CoDES: A Context-Efficient Framework for Enhancing Small Language Models via Domain-Specific Adaptation and Model Ensembling

Lan Hu [*,†] , Yuting Xin [†] , Binqi Shen , Hanyu Cai , Lier Jin

Posted Date: 16 March 2026

doi: 10.20944/preprints202603.1152.v1

Keywords: large language models (LLMs); parameter-efficient fine-tuning; low-rank adaptation (LoRA); model ensembling; domain adaptation

 doi:10.20944/preprints202603.1152.v1

Article

CoDES: A Context-Efficient Framework for Enhancing Small Language Models via Domain-Specific Adaptation and Model Ensembling

Lan Hu [1,*,†], Yuting Xin [2,†], Binqi Shen [3], Hanyu Cai [3] and Lier Jin [4]

[1] Carnegie Mellon University, USA;
[2] University of Minnesota, USA
[3] Northwestern University, USA
[4] Duke University, USA
* Correspondence: lanh@alumni.cmu.edu
† Equal contribution.

Abstract

Efficiently adapting large language models (LLMs) to specialized domains remains challenging due to substantial computational and memory requirements. In this work, we introduce CoDES (Context-efficient Domain Ensemble System), a framework designed to enhance small language models through context-efficient domain adaptation and weighted parameter ensembling. CoDES integrates context-specific fine-tuning, parameter-efficient adaptation using Low-Rank Adaptation (LoRA), and completion-only supervision to focus training on answer tokens while preserving pre-trained capabilities and reducing computational cost. To further improve performance and robustness, the framework combines multiple fine-tuned models through weighted parameter ensembling. We evaluate CoDES on biomedical multiple-choice question answering using the MedMCQA benchmark. Experimental results show that the ensemble of tuned small models achieves 74.8% accuracy, approaching the performance of a much larger 72B-parameter model (77.1%). While requiring substantially fewer computational resources. The proposed framework offers several practical advantages, including achieving comparable performance, lower energy consumption, faster inference, and flexible adaptation to specialized domains. By reducing the reliance on extremely large models, CoDES provides a scalable and resource-efficient pathway for deploying high-performing language model systems in knowledge-intensive environments where models must be frequently updated with evolving domain information.

Keywords: large language models (LLMs); parameter-efficient fine-tuning; low-rank adaptation (LoRA), model ensembling; domain adaptation

1. Introduction

Large language models (LLMs) have rapidly expanded the scope of what is possible in natural language processing, evolving at an unprecedented pace [1]. Modern state-of-the-art models demonstrate remarkable capabilities across a wide range of tasks, including reasoning, knowledge retrieval, and complex language understanding. However, these performance gains are often accompanied by substantial computational, environmental, and financial costs due to the scale and resource requirements of large models [2–4]. As organizations increasingly seek to deploy LLM-based systems in practical environments, selecting an appropriate model size has become a critical design decision [5].

In response to these challenges, researchers have begun exploring smaller language models that require significantly fewer computational resources. Typically ranging from a few million to several billion parameters, small language models offer faster training cycles, lower inference latency, and more accessible deployment compared with their larger counterparts [6]. These characteristics make them

 doi:10.20944/preprints202603.1152.v1

particularly attractive for applications requiring real-time responses or operating under constrained hardware environments [7,8]. As a result, small language models are increasingly considered for practical industrial deployments where scalability and cost efficiency are important factors [9,10].

A significant performance gap often remains between small language models and large LLMs [11] despite the above advantages. While large language models often improve performance through scaling [12], smaller models must rely on more efficient training strategies, targeted datasets, and specialized adaptation techniques to narrow the performance gap [13,14]. Although recent studies have explored parameter-efficient training techniques and domain-specific adaptation, there remains limited empirical evidence demonstrating whether carefully optimized small models can achieve performance comparable to large-scale LLMs in complex reasoning tasks. Research has shown that model performance can be significantly improved through carefully designed training frameworks and architectural strategies [15–20].

To address the performance limitation of small language models, we propose CoDES (Context-efficient Domain Ensemble System), a framework that enhances small language model performance through targeted domain adaptation and weighted parameter ensembling. Instead of relying on larger model scales, our approach focuses on improving how compact models are trained and utilized within specialized domains. By integrating parameter-efficient training strategies with model ensembling, the framework aims to narrow the performance gap between small language models and large LLMs while maintaining significantly lower computational requirements. To evaluate the proposed framework, we conduct experiments on the MedMCQA [21] biomedical question answering dataset, a domain well suited for examining the reasoning limitations of small language models.

