Predicting Task Performance with Context-aware Scaling Laws

Kyle Montgomery[1*], **David Park**[1*], **Jianhong Tu**[1],
Michael Bendersky[2], **Beliz Gunel**[2], **Dawn Song**[3], **Chenguang Wang**[1†]
[1]Washington University in St. Louis, [2]Google DeepMind, [3]UC Berkeley
{kylemontgomery, d.park, chenguangwang}@wustl.edu

Abstract

Scaling laws have transformed our understanding of large language models by linking upstream metrics like cross-entropy loss to design factors such as model size, training data, and compute. However, these conventional laws fail to capture downstream task performance, where context plays a critical role. In this work, we propose a straightforward, interpretable framework that jointly models downstream performance as a function of the training compute and the provided context. We empirically validate our framework by fitting it on the observed downstream performance of extended-context variants of Llama-2-7B and Llama-2-13B across 65,500 unique instances spanning three tasks: arithmetic reasoning, common sense reasoning, and machine translation. Our results demonstrate that our framework accurately models in-distribution downstream performance, generalizes across three orders of magnitude in training compute, and reliably extrapolates performance as the amount of context increases. These findings offer valuable insights into the interplay between training compute and context utilization, providing guidance for designing more efficient long-context LLMs for diverse downstream tasks. Our code is available at https://github.com/wang-research-lab/context-scaling.

1 Introduction

Neural scaling laws (Hestness et al., 2017; Kaplan et al., 2020), which describe how model performance scales with the number of model parameters, the size of the training dataset, or the amount of training compute, have shaped our understanding of how large language models (LLMs) (Brown et al., 2020; Touvron et al., 2023; Team et al., 2024; Grattafiori et al., 2024; OpenAI et al., 2024) improve with increased resources. These findings have guided the design and development of increasingly larger models, providing a blueprint to optimally scale up performance under a fixed compute budget (Hoffmann et al., 2022; OpenAI et al., 2024).

While upstream metrics like cross-entropy loss serve as convenient proxies during model development, in real-world applications, downstream performance often diverges from these upstream trends (Wei et al., 2022; Hu et al., 2024). Accurate upfront performance estimates for downstream tasks can help guide model development and identify emergence or saturation on certain tasks with fewer costly experiments. Existing works on predicting downstream performance often rely on overly complicated, less interpretable methods. For instance, Chen et al. (2024) utilizes a two-stage approach using upstream loss as an intermediary, while Ye et al. (2023) fits a multi-layered perceptron to predict performance on BIG-Bench (Srivastava et al., 2023).

In contrast, we propose a straightforward, interpretable framework that directly models the downstream performance of LLMs across a number of tasks. The key is to jointly model downstream performance as a function of the training compute and the provided context. Specifically, we develop a functional form (see Eq. (1)) which combines two saturating power-law terms (one in the amount of training compute and another in the amount of context) along with a penalty term to account for cases in which the context exceeds the model's context limit. This formulation is motivated by the intuition that downstream performance improves with increased training compute and longer, yet relevant, context until the benefits saturate or the context limit is exceeded. Figure 1 compares our fit to existing methods that do not consider context.

We empirically validate our scaling framework by fitting it on the observed downstream performance of extended-context variants of Llama-2-7B

[*] Equal contribution.
[†] Corresponding author.

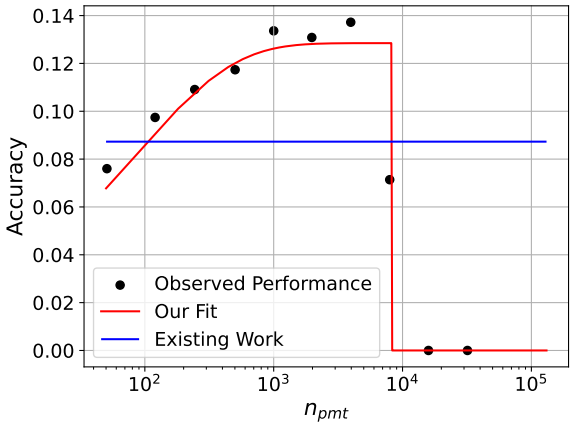


Figure 1: Existing approaches ignore the impact of context length and predict an average performance level regardless of the number of in-context demonstrations. In comparison, our context-aware fit closely tracks the observed performance as additional context is provided.

and Llama-2-13B (Touvron et al., 2023; Peng et al., 2024) across 65,500 unique instances spanning three tasks: arithmetic reasoning, common sense reasoning, and machine translation. Our results demonstrate that our framework accurately predicts downstream performance for both Llama-2-7B and Llama-2-13B (Sec. 4). Furthermore, we find that our fits generalize well on held-out models spanning 3 orders of magnitude in training compute (Sec. 4.1). Similarly, we demonstrate that our fits generalize to longer contexts, even as the context exceeds a model's context limit (Sec. 4.2). Lastly, we show that our fits generalize across different context-extension techniques (Sec. 4.3). These findings offer valuable insights into the interplay between training compute and context utilization, providing guidance for designing more efficient long-context LLMs for diverse downstream tasks.

Our main contributions are threefold:

- We propose a framework that extends conventional neural scaling laws to downstream tasks by incorporating the context length and context limit, providing a more accurate model of LLM performance across varying context lengths.
- We empirically fit this framework to Llama-2 models with extended context windows across 3 tasks: arithmetic reasoning, common sense reasoning, and machine translation. We demonstrate the generality of our approach by showing that our scaling laws hold across 3 orders of magnitude in training compute, 4 orders of magnitude in context length, and across different context-extension techniques.
- Our framework offers an interpretable tool for understanding the interplay between compute, context, and downstream performance, providing insights that can guide the design of future long-context LLMs.

