domain was applied to Math, Health, and Science while Other and Coding was allowed up to 100,000 entries per dataset. This decision was made because some datasets contained repetitive data, whereas the coding and other benchmarks featured more diverse and varied datasets. We downsampled some coding datasets because they are over represented in the training set. This methodology aimed to create a comprehensive training corpus that prevents any single source from dominating the learning process, thereby optimizing the model's ability to generalize across diverse tasks and knowledge domains. Table 2 outlines total number of training instances in each domain.

To further enhance the diversity and coverage of our dataset, we employed synthetic data generation using the Llama 3.1 405B model (Dubey et al., 2024) . This step was crucial in addressing a significant gap we identified in existing datasets: a scarcity of casual, conversational questions that were clearly classified by domain. We found that while many datasets provided structured, formal queries, they lacked the natural language and varied scenarios typical of real-world interactions. We first created a hand-crafted dataset of 100 examples of conversation-style questions for each domain.[1] We selected a wide array of question content within each domain. We then prompted Llama 405B to generate 100 questions for each hand-crafted examples, resulting in a total of 10,000 synthetic examples for each domain.[2] We found that incorporating hand-crafted examples into the model not only

[1] By "conversation-style", we refer to questions that simulate a more natural, interactive dialogue, as opposed to traditional fact-based or direct question-answer formats.

[2] Temperature set to 1.0 to ensure more diverse dataset (Jean Kaddour).

produced outputs closely aligned with our desired question format but also introduced a greater diversity of questions. When rerunning the same prompt without these hand-selected examples, the model would often generate similar outputs, lacking variety.

Here are some examples of the handcrafted dataset:

- **Math**: *"I'm out with 4 friends and our total bill is $137.50. We want to leave a 15% tip. How much should each person pay if we split it evenly?"*
- **Health**: *"I've had this annoying sore throat for about 4 days now. It's not super painful, but it's definitely there, especially when I swallow."*
- **Science**: *"Can you explain how microwaves work?"*

Given the training data (data in Table 1 and the synthetic data) for each domain, we fine-tuned DeBERTa to classify the domain given an input instance. The fine-tuning was performed with a configuration of 1 epoch, a batch size of 32, and a learning rate of 1e-5. The model was trained on an A100 GPU for 1 epoch.

3.3 Experts

3.3.1 Expert Selection

Our research use a combination of domain-specific and general-purpose models to create a system of expert agents. The selection of these models was primarily based on the availability of high-quality, open-source options that demonstrated superior performance in their respective domains. We utilized two sets of models: a "medium" set with larger parameter counts, and a "small" set with more compact models.

Medium Model Set ($\leq$73B parameters)

The following models were chosen as the experts for our medium model:

- **Health**: Palmyra-health-70B (Writer, 2024)
- **Math**: Qwen2.5-72B-Math-Instruct (Yang et al., 2024)
- **Science**: Qwen2.5-72B-Instruct (Yang et al., 2024)

Domain	Datasets
Math	TIGER-Lab/MathInstruct lighteval/MATH allenai/math_qa openai/gsm8k camel-ai/math meta-math/MetaMathQA deepmind/math_dataset/algebra__linear_1d deepmind/math_dataset/algebra__polynomial_roots deepmind/aqua_rat AI4Math/MathVerse
Health	nlpaueb/biomrc iari/HumGen_Clinical_Notes medmcqa lavita/ChatDoctor-HealthCareMagic-100k
Science	bigbio/pubmed_qa derek-thomas/ScienceQA allenai/sciq bigscience/P3 ai2_arc nlpaueb/biomrc allenai/scitldr tdiggelm/climate_fever medmcqa Idavidrein/gpqa allenai/scifact allenai/scirepeval
Coding	codeparrot/apps bigcode/the-stack nuprl/MultiPL-E code_x_glue_ct_code_to_text deepmind/code_contests huggingface/codecompetitions openai/openai_humaneval bigcode/humanevalpack defect_prediction google/code_x_glue_ct_code_to_text google-research-datasets/mbpp
Other	bigscience/P3 wiki_qa Anthropic/persuasion huggingface/cnn_dailymail allenai/qasper openai/summarize_from_feedback Salesforce/wikitext Anthropic/llm_global_opinions google-research-datasets/wiki_split google-research-datasets/aquamuse

Table 1: Datasets used for training router. Full citations can be found in Appendix A.