The main contributions of this work are summarized as follows:

- We propose CoDES, a context-efficient framework for improving small language model performance through domain-specific adaptation.
- We design a training pipeline that combines parameter-efficient fine-tuning with structured conversational preprocessing and completion-only supervision.
- We introduce a weighted parameter ensembling strategy that combines multiple fine-tuned models to further narrow the performance gap with large language models.
- Through experiments on biomedical question answering tasks, we demonstrate that our framework enables small models to approach the performance of substantially larger models while requiring significantly fewer computational resources.

2. Related Work

2.1. Domain Specialization

Existing work has begun to examine domain-specific modeling for high-stakes applications, showing that structured inputs and domain signals improve accuracy and interpretability; however, systematic studies that bring together compact model tuning, calibrated probabilistic outputs, and deployment constraints for the biomedical field remain limited [22].

2.2. Parameter- and Compute-efficient Adaptation

Adapting pretrained transformers under hardware and cost constraints typically updates only a small set of parameters or uses low-precision weight formats. Techniques such as low-rank adapters (LoRA) [23] and 4-bit/mixed-precision loading reduce memory and compute while keeping pretrained behavior, making them practical choices for fine-tuning compact models on domain data [24].

2.3. Model Ensembling and Calibration

Combining complementary small models can improve accuracy and calibration without switching to much larger models. Selective processing - routing computation to the most relevant input regions or scales - reduces wasted work and avoids introducing irrelevant or misleading background details [25]. Compact multistage fusion modules can integrate complementary feature streams before

 doi:10.20944/preprints202603.1152.v1

prediction [26]. Simple model-combination techniques such as weighted parameter averaging and light ensembling effectively blend strengths of different fine-tuned checkpoints with little extra cost; these ideas motivate our weighted parameter-averaging approach.

These strands motivated our design: we target a domain-focused biomedical setting, use 4-bit loading with LoRA to keep tuning cheap, format data as conversational templates and apply completion-only loss so fine-tuning concentrates on answer tokens, and merge two fine-tuned checkpoints by weighted parameter averaging to combine their complementary strengths.

3. Methodology

The objective of this study is to test the hypothesis that small-parameter language models, when augmented with domain-specific context in a context-efficient manner, can achieve performance comparable to large, generalized large language models (LLMs). Building on this hypothesis, we propose a context-efficient framework that is systematically evaluated in a high-impact real-world domain: medical question answering, and is designed to be easily extensible to other specialized domains.

Our framework emphasizes practical feasibility: reducing computational and GPU resource requirements while maintaining competitive performance relative to substantially larger models that may be costly, inaccessible, or impractical to deploy in resource-constrained, knowledge-intensive, and data-sensitive domains such as healthcare, education, and legal services.

3.1. Data Selection

All experiments used the MedMCQA dataset for training, context-based learning, and evaluation. The dataset consists of multiple-choice questions drawn from standardized medical entrance examinations (AIIMS and NEET-PG), covering 21 subject areas and 2.4k distinct healthcare topics.

The dataset is divided into 3 sets, which supports supervised learning while maintaining a sufficiently large held-out test set for reliable comparison:

1. Training set: 183K questions
2. Development/Validation Set: 4K questions
3. Test Set: 6K questions

MedMCQA provides several characteristics that make it particularly suitable for evaluating the proposed framework:

3.1.1. Domain Relevance to Medical LLM Evaluation

MedMCQA was selected primarily for its strong alignment with the focus of this work to test the novel framework. As large language models are increasingly applied in high-stakes and complex environments such as healthcare, evaluating model behavior in a medically grounded benchmark is important to assess whether domain-specific tuning can support reliable performance and calibrated user trust [27–29]. At the same time, medical applications often operate under practical deployment constraints and have up-to-the-minute knowledge that general LLMs lack deep, nuanced expertise for, making the domain well suited for studying whether targeted contextual tuning can compensate for reduced model capacity.

3.1.2. Benchmark Popularity and Reproducibility

The dataset is publicly available and widely adopted as a benchmark for evaluating medical question-answering systems and medical language models, including studies on domain-specialized LLMs, retrieval-augmented medical QA systems, and multilingual medical reasoning frameworks [30,31]. It thereby enables straightforward replication and transparent comparison with future work.

 doi:10.20944/preprints202603.1152.v1

3.1.3. Objective Evaluation via Multiple-Choice Format

its multiple-choice question (MCQA) format allows for clear and objective evaluation through direct accuracy measurement, avoiding the ambiguity often associated with open-ended generative responses.

3.1.4. Diverse Knowledge and Difficulty Levels

The questions span varying levels of difficulty and test diverse forms of medical knowledge, including clinical reasoning, treatment and diagnostic judgment, and conceptual explanation. With an average question length of 12.77 tokens, the dataset also limits reliance on long prompt contexts and instead places greater emphasis on the model's internal understanding of domain knowledge [21].

