LoRA-Select: Adaptive Task-Specific Text Generation with Dynamic LoRA Switching

Vaibhav Thalanki[1*], **Lakshmi Vadhanie G**[1*], **Akshara Reddy Patlannagari**[2*]

[1] Khoury College of Computer Sciences, Northeastern University
[2] College of Engineering, Northeastern University
thalanki.v@northeastern.edu, ganesh.la@northeastern.edu, patlannagari.a@northeastern.edu

Abstract

This project addresses the critical challenges of efficiency, adaptability, and domain specialization in large language models (LLMs). While full fine-tuning requires prohibitive computational resources and prompt-based methods often fail in specialized domains, we propose a novel solution: a dynamic Mixture-of-Experts (MoE) framework utilizing Low-Rank Adaptation (LoRA). Our approach trains lightweight, domain-specific adapters and dynamically activates them during inference based on input classification. Using a strict Top-1 selection strategy for adapter loading, we demonstrate significant improvements across legal, financial, and healthcare domains compared to baseline models. Results indicate enhanced task performance with minimal inference latency overhead and reduced memory footprint by loading only domain-relevant components. This framework enables precise, resource-efficient AI specialization while supporting scalable integration of new domains without base model retraining.

Introduction

Large language models have revolutionized natural language processing, but their application across specialized domains presents significant challenges. Technical fields such as medicine, law, and finance demand domain expertise that general-purpose LLMs often lack. Current solutions to this problem fall into two categories, each with substantial limitations: (1) full model fine-tuning, which requires extensive computational resources and training time, making it impractical for most applications; and (2) prompt-based methods, which frequently lack the specificity and depth required for technical domains.

These limitations create a pressing need for more efficient adaptation techniques that can deliver specialized knowledge without the computational burden of full fine-tuning. This project addresses this challenge through a novel dynamic Mixture-of-Experts framework using Low- Rank Adaptation. Rather than fine-tuning the entire model for each domain, we train lightweight, domain-specific adapters that can be selectively activated based on input classification. This approach offers several advantages:

- Efficient resource utilization by loading only domain-relevant components during inference
- Scalable integration of new domains without retraining the base model
- Reduced memory footprint compared to maintaining multiple specialized models

By focusing on the legal, financial, and healthcare domains, we demonstrate the framework's ability to deliver domain-specific expertise while maintaining operational efficiency.

Background

Large Language Models and Domain Adaptation

Large language models have demonstrated remarkable general capabilities across various tasks, but often lack the specialized knowledge required for technical domains. Domain adaptation techniques aim to address this limitation by tailoring models to specific fields without losing their general capabilities. Current approaches include prompt engineering, few-shot learning, and parameter-efficient fine-tuning methods.

LLM Fine-Tuning

Fine-tuning updates a pre-trained model's weights using domain-specific data, typically through:

- **Full Fine-Tuning:** Adjusts all parameters of the model. While effective, this method is computationally expensive and risks catastrophic forgetting (where the model loses general capabilities).
- **Partial Fine-Tuning**: Updates only specific layers (e.g., the top layers) to preserve most of the pre-trained knowledge. This reduces computational overhead but may limit adaptability.
- **Adapter-Based Methods**: Inserts small, trainable modules between layers of a frozen base model. Adapters are parameter-efficient but can introduce inference latency.

Recent advances in PEFT, such as Low-Rank Adaptation (LoRA), have made fine-tuning more scalable by decomposing weight updates into low-rank matrices, drastically reducing trainable parameters.

*These authors contributed equally.

Low-Rank Adaptation (LoRA)

Low-Rank Adaptation (LoRA), introduced by (Hu et al. 2021), represents a significant advancement in parameter-efficient fine-tuning. LoRA freezes the pre-trained model weights and injects trainable rank decomposition matrices into each layer of the Transformer architecture. The key insight behind LoRA is that weight updates during adaptation have low intrinsic rank, parameterized as:

$$\Delta W = A \times B$$

Where $A \in R^{d \times r}$ and $B \in R^{r \times k}$, and $r < min(d, k)$. This approach significantly reduces the number of trainable parameters while maintaining performance comparable to full fine-tuning, making it particularly suitable for domain specialization in resource- constrained environments.