Domain	Number of Entries
Health	100,000
Math	113,611
Science	224,885
Coding	572,636
Other	700,000

Table 2: Final data distribution across domains from datasets

- **Coding**: Qwen2.5-72B-Instruct (Yang et al., 2024)
- **Other**: Meta-Llama-3.1-70B-Instruct (Dubey et al., 2024)

Small MoDEM Model Set (≤8B parameters)

We also explored a set of smaller models, each with less than 8B parameters:

- **Health**: Meta-Llama-3.1-8B-Instruct (Dubey et al., 2024)
- **Math**: Qwen2.5-Math-7B-Instruct (Yang et al., 2024)
- **Science**: Qwen2.5-7B-Instruct (Yang et al., 2024)
- **Coding**: Qwen2.5-Coder-7B (Hui et al., 2024)
- **Other**: Meta-Llama-3.1-8B-Instruct (Dubey et al., 2024)

The selection of models was based on evaluating across different domains, where we chose the best-performing models for each domain. In almost all cases, we found that modern models specialized in a specific domain significantly outperformed general-purpose models of the same size (Yang et al., 2024). For instance, the Palmyra models excelled in health (Writer, 2024), while the Qwen2.5-Math model proved to be the most effective for mathematical tasks (Yang et al., 2024).

In cases where domain-specific models were not available, we defaulted to strong general-purpose models to maintain consistency across the system. Models like Meta-Llama-3.1 served as reliable baselines, ensuring good performance even in the absence of specialized options.

3.4 Prompting

We use zero-shot prompting with chain of thought (Wei et al., 2023) to prompt each expert to answer questions in the benchmarks (Section 3.1).[3] Full prompts can be found in appendix B

Category	Accuracy
Health	81.18%
Math	96.63%
Science	83.02%
Coding	77.42%
Other	52.94%
Overall	81.00%

Table 3: Router Classification Results on MMLU.

4 Results

4.1 Router Performance

We evaluated our router on the test set of the datasets used for training, and it achieved an average accuracy of 97%, illustrating its high reliability in routing queries for tasks similar to those it was fine-tuned on. We next assessed the router's performance on the MMLU to test its ability to generalize to out-of-distribution data. We manually mapped the MMLU domains into our chosen domains.[4] Table 3 presents the results. We generally see strong performance for the specialised domains, although for "Other" the performance is a little lower. The latter observation is perhaps not too surprising, it's a "catch all" domain that doesn't have a concrete definition and so it's difficult to have training data that captures the full data distribution. Overall these results suggest that the router generalises well and is sufficiently reliable as a domain router.

We manually assessed some of the error cases and found that some mis-classifications are due to domain-ambiguity. To give an example:

- **Example** "A burial site where the body is allowed to decompose naturally without a casket is called a ____ cemetery."
  True Domain: Health, **Predicted:** Other

4.2 MoDEM Performance

We present the full results in Table 4 and 5 for medium and small MoDEM respectively. For baseline comparisons, we used the Llama 3.1 instruct models, which are generally considered SoTA for

[3]We use the following prompt: *Solve the following problem step by step, explaining each step clearly to ensure the reasoning process is well-justified.* For multiple-choice questions, we have an additional sentence appended to the previous prompt: *Clearly state which multiple choice option you pick.*

[4]Recall that MMLU was not used in the training data for the router.

Domain	Benchmark	Llama 3.1 70B	Medium (<73B)	Improvement
Multi-domain	MMLU	86.0%	**87.7%**	+1.7%
	MMLU Pro	58.0%	**63.4%**	+5.4%
Coding	HumanEval	80.5%*	**86.5%***	+6.0%
Science	GPQA	46.1%	**48.4%**	+2.3%
Math	College Math	42.5%*	**49.5%***	+7.0%
	MATH	65.7%*	**85.9%***	+20.2%
	GSM8k	94.1%*	**95.9%***	+1.8%
	Olympiad Bench	27.7%*	**49.0%***	+21.3%

Table 4: Comparison of Llama 3.1 70B vs. medium MoDEM ($\leq$73B) on various benchmarks. An asterisk (*) indicates numbers sourced from another paper. See Section 4.2 for further explanation.