3.2. Model Selection

To study performance across model scales, we include both high-capacity generalized models and smaller, more computationally economical alternatives.

Large-Scale Reference Model: We chose Qwen2.5-72B as the large-model reference point [32]. The model is widely adopted across a broad set of tasks and offers strong general-language capabilities, maki ng it a practical benchmark in scenarios where domain-specialized medical models are unavailable. Its role in this study is not to represent an upper bound of medical reasoning performance, but to serve as a consistent, high-capacity comparison target.

Smaller Models: For the smaller-scale models, selection was guided by three considerations: accessibility, community adoption (or popularity), and suitability for controlled comparison.

1. Qwen2.5-14B was chosen as a reduced-scale counterpart to Qwen-72B. Sharing a similar architectural foundation allows for a more direct examination of how parameter count influences performance under comparable conditions.
2. LLaMA3.1-8B [33] was included as a compact, open-source model with strong general-language proficiency and significantly lower computational requirements. Its smaller size makes it particularly relevant for assessing whether structured contextual input can compensate for limited capacity.

For both models, we report results under two conditions:

1. Performance without task-specific adaptation (i.e., zero-shot evaluation)
2. Performance after applying context-efficient fine-tuning and augmentation.

This comparison isolates the effect of the proposed framework from that of model size alone. In addition, we examine whether combining multiple context-enhanced small models through ensemble techniques can yield further improvements. The ensemble strategy and training details are described in the following section.

3.3. Training Method

The overall methodology is illustrated in Figure 1. Our framework processes domain-specific data through targeted preprocessing, then fine-tunes small language models using parameter-efficient adaptation, which are further combined via weighted parameter ensembling. The resulting models are evaluated on downstream tasks using standard metrics such as accuracy and log loss, allowing assessment of both predictive quality and computational efficiency.

Data Preprocessing: All datasets were preprocessed to support completion-only supervision. Each data was formatted into a conversational chat template, including system and user messages with question-answer pairs suitable for instruction tuning. The sequences were then tokenized using the model tokenizer, truncated to a maximum length, and converted into input IDs compatible with the language model. Labels were generated so that only the tokens corresponding to the answer contributed to the loss, while all prompt tokens were masked.

Parameter-Efficient Model Preparation: We conducted experiments using Llama3.1-8B and Qwen2.5-14B. To reduce memory consumption, we loaded the models in 4-bit precision, lowering GPU memory requirements while preserving performance during training.

 doi:10.20944/preprints202603.1152.v1

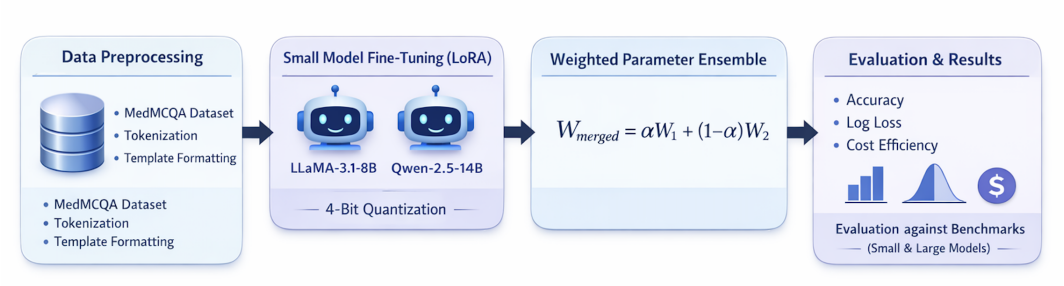


Figure 1. Overview of the Context-efficient Domain Ensemble System (CoDES) Framework.

We then applied LoRA (Low-Rank Adaptation)[23] to introduce trainable low-rank matrices into selected layers, keeping the original weights frozen. This parameter-efficient adaptation allowed the models to learn effectively with only a small subset of parameters, reducing both memory and computational demands. By fine-tuning only these additional matrices, the models retained their pretrained linguistic and reasoning capabilities while adapting efficiently to the downstream task.

Completion-Only Fine-Tuning: We fine-tuned the model using the SFTTrainer provided in the TRL (Transformer Reinforcement Learning) library from Hugging Face [34], including the AdamW optimizer, mixed-precision training, and gradient accumulation. Training hyperparameters such as batch size, learning rate, and number of epochs were chosen based on preliminary experiments and adjusted for each dataset and model to ensure stable convergence. The loss was computed only on the answer portion of each sequence, excluding prompt tokens.

By restricting updates to the answer tokens, the model learned to map context to the correct response while retaining its pretrained capabilities for understanding and reasoning. This approach reduced unnecessary gradient noise, sped up convergence, and improved the model's ability to produce concise, discrete answers in downstream tasks.