Mixture-of-Experts (MoE) Architecture

The Mixture-of-Experts architecture employs multiple specialized neural networks (experts) and a gating mechanism to route inputs to the most appropriate expert. While traditional MoE approaches incorporate experts within the model architecture, our approach combines MoE principles with external LoRA adapters.

Challenges in Domain-Specialized LLMs

- **Computational Cost**: Full fine-tuning is infeasible for many applications due to hardware constraints.
- **Scalability:** Maintaining separate models for each domain is memory-intensive.
- **Knowledge Interference**: Simultaneous adaptation to multiple domains can degrade performance.

Related Work

MoA (Feng et al. 2024) introduces Mixture-of-LoRAs, a framework enhancing LLM adaptability and efficiency via domain-specific LoRA modules (e.g., medicine, finance) and dynamic Mixture-ofExperts routing. A sequence-level strategy, guided by domain metadata, achieves 99.9% routing accuracy, reduces memory overhead by activating only relevant adapters, and mitigates task interference. Similarly, our approach uses the same strategy to prevent interference between tasks and ultimately enhances the performance of each individual prompt response.

HDMoLE (Mu et al. 2025) combines LoRA with MoE to efficiently fine-tune LLM-based ASR models across multiple accent domains without catastrophic forgetting. While both HDMoLE and our proposed approach utilize a Mixture of LoRA Experts architecture, a key difference lies in expert selection: HDMoLE employs dynamic thresholds to activate varying numbers of experts based on input characteristics, whereas our approach uses a strict Top-1 selection strategy, loading a single domain-specific adapter per prompt.

LoRA-Switch (Kong et al. 2024) presents an innovative system-algorithm co-design approach for efficiently managing multiple specialized adapters in large language models. This was developed to address the latency in inference time when using a MoE model based on dynamic adapter switching. They implement token-wise adapter routing where each token activates k-weighted adapter paths before processing. The fundamental difference between our approach and Kong et al. is granularity and dynamism. Our approach is like choosing a single specialist to summarize an entire document. LoRA-Switch is more like having a team of specialists collaborating word-by-word, with different experts weighing in more heavily depending on the specific word being processed.

LoRA+ (Hayou, Ghosh, and Yu 2024) optimizes learning rate allocation for LoRA matrices, addressing suboptimal feature learning in wide networks. By assigning differential learning rates to up- projection and down-projection matrices, LoRA+ achieves faster convergence and improved task performance without additional computational overhead. Our framework integrates this insight by implementing specialized learning rate schedules for domain-specific adapters, enhancing training efficiency while maintaining the parameter efficiency of standard LoRA approaches.

Alternative Approaches and Their Limitations

Several alternative approaches could address domain specialization in language models:

Full Model Fine-tuning provides strong performance but requires extensive computational resources and creates separate models for each domain. Despite optimization techniques like DeepSpeed-ZeRO (Rajbhandari et al. 2022), this approach remains impractical for multi-domain deployment in resource-constrained environments.

Prompt-based Methods like prompt tuning (Lester, Al-Rfou, and Constant 2021) and prefix tuning (Li and Liang, 2021) are extremely parameter-efficient but often lack the capacity to capture deep domain expertise, particularly in highly specialized fields like medicine and law.

Adapter-based Methods such as Adapter Fusion (Pfeiffer et al. 2021) add trainable modules between transformer layers. While effective, these methods introduce computational overhead during inference due to additional layers in the model architecture.

Embedded MoE Architectures like Switch Transformers (Fedus, Zoph, and Shazeer 2022) and Mixtral (Jiang et al. 2024) integrate experts within the model architecture but require specialized model architectures and substantial computational resources.

