
Position: Enough of Scaling LLMs! Lets Focus on *Downscaling*

Yash Goel^{* 1} Ayan Sengupta^{* 1} Tanmoy Chakraborty¹

Abstract

We challenge the dominant focus on neural scaling laws and advocate for a paradigm shift toward *downscaling* in the development of large language models (LLMs). While scaling laws have provided critical insights into performance improvements through increasing model and dataset size, we emphasize the significant limitations of this approach, particularly in terms of computational inefficiency, environmental impact, and deployment constraints. To address these challenges, we propose a holistic framework for downscaling LLMs that seeks to maintain performance while drastically reducing resource demands. This paper outlines practical strategies for transitioning away from traditional scaling paradigms, advocating for a more sustainable, efficient, and accessible approach to LLM development.

1. Introduction

The development of neural scaling laws has provided a foundational framework for understanding the performance trajectory of large language models (LLMs). These laws (Kaplun et al., 2020; Hoffmann et al., 2022) describe how model performance improves predictably with increases in parameters, dataset size, and compute resources. Initially, scaling laws seemed to offer a clear roadmap for the continuous and predictable advancement of LLMs. However, as the field has evolved — evidenced by the sharp rise in the number of proposed scaling laws between 2020 and 2024, as shown in Figure 1 — significant limitations and challenges have become apparent. These emerging concerns cast doubt on the long-term viability and effectiveness of scaling laws as the primary strategy for advancing AI.

One of the central criticisms of neural scaling laws lies in

^{*}Equal contribution ¹Indian Institute of Technology Delhi, India. Correspondence to: Ayan Sengupta <ayan.sengupta@ee.iitd.ac.in>, Yash Goel <ee1210984@ee.iitd.ac.in>.

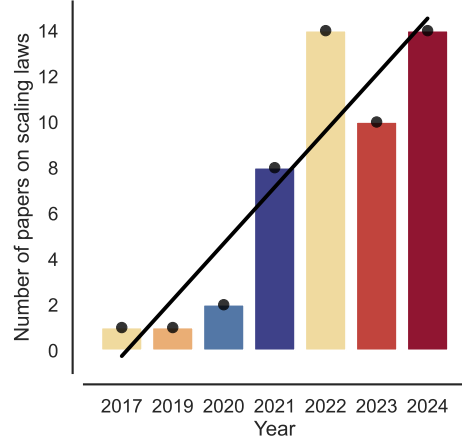


Figure 1. The growth of the number of papers on scaling laws for neural models over the past eight years.

their reliance on simplified power law relationships to predict model performance. While these laws capture broad trends, they often fail to account for nuanced factors that influence real-world outcomes. For instance, the assumption that increasing dataset size or compute will indefinitely yield proportional performance improvements ignores diminishing returns observed in practice (Diaz & Madaio, 2024). As models scale, the marginal gains from additional data and compute tend to decrease, leading to inefficient resource allocation (Muennighoff et al., 2023).

Another limitation of neural scaling laws is their emphasis on uniform scaling across model size, data, and compute. Jin et al. (2023) demonstrated that different abilities, such as fact recall and in-context learning, degrade at different rates under pruning or downsizing. This variability suggests that scaling laws may not provide a one-size-fits-all solution and that alternative approaches are needed to address diverse performance requirements. By focusing exclusively on scaling, researchers risk missing opportunities to develop more efficient and specialized models tailored to specific applications.

Neural scaling laws also neglect the broader implications of scaling on energy consumption and environmental sustainability. The computational requirements for training large-scale models are immense, resulting in significant carbon emissions (Faiz et al., 2024). Recent studies showed

that the energy efficiency of compute-intensive tasks varies based on factors such as hardware configuration, server location, and training duration (Zhang et al., 2023). Despite these insights, scaling laws do not incorporate considerations for energy efficiency, leading to strategies that may be technically effective but environmentally unsustainable.

As the limitations of neural scaling laws become more apparent, the development of downscaling laws has emerged as a critical area of focus. These laws aim to understand how to reduce model size, dataset requirements, and compute usage while preserving performance for specific tasks. Downscaling laws are essential for promoting efficiency, sustainability, and accessibility in AI research. **In this position paper, we enlighten on the importance of focused research on downscaling laws and propose strategies for leveraging the insights from neural scaling laws for downscaling LLMs to efficient, environment-friendly and sustainable foundational models.** Smaller models guided by downscaling laws can address resource constraints, enabling broader participation in model development by reducing the financial and computational barriers. Additionally, these laws can provide insights into optimizing task-specific performance, ensuring that models can meet diverse application requirements without excessive scaling.¹

2. Neural Scaling Laws

Neural scaling laws outline how the performance of neural networks improves predictably with increases in model size, data volume, and computational resources. These relationships typically follow power law patterns, providing a framework for optimizing model development.

2.1. Scaling Laws of LLMs

The evolution of scaling laws for LLMs began with Kaplan et al. (2020), which established power law relationships (described in Equation 1) between model performance and three key factors: model size, dataset size, and compute. This study revealed that larger models are more sample-efficient, suggesting that optimal compute efficiency can be achieved by training very large models on relatively modest data. Importantly, architectural variations like depth versus width were found to have minimal impact. Subsequent refinements emerged with Hoffmann et al. (2022), known as the *Chinchilla law* (described in Equation 2). This work argued against prioritizing model size alone and demonstrated that balancing model size and training data is critical. For instance, a 70B model trained on more data outperformed larger models such as Gopher-280B, while using the same compute budget. This insight was further reinforced by Tay

et al. (2023), affirming the superiority of vanilla Transformer architectures over novel designs when scaled, explaining the industry’s reliance on standard architectures.

Caballero et al. (2023) introduced the concept of Smoothly Broken Neural Scaling Laws (BNSL), offering a more nuanced framework for understanding scaling behavior. BNSL addressed phenomena like double descent and sharp capability transitions, which traditional scaling laws failed to predict. This work highlighted the limitations of existing models in predicting performance at extreme scales, broadening our understanding of scaling dynamics across domains. In vocabulary scaling, Tao et al. (2024) discovered power law relationship between vocabulary size and model parameters. This work demonstrated that larger models requires proportionally larger vocabularies to optimize performance, challenging existing practices.

Kaplan Scaling Law

$$L(N, D) = \left[\left(\frac{N_c}{N} \right)^{\frac{\alpha_N}{\alpha_D}} + \frac{D_c}{D} \right]^{\alpha_D} \quad (1)$$

where N denotes number of model parameters, D denotes the amount of training data, and N_c , D_c , α_N , α_D are fitting constants.

Chinchilla Scaling Law

$$L(N, D) = \frac{A}{N^\alpha} + \frac{B}{D^\beta} + E \quad (2)$$

where A , B , α , β , and E are fitting constants.

The key difference between the two scaling laws is that Kaplan suggested that if the model increases $8\times$, the data set must increase by $5\times$, whereas Chinchilla suggested an equal scaling of N and D . Also, Chinchilla proposed an irreducible term in the loss which doesn’t vanish on increasing N and D .

2.2. Scaling Laws Beyond LLMs

Studies on scaling small language models have shown promising results. Hu et al. (2024) demonstrated that optimized training strategies could enable small models (1.2B – 2.4B parameters) to rival larger counterparts (7B – 13B parameters). Techniques like the Warmup-Stable-Decay scheduler and extensive use of data (192 \times parameter size) improved efficiency and performance, paving the way for deploying models effectively on edge devices.

Hernandez et al. (2021) investigated transfer learning, revealing a power law relationship between pre-training and fine-tuning. Their findings demonstrated that pre-training significantly enhances the utility of smaller fine-tuning

¹The source code of our analysis can be found at <https://github.com/LCS2-IIITD/Downscaling>.

datasets. However, the phenomenon of “ossification” posed challenges in adapting pre-trained weights, particularly in high-data regimes. Complementary studies (Abnar et al., 2021; Zhang et al., 2024a) highlighted that higher upstream accuracy does not always translate to downstream performance improvements. Notably, the FLP Method (Chen et al., 2024c) merged as a critical advancement, enabling accurate predictions of downstream performance for larger models using smaller-scale data.

Sparse architectures, mainly Mixture of Experts (MoE) models, revolutionized scaling strategies. Clark et al. (2022) highlighted the potential of sparsity in parameter utilization, enabling larger capacities without proportional compute increases. Advances in MoE granularity (Krajewski et al., 2024) refined these models, achieving compute savings up to $40\times$ compared to dense Transformers. Subsequent studies (Yun et al., 2024) balanced expert quantity and inference efficiency, providing a blueprint for optimizing training and deployment.

Inference efficiency has become a focal point of scaling research. Chen et al. (2024a) challenged simple scaling assumptions, showing that sophisticated inference strategies could enable smaller models to outperform larger ones. Techniques like REBASE demonstrated this potential, with Llemma-7B rivaling its 34B counterpart at half the computational cost (Wu et al., 2025). Sardana et al. (2024) revealed that smaller models trained longer could meet high inference demands more efficiently. This paradigm shift emphasizes balancing model scale, training duration, and inference strategies to achieve cost-effective deployments.

2.3. Criticisms of Neural Scaling Laws

While scaling laws have gained significant popularity, recent studies have highlighted key limitations and uncovered opportunities to refine and extend these frameworks. Diaz & Madaio (2024) critiqued the assumption that larger datasets inherently lead to better model performance. The authors argued that this relationship breaks down when AI systems are deployed to serve diverse human populations. As datasets expand, they increasingly encompass distinct communities with varying, and often conflicting, perspectives on what constitutes a “good” AI system. Current evaluation metrics overlook this diversity, relying on universal benchmarks that may disadvantage certain groups. Villalobos et al. (2024) explored the potential limits of scaling LLMs due to the finite availability of public human-generated text data. It provides a detailed analysis of current trends in the growth of dataset size, the stock of available human-generated data, and the implications for the continued scaling of LLMs. The authors argued that not all human-generated text data is of equal quality, and using low-quality data could hinder model performance. In their analysis, the authors also

included adjustments for data quality and multi-epoch training, revealing that careful filtering and deduplication could enhance the utility of the data stock. However, even with these strategies, they emphasized the need for more efficient data utilization to mitigate the impending data bottleneck. Sorscher et al. (2023) addressed the inefficiency of power law scaling, where marginal performance gains require exponentially more data. The authors proposed data pruning as a solution to achieve exponential improvements in model performance relative to dataset size.

$$CO2eq_{oper} = \sum_i (P_i \cdot eff_i \cdot n_i \cdot t_i) \cdot PUE \cdot c_{inten} \quad (3)$$

$$CO2eq_{emb} = \sum_i \frac{t_i \cdot area_i \cdot CPA_i}{lifetime_i} \quad (4)$$

$$CO2eq = CO2eq_{oper} + CO2eq_{emb} \quad (5)$$

where $CO2eq_{oper}$ and $CO2eq_{emb}$ denote operational and embodied carbon footprints, respectively; P_i , eff_i , n_i , t_i , $area_i$, CPA_i , and $lifetime_i$ denote the peak power, hardware efficiency, count, execution time, area, carbon emitted per unit area and lifetime of hardware unit i , respectively; PUE denotes power usage efficiency of the specific data center; and c_{inten} denotes the carbon intensity of the specific data center.