Ensemble Strategy: Research has shown that model ensembling is a straightforward and effective technique for model enhancement [35,36]. To further improve performance, we combined two of the best-performing fine-tuned small-parameter models using a weighted average of their parameters, with a scaling factor controlling each model's contribution. Different weight configurations were tested to find the combination that produced the most accurate predictions, allowing the ensemble to benefit from the strengths of both models and achieve better results than either model alone.

3.4. Evaluation Metrics

For each question in the validation set, the model predicted a probability distribution over the four possible answer choices, A, B, C, and D. Let $\mathbf{p}_i = [p_{i,A}, p_{i,B}, p_{i,C}, p_{i,D}]$ denote the predicted probabilities for question i, and let y_i denote the correct answer.

The performance of the models was evaluated using accuracy and log loss.

Accuracy

measures the proportion of questions for which the predicted choice matched the correct answer:

$$\text{Accuracy} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\left[\arg\max_c p_{i,c} = y_i\right].$$

Log Loss

also known as cross-entropy loss, measures the quality of probabilistic predictions and is defined as:

$$L(y,p) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c\in\{A,B,C,D\}} \mathbf{1}[y_i = c]\log(p_{i,c}),$$

where:

 doi:10.20944/preprints202603.1152.v1

- N is the total number of evaluated questions,
- c indexes the answer choices,
- y_i is the correct label for question i,
- $p_{i,c}$ is the predicted probability for choice c of question i,
- $\mathbf{1}[\cdot]$ is the indicator function.

Models that assign high probability to incorrect answers result in a heavier penalty. Consequently, when two models have the same accuracy, the model with lower log loss exhibits more consistent and better-calibrated predictive behavior.

Examples with invalid or non-finite probabilities were excluded from evaluation. These metrics together captured both the correctness and the confidence of the model's predictions.

4. Experiments & Results

Building on the dataset preparation, model selection, and training techniques detailed in the Methodology section, we conducted a set of experiments aiming to improve performance of small language models.

4.1. Baseline Performance

To establish a baseline performance, we conducted zero-shot evaluation on one large language model and two smaller models on an identical dataset consisting of 10,000 randomly-selected examples from MedMCQA dataset. Specifically, we compare Qwen2.5-72B, LLaMA 3.1-8B, and Qwen2.5-14B under the same evaluation protocol. The results are reported in Table 1.

Table 1. Baseline Model Performance Comparison.

Baseline Model	Sample Size	Accuracy
Qwen2.5-72B	10k	77.1%
Llama3.1-8B	10k	63.5%
Qwen2.5-14B	10k	64.0%

The Qwen2.5-72B model achieves the highest accuracy of 77.1%, whereas Llama 3.1-8B and Qwen2.5-14B achieve 63.5% and 64.0%, respectively. This performance gap is attributable to the increased parameter capacity of large LLMs, which enables more expressive representations and improved generalization. The 77.1% accuracy achieved by the large model serves as a baseline reference and will be used for comparison in subsequent sections.

4.2. Small Model Fine Tuning with LoRA

In this section, we apply Low-Rank Adaptation (LoRA) to fine-tune the two small language models using a 10,000 examples randomly selected from the MedMCQA training dataset and a validation dataset of 3,000 examples. The accuracy is based on performance on the evaluation dataset.

LoRA introduces several hyperparameters that govern the fine-tuning process. Given the large hyperparameter search space, we focus on the learning rate and the number of training epochs, as these factors have the most significant impacts on evaluation metrics from early experiments.

Table 2 presents the performance of Llama3.1-8B and Qwen2.5-14B under different LoRA fine-tuning configurations. The results show that moderate learning rates around $5e^{-5}$ consistently produce the best performance across both models, while very small learning rates ($2e^{-5}$) lead to slower learning and reduced accuracy. Increasing the number of training epochs from one to two also improves performance, suggesting that additional exposure to domain-specific data enables more effective parameter adaptation. Furthermore, improvements in accuracy are accompanied by reductions in log loss, indicating better calibration of model predictions.

 doi:10.20944/preprints202603.1152.v1

Table 2. Performance of Llama3.1-8B and Qwen2.5-14B under different LoRA fine-tuning hyperparameter configurations.