Our dynamic LoRA-based framework synthesizes insights from these foundational works while addressing their limitations. By implementing sequence-level routing with Top-1 expert selection, incorporating differential learning rates, and optimizing adapter switching, we achieve domain specialization with minimal computational overhead. This approach balances the performance benefits of domain-specific adaptation with the practical constraints of resource-

Parameter	Value
r	16
LoRA α	32
LoRA Dropout	0.05
Bias	none
Task Type	CAUSAL_LM

Table 1: LoRA configuration for the adapters.

efficient deployment, enabling specialized language modeling across diverse technical fields.

Project Description

Timeline

Since the task of this project is to choose a base model to fine tune using Peft LoRA, the first task in hand was to choose a baseline model. The task here is to build somewhat a question-answering text generation model. Therefore, we looked at different Decoder only LLMs: Llama, Mistral-7b and GPTJ-6b. Initial experiments were done using the Llama model only to realise the free Google Colab (T4 GPU) could not effectively load the Llama pre-trained weights. Quantization was then performed to make the loading process easier. But the Llama-7b variant was running into low-level errors in CUDA and GPU mapping.

However, another teammate had tried 4-bit quantization on Mistral-7b model and successfully loaded the model weights. Since we were short on time, we decided to scrap Llama. We tested few samples from the dataset on the un-finetuned Mistral model, the predictions were sound, although not capturing enough domain knowledge but it captured the gist of the query. Mistral was trained on a massive, diverse corpus of publicly available text (books, websites, code repositories, etc.), therefore the baseline is really high and it is very capable for zero-shot. So, the improvements made using fine tuning is not going to be palpable.

For the weaker baseline model, we chose the GPTJ-6b model. The GPT-J-6B model is a 6-billion parameter open-source autoregressive language model developed by EleutherAI, designed as an alternative to OpenAI's GPT-3. This model has a lower baseline than Mistral but it learns quick. On this note, fine tuning was applied choosing this model as the base model. A subset of the ChatDoctor-200k dataset was used to fine tune the medical LoRA adapter with GPTJ. This test lasted about 450 steps with batch size 4 and gradient accumulation 4 (equivalent to 16 batch size) totalling about 6 hours. The convergence appeared to be very slow with unsatisfactory results.

Given the time constraints, we decided to go back to Mistral-7b model and we fine tuned 3 LoRA adapter weights for the domains: Finance, Medicine and Legal. The LoRA config and training hyperparameters are listed below in Tables 1 and 2. The legal adapter was trained on the dzunggg dataset for 500 steps, totalling 3.8 hours of training. The medical adapter was trained on the chatdoctor dataset for 400 steps, totalling 3.2 hours of training and the finance adapter on the gbharti dataset for 740 steps, taking 3.6 hours. The total number of parameters for the model (mistral-7b with the adapter following the specifications mentioned in Table 1) is 7,248,547,840 out of which only 0.094% are trainable equalling 6,815,744 params.

Datasets

Three datasets are used to fine tune three specialized adapters. The Chatdoctor dataset (Li et al. 2023) consists of 200,000 medical conversations between patients and providers. It is structured as detailed Q&A pairs that represent realistic patient-doctor interactions. This is used to fine tune the medical adapter. The legal adapter is fine tuned on a dataset which consists of professional legal QA pairs covering various topics (Nguyen 2024). It contains carefully curated content from legal professionals with domain expertise. Lastly, the finance adapter is fine tuned on finance-alpaca dataset by (Bharti 2024). It consists of 60,000 QA pairs on financial concepts and investments. It spans topics from personal finance to advanced market analysis and investment strategies.

Methodology

We fine-tune the Mistral-7b base language model using LoRA for specialized text generation across domains (Medicine, Finance, Law). The ComprehendIt zero-shot task classification model will determine the most relevant domain for each input, dynamically loading the corresponding LoRA adapter at inference. This approach enhances specialization while optimizing memory usage, enabling scalable and adaptive natural language understanding (NLU) and natural language generation (NLG). The full pipeline is shown below:

- **User Input** $\rightarrow$ Query is received.
- **Task Classification** $\rightarrow$ 0-shot classifier determines the domain.
- **LoRA Dispatching** $\rightarrow$ The relevant LoRA adapter is loaded dynamically.
- **Inference** $\rightarrow$ Mistral, with the fine-tuned LoRA adapter applied, generates a natural language response.