Proposition 2.1.

$$CO2eq(P, D) = (K_1 + K_2) \cdot N \cdot D \quad (6)$$

where $(K_1 + K_2)$ represents a compound constant that encapsulates all hardware, data center, and efficiency parameters. This shows that CO2 emissions scale linearly with both (i) the number of model parameters (N), and (ii) the amount of training data (D).

Computational Costs of Scaling Laws. LLMs require significant computational resources for training, inference, and storage, leading to substantial carbon emissions. These emissions can be divided into operational costs, stemming from energy usage during training and deployment, and embodied costs, which result from the manufacturing of hardware such as GPUs, TPUs, and storage devices. To address the growing computational costs of neural scaling, frameworks like LLMCarbon (Faiz et al., 2024) (See Appendix A for details) provide valuable tools for estimating and optimizing carbon footprints. We describe the estimated operational, embodied and total carbon footprint of an LLM pre-training in Equations 3, 4 and 5, respectively.

Further simplification, as shown in Equation 6, reveals that carbon footprints scale linearly with both LLM size and

pre-training token size². We empirically validate the scaling laws of carbon cost for different model and pre-training data sizes, as illustrated in Figures 2 and 3, respectively. While Kaplan scaling law suggests a logarithmic scaling of test loss with respect to model size ($L \sim N^{-0.08}$) and pre-training token size ($L \sim D^{-0.1}$), carbon cost increases linearly with both the factors. Moreover, both Kaplan and Chinchilla scaling laws suggest that test loss $L > \max(N^{-\alpha}, D^{-\beta})$, for some arbitrary constants, α and β . Therefore, the test loss of an LLM is always irreducible, *i.e.*, can not be brought down below a theoretical bound, irrespective of the model or pre-training data size.

By combining the Kaplan scaling law (Equation 1) with our CO_2 emission formula (Equation 6), we can establish a direct relationship between model performance and environmental impact: From Kaplan $L(N) \propto N^{-\alpha}$; where L is the loss and N is the number of parameters, with $\alpha \approx 0.08$. Solving for N in the Kaplan equation and substituting, we can express $N \propto L^{-1/\alpha}$. Therefore putting in Equation 6:

$$CO_2eq \propto L^{-1/\alpha} \quad (7)$$

Recent work by Chen et al. (2025) has shown that downstream performance P can be modeled as:

$$P = w_1 + w_2 \cdot L \quad (8)$$

Where w_1 and w_2 are task-specific constants. This gives us:

$$CO_2eq \propto (P)^{-1/\alpha} \quad (9)$$

Given that $\alpha \approx 0.08$, we can approximate:

$$P \propto CO_2eq^\alpha \approx CO_2eq^{0.08} \quad (10)$$

This demonstrates that performance improvements scale approximately with the 0.08 power of carbon emissions. Put differently, to achieve linear improvements in performance, carbon emissions must increase exponentially - a 10% improvement in performance would require approximately $1.1^{12.5} = 329\%$ more carbon emissions. Moreover, as the model’s performance increases, improving it further requires exponentially more computing.

3. Rise of Small and Efficient Language Models

Several converging practical and technical factors have driven the rise in popularity of Small Language Models (SLMs), which are popular for their size, efficiency, and ability to maintain competitive performance while offering cost-effective and resource-efficient solutions. Studies indicate that models like TinyLlama (Zhang et al., 2024b) (1.1B

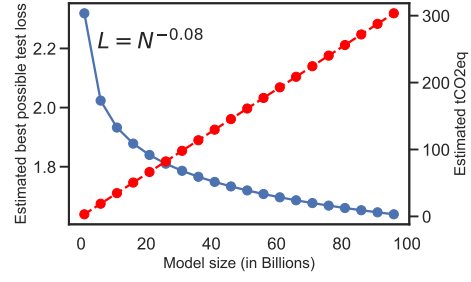


Figure 2. Test loss decreases logarithmically with model size, whereas the estimated training-time carbon emission increases linearly.

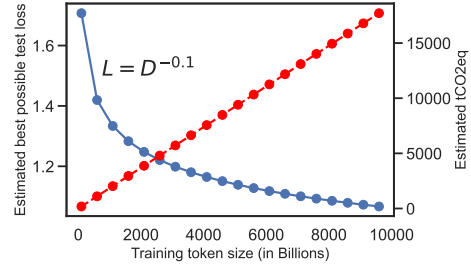


Figure 3. Test loss decreases logarithmically with training token size, whereas the estimated training-time carbon emission increases linearly.

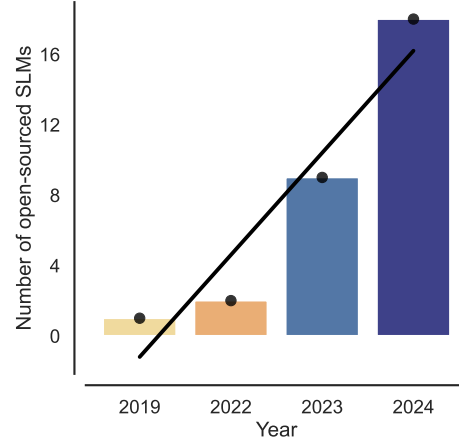


Figure 4. Number of open-sourced SLMs (size 100M-5B) developed over the years (statistics taken from Lu et al. (2024b)).

parameters), Mistral-7B (Jiang et al., 2023a), and Phi-4 (Abdin et al., 2024) (14B parameters) can rival larger models in performance despite their more compact architectures (Lu et al., 2024a).

Figure 4 highlights the sharp rise of SLM developments between 2023 and 2024. The momentum behind SLMs has been further accelerated by their practical advantages in real-world applications. The ability to run these models on consumer hardware, from laptops to edge devices, has made them particularly attractive for businesses and developers who need to balance performance with resource constraints. For instance, models like Gemini Nano (Team

²The proof is provided in Appendix B.

et al., 2024) have been specifically designed for mobile devices. In contrast, others like Code Llama (7B) (Rozière et al., 2024) have demonstrated that smaller models can excel in specialized tasks like code generation. Additionally, the cost-effectiveness of training and deploying SLMs, with their faster inference times and lower energy consumption, has made them increasingly appealing in an era where environmental impact and operational efficiency are key considerations. The success of open-source SLMs has also fostered a vibrant community of researchers and developers, who continue to push the boundaries of what is possible with smaller architectures. SLMs have emerged as strong competitors to larger models due to several defining characteristics:

- **High-quality, domain-specific training data.** Rather than relying solely on scale, SLMs prioritize data, emphasizing reasoning patterns and domain relevance.
- **Efficient attention mechanisms.** Techniques like grouped-query attention (GQA) and sliding window attention (SWA) capture long-range dependencies with minimal computational overhead.
- **Targeted architectural optimizations.** Custom architectures tailored for specific domains boost performance.
- **Knowledge distillation.** Advanced methods effectively transfer knowledge from larger models to smaller ones.
- **Model compression techniques.** Sophisticated quantization and pruning strategies reduce model size while preserving capabilities.

3.1. Data Quality Evolution

The early success of SLMs revealed that data quality can be more crucial than quantity. Recent models have demonstrated diverse approaches to data quality optimization. Phi-4 (Abdin et al., 2024) introduces a sophisticated multi-layered approach featuring plurality-based filtering, where multiple generated answers per question help filter out overly simplistic or ambiguous content, along with a self-revision mechanism that enables continuous improvement of training data quality. QWEN 2.5 (Qwen et al., 2025) balances scale with quality by implementing rigorous filtering through its instruction models while strategically integrating domain-specific datasets for mathematics and coding to achieve state-of-the-art performance in these areas. MobileLLaMA (Chu et al., 2023) demonstrates the effectiveness of combining high-quality multimodal datasets with supervised fine-tuning using carefully curated dialogue data. These approaches collectively highlight a shift from pure scale to sophisticated data curation techniques, suggesting that the future of language models depends more on intelligent data processing than model size alone.

3.2. Architectural Innovations

Several key architectural innovations have enabled SLMs to achieve impressive efficiency. Models like Mistral-7B (Jiang et al., 2023a) introduce improvements such as grouped-query attention (GQA) and sliding window attention (SWA), allowing them to outperform larger parameter models in specific tasks. Models like Mamba (Gu & Dao, 2025) use hybrid architectures, combining transformers with state space models, providing new approaches to achieving efficiency while maintaining performance. Also, the development of efficient attention mechanisms and optimization techniques has allowed models like TinyLlama (Zhang et al., 2024b) to achieve high training throughput while maintaining small parameter counts.

3.3. Advancements in Task Specialization and Training Methodology

The field has seen significant progress in developing specialized SLMs that excel in specific domains. For example, WizardMath (Luo et al., 2025) demonstrates strong mathematical reasoning capabilities and surpasses most open-source models with the size ranging between 7B to 40B. Similarly, WizardCoder (Luo et al., 2024) shows that smaller models could outperform larger ones in code generation. Advanced knowledge distillation techniques and targeted training approaches have enabled this specialization. The evolution of training methodologies has been crucial in the evolution of SLMs. As demonstrated by the Orca (Mukherjee et al., 2023) models, progressive learning approaches show how smaller models could effectively learn from larger ones through carefully structured training processes. Despite their smaller size, explanation tuning and chain-of-thought approaches have enabled SLMs to develop stronger reasoning capabilities. The development of more sophisticated fine-tuning techniques has allowed for better transfer of capabilities from larger to smaller models.