2 Background

Here, we introduce relevant preliminaries, including notation conventions and the process of extending the context window of the Llama-2 models (Touvron et al., 2023).

2.1 Notation

We adopt the following notation:

- $\mathcal{P}$ – aggregate performance on a downstream task. Occasionally, we'll use a subscript to denote the specific task (e.g., $\mathcal{P}_{\text{MT}}$ for machine translation).
- N – the number of model parameters, excluding embedding/de-embedding parameters.
- D – the number of tokens in the training dataset.
- C – the amount of non-embedding training compute. Following Kaplan et al. (2020), we estimate $C \approx 6N$ FLOPs per training token, or $C \approx 6ND$ FLOPs in total.
- n_{ctx} – the context limit of a model in tokens, i.e., the maximum number of positional embeddings computed for any training sequence. Often, we quote numerical values using k to denote units of 1024 tokens. For example, a context limit of "128k" corresponds to $128 \times 1024 = 131072$ tokens.
- n_{pmt} – the length (in tokens) of a given input query or context. For simplicity, n_{pmt} does not include generated/outputted tokens.

2.2 Extending Llama-2's Context Limit

Because the complexity of the self-attention layers grows quadratically in the sequence length (Duman Keles et al., 2023; Dao et al., 2022), LLMs are commonly pre-trained on short sequences (e.g., 4k tokens) rather than long sequences (e.g., 128k

Base Model	Non-embedding Params (N)	Context Limit (n_{ctx})	Dataset Size (D)	Training Compute (C)
Llama-2-7B	6,476,271,616	4k	2.0T	7.7719×10^{22}
		8k	2.0T + 0.210B	7.7723×10^{22}
		16k	2.0T + 0.419B	7.7732×10^{22}
		32k	2.0T + 0.836B	7.7748×10^{22}
		64k	2.0T + 1.678B	7.7780×10^{22}
		128k	2.0T + 3.355B	7.7846×10^{22}
Llama-2-13B	12,688,184,320	4k	2.0T	1.5227×10^{23}
		8k	2.0T + 0.210B	1.5227×10^{23}
		16k	2.0T + 0.419B	1.5229×10^{23}
		32k	2.0T + 0.836B	1.5232×10^{23}
		64k	2.0T + 1.678B	1.5239×10^{23}
		128k	2.0T + 3.355B	1.5251×10^{23}

Table 1: The 12 checkpoints against which we fit scaling curves. The 4k variants are the official Llama-2-7B and Llama-2-13B checkpoints. The additional training tokens and compute from extending the context limit via YaRN (Peng et al., 2024) are factored into D and C.

tokens). As a result, LLMs struggle to generalize to sequences longer than those seen during pre-training. Because we plan to explore how downstream performance varies with context length, Llama-2's original context limit of 4k tokens will not be sufficient. Fortunately, a number of techniques have been proposed that can extend the context window of LLMs for a fraction of the pre-training compute budget (Chen et al., 2023; Peng et al., 2024; Xiong et al., 2024).

YaRN (Peng et al., 2024) is our method of choice for extending Llama 2's context limit. We selected YaRN due to its high compute efficiency and strong empirical results compared to other techniques. YaRN involves fine-tuning the pre-trained model for a limited number of steps on sequences exceeding the pre-trained LLM's context limit in order to increase the effective size of the LLM's context limit so that it may better model long sequences.

We adopt the methodology from Peng et al. (2024) and fine-tune Llama-2-7B and Llama-2-13B (Touvron et al., 2023) for 400 steps with a global batch size of 64 on sequences of length n'_{ctx} (where $n'_{\text{ctx}} > n_{\text{ctx}}$) from the PG-19 corpus (Rae et al., 2020). We use the AdamW optimizer (Loshchilov and Hutter, 2019) with $\beta_1 = 0.9$ and $\beta_2 = 0.95$, and a learning rate of 2×10^{-5}. We train variants of Llama-2-7B and Llama-2-13B with $n_{\text{ctx}} \in \{8\text{k}, 16\text{k}, 32\text{k}\}$, and source checkpoints for $n_{\text{ctx}} \in \{64\text{k}, 128\text{k}\}$ from Peng et al. (2024).

In order to validate the effectiveness of the context extension training, we evaluate the performance of our 12 Llama-2 models in Table 1 on RULER (Hsieh et al., 2024), a synthetic needle-in-a-haystack benchmark developed to evaluate long-context LLMs. Specifically, we evaluate each model on 100 instances per length, for each of RULER's 13 tasks. Results are displayed in Table 2 and suggest that context extension via YaRN (Peng et al., 2024) is somewhat effective. Interestingly, models tend to underperform when evaluated at their extended context limit, suggesting that training with a context limit well beyond the target evaluation range can lead to improved performance within that desired range.

3 Method

We posit that aggregate task performance $\mathcal{P}$ can be modeled as the product of two saturating power laws in C and n_{pmt}, with a sigmoid penalty term for when $n_{\text{pmt}} > n_{\text{ctx}}$. This form provides a good fit for a range of tasks, including arithmetic reasoning, common sense reasoning, and machine translation tasks. Formally, we model $\mathcal{P}$ as

$$\mathcal{P}(C, n_{\text{pmt}}, n_{\text{ctx}}) = \overbrace{\left[1 - \exp\left(-A\left(\frac{C}{C^c}\right)^{\alpha}\right)\right]}^{\text{Saturating term in } C} \times \underbrace{\left[1 - \exp\left(-B\left(\frac{n_{\text{pmt}}}{n^c_{\text{pmt}}}\right)^{\beta}\right)\right]}_{\text{Saturating term in } n_{\text{pmt}}} \times \underbrace{\sigma\left(n_{\text{pmt}} - n_{\text{ctx}}\right)}_{\text{Penalty term}}, \quad (1)$$

where A, C^c, α, B, n^c_{pmt}, and β are parameters to be optimized.