Empirical Results

The adapters were trained using the Huggingface PEFT library. The training results are available on Tensorboard on the Huggingface Hub. From the graphs 2, 3 and 4, it can be seen that although a smoothing of 0.6 is applied, the plots are converging with decreasing eval and training loss. ROUGE and BERTScores were used for quantitative evaluation of the original model and model with the fine-tuned adapters. For the evaluation, a few 20 samples were used along with human references from the dataset for each of the domains. The evaluation results across domains are

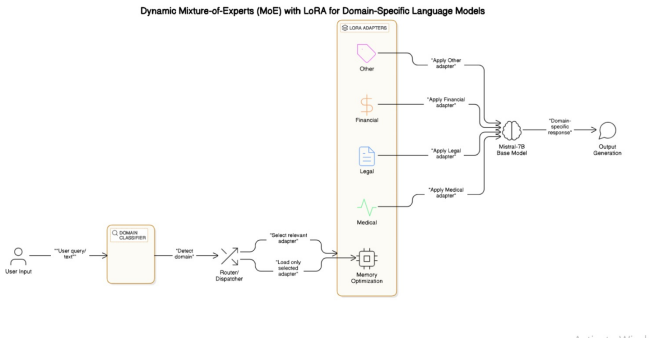

Figure 1: System Architecture

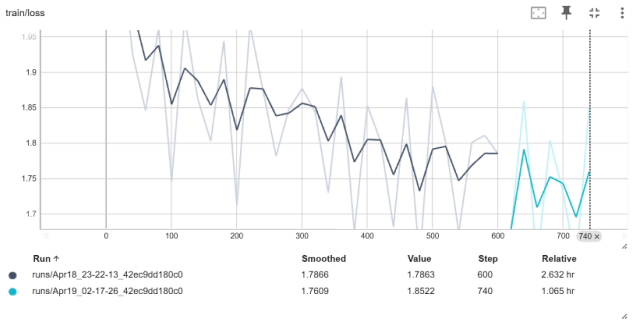

Figure 2: Training Plot for Finance

shown in Table 3.

The ROUGE score usually does not comprehend the semantics of the generated language with respect to the original human reference but merely compares the ngrams. To properly assess the relevance, BERT-Score also was used for performance evaluation. The BERTScores for the legal and finance domain improved compared to the original Mistral-7b model, however, there was a significant dip for the healthcare domain. Note: It can be seen that the original Mistral model performs very competitively relative to the fine tuned ones, this shows that the pre-trained mistral model has a very high baseline and the improvements made by PEFT fine tuning especially for very few steps/epochs as performed in this project are not super significant. In Figure 5, the performance comparison in terms for BERTScore and ROUGE2 is shown across domains.

User Interface

A user interface was developed using Gradio to allow users to enter queries and get results from the model with auto-dispatched domain-specific adapters. The LoRA adapters were pushed to hub and can be loaded from the hub using the Huggingface PEFT library efficiently. The model needs to be instantiated and running in session for the live link to be active. The live link can be sent and the UI can be accessed across web browsers.

The user interface allows the user to override the domain if needed and also allows different decoding strategies like top-p and temperature. Since the Comprehend-It model has to classify the user text into the labels: $[legal, medicine, finance]$, and the appropriate adapter needs to be loaded in inference time, the response time is a little longer than usual. From the experiments provided, the average time for response is about 10-15 seconds.