3.4. Efficiency Optimizations

Post-training optimizations have become increasingly sophisticated. Advanced quantization techniques like SmoothQuant (Xiao et al., 2023) have enabled significant model compression without substantial performance loss. Structured pruning approaches have allowed for systematic model size reduction while preserving key capabilities. Developing efficient inference techniques has made SLMs more practical for real-world deployment. These developments suggest that SLMs will continue to close the gap with larger models while offering significant advantages in terms of efficiency and practicality.

4. Our Recommendations

The conventional wisdom of neural scaling laws — that bigger models and more data inevitably lead to better performance, is fundamentally flawed and computationally inefficient. While recent LLMs have demonstrated impressive capabilities through scaling, this success comes at exponentially increasing computational and environmental costs. In this section, we describe our position on **downscaling large language model with domain adaptation** (our specific recommendations are highlighted with underlined text).

4.1. Downscaling Datasets for Better Knowledge Alignment

The wide success of SLMs has demonstrated the power of cleaned and high quality data. Also, the evidence from data pruning methods (Sorscher et al., 2023) demonstrates that strategic selection of training examples can achieve exponential rather than power law scaling, fundamentally undermining the assumption that we need ever-increasing amounts of data. This is especially important in light of findings from data-constrained scaling (Muennighoff et al., 2023), which demonstrate that in scenarios with limited data, training smaller models for more epochs outperforms increasing model size, challenging previous scaling assumptions. Future research should focus on optimizing the utilization of existing data over merely accumulating larger datasets.

The emergence of the Domain-Continual Pre-Training (D-CPT) law (Que et al., 2024) (see Appendix C for details) provides a systematic framework to find optimal mixture ratios between general and domain-specific data for domain adaptation, to counter catastrophic forgetting of general data. It eliminates wasteful trial-and-error approaches, achieving remarkable accuracy across six diverse domains. This scientific approach to model development offers a more sophisticated alternative to brute-force scaling.

4.2. Model Downscaling

A branch of research has focused on compressing LLMs to reduce computational and hardware requirements using various pruning techniques. Unstructured pruning (Sun et al., 2023) removes individual weights, producing sparse matrices that maintain performance but are less hardware-efficient. Semi-structured pruning (Frantar & Alistarh, 2023), such as the 2:4 sparsity pattern (Pool et al., 2021), introduces a hardware-friendly structured sparsity that accelerates computation. Structured pruning (Ashkboos et al., 2024; Yuan et al., 2023; Sengupta et al., 2025) takes a broader approach by removing entire components, such as Transformer layers (depth pruning) (Fan et al., 2019) or reducing embedding dimensions and attention heads (width

pruning) (Zhu et al., 2021). After pruning, post-training is crucial to mitigate performance degradation. This involves fine-tuning or continual pre-training on datasets tailored to enhance performance recovery while maintaining efficiency. The P2 law (Chen et al., 2024b), highlighted in Equation 11, provides a predictive framework for the post-training loss, considering factors such as pruning rate, model size, pre-pruning loss, and the number of training tokens. This law enables to balance computational costs with performance recovery by identifying optimal post-training data sizes. While higher pruning rates inevitably lead to larger initial losses, effective post-training strategies significantly minimize this impact, allowing smaller models to perform on par with their larger counterparts in many scenarios.

$$L(N_0, D, \rho, L_0) = L_0 + \left(\frac{1}{\rho}\right)^\gamma \left(\frac{1}{N_0}\right)^\delta \left(\frac{N_C}{N_0^\alpha} + \frac{D_C}{D^\beta} + E\right) \quad (11)$$

where L_0 is the uncompressed model loss, ρ is the pruning rate, N_0 is the model size before pruning, D is the number of post-training tokens, and $N_C, D_C, E, \alpha, \beta, \gamma$ are fitting constants.

4.3. Ensemble of LLMs

The recent advancements in model merging (Yang et al., 2024) provide an exciting opportunity to combine multiple trained models in an efficient and effective manner. This offers significant practical benefits, including reduced storage and inference costs, while also preserving privacy by eliminating the need for original training data. Although model merging is one way to integrate LLM capabilities, ensemble approaches have become more popular because they offer more consistent performance increases and greater flexibility. Ensemble approaches can effectively exploit diverse LLMs despite of their architectural differences, while merging needs models to share compatible architectures and parameter spaces and does not guarantee better performance after combination. Ensemble techniques are especially appealing for practical applications because of their flexibility and the fact that they tend to produce more consistent performance improvements.

Lu et al. (2024a) categorized LLM ensemble strategies into three main approaches: (i) before inference through router-based model selection, (ii) during inference via token-level integration, and (iii) after inference through output combination or selection. Pre-inference ensemble methods employ routers to select the most appropriate LLM for a given input, optimizing for both performance and computational efficiency. For example, ZOOPER (Lu et al., 2023) uses a reward model to train a router that selects the optimal LLM based on query characteristics. During-inference ensem-

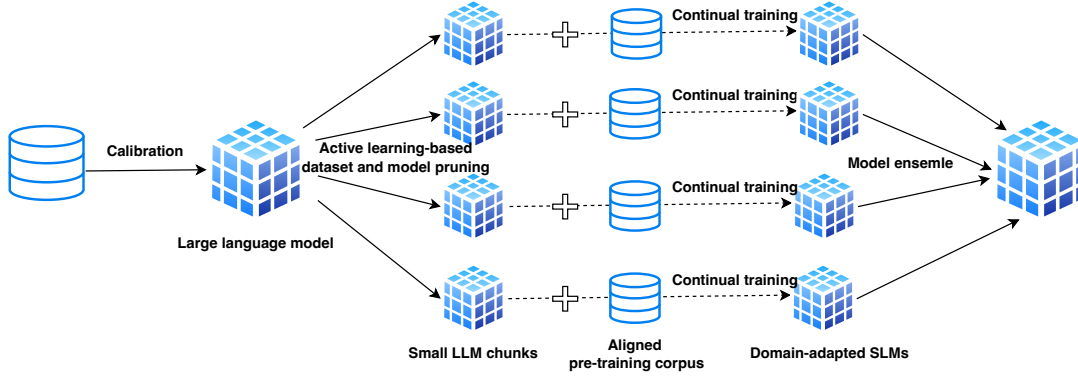


Figure 5. Our proposed framework for model downscaling with adaptive domain adaptation. We decompose an LLM pre-training into multiple stages – (1) calibration for compressing LLM into multiple SLM chunks, along with aligned pre-training corpus, (2) active learning-based dataset pruning, for optimal data downscaling, (3) Continual training of SLMs on aligned corpus and (4) model ensemble for merging SLM chunks to recover the larger model with original parameter size.

ble approaches operate at a finer granularity by combining outputs at each decoding step (Li et al., 2024), which can help reduce exposure bias and hallucinations. However, these methods often face challenges with heterogeneous vocabularies across different LLMs (Xu et al., 2024). Post-inference ensemble strategies combine multiple generated outputs or build model cascades where smaller models handle simpler queries and larger models are invoked only when necessary. This hierarchical approach, as demonstrated by Chen et al. (2023), can maintain quality while significantly reducing computational costs, showcasing how ensemble methods can deliver practical benefits that are harder to achieve through merging alone.

$$L(L_0, n) = L_0 - b + \frac{b}{n^a} \quad (12)$$

where L_0 is the base model loss, n is the number of models in the ensemble and a, b are fitting constants.

Lobacheva et al. (2021) introduced scaling law of deep ensemble, which can be analysed as shown in Equation 12 (see Appendix D for more details), which describes the expected ensemble loss as the number of models in the ensemble increases. Crucially, the authors revealed a “Memory Split Advantage” effect, where using multiple smaller networks can outperform a single large network for a given computational budget. Although Lobacheva et al. (2021) investigated the law primarily for CNN architectures, the key insights found in the study can be extended to other neural architectures as well. This finding suggests that practitioners can improve model performance by strategically splitting their computational resources across several medium-sized network models rather than investing in one large model.

4.4. Why Downscaling Law Might Work?

As we have all the necessary tools, such as post-training after pruning, and expected loss of ensemble, we can combine the knowledge from Equation 11 and Equation 12, to propose Equation 13 (see Appendix E for details), establishing a robust theoretical guarantee of efficient downscaling. It describes a condition where increasing the number of models in a deep ensemble can lead to lower expected ensemble loss $L(L_0, n)$ compared to the base model loss while maintaining the same overall computational cost. The key insight is that by increasing the number of models, as long as the condition in Equation 13 is satisfied, the ensemble can capture more diversity and reduce the overall loss. This is a noteworthy finding as deep ensembling enhances model performance without the need for additional computational resources.

Proposition 4.1. For $n \in \mathbb{Z}_+$ satisfying

$$\left(\frac{n^a - 1}{n^{a+\gamma}} \right) (n - 1)^\gamma \geq \frac{1}{bN_0^\delta} \quad (13)$$

The expected ensemble loss $L < L_0$, the loss by the base model, at the same computational cost C .

To show the implication of the equation in depth, let us take an example of the LLaMA-3-8B model. From Chen et al. (2024b), we obtain $\alpha = -1.57$ and $\gamma = 1.08$. Similarly, taking some loose assumptions from Lobacheva et al. (2021), we obtain $a = 0.83, b = 0.83, \delta = 0.29$. Putting these values in Equation 13, it holds $\forall n > 7$. This implies that we can use 8 or more pruned versions of LLaMA-3-8B, each with 1B parameters, in an ensemble that would outperform the original model without incurring any additional cost.

4.5. Proposed Downscaling Pipeline

Integrating these insights into model efficiency, scaling laws and benefits of SLMs, we propose a novel pipeline for developing more efficient and domain-adapted language models, as shown in Figure 5. This pipeline leverages techniques from dataset optimization, model compression, and ensemble methodologies to create smaller, more efficient, yet competent language models.

The first phase employs active learning-based techniques for dataset pruning (Sorscher et al., 2023) and model pruning. Using Bayesian active learning strategies (Bayer & Reuter, 2024), we identify the most informative samples from the training data while simultaneously determining optimal model architectures using unstructured, semi-structured pruning, or structured pruning (Sun et al., 2023; Frantar & Alistarh, 2023; Ashkboos et al., 2024; Yuan et al., 2023; Sengupta et al., 2025). This dual optimization approach allows us to create multiple smaller, more efficient model variants from a single LLM while maintaining essential capabilities. Also, we obtain a collection of high quality and carefully-curated datasets. The data pruning process is guided by theoretical frameworks such as Sorscher et al. (2023), suggesting that strategic selection of samples can help break the power law. The pruning of models should be done in a manner that it follows our proposition of downscaling (Equation 13).