Domain	Benchmark	Llama 8B	Small (<8B)	Improvement
Multi-domain	MMLU	73.0%	**76.2%**	+3.2%
	MMLU Pro	40.4%	**46.5%**	+6.1%
Coding	HumanEval	72.6%*	**88.4%***	+15.8%
Science	GPQA	32.6%	**35.0%**	+2.4%
Math	College Math	33.8%*	**46.8%***	+13.0%
	MATH	47.2%*	**83.6%***	+36.4%
	GSM8k	76.6%*	**95.2%***	+18.6%
	Olympiad Bench	15.4%*	**41.6%***	+26.2%

Table 5: Comparison of Llama 8B vs. small MoDEM ($\leq$8B) on various benchmarks. An asterisk (*) indicates numbers sourced from another paper. See Section 4.2 for further explanation.

open source models. In instances where the same prompting techniques (zero-shot with Chain of Thought) were employed, we use reported outcomes (denoted by an asterisk in the tables) due to computational limitations and challenges associated with evaluating certain benchmarks (e.g. the test set is not open-source).[5] Concretely, we ran the MMLU, MMLU-Pro and GPQA benchmark results ourselves for the baseline. But for all other benchmarks (HumanEval, College Math, Math, GSM8k and Olympiad Bench) we sourced the results from the Qwen-2.5 Technical Report (Yang et al., 2024) and the Llama 3.1 Technical Report (Dubey et al., 2024).

MoDEM demonstrate consistent performance gain across all evaluated benchmarks when compared to their respective baselines. This consistent improvement highlights the effectiveness of our domain-specialized models and the strength of the routing system in accurately selecting the appropriate expert for each task. For the math domain in particular, MoDEM delivered substantial improvements. The performance gains in these areas show the clear advantage of domain-specific training and highlight the effectiveness of our approach to model specialization. In tasks involving multi-domain knowledge and reasoning (MMLU and MMLU-Pro), both small and medium MoDEM still show improvement over the baseline, demonstrating MoDEM is versatile across different domains.

4.3 Cost and Efficiency Analysis

To evaluate the efficiency of our model, we compared its performance and inference costs with other leading models. All costs are based on Together AI (TogetherAI, 2024) figures where possible. For models not publicly hosted we based price off models of similar size. At the time of publishing the Qwen 2.5 models were not publicly hosted so we defaulted to the Qwen 2 prices. Palmyra-Health was also not hosted on TogetherAi so we use the price of the Writer API. For our router cost we assumed pricing based off other Bert based models of similar size being hosted. We assumed $0.03 per million tokens for the router cost. The reported cost for our models were based off the average over the MMLU dataset. Prices may vary slightly depending on dataset due to different experts models

[5] For these benchmarks, we found in practice over 98% of the prompts were routed to a single model (e.g. 98.4% of Math benchmark was routed to our math expert) and so the results would be reasonably close to those we would obtain if we ran them ourselves.

Model	MMLU Accuracy (%)	Parameters	Input Tokens ($/million tokens)
Llama 3.1 405B	88.6	405B	5.00
Medium MoDEM	**87.7**	**<73B**	**0.92**
Qwen 2.5-72B	86.1	72B	0.9
Llama 3.1 70B	86.0	70B	0.88
Mixtral-8x22B	77.5	8x22B	1.20

Table 6: Comparison of medium MoDEM vs. leading models in terms of estimated inference cost.

Model	MMLU Accuracy (%)	Parameters	Input Tokens ($/million tokens)
Llama 3.1 70B	86.0	70B	0.88
Small MoDEM	**76.2**	**<8B**	**0.22**
Llama 3.1 8B	73.0	8B	0.18
Mixtral-8x7B Instruct	70.6	8x7B	0.60
Gemma2-9B	69.2	9B	0.30
Mistral-7B	62.5	7B	0.20

Table 7: Comparison of small MoDEM vs. leading models in terms of estimated inference cost.

having different inference costs.