We select this form because we expect that the downstream performance $\mathcal{P}$ is proportional to diminishing terms in the amount of training compute C (which integrates both model size N and dataset size D) (Chen et al., 2024; Owen, 2024) and the context length (Brown et al., 2020; Caballero et al., 2023), assuming the context remains relevant as its

Model	n_{ctx}	$n_{\text{pmt}} = 4\text{k}$	$n_{\text{pmt}} = 8\text{k}$	$n_{\text{pmt}} = 16\text{k}$	$n_{\text{pmt}} = 32\text{k}$	$n_{\text{pmt}} = 64\text{k}$	$n_{\text{pmt}} = 128\text{k}$
Llama-2-7B	4k	0.822	0.000	0.000	0.000	0.000	0.000
	8k	0.829	0.586	0.000	0.000	0.001	0.005
	16k	0.795	0.58	0.378	0.000	0.000	0.002
	32k	0.746	0.599	0.517	0.317	0.000	0.000
	64k	0.794	0.647	0.593	0.530	0.225	0.000
	128k	0.776	0.663	0.552	0.439	0.383	0.129
Llama-2-13B	4k	0.861	0.000	0.000	0.000	0.000	0.000
	8k	0.870	0.625	0.000	0.000	0.000	0.000
	16k	0.865	0.679	0.392	0.000	0.000	0.000
	32k	0.848	0.727	0.622	0.378	0.000	0.000
	64k	0.860	0.734	0.612	0.511	0.282	0.001
	128k	0.819	0.684	0.586	0.484	0.447	0.163

Table 2: Accuracy of our extended Llama-2 models on RULER (Hsieh et al., 2024).

length increases and $n_{\text{pmt}} \leq n_{\text{ctx}}$. We saturate these terms via exponentiation to ensure our predicted performance remains below the maximum theoretical performance of 1.0. The product form arises because compute and context are complementary, not additive; a significant lack in one dimension limits the benefit derived from the other. For example, providing more context is only beneficial to the extent that the model is capable of leveraging that additional context. We impose a sharp sigmoid penalty term because $\mathcal{P}$ is measured only on the generated tokens, and if $n_{\text{pmt}} > n_{\text{ctx}}$, then any generated tokens will fall beyond the range in which the model can make reliable predictions, meaning $\mathcal{P}$ degrades rapidly, especially on tasks that require extended and coherent generations (e.g., reasoning through a math word problem or translating an entire sentence).

3.1 Datasets

We evaluate our 12 models in Table 1 on 65,500 instances of varying lengths that span 3 tasks:

- **Arithmetic reasoning** We collect 3550 testing instances across GSM8K (Cobbe et al., 2021), MATH (Hendrycks et al., 2021), AQUA-RAT (Ling et al., 2017), and Deepmind Math (Saxton et al., 2019). Because the instances are rather short, we pack the context with up to 511 demonstrations sampled from the training splits of each dataset.
- **Common sense reasoning** We sample 1750 testing instances across PIQA (Bisk et al., 2020), SIQA (Sap et al., 2019), OpenBookQA (Mihaylov et al., 2018), HellaSwag (Zellers et al., 2019), WinoGrande (Sakaguchi et al., 2020), ARC-Easy/Challenge (Clark et al., 2018), and CommonSenseQA (Talmor et al., 2019), and pack the context with up to 511 demonstrations from their respective training splits.
- **Machine translation** We sample 250 translation instances from WMT-14 (Bojar et al., 2014) from each of German, French, Hindi, Czech, and Russian to English. As before, we pack the context with up to 511 demonstrations (of the same source language) and measure the BLEU-4 (Papineni et al., 2002) score of the generation against the reference translation.

Additional details can be found in Appendix A.

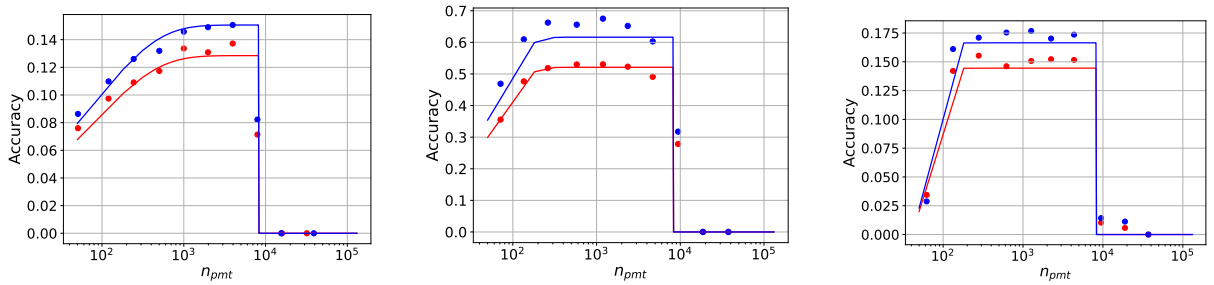
3.2 Fitting Procedure

For each task, we aggregate the results for each model by the context length, using the number of in-context demonstrations as a proxy for length. Within each group, we average over the context length and metric value for each instance. In doing so, we collect a number of records of the form (C, n_{pmt}, n_{ctx}, avg. metric value) on which we fit Eq. (1) for each of our 3 tasks.