These pruned models are designed to specialize in specific domains using domain continual pre-training. The parallel streams shown in the pipeline (Figure 5) represent these different SLM variants, each paired with an aligned pre-training corpus carefully curated for their target domain. This is crucial for efficient specialization and follows the D-CPT law for systematic domain adaptation, without catastrophic forgetting, and guided by theoretical frameworks such as the P2 law, which helps predict and minimize post-compression performance degradation. This allows each model to develop deep expertise in its designated area while maintaining computational efficiency. This targeted training approach eliminates wasteful trial-and-error methods typically associated with domain adaptation, as each model receives only the most relevant data for its specialization.

The final stage of our pipeline implements a model ensemble methodology to combine these domain-adapted SLMs into a unified system. This ensemble approach preserves the specialized capabilities of individual SLMs while creating a more versatile final model. By leveraging advanced ensemble techniques (He et al., 2020; Yadav et al., 2024), we can effectively aggregate the domain expertise of each component model while maintaining a smaller computational footprint compared to traditional large-scale models. This approach is fundamentally different from mixture of experts, where the primary motivation is usually to encour-

age sparsity within different experts. Rather, we focus on combining multiple smaller experts together in a more collaborative manner, ensuring better combined performance. As shown earlier in Proposition 4.1, this ensemble would give a better performance than the original model at the same computational cost.

The success of this pipeline relies on careful orchestration between components and systematic empirical validation. Our approach addresses three critical challenges in modern language model development: reducing computational requirements through strategic pruning, maintaining performance through targeted domain adaptation, and combining specialized capabilities through sophisticated ensemble methods. The result is a more efficient, domain-aware language model that achieves high performance without excessive computational demands or model scale.

5. Alternative Views

5.1. Limitations of Downscaling

Despite their wide success, SLMs frequently have difficulties with complex language understanding and contextual intricacies. Also, they may not attain the same degree of precision in multifaceted reasoning or complex data patterns as their larger counterparts which leads to diminished performance on activities necessitating profound comprehension or considerable expertise across multiple fields. Furthermore, Jin et al. (2023) revealed crucial insights into how language model capabilities degrade during downscaling, challenging the assumption that reduction in model size uniformly affects all capabilities. Their findings demonstrate a stark contrast in how different cognitive abilities deteriorate. While fact recall begins to significantly degrade when more than 30% of weights are removed through pruning, in-context learning capabilities show remarkable resilience, maintaining performance even with aggressive pruning of up to 60-70% of weights. This pattern remains consistent across various model reduction approaches, including pruning and dense scaling, and has been observed across different model families like OPT and LLaMA. This suggests that the neural mechanisms responsible for storing factual knowledge require significantly more parameters than those supporting in-context learning capabilities. This insight improves SLM deployments by tailoring model reduction strategies to the target application’s fact recall versus in-context learning needs.

5.2. Limitations of Ensembling

Despite their advantages, ensemble techniques for LLMs face several significant limitations that affect their practical deployment and effectiveness. As highlighted in the survey by Lu et al. (2024a), each ensemble approach has

its own set of challenges. Pre-inference ensemble methods heavily rely on the accuracy of routing mechanisms, and the router’s ability to generalize to new queries (Shnitzer et al., 2023). During-inference ensemble methods struggle with heterogeneous model architectures and vocabularies, making it challenging to directly combine outputs from different LLM families (Huang et al., 2024). Post-inference ensemble approaches face challenges related to the quality and diversity of candidate pools (Jiang et al., 2023b). Additionally, they are particularly resource-intensive as they require both multiple model executions and additional computation for output selection or fusion. A practical limitation across all ensemble approaches is the increased latency in response generation, which can be particularly problematic in real-time applications where quick responses are crucial.

6. Conclusion

In this position paper, we critiqued the underlying flaws of neural scaling laws and presented proof that the Carbon Dioxide emissions of training LLMs scale linearly with both the number of parameters and dataset size. This indicates that the scaling of LLMs is not sustainable in the long term, given the growing emphasis on climate change. Our study encouraged exploration of downscaling laws, where instead of training larger models on larger datasets, we proposed training smaller models on aligned pre-training datasets for robust performance without introducing additional computational costs. Utilising insights from recent research, we offered a downscaling law that estimates the number of smaller models in an ensemble that may outperform the original model. The study paves way for more sustainable approaches for training language models.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

Acknowledgment

T. Chakraborty acknowledges the support of the IBM-IITD AI Horizons network and Rajiv Khemani Young Faculty Chair Professorship in Artificial Intelligence.

References

Abdin, M., Aneja, J., Behl, H., Bubeck, S., Eldan, R., Gunasekar, S., Harrison, M., Hewett, R. J., Javaheripi, M., Kauffmann, P., Lee, J. R., Lee, Y. T., Li, Y., Liu, W., Mendes, C. C. T., Nguyen, A., Price, E., Rosa, G. d., Saarikivi, O., Salim, A., Shah, S., Wang, X., Ward, R.,

Wu, Y., Yu, D., Zhang, C., and Zhang, Y. Phi-4 technical report, December 2024. URL <http://arxiv.org/abs/2412.08905>. arXiv:2412.08905.

Abnar, S., Dehghani, M., Neyshabur, B., and Sedghi, H. Exploring the limits of large scale pre-training, October 2021. URL <http://arxiv.org/abs/2110.02095>. arXiv:2110.02095.

Ashkboos, S., Croci, M. L., do Nascimento, M. G., Hoefler, T., and Hensman, J. Slicept: Compress large language models by deleting rows and columns. In *International Conference on Learning Representations*, 2024.

Bayer, M. and Reuter, C. Activellm: Large language model-based active learning for textual few-shot scenarios, 2024. URL <https://arxiv.org/abs/2405.10808>.

Caballero, E., Gupta, K., Rish, I., and Krueger, D. Broken neural scaling laws, July 2023. URL <http://arxiv.org/abs/2210.14891>. arXiv:2210.14891.

Chen, L., Zaharia, M., and Zou, J. Frugalpt: How to use large language models while reducing cost and improving performance, 2023. URL <https://arxiv.org/abs/2305.05176>.

Chen, L., Davis, J. Q., Hanin, B., Bailis, P., Stoica, I., Zaharia, M. A., and Zou, J. Y. Are more llm calls all you need? towards the scaling properties of compound ai systems. *Advances in Neural Information Processing Systems*, 37:45767–45790, 2024a.

Chen, X., Hu, Y., Zhang, J., Zhang, X., Li, C., and Chen, H. Scaling law for post-training after model pruning. *arXiv preprint arXiv:2411.10272*, 2024b.

Chen, Y., Huang, B., Gao, Y., Wang, Z., Yang, J., and Ji, H. Scaling laws for predicting downstream performance in llms, October 2024c. URL <http://arxiv.org/abs/2410.08527>. arXiv:2410.08527.

Chen, Y., Huang, B., Gao, Y., Wang, Z., Yang, J., and Ji, H. Scaling laws for predicting downstream performance in llms, 2025. URL <https://arxiv.org/abs/2410.08527>.

Chu, X., Qiao, L., Lin, X., Xu, S., Yang, Y., Hu, Y., Wei, F., Zhang, X., Zhang, B., Wei, X., and Shen, C. Mobilevlm : a fast, strong and open vision language assistant for mobile devices, December 2023. URL <http://arxiv.org/abs/2312.16886>. arXiv:2312.16886.

Clark, A., de Las Casas, D., Guy, A., Mensch, A., Paganini, M., Hoffmann, J., Damoc, B., Hechtman, B., Cai, T., Borgeaud, S., et al. Unified scaling laws for routed language models. In *International conference on machine learning*, pp. 4057–4086. PMLR, 2022.

- Diaz, F. and Madaio, M. Scaling laws do not scale, July 2024. URL <http://arxiv.org/abs/2307.03201>. arXiv:2307.03201.
- Faiz, A., Kaneda, S., Wang, R., Osi, R. C., Sharma, P., Chen, F., and Jiang, L. Llmcarbon: Modeling the end-to-end carbon footprint of large language models. In *The Twelfth International Conference on Learning Representations*, 2024.
- Fan, A., Grave, E., and Joulin, A. Reducing transformer depth on demand with structured dropout, 2019. URL <https://arxiv.org/abs/1909.11556>.
- Frantar, E. and Alistarh, D. Sparsegpt: Massive language models can be accurately pruned in one-shot. In *Proceedings of the 40th International Conference on Machine Learning*, 2023. URL <https://arxiv.org/abs/2301.00774>.
- Gu, A. and Dao, T. Mamba: Linear-time sequence modeling with selective state spaces. In *First Conference on Language Modeling*, 2025.
- He, B., Lakshminarayanan, B., and Teh, Y. W. Bayesian deep ensembles via the neural tangent kernel. *Advances in neural information processing systems*, 33:1010–1022, 2020.
- Hernandez, D., Kaplan, J., Henighan, T., and McCandlish, S. Scaling laws for transfer, February 2021. URL <http://arxiv.org/abs/2102.01293>. arXiv:2102.01293.
- Hoffmann, J., Borgeaud, S., Mensch, A., Buchatskaya, E., Cai, T., Rutherford, E., Casas, D. d. L., Hendricks, L. A., Welbl, J., Clark, A., Hennigan, T., Noland, E., Millican, K., Driessche, G. v. d., Damoc, B., Guy, A., Osindero, S., Simonyan, K., Elsen, E., Rae, J. W., Vinyals, O., and Sifre, L. Training compute-optimal large language models. In *Conference on Neural Information Processing Systems*. arXiv, March 2022. doi: 10.48550/arXiv.2203.15556. URL <http://arxiv.org/abs/2203.15556>. arXiv:2203.15556.
- Hu, S., Tu, Y., Han, X., Cui, G., He, C., Zhao, W., Long, X., Zheng, Z., Fang, Y., Huang, Y., et al. Minicpm: Unveiling the potential of small language models with scalable training strategies. In *First Conference on Language Modeling*, 2024.
- Huang, Y., Feng, X., Li, B., Xiang, Y., Wang, H., Qin, B., and Liu, T. Ensemble learning for heterogeneous large language models with deep parallel collaboration. In *Conference on Neural Information Processing Systems*, 2024. URL <https://arxiv.org/abs/2404.12715>.
- Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., de las Casas, D., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., Lavaud, L. R., Lachaux, M.-A., Stock, P., Scao, T. L., Lavril, T., Wang, T., Lacroix, T., and Sayed, W. E. Mistral 7b, 2023a. URL <https://arxiv.org/abs/2310.06825>.
- Jiang, D., Ren, X., and Lin, B. Y. LLM-blender: Ensembling large language models with pairwise ranking and generative fusion. In Rogers, A., Boyd-Graber, J., and Okazaki, N. (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14165–14178, Toronto, Canada, July 2023b. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.792. URL <https://aclanthology.org/2023.acl-long.792/>.
- Jin, T., Clement, N., Dong, X., Nagarajan, V., Carbin, M., Ragan-Kelley, J., and Dziugaite, G. K. The cost of downscaling language models: fact recall deteriorates before in-context learning, October 2023. URL <http://arxiv.org/abs/2310.04680>. arXiv:2310.04680.
- Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., and Amodei, D. Scaling laws for neural language models, January 2020. URL <http://arxiv.org/abs/2001.08361>. arXiv:2001.08361.
- Krajewski, J., Ludziejewski, J., Adamczewski, K., Pióro, M., Krutul, M., Antoniak, S., Ciebiera, K., Król, K., Odrzygóźdź, T., Sankowski, P., Cygan, M., and Jaszczur, S. Scaling laws for fine-grained mixture of experts, February 2024. URL <http://arxiv.org/abs/2402.07871>. arXiv:2402.07871.
- Li, T., Liu, Q., Pang, T., Du, C., Guo, Q., Liu, Y., and Lin, M. Purifying large language models by ensembling a small language model, 2024. URL <https://arxiv.org/abs/2402.14845>.
- Lobacheva, E., Chirkova, N., Kodryan, M., and Vetrov, D. On power laws in deep ensembles, 2021. URL <https://arxiv.org/abs/2007.08483>.
- Lu, J., Pang, Z., Xiao, M., Zhu, Y., Xia, R., and Zhang, J. Merge, ensemble, and cooperate! a survey on collaborative strategies in the era of large language models, 2024a. URL <https://arxiv.org/abs/2407.06089>.
- Lu, K., Yuan, H., Lin, R., Lin, J., Yuan, Z., Zhou, C., and Zhou, J. Routing to the expert: Efficient reward-guided ensemble of large language models, 2023. URL <https://arxiv.org/abs/2311.08692>.