MMLU results are in Table 6 and 7 for medium and small MoDEM respectively. Our models demonstrate a superior price-to-performance ratio compared to the leading models. Both medium and small MoDEM deliver higher accuracies across benchmarks while maintaining competitive or lower inference costs, showcasing significant improvements in cost-effectiveness. For small MoDEM in particular, we see that it has a much better performance compared to similar sized models. For medium MoDEM, its performance is close to a much larger model (Llama 405B), even though it is 5-6 times smaller and cheaper. Together these results illustrate the scalability and effectiveness of our approach across a range of model sizes.

5 Discussion

The results of our study on mixture of experts with domain-specific routing suggest a potential shift in the development and deployment of large language models (LLMs). This section explores the implications of our findings, their broader impact on the field of artificial intelligence, and potential directions for future research.

5.1 Potential Shift in Model Development

Our research demonstrates that combining domain routing with models fine-tuned for specific domains can significantly outperform base models of the same size, leading to a more favorable performance-to-cost ratio. This challenges the current trend of developing increasingly large, general-purpose models and instead points towards a future where AI systems consist of an ecosystem of smaller, highly specialized models coupled with intelligent routing mechanisms.

This shift parallels how human expertise is organized in society, where specialists in various fields collaborate to solve complex problems. In the context of AI, this approach could result in:

- More efficient resource utilization
- Reduced computational costs
- Superior performance in domain-specific tasks
- Increased interpretability and control over model outputs

As compute bottlenecks continue to constrain the development of ever-larger models, the transition towards domain-specific models may become necessary to sustain progress in LLM capabilities and performance. By optimizing resources and leveraging domain expertise, this approach holds promise for maintaining the current rate of advancements in the field.

Our approach holds significant potential for future improvement. As the AI community develops more specialized, high-performance models, we

anticipate substantial increases in the overall capabilities of our system. The current performance represents a lower bound of what's achievable, and as specialized models trained on domain-specific data emerge, it will benefit our mixture of experts routing approach.

We want to also highlight that MoDEM's domain set is adaptable. As new specialized models in fields like legal or environmental science become available, they can be easily integrated by updating the router and adding relevant expert models. Existing domains can also be refined or consolidated based on performance analysis, ensuring continued efficiency. Additionally, hierarchical domain structures, such as broad categories with more specific sub-domains, could further enhance routing precision. This adaptable approach ensures our system evolves with AI developments, providing a scalable framework for continuous improvement aligned with real-world needs.

5.2 Implications for AI Deployment

Our findings reveal that domain-specific models with fewer parameters can match or outperform larger general-purpose models like Llama 405B, carrying important implications for AI deployment. This approach delivers state-of-the-art performance at a fraction of the inference cost, drastically reducing computational overhead while maintaining high-quality results. It opens opportunities for cost-effective AI deployment, particularly in resource-constrained settings where large models are impractical.

5.3 Future Research Directions

Our findings highlight several promising research directions using mixture of experts. Key challenges include developing better routing techniques, such as improving domain selection accuracy and scaling to more domains. Expanding domain-specific models to cover a wider range of tasks will also increase the system's applicability across industries. Cross-domain integration and dynamic model selection could enhance handling of complex queries by combining outputs from multiple experts in real time. Additionally, introducing difficulty-based routing within each domain could optimize resource use, directing simpler queries to smaller models and complex ones to larger models, improving cost-effectiveness and performance.

6 Conclusion

This study demonstrates the effectiveness of combining domain-specific expert models with routing to enhance the performance and efficiency of large language models. Our approach consistently outperformed baseline models across various benchmarks, with strong improvement in specialized domains such as mathematics. Both our small and medium MoDEM achieved superior performance-to-cost ratios compared to larger, general-purpose models, highlighting the potential for significant efficiency gains in AI deployment.

This research demonstrates a promising new direction in the field of artificial intelligence: the combination of domain-specific models with intelligent routing systems. The study's findings suggest that this approach can lead to significant improvements in both performance and cost-efficiency compared to traditional large language models. These findings point to a potential shift in AI development and deployment. Rather than focusing solely on creating increasingly large general-purpose models, the future may lie in developing ecosystems of smaller, highly specialized models coupled with sophisticated routing systems. This approach could lead to more efficient resource utilization, reduced computational costs, and superior performance in domain-specific tasks.