To fit the scaling curves, we use a two-stage optimization procedure that combines global search with local refinement. First, we use an out-of-the-box global optimizer to perform a broad

Parameter	Lower Bound	Upper Bound
A	0	100
C^c	0	10^{30}
α	0	10
B	0	100
n^c_{pmt}	0	131,072
β	0	10

Table 3: Upper and lower bounds on A, C^c, α, B, n^c_{pmt}, and β.



(a) Arithmetic reasoning. (b) Common sense reasoning. (c) Machine translation.

Figure 2: Contours of fits at $C = 7.8 \times 10^{22}$ (red) and $C = 1.5 \times 10^{23}$ (blue) for $n_{\text{ctx}} = 8\text{k}$ on three tasks: arithmetic reasoning (left), common sense reasoning (middle) and machine translation (right).

Task	A	C^c	α	B	n^c_{pmt}	β
Arithmetic reasoning	9.96	9.7×10^{29}	0.26	62.24	1.3×10^5	0.56
Common sense reasoning	99.39	1.5×10^{28}	0.40	96.31	3.5×10^3	1.12
Machine translation	5.55	5.4×10^{29}	0.23	31.82	3.0×10^2	2.97

Table 4: Fits for $\mathcal{P}(C, n_{\text{pmt}}, n_{\text{ctx}})$ on 3 downstream tasks: arithmetic reasoning, common sense reasoning, and machine translation.

search over the parameter space. Specifically, we use SciPy's `differential_evolution` global optimization method, an evolutionary algorithm well suited for non-convex, non-linear optimization problems such as this (Storn and Price, 1997). We define finite upper and lower bounds for each parameter, informed by Kaplan et al. (2020) and Xiong et al. (2024). We use the same bounds across all tasks, which are listed in Table 3. Finally, we do a pass through a local optimizer (SciPy's `curve_fit`), using the estimate from the global optimizer as a starting point, to achieve a precise fit.

4 Empirical Results

We model the aggregate performance $\mathcal{P}$ on each of our 3 tasks (arithmetic reasoning, common sense reasoning, and machine translation) using Eq. (1). Unless otherwise noted, scaling laws are fit on the results of all 12 Llama-2 models in Table 1 using the procedure outlined in Section 3.2. Table 4 includes the parameter values which we found to be optimal for each task. Contours of our fits at $C = 7.8 \times 10^{22}$ and $C = 1.5 \times 10^{23}$ for $n_{\text{ctx}} = 8\text{k}$ are provided in Figure 2. Additional contours are provided in Appendix B. We report the mean absolute prediction error $|\mathcal{P} - \hat{\mathcal{P}}|$, which is the average of residuals (in absolute value). When discussing individual residuals, we'll often include the sign of the residual to indicate the direction (i.e., whether we're under- or over-predicting).

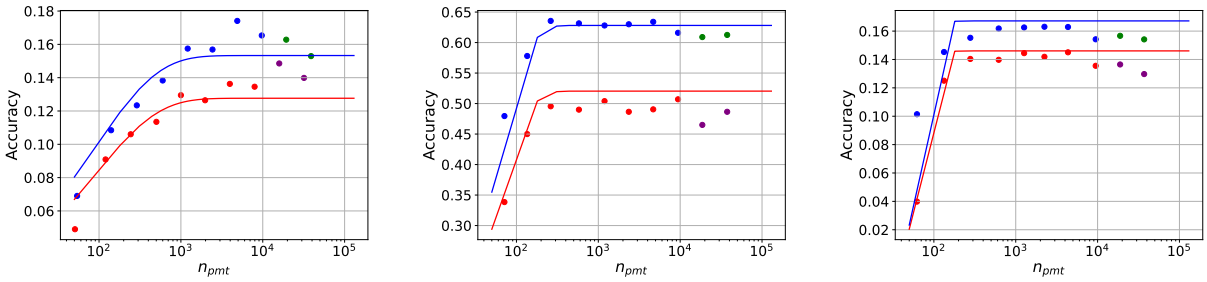
On the arithmetic reasoning task, we achieve an excellent fit, with an average prediction error $|\mathcal{P} - \hat{\mathcal{P}}|$ of just 0.010. Similarly, on common sense reasoning and machine translation, we observe average prediction errors of 0.037 and 0.007, respectively. Additionally, we model the behavior around the boundary condition at $n_{\text{pmt}} = n_{\text{ctx}}$ surprisingly well.

Our results confirm that $\mathcal{P}$ can be jointly determined by the training compute and context length. Increasing C corresponds to an increase in $\mathcal{P}$, in effect shifting up the contour by some diminishing amount in C in the region where $n_{\text{pmt}} < n_{\text{ctx}}$. Similarly, increasing n_{pmt} when n_{pmt} is small leads to significant gains in $\mathcal{P}$, which diminish (sub-linearly for arithmetic reasoning and super-linearly for common sense reasoning and machine translation) and saturate quickly. In the context of our task construction, this makes a lot of sense; the first few in-context task demonstrations go far in improving the generated responses, but once the model has seen enough context to sufficiently capture the task structure, additional demonstrations provide little marginal benefit (Brown et al., 2020). Additionally, the optimal number of demonstrations is task-dependent; our results suggest that models make better use of additional demonstrations on arithmetic reasoning tasks than they do on common sense reasoning or machine translation tasks.