- Lu, Z., Li, X., Cai, D., Yi, R., Liu, F., Zhang, X., Lane, N. D., and Xu, M. Small language models: survey, measurements, and insights, September 2024b. URL <http://arxiv.org/abs/2409.15790>. arXiv:2409.15790.
- Luo, H., Sun, Q., Xu, C., Zhao, P., Lou, J., Tao, C., Geng, X., Lin, Q., Chen, S., Tang, Y., and Zhang, D. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. In *International Conference on Learning Representations*, 2025. URL <https://arxiv.org/abs/2308.09583>.
- Luo, Z., Xu, C., Zhao, P., Sun, Q., Geng, X., Hu, W., Tao, C., Ma, J., Lin, Q., and Jiang, D. Wizardcoder: Empowering code large language models with evol-instruct. In *The Twelfth International Conference on Learning Representations*, 2024.
- Muennighoff, N., Rush, A. M., Barak, B., Scao, T. L., Piktus, A., Tazi, N., Pyysalo, S., Wolf, T., and Raffel, C. Scaling data-constrained language models. In *Conference on Neural Information Processing Systems*. arXiv, October 2023. doi: 10.48550/arXiv.2305.16264. URL <http://arxiv.org/abs/2305.16264>. arXiv:2305.16264.
- Mukherjee, S., Mitra, A., Jawahar, G., Agarwal, S., Palangi, H., and Awadallah, A. Orca: Progressive learning from complex explanation traces of gpt-4, 2023. URL <https://arxiv.org/abs/2306.02707>.
- Pool, J., Sawarkar, A., and Rodge, J. Accelerating inference with sparsity using the nvidia ampere architecture and nvidia tensorrt, 2021. URL <https://developer.nvidia.com/blog/accelerating-inference-with-sparsity-using-ampere-and-tensorrt/>.
- Que, H., Liu, J., Zhang, G., Zhang, C., Qu, X., Ma, Y., Duan, F., Bai, Z., Wang, J., Zhang, Y., Tan, X., Fu, J., Su, W., Wang, J., Qu, L., and Zheng, B. D-cpt law: domain-specific continual pre-training scaling law for large language models. In *Conference on Neural Information Processing Systems*. arXiv, June 2024. doi: 10.48550/arXiv.2406.01375. URL <http://arxiv.org/abs/2406.01375>. arXiv:2406.01375.
- Qwen, Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Li, C., Liu, D., Huang, F., Wei, H., Lin, H., Yang, J., Tu, J., Zhang, J., Yang, J., Yang, J., Zhou, J., Lin, J., Dang, K., Lu, K., Bao, K., Yang, K., Yu, L., Li, M., Xue, M., Zhang, P., Zhu, Q., Men, R., Lin, R., Li, T., Tang, T., Xia, T., Ren, X., Ren, X., Fan, Y., Su, Y., Zhang, Y., Wan, Y., Liu, Y., Cui, Z., Zhang, Z., and Qiu, Z. Qwen2. 5 technical report, January 2025. URL <http://arxiv.org/abs/2412.15115>. arXiv:2412.15115.
- Rozière, B., Gehring, J., Gloeckle, F., Sootla, S., Gat, I., Tan, X. E., Adi, Y., Liu, J., Sauvestre, R., Remez, T., Rapin, J., Kozhevnikov, A., Evtimov, I., Bitton, J., Bhatt, M., Ferrer, C. C., Grattafiori, A., Xiong, W., Défossez, A., Copet, J., Azhar, F., Touvron, H., Martin, L., Usunier, N., Scialom, T., and Synnaeve, G. Code llama: Open foundation models for code, 2024. URL <https://arxiv.org/abs/2308.12950>.
- Sardana, N., Portes, J., Doubov, S., and Frankle, J. Beyond chinchilla-optimal: Accounting for inference in language model scaling laws. In *International Conference on Machine Learning*, pp. 43445–43460. PMLR, 2024.
- Sengupta, A., Chaudhary, S., and Chakraborty, T. You only prune once: Designing calibration-free model compression with policy learning. In *The Thirteenth International Conference on Learning Representations*, 2025.
- Shnitzer, T., Ou, A., Silva, M., Soule, K., Sun, Y., Solomon, J., Thompson, N., and Yurochkin, M. Large language model routing with benchmark datasets, 2023. URL <https://arxiv.org/abs/2309.15789>.
- Sorscher, B., Geirhos, R., Shekhar, S., Ganguli, S., and Morcos, A. S. Beyond neural scaling laws: beating power law scaling via data pruning, April 2023. URL <http://arxiv.org/abs/2206.14486>. arXiv:2206.14486.
- Sun, M., Liu, Z., Bair, A., and Kolter, J. Z. A simple and effective pruning approach for large language models. *arXiv preprint arXiv:2306.11695*, 2023.
- Tao, C., Liu, Q., Dou, L., Muennighoff, N., Wan, Z., Luo, P., Lin, M., and Wong, N. Scaling laws with vocabulary: Larger models deserve larger vocabularies. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.
- Tay, Y., Dehghani, M., Abnar, S., Chung, H., Fedus, W., Rao, J., Narang, S., Tran, V., Yogatama, D., and Metzler, D. Scaling laws vs model architectures: How does inductive bias influence scaling? In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 12342–12364, 2023.
- Team, G., Anil, R., Borgeaud, S., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., Millican, K., Silver, D., Johnson, M., Antonoglou, I., Schrittwieser, J., Glaese, A., Chen, J., Pitler, E., Lillicrap, T., Lazaridou, A., Firat, O., Molloy, J., Isard, M., Barham, P. R., Hennigan, T., Lee, B., Viola, F., Reynolds, M., Xu, Y., Doherty, R., Collins, E., Meyer, C., Rutherford, E., Moreira, E., Ayoub, K., Goel, M., Krawczyk, J., Du, C., Chi, E., Cheng, H.-T., Ni, E., Shah, P., Kane, P., Chan, B., Faruqui, M., Severyn, A., Lin, H., Li, Y., Cheng, Y., Ittycheriah, A., Mahdiah, M., Chen, M., Sun, P., Tran, D., Bagri, S., Lakshminarayanan, B., Liu, J., Orban,