Model	Learning Rate	Epoch	Accuracy	Log Loss
Llama3.1-8B[1]	$1e^{-4}$	1	72.5%	1.09
Llama3.1-8B[2]	$5e^{-5}$	1	70.6%	1.10
Llama3.1-8B[3]	$5e^{-5}$	2	73.2%	1.09
Llama3.1-8B[4]	$2e^{-5}$	1	63.6%	1.19
Llama3.1-8B[5]	$2e^{-5}$	2	68.5%	1.14
Qwen2.5-14B[1]	$1e^{-4}$	1	63.7%	1.19
Qwen2.5-14B[2]	$5e^{-5}$	2	69.5%	1.12
Qwen2.5-14B[3]	$2e^{-5}$	1	63.5%	1.19

The highest achieved accuracies increased by 9.7% and 5.5% for Llama3.1-8B and Qwen2.5-14B compared with the baseline accuracies shown in TABLE I, respectively. The results demonstrate that, with appropriate hyperparameter selection, small models can attain substantial performance gains through domain-specific context learning. The Llama3.1-8B model achieves higher accuracy than the Qwen2.5-14B model, indicating that Llama3.1-8B exhibits superior adaptability to LoRA-based fine-tuning.

4.3. Ensemble of Small Models v.s. Large model

We ensembled two best performed fine tuned model (Llama3.1-8B[3] and Qwen2.5-14B[2]) by taking weighted averaging of their parameters. The formula for ensembling two models is as below, where alpha is a scaling factor that determines the contribution of each model.

$$W_{\text{merged}} = \alpha W_1 + (1 - \alpha) W_2$$

Table 3. Performance of weighted parameter ensembles combining fine-tuned Llama3.1-8B and Qwen2.5-14B models.

Llama Weight	Qwen Weight	Accuracy	Log Loss
0.80	0.20	73.4%	0.70
0.70	0.30	74.1%	0.68
0.65	**0.35**	**74.8%**	**0.66**
0.60	0.40	74.3%	0.67
0.50	0.50	73.8%	0.68

In Table 3, we report the performance of ensemble model with different weights. The results show that, despite performances vary differently, all ensemble configurations show improved accuracy and lower log loss compared to a single model. When compared with the baseline large model, performance peaks at the weight configuration of 0.65 / 0.35, after which further increases in the Qwen weight lead to slight declines in both accuracy and calibration. The best-performing ensemble model achieves an accuracy of 74.8%, which approaches the performance of the baseline large model, Qwen2.5-72B (77.1%).

Another observation is that the ensemble model performances remain relatively stable across a range of weight configurations. Accuracy ranges from 73.4% and 74.8% with difference of 1.4% , indicating that the ensemble approach is robust to moderate changes in the weight allocation between the two models. All ensemble weight configurations listed in Table 3 yield higher accuracy and lower log loss compared with the individual Llama3.1-8B and Qwen2.5-14B models. This improvement suggests that the two fine-tuned models capture complementary patterns in the data. By combining their parameters through weighted averaging, the ensemble is able to leverage strengths from both models, resulting in improved predictive performance.

 doi:10.20944/preprints202603.1152.v1

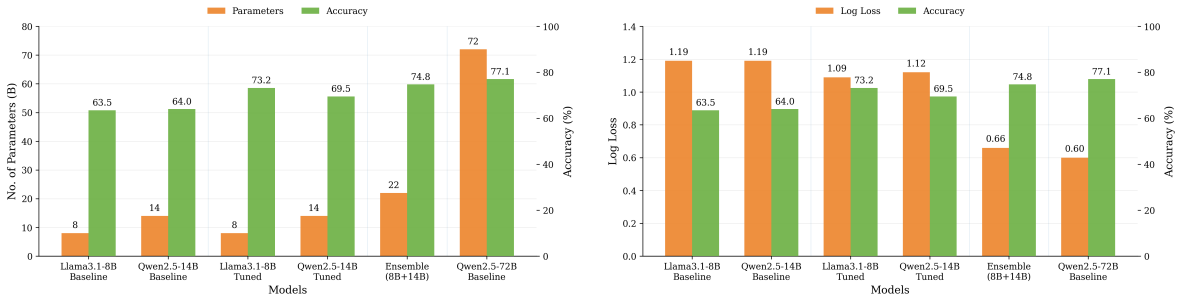
In addition to accuracy gains, the ensemble also produces lower log loss values. The log loss of the ensemble model ranges from 0.66 to 0.70, compared with the individual model log loss values between 1.09 and 1.19. This reduction indicates better calibration of predicted probabilities, suggesting that the ensemble not only improves correctness but also produces more reliable confidence estimates.

4.4. Practical Implications of the Framework

We validate the feasibility and effectiveness of the newly proposed framework that, combining context-specific tuning, parameter-efficient adaptation, and model ensembling can enable small models to achieve competitive performance while maintaining practical advantages in deployment.