The remainder of this section aims to study the extent to which our fits generalize to out-of-

Model	C	n_{ctx}	$\mathcal{P}_{\text{AR}} - \hat{\mathcal{P}}_{\text{AR}}$	$\mathcal{P}_{\text{CSR}} - \hat{\mathcal{P}}_{\text{CSR}}$	$\mathcal{P}_{\text{MT}} - \hat{\mathcal{P}}_{\text{MT}}$
Qwen-2.5-0.5B	3.8×10^{22}	32k	+0.057	+0.008	-0.057
Gemma-2-2B	2.4×10^{22}	4k	+0.066	+0.260	+0.059
Gemma-2-9B	4.0×10^{23}	4k	+0.069	+0.051	+0.017
Gemma-2-27B	2.0×10^{24}	4k	+0.024	-0.099	-0.054
Llama-2-70B	8.2×10^{23}	4k	-0.002	-0.031	-0.025

Table 5: Generalization of fit on test models for arithmetic reasoning (AR), common sense reasoning (CSR), and machine translation (MT).



(a) Arithmetic reasoning. (b) Common sense reasoning. (c) Machine translation.

Figure 3: Contours of fits at $C = 7.8 \times 10^{22}$ (red) and $C = 1.5 \times 10^{23}$ (blue) for $n_{\text{ctx}} = 128\text{k}$ on three tasks: arithmetic reasoning (left), common sense reasoning (middle) and machine translation (right). Held-out observations are colored in purple and green for Llama-2-7b and Llama-2-13b, respectively.

distribution amounts of training compute (Section 4.1), context length (Section 4.2), and context-extension method (Section 4.3). Finally, Section 4.4 analyzes the role of the sigmoid penalty term.

4.1 Generalization along C

Our scaling laws are fit over a narrow range of C, specifically $7.8 \times 10^{22} \leq C \leq 1.5 \times 10^{23}$. To test how well our fits generalize outside of this range, we evaluate several testing models (namely, Qwen2.5-0.5B (Yang et al., 2025), Gemma-2 (Team et al., 2024), and Llama-2-70B (Touvron et al., 2023)) ranging between 0.5B to 70B parameters and spanning 3 orders of magnitude in C. We evaluate these models at their respective context limits[†], and report the prediction error on each task in Table 5.

We observe good generalization across these 5 testing models, with many of the prediction errors falling near or below 5 points. Interestingly, our fits generalize the worst to Gemma-2-2B, despite generalizing well to Gemma-2-9B and Gemma-2-27B. Moreover, we achieve stronger generalization on arithmetic reasoning and machine translation tasks compared to common sense reasoning, which aligns with our in-distribution results. Finally, these results suggest we tend to underestimate the performance when C is small, and slightly overestimate performance when C is large.

4.2 Generalization along n_{pmt}

In order to measure how well our scaling laws generalize to longer contexts, we refit our scaling curves, this time holding out observations where the context length exceeds 10,000 tokens. Figure 3 displays contours of our fits at $C = 7.8 \times 10^{22}$ and $C = 1.5 \times 10^{23}$ for $n_{\text{ctx}} = 128\text{k}$ for each task. Again, we see strong generalization along n_{pmt}, achieving prediction errors of just 0.017, 0.067, and 0.006 across the held-out observations on arithmetic reasoning, common sense reasoning, and machine translation, respectively. These low error rates across diverse tasks demonstrate that our joint scaling framework can reliably extrapolate to longer context lengths, making it particularly suitable for long-context LLM design.

Interestingly, on common sense reasoning and machine translation tasks, we observe that $\mathcal{P}$ is inversely proportional to n_{ctx} for some fixed n_{pmt}. That is, as we extend the context, performance slightly worsens. We hypothesize that this decline is not due to an intrinsic scaling trend but rather because the training mix used to extend the context is

[†]While Gemma-2 has a context limit of 8k tokens, it uses sliding window attention for every odd layer. Since this behavior is not supported in vLLM (Kwon et al., 2023), we treat Gemma-2 as if its context limit is 4k tokens.

misaligned with these tasks. For example, our training mix is sourced from PG-19 (Rae et al., 2020), which includes predominantly English text, so it's unsurprising that machine translation performance worsens with increased training.

4.3 Does the choice of context extension technique matter?

A number of different techniques have been proposed for extending the context length of a model with rotary positional embeddings (Chen et al., 2023; Peng et al., 2024; Xiong et al., 2024). It's natural to wonder how sensitive our fit scaling curves are to one's choice of context extension technique. To test this, we evaluate Together's Llama-2-7B model (Together.ai, 2023) extended to 32k context via positional interpolation (Chen et al., 2023). We evaluate this model at its context limit of 32k tokens across our 3 tasks and compute the prediction error for each, that is, the difference between the observed performance $\mathcal{P}$ and the predicted performance $\hat{\mathcal{P}}$. We compare against the prediction error on the Llama-2-7B checkpoint extended to $n_{\text{ctx}} = 32\text{k}$ via YaRN (Peng et al., 2024). It's worth noting the training mix and quantity are different between these two models; Together's was trained on 1.5B tokens of a diverse data mix, while we follow Peng et al. (2024) and train on just 0.836B tokens from PG-19 (Rae et al., 2020). Still, our compute estimates for both models are sufficiently similar (7.777×10^{22} vs 7.775×10^{22} FLOPs, respectively).

Table 6 lists the results. In general, the prediction errors we observe on Together's Llama-2-7B model extended via positional interpolation are similar to the prediction errors we observe on our Llama-2-7B model extended via YaRN. These results suggest that the choice of context extension technique has little impact on the scaling properties of downstream performance.