- A., Güra, F., Zhou, H., Song, X., Boffy, A., Ganapathy, H., Zheng, S., Choe, H., Ágoston Weisz, Zhu, T., Lu, Y., Gopal, S., Kahn, J., Kula, M., Pitman, J., Shah, R., Taropa, E., Merey, M. A., Baeuml, M., Chen, Z., Shafey, L. E., Zhang, Y., Sercinoglu, O., Tucker, G., Piqueras, E., Krikun, M., Barr, I., Savinov, N., Danihelka, I., Roelofs, B., White, A., Andreassen, A., von Glehn, T., Yagati, L., Kazemi, M., Gonzalez, L., Khalman, M., Sygnowski, J., Frechette, A., Smith, C., Culp, L., Proleev, L., Luan, Y., Chen, X., Lottes, J., Schucher, N., Lebron, F., Rustemi, A., Clay, N., Crone, P., Kocisky, T., Zhao, J., Perz, B., Yu, D., Howard, H., Bloniarz, A., Rae, J. W., Lu, H., Sifre, L., Maggioni, M., Alcober, F., Garrette, D., Barnes, M., Thakoor, S., Austin, J., Barth-Maron, G., Wong, W., Joshi, R., Chaabouni, R., Fatiha, D., Ahuja, A., Tomar, G. S., Senter, E., Chadwick, M., Kornakov, I., Attaluri, N., Iturrate, I., Liu, R., Li, Y., Cogan, S., Chen, J., Jia, C., Gu, C., Zhang, Q., Grimstad, J., Hartman, A. J., Garcia, X., Pillai, T. S., Devlin, J., Laskin, M., de Las Casas, D., Valter, D., Tao, C., Blanco, L., Badia, A. P., Reitter, D., Chen, M., Brennan, J., Rivera, C., Brin, S., Iqbal, S., Surita, G., Labanowski, J., Rao, A., Winkler, S., Parisotto, E., Gu, Y., Olszewska, K., Addanki, R., Miech, A., Louis, A., Teplyashin, D., Brown, G., Catt, E., Balaguer, J., Xiang, J., Wang, P., Ashwood, Z., Briukhov, A., Webson, A., Ganapathy, S., Sanghavi, S., Kannan, A., Chang, M.-W., Stjerngren, A., Djolonga, J., Sun, Y., Bapna, A., Aitchison, M., Pejman, P., Michalewski, H., Yu, T., Wang, C., Love, J., Ahn, J., Bloxwich, D., Han, K., Humphreys, P., Sellam, T., Bradbury, J., Godbole, V., Samangooei, S., Damoc, B., Kaskasoli, A., Arnold, S. M. R., Vasudevan, V., Agrawal, S., Riesa, J., Lepikhin, D., Tanburn, R., Srinivasan, S., Lim, H., Hodgkinson, S., Shyam, P., Ferret, J., Hand, S., Garg, A., Paine, T. L., Li, J., Li, Y., Giang, M., Neitz, A., Abbas, Z., York, S., Reid, M., Cole, E., Chowdhery, A., Das, D., Rogozińska, D., Nikolaev, V., Sprechmann, P., Nado, Z., Zilka, L., Prost, F., He, L., Monteiro, M., Mishra, G., Welty, C., Newlan, J., Jia, D., Allamanis, M., Hu, C. H., de Liedekerke, R., Gilmer, J., Saroufim, C., Rijhwani, S., Hou, S., Shrivastava, D., Baddepudi, A., Goldin, A., Oztirel, A., Cassirer, A., Xu, Y., Sohn, D., Sachan, D., Amplayo, R. K., Swanson, C., Petrova, D., Narayan, S., Guez, A., Brahma, S., Landon, J., Patel, M., Zhao, R., Villela, K., Wang, L., Jia, W., Rahtz, M., Giménez, M., Yeung, L., Keeling, J., Georgiev, P., Mincu, D., Wu, B., Haykal, S., Saputro, R., Vodrahalli, K., Qin, J., Cankara, Z., Sharma, A., Fernando, N., Hawkins, W., Neyshabur, B., Kim, S., Hutter, A., Agrawal, P., Castro-Ros, A., van den Driessche, G., Wang, T., Yang, F., Yiin Chang, S., Komarek, P., McIlroy, R., Lučić, M., Zhang, G., Farhan, W., Sharman, M., Natsev, P., Michel, P., Bansal, Y., Qiao, S., Cao, K., Shakeri, S., Butterfield, C., Chung, J., Rubenstein, P. K., Agrawal, S., Mensch, A., Soparkar, K., Lenc, K., Chung, T., Pope, A., Maggiore, L., Kay, J., Jhakra, P., Wang, S., Maynez, J., Phuong, M., Tobin, T., Tacchetti, A., Trebacz, M., Robinson, K., Katariya, Y., Riedel, S., Bailey, P., Xiao, K., Ghelani, N., Aroyo, L., Slone, A., Houlisby, N., Xiong, X., Yang, Z., Gribovskaya, E., Adler, J., Wirth, M., Lee, L., Li, M., Kagohara, T., Pavagadhi, J., Bridgers, S., Bortsova, A., Ghemawat, S., Ahmed, Z., Liu, T., Powell, R., Bolina, V., Iinuma, M., Zablotskaia, P., Besley, J., Chung, D.-W., Dozat, T., Comanescu, R., Si, X., Greer, J., Su, G., Polacek, M., Kaufman, R. L., Tokumine, S., Hu, H., Buchatskaya, E., Miao, Y., Elhawaty, M., Siddhant, A., Tomasev, N., Xing, J., Greer, C., Miller, H., Ashraf, S., Roy, A., Zhang, Z., Ma, A., Filos, A., Besta, M., Blevins, R., Klimenko, T., Yeh, C.-K., Changpinyo, S., Mu, J., Chang, O., Pajarskas, M., Muir, C., Cohen, V., Lan, C. L., Haridasan, K., Marathe, A., Hansen, S., Douglas, S., Samuel, R., Wang, M., Austin, S., Lan, C., Jiang, J., Chiu, J., Lorenzo, J. A., Sjöstrand, L. L., Cevey, S., Gleicher, Z., Avrahami, T., Boral, A., Srinivasan, H., Selo, V., May, R., Aisopos, K., Hussenot, L., Soares, L. B., Baumli, K., Chang, M. B., Recasens, A., Caine, B., Pritzel, A., Pavetic, F., Pardo, F., Gergely, A., Frye, J., Ramasesh, V., Horgan, D., Badola, K., Kassner, N., Roy, S., Dyer, E., Campos, V. C., Tomala, A., Tang, Y., Badawy, D. E., White, E., Mustafa, B., Lang, O., Jindal, A., Vikram, S., Gong, Z., Caelles, S., Hemsley, R., Thornton, G., Feng, F., Stokowiec, W., Zheng, C., Thacker, P., Çağlar Ünlü, Zhang, Z., Saleh, M., Svensson, J., Bileschi, M., Patil, P., Anand, A., Ring, R., Tshilas, K., Vezer, A., Selvi, M., Shevlane, T., Rodriguez, M., Kwiatkowski, T., Daruki, S., Rong, K., Dafoe, A., FitzGerald, N., Gu-Lemberg, K., Khan, M., Hendricks, L. A., Pellat, M., Feinberg, V., Cobon-Kerr, J., Sainath, T., Rauh, M., Hashemi, S. H., Ives, R., Hasson, Y., Noland, E., Cao, Y., Byrd, N., Hou, L., Wang, Q., Sottiaux, T., Paganini, M., Lespiau, J.-B., Moufarek, A., Hassan, S., Shivakumar, K., van Amersfoort, J., Mandhane, A., Joshi, P., Goyal, A., Tung, M., Brock, A., Sheahan, H., Misra, V., Li, C., Rakićević, N., Dehghani, M., Liu, F., Mittal, S., Oh, J., Noury, S., Sezener, E., Huot, F., Lamm, M., Cao, N. D., Chen, C., Mudgal, S., Stella, R., Brooks, K., Vasudevan, G., Liu, C., Chain, M., Melinkeri, N., Cohen, A., Wang, V., Seymore, K., Zubkov, S., Goel, R., Yue, S., Krishnakumaran, S., Albert, B., Hurley, N., Sano, M., Mohananey, A., Joughin, J., Filonov, E., Kępa, T., Eldawy, Y., Lim, J., Rishi, R., Badiezadegan, S., Bos, T., Chang, J., Jain, S., Padmanabhan, S. G. S., Puttagunta, S., Krishna, K., Baker, L., Kalb, N., Bedapudi, V., Kurzrok, A., Lei, S., Yu, A., Litvin, O., Zhou, X., Wu, Z., Sobell, S., Siciliano, A., Papir, A., Neale, R., Bragagnolo, J., Toor, T., Chen, T., Anklin, V., Wang, F., Feng, R., Gholami, M., Ling, K., Liu, L., Walter, J., Moghaddam, H., Kishore, A., Adamek, J., Mercado, T., Mallinson, J., Wandekar, S., Cagle, S., Ofek, E., Garrido, G., Lombriser, C., Mukha, M., Sun,