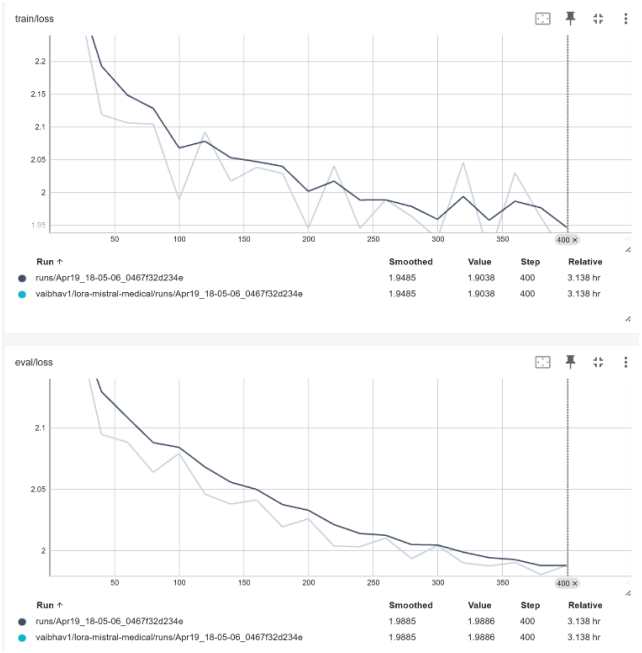

Figure 3: Eval and Train loss Plot for Medicine

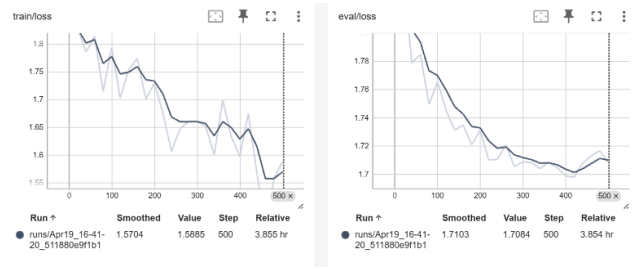

Figure 4: Eval and Train loss Plot for Legal

Broader Implications

The project framework offers a meaningful contribution to the ongoing challenge of domain specialization in large language models (LLMs). As LLMs become increasingly integrated into high-stakes domains such as medicine, finance, and law, ensuring that they deliver precise, contextually accurate, and resource-efficient outputs becomes imperative. The proposed dynamic Mixture-of-Experts (MoE) strategy, utilizing Low-Rank Adaptation (LoRA), effectively addresses the dual concerns of computational scalability and domain adaptability. One giant leap forward towards this approach is the unveil of the deepseek LLM model that rattled the global stock market, which uses a MoE architecture,

Parameter	Value
Max Steps	1000
Batch Size (per device)	4
Gradient Accumulation	4
Optimizer	adamw_torch
Learning Rate	0.0002
Weight Decay	0.001
Mixed Precision	fp16: True, bf16: False
Scheduler	cosine
Warmup Ratio	0.03
Max Grad Norm	0.3

Table 2: Essential training arguments for replication.

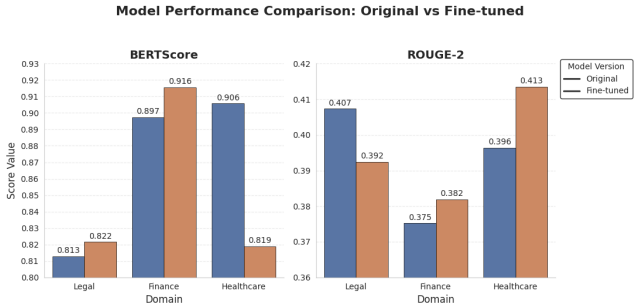

Figure 5: Comparison of Performance

each model specializing in a singular domain.

Societal Implications

The deployment of LLMs in sensitive domains raises critical concerns related to the reliability, accuracy, and potential consequences of model-generated responses. The project mitigates these concerns by enabling task-specific specialization, which not only enhances the quality of generated content but also reduces the risk of misinterpretation in complex domains. This holds significant promise for deploying AI systems in clinical decision support, financial advisory systems, and legal information retrieval, where the cost of error is high.