4.4 Ablation over the penalty term

To quantify the impact of our sigmoid penalty for prompt lengths exceeding the model's context limit, we fit Eq. (1) on the arithmetic reasoning task with and without the penalty term. Table 7 reports the resulting prediction errors. We observe that without the penalty term, the fit underestimates performance when $n_{\text{pmt}} \leq n_{\text{ctx}}$ and overestimates performance when $n_{\text{pmt}} > n_{\text{ctx}}$, confirming the importance of the penalty term.

5 Related Work

Hestness et al. (2017) and Kaplan et al. (2020) introduce scaling laws which describe the relationship between upstream model performance (e.g., cross-entropy loss) and model design features (e.g., the number of model parameters, the size of the training dataset, or the total amount of training compute). Henighan et al. (2020) extends this analysis to other types of autoregressive models (e.g., generative image and video modeling). Hoffmann et al. (2022) and OpenAI et al. (2024) describe the use of scaling laws to train compute-optimal LLMs, and Caballero et al. (2023) introduces a form of smoothly broken neural scaling laws to better capture non-monotonic scaling.

Several works have focused on scaling laws for predicting downstream performance. Wei et al. (2022) and Hu et al. (2024) focus on predicting abilities that "emerge" in LLMs when trained on enough compute. Isik et al. (2024) explores scaling laws for transfer learning on machine translation tasks, while Schaeffer et al. (2025) studies scaling laws for downstream multiple-choice tasks. Other works have employed a collaborative approach and source performance data from public benchmarks to better generalize across different model families (Zhang et al., 2024; Ruan et al., 2024; Polo et al., 2025; Gadre et al., 2024). Chen et al. (2024) and Ruan et al. (2024) employ a two-stage approach, using an intermediary (e.g., upstream loss) for predicting downstream performance. Both Owen (2024) and Ye et al. (2023) aim to predict aggregate performance on benchmarks such as BIG-Bench (Srivastava et al., 2023). Comparatively, this work introduces a dependence on the context length and suggests that you can predict downstream performance and obtain strong generalization (even across model families) with a straightforward, interpretable functional form.

Both Kaplan et al. (2020) and Caballero et al. (2023) briefly explore the scaling of upstream performance as it relates to context length. Xiong et al. (2024) extends the context limit of Llama-2 and finds that validation loss scales as a power law in the context length, but stops short of exploring the relationship between downstream performance and context length. Caballero et al. (2023) and Brown et al. (2020) explore the diminishing returns of increasing the number of in-context demonstrations. To the best of our knowledge, our work is the first to explicitly focus on the scaling relationship be-

Model	C	n_{pmt}	$\mathcal{P}_{\text{AR}} - \hat{\mathcal{P}}_{\text{AR}}$	$\mathcal{P}_{\text{CSR}} - \hat{\mathcal{P}}_{\text{CSR}}$	$\mathcal{P}_{\text{MT}} - \hat{\mathcal{P}}_{\text{MT}}$
Llama-2-7B (PI)	7.777×10^{22}	32k	+0.014	+0.079	-0.005
Llama-2-7B (YaRN)	7.775×10^{22}	32k	+0.005	+0.014	-0.005

Table 6: Generalization of fit on test models for arithmetic reasoning (AR), common sense reasoning (CSR), and machine translation (MT) at $n_{\text{ctx}} = 32$k.

	$\lvert P - \hat{P}\rvert_{n_{\text{pmt}} \leq n_{\text{ctx}}}$	$\lvert P - \hat{P}\rvert_{n_{\text{pmt}} > n_{\text{ctx}}}$	$\lvert P - \hat{P}\rvert$
With penalty term	0.010	0.014	0.010
Without penalty term	0.019	0.104	0.029

Table 7: Prediction errors on the arithmetic reasoning task, with and without the sigmoid penalty term.

tween downstream performance and context length, and the first attempt to unify the understanding of scaling with respect to both context and compute.

The ability of an LLM to extrapolate to longer sequences depends heavily on its positional encodings. While some positional encoding techniques (e.g., ALiBi (Press et al., 2022)) offer limited length extrapolation, other common techniques (e.g., RoPE (Su et al., 2024)) don't. As a result, a number of techniques to efficiently extend the context window of LLMs have been proposed.

Some techniques offer training-free context extension, typically by adjusting the attention mechanism itself. Jin et al. (2024) leverages a bi-level attention mechanism, applying standard self-attention to adjacent tokens and grouped attention for distant tokens. InfLLM is a memory-based technique that integrates sliding-window attention with block-level context memory (Xiao et al., 2024). Similarly, LM-Infinite employs a Λ-shaped attention mask, effectively masking attention over tokens in the middle, and restricts the maximum positional difference between any two tokens to the maximum sequence length seen during pre-training (Han et al., 2024). On the other hand, An et al. (2024) introduces dual-chunk attention, which decomposes the attention computation into chunk-based modules to better capture the relative positional information between distant tokens.

Additionally, a number of techniques have been proposed that focus on rescaling the positional encodings. Concurrently, Chen et al. (2023) and kaiokendev (2023) introduced position interpolation, which extends the context window by linearly interpolating the position indices to be within the pre-trained context limit. Xiong et al. (2024) proposes decreasing the rotational angle (base frequency) of RoPE to prevent the relative positional information from decaying. Building on this, NTK-aware interpolation (bloc97, 2023b) adjusts the scaling for each RoPE dimension based on its frequency, thereby mitigating the loss of high-frequency details. bloc97 (2023a) introduces NTK-by-parts interpolation, which selectively interpolates lower-frequency dimensions while preserving higher-frequency components to maintain local relative positioning. YaRN (Peng et al., 2024) combines NTK-by-parts with a mechanism to rescale the logits in the attention softmax to further improve performance on long sequences. In this work, we utilize YaRN to extend the context limit of the Llama-2 models due to its high compute efficiency and strong empirical results compared to other techniques.