Limitations

It's important to note that our selection was constrained by the current landscape of available open-source, domain-specific models. The field of AI is rapidly evolving, and the development of specialized models is a relatively recent trend. As such, our study represents an initial exploration into the potential of combining domain experts with intelligent routing.

Additionally, we were somewhat limited by the lack of public APIs for certain models, making it challenging to run direct benchmarks. This constraint forced us to rely on benchmarks reported in other studies, which may not have fully captured the performance nuances in our specific use case. As more models become accessible and standardized benchmarking tools evolve, future iterations of our research will likely benefit from more comprehensive and direct performance evaluations.

Acknowledgments

We thank the reviewers from ALTA for their valuable feedback and constructive comments on this paper.

References

Mistral AI. 2024. Codestral: Hello, World! Section: news.

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, Minneapolis, Minnesota. Association for Computational Linguistics.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. 2021. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*.

Jan A. Botha, Manaal Faruqui, John Alex, Jason Baldridge, and Dipanjan Das. 2018. Learning to split and rephrase from Wikipedia edit history. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 732–737, Brussels, Belgium. Association for Computational Linguistics.

Isabel Cachola, Kyle Lo, Arman Cohan, and Daniel S. Weld. 2020. TLDR: Extreme summarization of scientific documents. *arXiv:2004.15011*.

Federico Cassano, John Gouwar, Francesca Lucchetti, Claire Schlesinger, Carolyn Jane Anderson, Michael Feldman, Molly Q Greenberg, Abhinav Jangda, and Arjun Guha. 2024. Knowledge Transfer from High-Resource to Low-Resource Programming Languages for Code LLMs. *Proceedings of the ACM on Programming Languages (PACMPL)*, 8(OOPSLA).

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. 2021a. Evaluating large language models trained on code. *Preprint*, arXiv:2107.03374.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. 2021b. Evaluating Large Language Models Trained on Code. *arXiv preprint*. ArXiv:2107.03374 [cs].

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv:1803.05457v1*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021a. Training Verifiers to Solve Math Word Problems. *arXiv preprint*. ArXiv:2110.14168 [cs].

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

Pradeep Dasigi, Kyle Lo, Iz Beltagy, Arman Cohan, Noah A. Smith, and Matt Gardner. 2021. A dataset of information-seeking questions and answers anchored in research papers.

Thomas Diggelmann, Jordan Boyd-Graber, Jannis Bulian, Massimiliano Ciaramita, and Markus Leippold. 2020. Climate-fever: A dataset for verification of real-world climate claims. *Preprint*, arXiv:2012.00614.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien

Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaoqing Ellen Tan, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aaron Grattafiori, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alex Vaughan, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Franco, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, Danny Wyatt, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Firat Ozgenel, Francesco Caggioni, Francisco Guzmán, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Govind Thattai, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Karthik Prasad, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kun Huang, Kunal Chawla, Kushal Lakhotia, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Maria Tsimpoukelli, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikolay Pavlovich Laptev, Ning Dong, Ning Zhang, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre

Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Rohan Maheswari, Russ Howes, Ruty Rinott, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Kohler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vítor Albiero, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaofang Wang, Xiaojian Wu, Xiaolan Wang, Xide Xia, Xilun Wu, Xinbo Gao, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yuchen Hao, Yundi Qian, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, and Zhiwei Zhao. 2024. The Llama 3 Herd of Models. *arXiv preprint*. ArXiv:2407.21783 [cs].

Esin Durmus, Liane Lovitt, Alex Tamkin, Stuart Ritchie, Jack Clark, and Deep Ganguli. 2024. Measuring the persuasiveness of language models.

Esin Durmus, Karina Nyugen, Thomas I. Liao, Nicholas Schiefer, Amanda Askell, Anton Bakhtin, Carol Chen, Zac Hatfield-Dodds, Danny Hernandez, Nicholas Joseph, Liane Lovitt, Sam McCandlish, Orowa Sikder, Alex Tamkin, Janel Thamkul, Jared Kaplan, Jack Clark, and Deep Ganguli. 2023. Towards measuring the representation of subjective global opinions in language models. *Preprint*, arXiv:2306.16388.