In particular, the framework offers the following practical advantages over deploying a single large-parameter model:

- **Comparable Performance:** The experimental results in Figure 2 further demonstrate that context-specific tuning can substantially improve the performance of relatively small language models and significantly narrow the gap with much larger models. For instance, the Llama3.1 8B model achieved 73.2% accuracy after contextual tuning, compared with a baseline accuracy of 63.5%. Similarly, the Qwen2.5 14B model improved from an initial baseline of 64.0% to 69.5% accuracy. When combined using the ensemble strategy within the proposed framework, the tuned small models achieved 74.8% accuracy, approaching the benchmark performance of the much larger Qwen2.5 72B model, which achieved 77.1% accuracy. These results highlight that carefully incorporating domain-relevant contextual information can significantly enhance model capability without relying solely on scaling model size.



(**a**) Accuracy vs. Parameter Size (**b**) Accuracy vs. Log Loss

Figure 2. Performance comparison across baseline, fine-tuned, and ensemble model configurations.

- **Lower Cost and Resource Requirements:** The framework enables high performance while operating with substantially smaller models. Training and inference with small language models require significantly fewer computational resources due to reduced parameter counts. Table 4 shows that the Qwen2.5-72B model consumes approximately 0.2 kWh of energy to analyze 10k multiple-choice questions, which is about seven times more than the 8B model and four times more than the 14B model. Even when using an ensemble of small models, the large model still consumes approximately 2.5 times more energy. This highlights the potential of the proposed framework to reduce operational costs and energy consumption.

Table 4. Model Energy Consumption Comparison Analyzing 10K Questions.

Model	Energy Consumption (kWh)
Qwen2.5-72B	0.20
Llama3.1-8B	0.03
Qwen2.5-14B	0.05
Qwen2.5-14B + Llama3.1-8B	0.08

 doi:10.20944/preprints202603.1152.v1

- **Faster Inference Performance:** Smaller models also enable faster response times. Large models require substantially more matrix multiplications and memory transfers during inference, leading to higher latency. By leveraging multiple tuned small models within the proposed framework, systems can maintain strong predictive performance while achieving faster inference speeds, which is particularly beneficial for real-time or large-scale applications.
- **Flexible Domain Customization:** The framework facilitates efficient adaptation of models to specialized domains. Through experiments on the MedMCQA dataset, we demonstrate that small models can be effectively tailored to domain-specific tasks through contextual fine-tuning. This modular approach allows organizations to customize models for specific knowledge domains without the need to train or deploy extremely large models.

Overall, the results suggest that the proposed framework provides a practical pathway for deploying high-performing language model systems under realistic computational and resource constraints while enabling scalable and sustainable AI adoption across organizations with varying levels of computational resources.

Such an approach is particularly important in knowledge-intensive domains where information evolves rapidly, including healthcare, finance, and legal. In these settings, organizations must frequently update models with new domain knowledge, and the ability to efficiently adapt smaller models through contextual tuning provides a practical and scalable pathway for maintaining accurate and up-to-date AI systems.

5. Conclusions

This work investigated whether language models with relatively modest parameter counts can achieve strong performance when supplied with domain-specific context in a targeted and efficient way. We designed CoDES, a framework that combines context-specific fine-tuning, parameter-efficient adaptation, and model ensembling, leveraging the complementary strengths of multiple tuned models while emphasizing effective use of contextual information rather than relying solely on model scale.

Our experiments focused on biomedical multiple-choice question answering. The results show that carefully tuned small models can achieve competitive accuracy and low loss while using substantially fewer parameters than larger models, highlighting the practical potential of the approach. In particular, context-specific tuning and ensemble strategies allow small models to scale effectively under realistic computational resource constraints. This suggests that for organizations or tasks targeting specific domains, leveraging small models with ensemble learning can provide strong performance without requiring the computational resources of very large models.

While our evaluation focused on biomedical multiple-choice question answering using a limited set of model architectures and parameter sizes, the framework's applicability to other domains remains to be fully explored. Future work will extend the approach to additional domains such as legal and finance industries that require accurate and context-aware reasoning, and to Non-Profits and small organizations with limited computational resources, applying the framework to solve the real-world problems. We also plan to explore broader domain-level tuning, incorporate more model families and sizes, and investigate open-ended or generative question answering tasks. Integrating retrieval-augmented context and analyzing the trade-offs between computational cost and performance will further enhance the framework's practical utility and scalability across diverse applications.

References

1. Tariq Shahzad, Tehseen Mazhar, Muhammad Usman Tariq, Wasim Ahmad, Khmaies Ouahada, and Habib Hamam. A comprehensive review of large language models: Issues and solutions in learning environments. *Discov. Sustain.*, 6(1), January 2025.
2. Maximilian Dauner and Gudrun Socher. Energy costs of communicating with ai. *Frontiers in Communication*, Volume 10 - 2025, 2025.
3. Xinjin Li, Yu Ma, Yangchen Huang, Xingqi Wang, Yuzhen Lin, and Chenxi Zhang. Synergized data efficiency and compression (sec) optimization for large language models. In *2024 4th International Con-*

 doi:10.20944/preprints202603.1152.v1

ference on Electronic Information Engineering and Computer Science (EIECS), pages 586–591, 2024. doi: 10.1109/EIECS63941.2024.10800533.