- B., Mohammad, H. R., Matak, J., Qian, Y., Peswani, V., Janus, P., Yuan, Q., Schelin, L., David, O., Garg, A., He, Y., Duzhyi, O., Älgmyr, A., Lottaz, T., Li, Q., Yadav, V., Xu, L., Chinien, A., Shivanna, R., Chuklin, A., Li, J., Spadine, C., Wolfe, T., Mohamed, K., Das, S., Dai, Z., He, K., von Dincklage, D., Upadhyay, S., Maurya, A., Chi, L., Krause, S., Salama, K., Rabinovitch, P. G., M., P. K. R., Selvan, A., Dektiarev, M., Ghiasi, G., Guven, E., Gupta, H., Liu, B., Sharma, D., Shtacher, I. H., Paul, S., Akerlund, O., Aubet, F.-X., Huang, T., Zhu, C., Zhu, E., Teixeira, E., Fritze, M., Bertolini, F., Marinescu, L.-E., Bölle, M., Paulus, D., Gupta, K., Latkar, T., Chang, M., Sanders, J., Wilson, R., Wu, X., Tan, Y.-X., Thiet, L. N., Doshi, T., Lall, S., Mishra, S., Chen, W., Luong, T., Benjamin, S., Lee, J., Andrejczuk, E., Rabiej, D., Ranjan, V., Styrk, K., Yin, P., Simon, J., Harriott, M. R., Bansal, M., Robsky, A., Bacon, G., Greene, D., Mirylenka, D., Zhou, C., Sarvana, O., Goyal, A., Andermatt, S., Siegler, P., Horn, B., Israel, A., Pongetti, F., Chen, C.-W. L., Selvatici, M., Silva, P., Wang, K., Tolins, J., Guu, K., Yogev, R., Cai, X., Agostini, A., Shah, M., Nguyen, H., Donnaile, N. O., Pereira, S., Friso, L., Stambler, A., Kurzrok, A., Kuang, C., Romanikhin, Y., Geller, M., Yan, Z., Jang, K., Lee, C.-C., Fica, W., Malmi, E., Tan, Q., Banica, D., Balle, D., Pham, R., Huang, Y., Avram, D., Shi, H., Singh, J., Hidey, C., Ahuja, N., Saxena, P., Dooley, D., Potharaju, S. P., O'Neill, E., Gokulchandran, A., Foley, R., Zhao, K., Dusenberry, M., Liu, Y., Mehta, P., Kotikalapudi, R., Safranek-Shrader, C., Goodman, A., Kessinger, J., Globen, E., Kolhar, P., Gorgolewski, C., Ibrahim, A., Song, Y., Eichenbaum, A., Brovelli, T., Potluri, S., Lahoti, P., Baetu, C., Ghorbani, A., Chen, C., Crawford, A., Pal, S., Sridhar, M., Gurita, P., Mujika, A., Petrovski, I., Cedoz, P.-L., Li, C., Chen, S., Santo, N. D., Goyal, S., Punjabi, J., Kappaganthu, K., Kwak, C., LV, P., Velury, S., Choudhury, H., Hall, J., Shah, P., Figueira, R., Thomas, M., Lu, M., Zhou, T., Kumar, C., Jurdi, T., Chikkerur, S., Ma, Y., Yu, A., Kwak, S., Ähdel, V., Rajayogam, S., Choma, T., Liu, F., Barua, A., Ji, C., Park, J. H., Hellendoorn, V., Bailey, A., Bilal, T., Zhou, H., Khatir, M., Sutton, C., Rzakowski, W., Macintosh, F., Shagin, K., Medina, P., Liang, C., Zhou, J., Shah, P., Bi, Y., Dankovics, A., Banga, S., Lehmann, S., Bredesen, M., Lin, Z., Hoffmann, J. E., Lai, J., Chung, R., Yang, K., Balani, N., Bražinskas, A., Sozanschi, A., Hayes, M., Alcalde, H. F., Makarov, P., Chen, W., Stella, A., Snijders, L., Mandl, M., Kärrman, A., Nowak, P., Wu, X., Dyck, A., Vaidyanathan, K., R. R., Mallet, J., Rudominer, M., Johnston, E., Mittal, S., Udathu, A., Christensen, J., Verma, V., Irving, Z., Santucci, A., Elsayed, G., Davoodi, E., Georgiev, M., Tenney, I., Hua, N., Cideron, G., Leurent, E., Alnahlawi, M., Georgescu, I., Wei, N., Zheng, I., Scandinaro, D., Jiang, H., Snoek, J., Sundararajan, M., Wang, X., Ontiveros, Z., Karo, I., Cole, J., Rajashekhar, V., Tume, L., Ben-David, E., Jain, R., Uesato, J., Datta, R., Bunyan, O., Wu, S., Zhang, J., Stanczyk, P., Zhang, Y., Steiner, D., Naskar, S., Azzam, M., Johnson, M., Paszke, A., Chiu, C.-C., Elias, J. S., Mohiuddin, A., Muhammad, F., Miao, J., Lee, A., Vieillard, N., Park, J., Zhang, J., Stanway, J., Garmon, D., Karmarkar, A., Dong, Z., Lee, J., Kumar, A., Zhou, L., Evens, J., Isaac, W., Irving, G., Loper, E., Fink, M., Arkatkar, I., Chen, N., Shafran, I., Petrychenko, I., Chen, Z., Jia, J., Levskaya, A., Zhu, Z., Grabowski, P., Mao, Y., Magni, A., Yao, K., Snaider, J., Casagrande, N., Palmer, E., Suganthan, P., Castaño, A., Giannoumis, I., Kim, W., Rybiński, M., Sreevatsa, A., Prendki, J., Soergel, D., Goedeckemeyer, A., Gierke, W., Jafari, M., Gaba, M., Wiesner, J., Wright, D. G., Wei, Y., Vashisht, H., Kulizhskaya, Y., Hoover, J., Le, M., Li, L., Iwuanyanwu, C., Liu, L., Ramirez, K., Khorlin, A., Cui, A., LIN, T., Wu, M., Aguilar, R., Pallo, K., Chakladar, A., Perng, G., Abellan, E. A., Zhang, M., Dasgupta, I., Kushman, N., Penchev, I., Repina, A., Wu, X., van der Weide, T., Ponnappalli, P., Kaplan, C., Simsa, J., Li, S., Dousse, O., Yang, F., Piper, J., Ie, N., Pasumarthi, R., Lintz, N., Vijayakumar, A., Andor, D., Valenzuela, P., Lui, M., Paduraru, C., Peng, D., Lee, K., Zhang, S., Greene, S., Nguyen, D. D., Kurylowicz, P., Hardin, C., Dixon, L., Janzer, L., Choo, K., Feng, Z., Zhang, B., Singhal, A., Du, D., McKinnon, D., Antropova, N., Bolukbasi, T., Keller, O., Reid, D., Finchelstein, D., Raad, M. A., Crocker, R., Hawkins, P., Dadashi, R., Gaffney, C., Franko, K., Bulanova, A., Leblond, R., Chung, S., Askham, H., Cobo, L. C., Xu, K., Fischer, F., Xu, J., Sorokin, C., Alberti, C., Lin, C.-C., Evans, C., Dimitriev, A., Forbes, H., Barnase, D., Tung, Z., Omernick, M., Bishop, C., Sterneck, R., Jain, R., Xia, J., Amid, E., Piccinno, F., Wang, X., Banzal, P., Mankowitz, D. J., Polozov, A., Krakovna, V., Brown, S., Bateni, M., Duan, D., Firoiu, V., Thotakuri, M., Natan, T., Geist, M., tan Girgin, S., Li, H., Ye, J., Roval, O., Tojo, R., Kwong, M., Lee-Thorp, J., Yew, C., Sinopalnikov, D., Ramos, S., Mellor, J., Sharma, A., Wu, K., Miller, D., Sonnerat, N., Vnukov, D., Greig, R., Beattie, J., Caveness, E., Bai, L., Eisenschlos, J., Korchemniy, A., Tsai, T., Jasarevic, M., Kong, W., Dao, P., Zheng, Z., Liu, F., Yang, F., Zhu, R., Teh, T. H., Sanmiya, J., Gladchenko, E., Trdin, N., Toyama, D., Rosen, E., Tavakkol, S., Xue, L., Elkind, C., Woodman, O., Carpenter, J., Papamakarios, G., Kemp, R., Kafle, S., Grunina, T., Sinha, R., Talbert, A., Wu, D., Owusu-Afriyie, D., Du, C., Thornton, C., Pont-Tuset, J., Narayana, P., Li, J., Fatehi, S., Wieting, J., Ajmeri, O., Uria, B., Ko, Y., Knight, L., Héliou, A., Niu, N., Gu, S., Pang, C., Li, Y., Levine, N., Stolovich, A., Santamaria-Fernandez, R., Goenka, S., Yustalim, W., Strudel, R., Elqursh, A., Deck, C., Lee, H., Li, Z., Levin, K., Hoffmann, R., Holtmann-Rice, D., Bachem, O., Arora, S., Koh, C., Yeganeh, S. H., Pöder, S., Tariq, M., Sun, Y., Ionita, L., Seyedhosseini, M., Tafti, P.

- Liu, Z., Gulati, A., Liu, J., Ye, X., Chrzaszcz, B., Wang, L., Sethi, N., Li, T., Brown, B., Singh, S., Fan, W., Parisi, A., Stanton, J., Koverkathu, V., Choquette-Choo, C. A., Li, Y., Lu, T., Ittycheriah, A., Shroff, P., Varadarajan, M., Bahargam, S., Willoughby, R., Gaddy, D., Desjardins, G., Cornero, M., Robenek, B., Mittal, B., Albrecht, B., Shenoy, A., Moiseev, F., Jacobsson, H., Ghaffarkhah, A., Rivière, M., Walton, A., Crepy, C., Parrish, A., Zhou, Z., Farabet, C., Radebaugh, C., Srinivasan, P., van der Salm, C., Fidjeland, A., Scellato, S., Latorre-Chimoto, E., Klimczak-Plucińska, H., Bridson, D., de Cesare, D., Hudson, T., Mendolicchio, P., Walker, L., Morris, A., Mauger, M., Guseynov, A., Reid, A., Odoom, S., Loher, L., Cotruta, V., Yenugula, M., Grewe, D., Petrushkina, A., Duerig, T., Sanchez, A., Yadlowsky, S., Shen, A., Globerson, A., Webb, L., Dua, S., Li, D., Bhupatiraju, S., Hurt, D., Qureshi, H., Agarwal, A., Shani, T., Eyal, M., Khare, A., Belle, S. R., Wang, L., Tekur, C., Kale, M. S., Wei, J., Sang, R., Saeta, B., Liechty, T., Sun, Y., Zhao, Y., Lee, S., Nayak, P., Fritz, D., Vuyyuru, M. R., Aslanides, J., Vyas, N., Wicke, M., Ma, X., Eltyshv, E., Martin, N., Cate, H., Manyika, J., Amiri, K., Kim, Y., Xiong, X., Kang, K., Luisier, F., Tripuraneni, N., Madras, D., Guo, M., Waters, A., Wang, O., Ainslie, J., Baldrige, J., Zhang, H., Pruthi, G., Bauer, J., Yang, F., Mansour, R., Gelman, J., Xu, Y., Polovets, G., Liu, J., Cai, H., Chen, W., Sheng, X., Xue, E., Ozair, S., Angermueller, C., Li, X., Sinha, A., Wang, W., Wiesinger, J., Koukoumidis, E., Tian, Y., Iyer, A., Gurumurthy, M., Goldenson, M., Shah, P., Blake, M., Yu, H., Urbanowicz, A., Palomaki, J., Fernando, C., Durden, K., Mehta, H., Momchev, N., Rahimtoroghi, E., Georgaki, M., Raul, A., Ruder, S., Redshaw, M., Lee, J., Zhou, D., Jalan, K., Li, D., Hechtman, B., Schuh, P., Nasr, M., Milan, K., Mikulik, V., Franco, J., Green, T., Nguyen, N., Kelley, J., Mahendru, A., Hu, A., Howland, J., Vargas, B., Hui, J., Bansal, K., Rao, V., Ghiya, R., Wang, E., Ye, K., Sarr, J. M., Preston, M. M., Elish, M., Li, S., Kaku, A., Gupta, J., Pasupat, I., Juan, D.-C., Someswar, M., M., T., Chen, X., Amini, A., Fabrikant, A., Chu, E., Dong, X., Muthal, A., Buthpitiya, S., Jauhari, S., Hua, N., Khandelwal, U., Hitron, A., Ren, J., Rinaldi, L., Drath, S., Dabush, A., Jiang, N.-J., Godhia, H., Sachs, U., Chen, A., Fan, Y., Taitelbaum, H., Noga, H., Dai, Z., Wang, J., Liang, C., Hamer, J., Ferng, C.-S., Elkind, C., Atias, A., Lee, P., Listík, V., Carlen, M., van de Kerkhof, J., Pikus, M., Zaher, K., Müller, P., Zykova, S., Stefanec, R., Gatsko, V., Hirnschall, C., Sethi, A., Xu, X. F., Ahuja, C., Tsai, B., Stefanoiu, A., Feng, B., Dhandhanian, K., Katyal, M., Gupta, A., Parulekar, A., Pitta, D., Zhao, J., Bhatia, V., Bhavnani, Y., Alhadlaq, O., Li, X., Danenberg, P., Tu, D., Pine, A., Filippova, V., Ghosh, A., Limonchik, B., Urala, B., Lanka, C. K., Clive, D., Sun, Y., Li, E., Wu, H., Hongtongsak, K., Li, I., Thakkar, K., Omarov, K., Majmundar, K., Alver-son, M., Kucharski, M., Patel, M., Jain, M., Zabelin, M., Pelagatti, P., Kohli, R., Kumar, S., Kim, J., Sankar, S., Shah, V., Ramachandruni, L., Zeng, X., Bariach, B., Weidinger, L., Vu, T., Andreev, A., He, A., Hui, K., Kashem, S., Subramanya, A., Hsiao, S., Hassabis, D., Kavukcuoglu, K., Sadovsky, A., Le, Q., Strohman, T., Wu, Y., Petrov, S., Dean, J., and Vinyals, O. Gemini: A family of highly capable multimodal models, 2024. URL <https://arxiv.org/abs/2312.11805>.
- Villalobos, P., Ho, A., Sevilla, J., Besiroglu, T., Heim, L., and Hobbhahn, M. Will we run out of data? Limits of LLM scaling based on human-generated data, June 2024. URL <http://arxiv.org/abs/2211.04325>. arXiv:2211.04325.
- Wu, Y., Sun, Z., Li, S., Welleck, S., and Yang, Y. Inference scaling laws: An empirical analysis of compute-optimal inference for llm problem-solving. In *The Thirteenth International Conference on Learning Representations*, 2025.
- Xiao, G., Lin, J., Seznec, M., Wu, H., Demouth, J., and Han, S. Smoothquant: Accurate and efficient post-training quantization for large language models. In *International Conference on Machine Learning*, pp. 38087–38099. PMLR, 2023.
- Xu, Y., Lu, J., and Zhang, J. Bridging the gap between different vocabularies for llm ensemble. In *Nations of the Americas Chapter of the Association for Computational Linguistics*, 2024. URL <https://arxiv.org/abs/2404.09492>.
- Yadav, P., Tam, D., Choshen, L., Raffel, C. A., and Bansal, M. Ties-merging: Resolving interference when merging models. *Advances in Neural Information Processing Systems*, 36, 2024.
- Yang, E., Shen, L., Guo, G., Wang, X., Cao, X., Zhang, J., and Tao, D. Model merging in llms, mllms, and beyond: Methods, theories, applications and opportunities, 2024. URL <https://arxiv.org/abs/2408.07666>.
- Yuan, Z., Shang, Y., Song, Y., Wu, Q., Yan, Y., and Sun, G. Asvd: Activation-aware singular value decomposition for compressing large language models. *arXiv preprint arXiv:2312.05821*, 2023.
- Yun, L., Zhuang, Y., Fu, Y., Xing, E. P., and Zhang, H. Toward inference-optimal mixture-of-expert large language models, April 2024. URL <http://arxiv.org/abs/2404.02852>. arXiv:2404.02852.
- Zhang, B., Liu, Z., Cherry, C., and Firat, O. When scaling meets llm finetuning: The effect of data, model and finetuning method. In *The Twelfth International Conference on Learning Representations*, 2024a.