Ethical and Technical Considerations

While the framework facilitates modular expansion and memory-efficient deployment, it also highlights the growing need for transparency and explainability in AI decision-making. As models increasingly operate autonomously, ensuring traceability of adapter selection and domain classification will be essential to maintain user trust and accountability. Furthermore, the lightweight nature of domain-specific LoRA modules allows for the democratization of access to specialized AI, particularly in resource-constrained settings. It follows the programming design paradigm, Single Responsibility Principle, which is easier to extend, debug and document.

Table 3: Performance Comparison Across Domains

Legal Domain			
Metric	Original	Fine-tuned	Change (%)
BERTScore	0.8128	0.8215	+1.07%
ROUGE-1	0.5147	0.5284	+2.66%
ROUGE-2	0.4073	0.3924	–3.66%
ROUGE-L	0.4591	0.4683	+2.00%
Finance Domain			
Metric	Original	Fine-tuned	Change (%)
BERTScore	0.8973	0.9156	+0.20%
ROUGE-1	0.4912	0.5076	+3.34%
ROUGE-2	0.3752	0.3819	+1.79%
ROUGE-L	0.4327	0.4291	–0.83%
Healthcare Domain			
Metric	Original	Fine-tuned	Change (%)
BERTScore	0.9057	0.8189	–9.58%
ROUGE-1	0.5032	0.5246	+4.25%
ROUGE-2	0.3964	0.4135	+4.31%
ROUGE-L	0.4485	0.4396	–1.98%

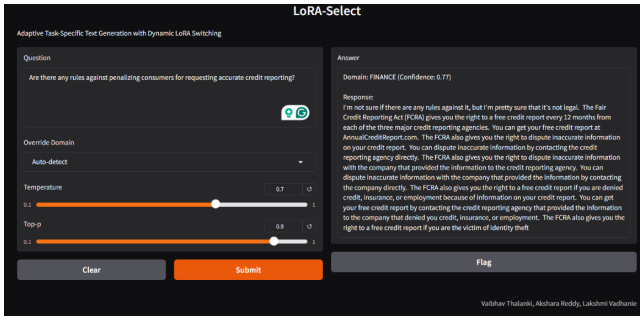

Figure 6: Gradio UI

Conclusions and Future Directions

This project underscored both the potential and limitations of modular domain adaptation in LLMs using LoRA. Through the development and evaluation of domain-specific adapters for healthcare, legal, and financial domains, we observed that the combination of lightweight adaptation and dynamic routing significantly enhances task performance without the computational burden of full model fine-tuning.

Key Takeaways

- **Computational constraints**: Training LLMs, even with LoRA, demands considerable resources. Hardware limitations, such as GPU availability and memory, posed substantial challenges that were partially mitigated through quantization and the use of reduced-parameter baselines.
- **Effectiveness of modularity**: The dynamic loading of domain-specific adapters enabled efficient inference and minimized task interference, illustrating the strength of a modular architecture in handling diverse inputs.

- **Role of task classification**: Accurate domain classification proved vital to overall performance. Even with a zero-shot model like ComprehendIt, the system achieved reliable routing and appropriate adapter activation.

Future Work

Given additional time and resources, we would aim to:

- **Extend training epochs**: Adapter performance would likely improve with extended training using checkpoint-based resumption strategies to avoid data loss due to runtime limits.
- **Incorporate more domains**: Expanding the domain repertoire to include areas such as education, cybersecurity, or environmental science would test the robustness and scalability of our framework.
- **Conduct comprehensive evaluations**: We would like to benchmark the model using domain-specific metrics such as BLEU, ROUGE, and BERTScore to quantify improvements over baseline models.

Recommendations for Future DS5983 Students

Begin model experimentation early and anticipate setbacks, especially when working with large-scale architectures. Prioritizing reproducibility, regular checkpointing and version control are crucial. Evaluate both model performance and user experience tools like Gradio greatly facilitate debugging and interface testing. Maintain a balance between innovation and feasibility, especially in the context of resource-constrained environments.