6 Conclusion

In this work, we introduce a straightforward, interpretable framework that jointly models downstream performance as a function of the training compute and the provided context. Extensive experiments on arithmetic reasoning, common-sense reasoning, and machine translation tasks demonstrate that our framework not only fits the in-distribution performance accurately but also generalizes well across 3 orders of magnitude in the amount of non-embedding training compute C, 4 orders of magnitude in the amount of input context length, and even to other context-extension techniques. These findings reveal that downstream performance benefits from increased compute and longer, relevant context, but only up to a saturation point. Our work thus provides actionable insights for designing more effective long-context LLMs and bridges the gap between upstream scaling metrics and real-world task performance.

Limitations

While our proposed context-aware scaling framework provides an interpretable approach to modeling downstream performance, it does come with limitations. Specifically, our formulation relies on a set of assumptions (e.g., performance scales with training compute and context) that may not hold under extreme scaling regimes or in the presence of adversarial attacks like many-shot jailbreaking (Anil et al., 2024). Moreover, factors such as the pre-training data mix, post-training and alignment, and architectural choices, which can all influence downstream model performance, are not explicitly accounted for. However, these factors likely affect the optimal parameters of a fit without necessarily changing the structure of Eq. (1). For example, post-training alignment (e.g., instruction tuning) might improve a model's zero-shot performance, resulting in a higher value for the parameter A compared to a non-aligned base model. Future work could investigate how these factors and others influence the identified parameters, enhancing the framework's predictive power while retaining its interpretable form. Lastly, our scaling curves are fit to a narrow range of training compute, and may fail to generalize well to LLMs trained on an amount of compute that extends far beyond this range.

References

Chenxin An, Fei Huang, Jun Zhang, Shansan Gong, Xipeng Qiu, Chang Zhou, and Lingpeng Kong. 2024. Training-free long-context scaling of large language models. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Cem Anil, Esin DURMUS, Nina Panickssery, Mrinank Sharma, Joe Benton, Sandipan Kundu, Joshua Batson, Meg Tong, Jesse Mu, Daniel Ford, Francesco Mosconi, Rajashree Agrawal, Rylan Schaeffer, Naomi Bashkansky, Samuel Svenningsen, Mike Lambert, Ansh Radhakrishnan, Carson Denison, Evan Hubinger, Yuntao Bai, Trenton Bricken, Timothy Maxwell, Nicholas Schiefer, James Sully, Alex Tamkin, Tamera Lanham, Karina Nguyen, Tomek Korbak, Jared Kaplan, Deep Ganguli, Samuel Bowman, Ethan Perez, Roger B Grosse, and David K Duvenaud. 2024. Many-shot jailbreaking. In *Advances in Neural Information Processing Systems*, volume 37, pages 129696–129742. Curran Associates, Inc.

Yonatan Bisk, Rowan Zellers, Ronan Le bras, Jianfeng Gao, and Yejin Choi. 2020. Piqa: Reasoning about physical commonsense in natural language. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):7432–7439.

bloc97. 2023a. Add NTK-Aware interpolation "by parts" correction.

bloc97. 2023b. NTK-Aware Scaled RoPE allows LLaMA models to have extended (8k+) context size without any fine-tuning and minimal perplexity degradation.

Ondrej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Ale s Tamchyna. 2014. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 12–58, Baltimore, Maryland, USA. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Ethan Caballero, Kshitij Gupta, Irina Rish, and David Krueger. 2023. Broken neural scaling laws. In *ICLR 2023 Workshop on Mathematical and Empirical Understanding of Foundation Models*.

Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. 2023. Extending context window of large language models via positional interpolation. *Preprint*, arXiv:2306.15595.

Yangyi Chen, Binxuan Huang, Yifan Gao, Zhengyang Wang, Jingfeng Yang, and Heng Ji. 2024. Scaling laws for predicting downstream performance in llms. *Preprint*, arXiv:2410.08527.

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

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

Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. Flashattention: Fast and memory-efficient exact attention with io-awareness. In *Advances in Neural Information Processing Systems*, volume 35, pages 16344–16359. Curran Associates, Inc.

Feyza Duman Keles, Pruthuvi Mahesakya Wijewardena, and Chinmay Hegde. 2023. On the computational complexity of self-attention. In *Proceedings of The 34th International Conference on Algorithmic Learning Theory*, volume 201 of *Proceedings of Machine Learning Research*, pages 597–619. PMLR.

Samir Yitzhak Gadre, Georgios Smyrnis, Vaishaal Shankar, Suchin Gururangan, Mitchell Wortsman, Rulin Shao, Jean Mercat, Alex Fang, Jeffrey Li, Sedrick Keh, Rui Xin, Marianna Nezhurina, Igor Vasiljevic, Jenia Jitsev, Luca Soldaini, Alexandros G. Dimakis, Gabriel Ilharco, Pang Wei Koh, Shuran Song, Thomas Kollar, Yair Carmon, Achal Dave, Reinhard Heckel, Niklas Muennighoff, and Ludwig Schmidt. 2024. Language models scale reliably with over-training and on downstream tasks. *Preprint*, arXiv:2403.08540.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry As-

pegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Chi Han, Qifan Wang, Hao Peng, Wenhan Xiong, Yu Chen, Heng Ji, and Sinong Wang. 2024. LM-infinite: Zero-shot extreme length generalization for large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3991–4008, Mexico City, Mexico. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the MATH dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Tom Henighan, Jared Kaplan, Mor Katz, Mark Chen, Christopher Hesse, Jacob Jackson, Heewoo Jun, Tom B. Brown, Prafulla Dhariwal, Scott Gray, Chris Hallacy, Benjamin Mann, Alec Radford, Aditya Ramesh, Nick Ryder, Daniel M. Ziegler, John Schulman, Dario Amodei, and Sam McCandlish. 2020. Scaling laws for autoregressive generative modeling. *Preprint*, arXiv:2010.14701.