Zhang, H., Ning, A., Prabhakar, R., and Wentzlaff, D. A hardware evaluation framework for large language model inference. *arXiv preprint arXiv:2312.03134*, 2023.

Zhang, P., Zeng, G., Wang, T., and Lu, W. Tinyllama: An open-source small language model, 2024b. URL <https://arxiv.org/abs/2401.02385>.

Zhu, M., Tang, Y., and Han, K. Vision transformer pruning, 2021. URL <https://arxiv.org/abs/2104.08500>.

A. CO2 Emission Formulas from LLMCarbon (Faiz et al., 2024)

FLOP Count Formulas:

$$\text{Training: } TC \approx 6ND \quad (14)$$

$$\text{Inference: } IC \approx 2ND \quad (15)$$

where N denotes number of parameters, and D denotes amount of training data.

Device Execution Time:

$$t_{dev} = \frac{TFLOP}{n_{dev} \cdot FLOP_{peak} \cdot eff} \quad (16)$$

where $TFLOP$ denotes FLOP count, n_{dev} denotes number of computing devices, eff denotes the hardware efficiency, and the computing device number, and $FLOP_{peak}$ represents the device peak throughput.

Hardware Energy:

$$energy_{hard} = \sum_{i \in hardware_set} (P_i \cdot eff_i \cdot n_i \cdot t_i) \quad (17)$$

where P_i , eff_i , n_i , t_i denote the peak power, hardware efficiency, count, and execution time of hardware unit i , respectively.

Operational Energy:

$$energy_{oper} = energy_{hard} \cdot PUE \quad (18)$$

where PUE denotes Power Usage Efficiency of the specific data center.

Operational carbon footprint:

$$CO2eq_{oper} = energy_{oper} \cdot c_{inten} \quad (19)$$

where c_{inten} denotes the carbon intensity of the specific data center.

Embodied Carbon Footprint:

$$CO2eq_{chip} = area \cdot CPA \quad (20)$$

$$CO2eq_{emb} = \sum_{i \in hardware_set} \frac{t_i \cdot CO2eq_{chip_i}}{lifetime_i} \quad (21)$$

where $CO2eq_{chip}$ denotes chip's embodied carbon footprint, CPA denotes Carbon emitted Per unit Area and $lifetime$ denotes lifetime of hardware unit.

Total Carbon Footprint:

$$CO2eq = CO2eq_{oper} + CO2eq_{emb} \quad (22)$$

B. Proof of Proposition 2.1

Equation 16 can be expressed in terms of Equation 14:

$$t = \frac{6ND}{n \cdot FLOP_{peak} \cdot eff} \text{ [for training]} \quad (23)$$

Starting with the basic formula and substituting each component:

$$CO2eq_{oper} = energy_{oper} \cdot c_{inten} \quad (\text{from 19})$$

$$= (energy_{hard} \cdot PUE) \cdot c_{inten} \quad (\text{from 18})$$

$$= \left(\sum_i P_i \cdot eff_i \cdot n_i \cdot t_i \right) \cdot PUE \cdot c_{inten} \quad (\text{from 17})$$

$$= \left(\sum_i P_i \cdot eff_i \cdot n_i \cdot \frac{6ND}{n \cdot FLOP_{peak} \cdot eff_i} \right) \cdot PUE \cdot c_{inten} \quad (\text{from 23})$$

Similarly for embodied CO2 from Equation 21 and Equation 23:

$$CO2eq_{emb} = \sum_i \frac{6PD}{n \cdot FLOP_{peak} \cdot eff_i} \cdot \frac{area_i \cdot CPA_i}{lifetime_i}$$

Combining operational and embodied CO2:

$$CO2eq(N, D) = CO2eq_{oper} + CO2eq_{emb}$$

Grouping terms with P and D and further simplifying:

$$CO2eq(N, D) = (K_1 + K_2) \cdot N \cdot D \quad (24)$$

where

$$K_1 = \left(\sum_i P_i \cdot eff_i \cdot n_i \cdot \frac{6}{n \cdot FLOP_{peak} \cdot eff_i} \right) \cdot PUE \cdot c_{inten}$$

$$K_2 = \sum_i \frac{6}{n \cdot FLOP_{peak} \cdot eff_i} \cdot \frac{area_i \cdot CPA_i}{lifetime_i}$$

$K_1 + K_2$ represents a compound constant that encapsulates all hardware, data center, and efficiency parameters.

C. Domain-Continual Pre-Training Scaling Law

Que et al. (2024) provided a mathematical framework to determine the optimal mixture ratio between general and domain-specific data during continual pre-training. It eliminates the need for expensive grid search experiments to find the best ratio and enables more efficient resource allocation in domain adaptation of LLMs. They validated its effectiveness across code, math, law, chemistry, music and medical domains. The law is as follows:

$$L(N, D, r) = E + \frac{A}{N^\alpha} + \frac{B \cdot r^\eta}{D^\beta} + \frac{C}{(r + \epsilon)^\gamma} \quad (25)$$

where N denotes number of parameters, D denotes dataset size, r denotes mixture ratio and $E, A, B, C, \alpha, \beta, \gamma, \eta, \epsilon$ are fitting parameters.

D. Proof of Equation 12

The scaling law of model ensemble proposed by Lobacheva et al. (2021) is:

$$L = c + \frac{b}{n^a} \quad (26)$$

where L is the ensemble loss, n is the number of models in the ensemble, and c, b are fitting parameters.

For getting a single base model loss L_0 , we can put $n = 1$:

$$L_0 = L(1) = c + b \implies c = L_0 - b$$

Putting this back in Equation 26, we get our final form:

$$L = L_0 - b + \frac{b}{n^a}$$

E. Proof of Proposition 4.1

For P2 scaling law in Equation 11, we get

$$L_{compressed} = L_0 + \left(\frac{1}{\rho}\right)^\gamma \left(\frac{1}{N_0}\right)^\delta \left(\frac{N_C}{N_0^\alpha} + \frac{D_C}{D^\beta} + E\right) \quad (27)$$

$$\approx L_0 + \frac{1}{\rho^\gamma} \cdot \frac{1}{N_0^\delta} \quad (28)$$

For simplicity, we assume the term $\left(\frac{N_C}{N_0^\alpha} + \frac{D_C}{D^\beta} + E\right)$ remains approximately constant when comparing the base model and the ensemble model at the same computational budget. This is justified as both the original model and ensemble use similar total parameter counts and similar computational resources, merely distributed differently.

Therefore, this term appears in both sides of our inequality and cancels out, allowing us to focus on the essential components of the inequality that determine when the ensemble outperforms the base model.

We can know that $\rho \rightarrow \text{pruning rate} = \left(\frac{n-1}{n}\right)$.

The ensemble loss is given by Equation 12:

$$L_{ensemble} = L - b + \frac{b}{n^a} \quad (29)$$

We need $L_{ensemble} \leq L_0$. So,

$$L_0 + \frac{1}{\rho^\gamma N_0^\delta} - b + \frac{b}{n^a} \leq L_0 \quad (30)$$

This implies:

$$\frac{1}{\delta^\gamma N_0^\delta} - b + \frac{b}{n^a} \leq 0 \quad (31)$$

Substituting $\rho = \frac{n-1}{n}$ and rearranging:

$$\left(\frac{n^a - 1}{n^{a+\gamma}}\right) (n-1)^\gamma \geq \frac{1}{bN_0^\delta} \quad (32)$$

The computational cost of the ensemble is $n \cdot K \cdot (1 - \rho) \cdot N \cdot D = K \cdot N \cdot D$, same as the base model computational cost.