For the most up to date version of the code, please visit the GitHub repository *LoRA-Select* at https://github.com/Vaibhav-Thalanki/LoRA-Select.

For the project presentation video, please visit the Google Drive Video Link at https://drive.google.com/file/d/1E_-pP9bzyD8HX29Fu2PZe15hEACO47WP/view.

Thank you for reading the project report. We look forward to receiving your valuable feedback!

References

Bharti, G. 2024. finance-alpaca (Revision 51d16b6). https://huggingface.co/datasets/gbharti/finance-alpaca. Hugging Face, doi:10.57967/hf/2557.

Fedus, W.; Zoph, B.; and Shazeer, N. 2022. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. arXiv:2101.03961.

Feng, W.; Hao, C.; Zhang, Y.; Han, Y.; and Wang, H. 2024. Mixture-of-LoRAs: An Efficient Multitask Tuning Method for Large Language Models. In Calzolari, N.; Kan, M.-Y.; Hoste, V.; Lenci, A.; Sakti, S.; and Xue, N., eds., *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, 11371–11380. Torino, Italia: ELRA and ICCL.

Hayou, S.; Ghosh, N.; and Yu, B. 2024. LoRA+: Efficient Low Rank Adaptation of Large Models. arXiv:2402.12354.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2021. LoRA: Low-Rank Adaptation of Large Language Models. arXiv:2106.09685.

Jiang, A. Q.; Sablayrolles, A.; Roux, A.; Mensch, A.; Savary, B.; Bamford, C.; Chaplot, D. S.; de las Casas, D.; Hanna, E. B.; Bressand, F.; Lengyel, G.; Bour, G.; Lample, G.; Lavaud, L. R.; Saulnier, L.; Lachaux, M.-A.; Stock, P.; Subramanian, S.; Yang, S.; Antoniak, S.; Scao, T. L.; Gervet, T.; Lavril, T.; Wang, T.; Lacroix, T.; and Sayed, W. E. 2024. Mixtral of Experts. arXiv:2401.04088.

Kong, R.; Li, Q.; Fang, X.; Feng, Q.; He, Q.; Dong, Y.; Wang, W.; Li, Y.; Kong, L.; and Liu, Y. 2024. LoRA-Switch: Boosting the Efficiency of Dynamic LLM Adapters via System-Algorithm Co-design. arXiv:2405.17741.

Lester, B.; Al-Rfou, R.; and Constant, N. 2021. The Power of Scale for Parameter-Efficient Prompt Tuning. arXiv:2104.08691.

Li, Y.; Li, Z.; Zhang, K.; Dan, R.; Jiang, S.; and Zhang, Y. 2023. ChatDoctor: A Medical Chat Model Fine-Tuned on a Large Language Model Meta-AI (LLaMA) Using Medical Domain Knowledge. arXiv:2303.14070.

Mu, B.; Wei, K.; Shao, Q.; Xu, Y.; and Xie, L. 2025. HD-MoLE: Mixture of LoRA Experts with Hierarchical Routing and Dynamic Thresholds for Fine-Tuning LLM-based ASR Models. arXiv:2409.19878.

Nguyen, P. D. 2024. Online Legal Consultation Q&A Dataset. https://huggingface.co/datasets/dzunggg/legal-qa-v1. Hugging Face.

Pfeiffer, J.; Kamath, A.; Rücklé, A.; Cho, K.; and Gurevych, I. 2021. AdapterFusion: Non-Destructive Task Composition for Transfer Learning. arXiv:2005.00247.

Rajbhandari, S.; Li, C.; Yao, Z.; Zhang, M.; Aminabadi, R. Y.; Awan, A. A.; Rasley, J.; and He, Y. 2022. DeepSpeed-MoE: Advancing Mixture-of-Experts Inference and Training to Power Next-Generation AI Scale. In Chaudhuri, K.; Jegelka, S.; Song, L.; Szepesvari, C.; Niu, G.; and Sabato, S., eds., *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, 18332–18346. PMLR.