William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. *arXiv preprint*. ArXiv:2101.03961 [cs].

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. 2024. OlympiadBench: A Challenging Benchmark for Promoting AGI with Olympiad-Level Bilingual Multimodal Scientific Problems. *arXiv preprint*. ArXiv:2402.14008 [cs].

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. DeBERTaV3: Improving DeBERTa using ELECTRA-Style Pre-Training with Gradient-Disentangled Embedding Sharing. *arXiv preprint*. ArXiv:2111.09543 [cs].

Dan Hendrycks, Steven Basart, Saurav Kadavath, Mantas Mazeika, Akul Arora, Ethan Guo, Collin Burns, Samir Puranik, Horace He, Dawn Song, and Jacob Steinhardt. 2021a. Measuring coding challenge competence with apps. *NeurIPS*.

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

Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Kai Dang, An Yang, Rui Men, Fei Huang, Xingzhang Ren, Xuancheng Ren, Jingren Zhou, and Junyang Lin. 2024. Qwen2.5-Coder Technical Report. *arXiv preprint*. ArXiv:2409.12186 [cs].

Hamel Husain, Ho-Hsiang Wu, Tiferet Gazit, Miltiadis Allamanis, and Marc Brockschmidt. 2019. Codesearchnet challenge: Evaluating the state of semantic code search. *arXiv preprint arXiv:1909.09436*.

Qi Liu Jean Kaddour. Synthetic Data Generation in Low-Resource Settings via Fine-Tuning of Large Language Models.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. Pubmedqa: A dataset for biomedical research question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2567–2577.

Matt Gardner Johannes Welbl, Nelson F. Liu. 2017. Crowdsourcing multiple choice science questions.

Denis Kocetkov, Raymond Li, Loubna Ben Allal, Jia Li, Chenghao Mou, Carlos Muñoz Ferrandis, Yacine Jernite, Margaret Mitchell, Sean Hughes, Thomas Wolf, Dzmitry Bahdanau, Leandro von Werra, and Harm de Vries. 2022. The stack: 3 tb of permissively licensed source code. *Preprint*.

Sayali Kulkarni, Sheide Chammas, Wan Zhu, Fei Sha, and Eugene Ie. 2020. Aquamuse: Automatically generating datasets for query-based multi-document summarization. *Preprint*, arXiv:2010.12694.

Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. 2020. GShard: Scaling Giant Models with Conditional Computation and Automatic Sharding. *arXiv preprint*. ArXiv:2006.16668 [cs, stat].

Guohao Li, Hasan Abed Al Kader Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. 2023. Camel: Communicative agents for "mind" exploration of large scale language model society. *Preprint*, arXiv:2303.17760.

Jing Li, Zhijie Sun, Xuan He, Li Zeng, Yi Lin, Entong Li, Binfan Zheng, Rongqian Zhao, and Xin Chen. 2024. LocMoE: A Low-Overhead MoE for Large Language Model Training. *arXiv preprint*. ArXiv:2401.13920 [cs].

Yujia Li, David Choi, Junyoung Chung, Nate Kushman, Julian Schrittwieser, Rémi Leblond, Tom Eccles, James Keeling, Felix Gimeno, Agustin Dal Lago, Thomas Hubert, Peter Choy, Cyprien de Masson d'Autume, Igor Babuschkin, Xinyun Chen, Po-Sen Huang, Johannes Welbl, Sven Gowal, Alexey Cherepanov, James Molloy, Daniel Mankowitz, Esme Sutherland Robson, Pushmeet Kohli, Nando de Freitas, Koray Kavukcuoglu, and Oriol Vinyals. 2022. Competition-level code generation with alphacode. *arXiv preprint arXiv:2203.07814*.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program induction by rationale generation: Learning to solve and explain algebraic word problems. *ACL*.

Hongwei Liu, Zilong Zheng, Yuxuan Qiao, Haodong Duan, Zhiwei Fei, Fengzhe Zhou, Wenwei Zhang, Songyang Zhang, Dahua Lin, and Kai Chen. 2024. MathBench: Evaluating the Theory and Application Proficiency of LLMs with a Hierarchical Mathematics Benchmark. *arXiv preprint*. ArXiv:2405.12209 [cs].

Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *The 36th Conference on Neural Information Processing Systems (NeurIPS)*.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Pointer sentinel mixture models. *Preprint*, arXiv:1609.07843.

Niklas Muennighoff, Qian Liu, Armel Zebaze, Qinkai Zheng, Binyuan Hui, Terry Yue Zhuo, Swayam Singh, Xiangru Tang, Leandro von Werra, and Shayne Longpre. 2023. Octopack: Instruction tuning code large language models. *arXiv preprint arXiv:2308.07124*.

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E. Gonzalez, M. Waleed Kadous, and Ion Stoica. 2024. RouteLLM: Learning to Route LLMs with Preference Data. *arXiv preprint*. ArXiv:2406.18665 [cs].

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2022. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Proceedings of the Conference on Health, Inference, and Learning*, volume 174 of *Proceedings of Machine Learning Research*, pages 248–260. PMLR.

Dimitris Pappas, Petros Stavropoulos, Ion Androutsopoulos, and Ryan McDonald. 2020. BioMRC: A dataset for biomedical machine reading comprehension. In *Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing*, pages 140–149, Online. Association for Computational Linguistics.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. 2023. GPQA: A Graduate-Level Google-Proof Q&A Benchmark. *arXiv preprint*. ArXiv:2311.12022 [cs].

Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Stella Biderman, Leo Gao, Tali Bers, Thomas Wolf, and Alexander M. Rush. 2021. Multitask prompted training enables zero-shot task generalization. *Preprint*, arXiv:2110.08207.

Saxton, Grefenstette, and Kohli Hill. 2019. Analysing mathematical reasoning abilities of neural models. *arXiv:1904.01557*.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. *arXiv preprint*. ArXiv:2402.03300 [cs].

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer. *arXiv preprint*. ArXiv:1701.06538 [cs, stat].

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. 2023. HuggingGPT: Solving AI Tasks with ChatGPT and its Friends in Hugging Face. *arXiv preprint*. ArXiv:2303.17580 [cs].

Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. 2020. Learning to summarize from human feedback. In *NeurIPS*.

TogetherAI. 2024. [link].

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7534–7550, Online. Association for Computational Linguistics.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. 2024. MMLU-Pro: A More Robust and Challenging Multi-Task Language Understanding Benchmark. *arXiv preprint*. ArXiv:2406.01574 [cs].

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. *arXiv preprint*. ArXiv:2201.11903 [cs].

Writer Engineering Writer. 2024. Palmyra-Med-70b: A powerful LLM designed for healthcare. https://dev.writer.com.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. 2024. Qwen2 Technical Report. *arXiv preprint*. ArXiv:2407.10671 [cs].

Yi Yang, Wen-tau Yih, and Christopher Meek. 2015. WikiQA: A challenge dataset for open-domain question answering. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2013–2018, Lisbon, Portugal. Association for Computational Linguistics.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2023. Metamath: Bootstrap your own mathematical questions for large language models. *arXiv preprint arXiv:2309.12284*.

Xiang Yue, Xingwei Qu, Ge Zhang, Yao Fu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. 2023a. Mammoth: Building math generalist models through hybrid instruction tuning. *arXiv preprint arXiv:2309.05653*.

Xiang Yue, Xingwei Qu, Ge Zhang, Yao Fu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. 2023b. MAmmoTH: Building Math Generalist Models through Hybrid Instruction Tuning.

Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei Chang, Peng Gao, and Hongsheng Li. 2024. Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems? In *arXiv*.

Justin Zhao, Timothy Wang, Wael Abid, Geoffrey Angus, Arnav Garg, Jeffery Kinnison, Alex Sherstinsky, Piero Molino, Travis Addair, and Devvret Rishi. 2024. LoRA Land: 310 Fine-tuned LLMs that Rival GPT-4, A Technical Report. *arXiv preprint*. ArXiv:2405.00732 [cs].

Yaqin Zhou, Shangqing Liu, Jingkai Siow, Xiaoning Du, and Yang Liu. 2019. Devign: Effective vulnerability identification by learning comprehensive program semantics via graph neural networks. In *Advances in Neural Information Processing Systems*, pages 10197–10207.