4. Chao Wu, Baoheng Li, Mingchen Gao, and Zhenyi Wang. From efficiency to adaptivity: A deeper look at adaptive reasoning in large language models, 2025. doi:10.48550/arXiv.2511.10788.
5. Peter Belcak, Greg Heinrich, Shizhe Diao, Yonggan Fu, Xin Dong, Saurav Muralidharan, Yingyan Celine Lin, and Pavlo Molchanov. Small language models are the future of agentic ai, 2025.
6. Fali Wang, Zhiwei Zhang, Xianren Zhang, Zongyu Wu, TzuHao Mo, Qiuhao Lu, Wanjing Wang, Rui Li, Junjie Xu, Xianfeng Tang, Qi He, Yao Ma, Ming Huang, and Suhang Wang. A comprehensive survey of small language models in the era of large language models: Techniques, enhancements, applications, collaboration with llms, and trustworthiness. *ACM Trans. Intell. Syst. Technol.*, 16(6), November 2025.
7. Ismail Lamaakal, Yassine Maleh, Khalid El Makkaoui, Ibrahim Ouahbi, Paweł Pławiak, Osama Alfarraj, May Almousa, and Ahmed A Abd El-Latif. Tiny language models for automation and control: Overview, potential applications, and future research directions. *Sensors (Basel)*, 25(5):1318, February 2025.
8. Jiantong Jiang, Peiyu Yang, Rui Zhang, and Feng Liu. Towards efficient large language model serving: A survey on system-aware kv cache optimization. *Authorea Preprints*, 2025. doi:10.36227/techrxiv.176046306.66521015/v3.
9. Hyunjae Kim, Hyeon Hwang, Jiwoo Lee, Sihyeon Park, Dain Kim, Taewhoo Lee, Chanwoong Yoon, Jiwoong Sohn, Jungwoo Park, Olga Reykhart, Thomas Fetherston, Donghee Choi, Soo Heon Kwak, Qingyu Chen, and Jaewoo Kang. Small language models learn enhanced reasoning skills from medical textbooks. *NPJ Digit. Med.*, 8(1):240, May 2025.
10. Lexin Zhou, Wout Schellaert, Fernando Martínez-Plumed, Yael Moros-Daval, Cèsar Ferri, and José Hernández-Orallo. Larger and more instructable language models become less reliable. *Nature*, 634(8032):61–68, 2024.
11. Martin Juan José Bucher and Marco Martini. Fine-tuned'small'llms (still) significantly outperform zero-shot generative ai models in text classification. *arXiv preprint arXiv:2406.08660*, 2024.
12. Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.
13. Jinming Xing, Dongwen Luo, Chang Xue, and Ruilin Xing. Comparative analysis of pooling mechanisms in llms: A sentiment analysis perspective, 2025. doi: 10.48550/arXiv.2411.14654.
14. Chaojun Xiao, Jie Cai, Weilin Zhao, Biyuan Lin, Guoyang Zeng, Jie Zhou, Zhi Zheng, Xu Han, Zhiyuan Liu, and Maosong Sun. Densing law of LLMs. *Nature Machine Intelligence*, 7(11):1823–1833, November 2025.
15. Yisu Wang, Ruilong Wu, Xinjiao Li, and Dirk Kutscher. Pactrain: Pruning and adaptive sparse gradient compression for efficient collective communication in distributed deep learning. In *2025 62nd ACM/IEEE Design Automation Conference (DAC)*, pages 1–7, 2025. doi:10.1109/DAC63849.2025.11133419.
16. Yixiao Zhou, Ziyu Zhao, Dongzhou Cheng, Zhiliang Wu, Jie Gui, Yi Yang, Fei Wu, Yu Cheng, and Hehe Fan. Dropping experts, recombining neurons: Retraining-free pruning for sparse mixture-of-experts llms. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 15169–15186, 2025. doi:10.48550/arXiv.2509.10377.
17. Shuang Zeng, Dekang Qi, Xinyuan Chang, Feng Xiong, Shichao Xie, Xiaolong Wu, Shiyi Liang, Mu Xu, Xing Wei, and Ning Guo. Janusvln: Decoupling semantics and spatiality with dual implicit memory for vision-language navigation, 2026. doi:10.48550/arXiv.2509.22548.
18. Zhangquan Chen, Manyuan Zhang, Xinlei Yu, Xufang Luo, Mingze Sun, Zihao Pan, Yan Feng, Peng Pei, Xunliang Cai, and Ruqi Huang. Think with 3d: Geometric imagination grounded spatial reasoning from limited views, 2025. doi:10.48550/arXiv.2510.18632.
19. Ruihan Luo, Xuanjing Chen, and Ziyang Ding. Sequda-rec: Sequential user behavior enhanced recommendation via global unsupervised data augmentation for personalized content marketing. *arXiv preprint arXiv:2509.17361*, 2025. doi:10.48550/arXiv.2509.17361.
20. Jiahao Tian, Zhenkai Wang, Jinman Zhao, and Zhicheng Ding. Mmrec: Llm based multi-modal recommender system. In *2024 19th International Workshop on Semantic and Social Media Adaptation & Personalization (SMAP)*, pages 105–110. IEEE, 2024. doi:10.48550/arXiv.2408.04211.
21. Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. Medmcqa : A large-scale multi-subject multi-choice dataset for medical domain question answering, 2022.
22. Dingyuan Liu, Qiannan Shen, and Jiaci Liu. The health-wealth gradient in labor markets: Integrating health, insurance, and social metrics to predict employment density. *Computation*, 14(1):22, 2026. doi:10.3390/computation14010022.

 doi:10.20944/preprints202603.1152.v1

23. Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021.
24. Tianyi Zhang, Kishore Kasichainula, Yaoxin Zhuo, Baoxin Li, Jae-Sun Seo, and Yu Cao. Transformer-based selective super-resolution for efficient image refinement. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 7305–7313, 2024. doi:10.48550/arXiv.2312.05803.
25. Qiannan Shen and Jing Zhang. Mftformer: Meteorological-frequency-temporal transformer with block-aligned fusion for traffic flow prediction. *Research Square*, 2026. Preprint, doi:10.21203/rs.3.rs-8770196/v1.
26. Qiang Wang, Feng Liu, Bo Zhang, Jiaci Liu, Fang Xu, and Yifan Wang. Siamctca: Cross-temporal correlation aggregation siamese network for uav tracking. *Drones*, 9(4):294, 2025. doi:10.3390/drones9040294.
27. Dandan Wang and Shiqing Zhang. Large language models in medical and healthcare fields: Applications, advances, and challenges. *Artificial Intelligence Review*, 2024.
28. Ziheng Chen, Chuqing Zhao, Mengyu Zhao, Qishi Zhan, Yushen Dong, Siqiao Zhao, Sijing Yu, Chenxi Yao, Yongyu Xie, Zhikang Dong, Qianyi Sun, Yunbo Liu, Cheng-Han Yu, Haochen Yang, and Guansu Wang. Rethinking subjective trust in llm: Actualizing tangibility from uncertainty. December 2025. doi: 10.36227/techrxiv.176463545.53813864/v1.
29. Jinhua Yang, Ting Liu, Yiming Taclis Luo, Tianyue Niu, Patrick Cheong-Iao Pang, Ao Xiang, and Qin Yang. Exploring the application boundaries of llms in mental health: A systematic scoping review. *Frontiers in Psychology*, 16:1715306, 2025. doi: 10.1109/ACCESS.2024.3406469.
30. Augustin Toma, Patrick R. Lawler, Jimmy Ba, Rahul G. Krishnan, Barry B. Rubin, and Bo Wang. Clinical camel: An open expert-level medical language model with dialogue-based knowledge encoding. *arXiv preprint arXiv:2305.12031*, 2023.
31. Tasnimul Hassan, Md Faisal Karim, Haziq Jeelani, Elham Behnam, Robert Green, and Fayeq Jeelani Syed. Optimizing medical question-answering systems: A comparative study of fine-tuned and zero-shot large language models with rag framework. *arXiv preprint arXiv:2512.05863*, 2025.
32. 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. Qwen2 technical report, 2024.
33. Meta AI. Introducing llama 3.1: Open source ai models. https://ai.meta.com/blog/meta-llama-3-1/, 2024. Accessed: 2026-03-07.
34. Younes Belkada and Hugging Face Contributors. Trl: Transformer reinforcement learning library. https://huggingface.co/docs/trl/sft_trainer, 2023. Accessed: 2026.
35. Enneng Yang, Li Shen, Guibing Guo, Xingwei Wang, Xiaochun Cao, Jie Zhang, and Dacheng Tao. Model merging in llms, mllms, and beyond: Methods, theories, applications and opportunities, 2025.
36. Jiahao Tian and Zhenkai Wang. Dlrrec: Denoising latent representations via multi-modal knowledge fusion in deep recommender systems. *arXiv preprint arXiv:2512.00596*, 2025. doi:10.48550/arXiv.2512.00596.