Appendix A: Dataset Citations

Below is a list of citations for the datasets used in our study, organized by domain:

- **Math**
  - TIGER-Lab/MathInstruct: (Yue et al., 2023a)
  - lighteval/MATH: (Yue et al., 2023b)
  - allenai/math_qa: (Amini et al., 2019)
  - openai/gsm8k: (Cobbe et al., 2021b)
  - camel-ai/math: (Li et al., 2023)
  - meta-math/MetaMathQA: (Yu et al., 2023)
  - deepmind/math_dataset/algebra__linear_1d: (Saxton et al., 2019)
  - deepmind/math_dataset/algebra__polynomial_roots: (Saxton et al., 2019)
  - deepmind/aqua_rat: (Ling et al., 2017)
  - AI4Math/MathVerse: (Zhang et al., 2024)
- **Health**
  - nlpaueb/biomrc: (Pappas et al., 2020)
  - iari/HumGen_Clinical_Notes: augmented-clinical notes
  - medmcqa: (Pal et al., 2022)
  - lavita/ChatDoctor-HealthCareMagic-100k: https://huggingface.co/datasets/lavita/ChatDoctor-HealthCareMagic-100k

- **Science**
  - bigbio/pubmed_qa: (Jin et al., 2019)
  - derek-thomas/ScienceQA: (Lu et al., 2022)
  - allenai/sciq: (Johannes Welbl, 2017)
  - bigscience/P3: (Sanh et al., 2021)
  - ai2_arc: (Clark et al., 2018)
  - nlpaueb/biomrc: (Pappas et al., 2020)
  - allenai/scitldr: (Cachola et al., 2020)
  - tdiggelm/climate_fever: (Diggelmann et al., 2020)
  - medmcqa: (Pal et al., 2022)
  - Idavidrein/gpqa: (Rein et al., 2023)
  - allenai/scifact: (Wadden et al., 2020)
  - allenai/scirepeval: (Wadden et al., 2020)
- **Coding**
  - codeparrot/apps: (Hendrycks et al., 2021a)
  - bigcode/the-stack: (Kocetkov et al., 2022)
  - nuprl/MultiPL-E: (Cassano et al., 2024)
  - code_x_glue_ct_code_to_text: (Husain et al., 2019)
  - deepmind/code_contests: (Li et al., 2022)
  - huggingface/codecompetitions: (Li et al., 2022)
  - openai/openai_humaneval: (Chen et al., 2021a)
  - bigcode/humanevalpack: (Muennighoff et al., 2023)
  - defect_prediction: (Zhou et al., 2019)
  - google/code_x_glue_ct_code_to_text: (Husain et al., 2019)
  - google-research-datasets/mbpp: (Austin et al., 2021)
- **Other**
  - bigscience/P3: (Sanh et al., 2021)
  - wiki_qa: (Yang et al., 2015)
  - Anthropic/persuasion: (Durmus et al., 2024)
  - huggingface/cnn_dailymail: (See et al., 2017)
  - allenai/qasper: (Dasigi et al., 2021)
  - openai/summarize_from_feedback: (Stiennon et al., 2020)
  - Salesforce/wikitext: (Merity et al., 2016)
  - Anthropic/llm_global_opinions: (Durmus et al., 2023)
  - google-research-datasets/wiki_split: (Botha et al., 2018)
  - google-research-datasets/aquamuse: (Kulkarni et al., 2020)

Appendix B: Prompting Techniques

For Prompting the Model

Prompt:
Solve the following problem step by step, explaining each step clearly to ensure the reasoning process is well-justified. Clearly state which multiple choice option you pick.

Input:

```
{question}
```

For Our LLM Evaluation

Prompt: *You will be given a ground truth answer and a model answer. Please output ACCURATE if the model answer matches the ground truth answer or INACCURATE otherwise. Please only return ACCURATE or INACCURATE. It is very important for my job that you do this.*

Input Format:

```
<GroundTruthAnswer>
{correctAnswer}
</GroundTruthAnswer>

<ModelAnswer>
{predictedAnswer}
</ModelAnswer>
```