Joel Hestness, Sharan Narang, Newsha Ardalani, Gregory Diamos, Heewoo Jun, Hassan Kianinejad, Md. Mostofa Ali Patwary, Yang Yang, and Yanqi Zhou. 2017. Deep learning scaling is predictable, empirically. *Preprint*, arXiv:1712.00409.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. 2022. Training compute-optimal large language models. *Preprint*, arXiv:2203.15556.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, and Boris Ginsburg. 2024. RULER: What's the real context size of your long-context language models? In *First Conference on Language Modeling*.

Shengding Hu, Xin Liu, Xu Han, Xinrong Zhang, Chaoqun He, Weilin Zhao, Yankai Lin, Ning Ding, Zebin Ou, Guoyang Zeng, Zhiyuan Liu, and Maosong Sun. 2024. Predicting emergent abilities with infinite resolution evaluation. In *The Twelfth International Conference on Learning Representations*.

Berivan Isik, Natalia Ponomareva, Hussein Hazimeh, Dimitris Paparas, Sergei Vassilvitskii, and Sanmi Koyejo. 2024. Scaling laws for downstream task performance of large language models. *Preprint*, arXiv:2402.04177.

Hongye Jin, Xiaotian Han, Jingfeng Yang, Zhimeng Jiang, Zirui Liu, Chia-Yuan Chang, Huiyuan Chen,

and Xia Hu. 2024. LLM maybe LongLM: SelfExtend LLM context window without tuning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 22099–22114. PMLR.

kaiokendev. 2023. Things I'm learning while training superhot.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *Preprint*, arXiv:2001.08361.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 158–167, Vancouver, Canada. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2381–2391, Brussels, Belgium. Association for Computational Linguistics.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Bar-

ret Zoph. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

David Owen. 2024. How predictable is language model benchmark performance? *Preprint*, arXiv:2401.04757.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. 2024. YaRN: Efficient context window extension of large language models. In *The Twelfth International Conference on Learning Representations*.

Felipe Maia Polo, Seamus Somerstep, Leshem Choshen, Yuekai Sun, and Mikhail Yurochkin. 2025. Sloth: scaling laws for llm skills to predict multi-benchmark performance across families. *Preprint*, arXiv:2412.06540.

Ofir Press, Noah Smith, and Mike Lewis. 2022. Train short, test long: Attention with linear biases enables input length extrapolation. In *International Conference on Learning Representations*.

Jack W. Rae, Anna Potapenko, Siddhant M. Jayakumar, Chloe Hillier, and Timothy P. Lillicrap. 2020. Compressive transformers for long-range sequence modelling. In *International Conference on Learning Representations*.

Yangjun Ruan, Chris J. Maddison, and Tatsunori Hashimoto. 2024. Observational scaling laws and the predictability of language model performance. *Preprint*, arXiv:2405.10938.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Winogrande: An adversarial winograd schema challenge at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8732–8740. AAAI Press.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. 2019. Social IQa: Commonsense reasoning about social interactions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4463–4473, Hong Kong, China. Association for Computational Linguistics.

David Saxton, Edward Grefenstette, Felix Hill, and Pushmeet Kohli. 2019. Analysing mathematical reasoning abilities of neural models. In *International Conference on Learning Representations*.

Rylan Schaeffer, Hailey Schoelkopf, Brando Miranda, Gabriel Mukobi, Varun Madan, Adam Ibrahim, Herbie Bradley, Stella Biderman, and Sanmi Koyejo. 2025. Why has predicting downstream capabilities of frontier ai models with scale remained elusive? *Preprint*, arXiv:2406.04391.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R. Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W. Kocurek, Ali Safaya, Ali Tazarv, Alice Xiang, Alicia Parrish, Allen Nie, Aman Hussain, Amanda Askell, Amanda Dsouza, Ambrose Slone, Ameet Rahane, Anantharaman S. Iyer, Anders Andreassen, Andrea Madotto, Andrea Santilli, Andreas Stuhlmüller, Andrew Dai, Andrew La, Andrew Lampinen, Andy Zou, Angela Jiang, Angelica Chen, Anh Vuong, Animesh Gupta, Anna Gottardi, Antonio Norelli, Anu Venkatesh, Arash Gholamidavoodi, Arfa Tabassum, Arul Menezes, Arun Kirubarajan, Asher Mullokandov, Ashish Sabharwal, Austin Herrick, Avia Efrat, Aykut Erdem, Ayla Karakaş, B. Ryan Roberts, Bao Sheng Loe, Barret Zoph, Bartłomiej Bojanowski, Batuhan Özyurt, Behnam Hedayatnia, Behnam Neyshabur, Benjamin Inden, Benno Stein, Berk Ekmekci, Bill Yuchen Lin, Blake Howald, Bryan Orinion, Cameron Diao, Cameron Dour, Catherine Stinson, Cedrick Argueta, César Ferri Ramírez, Chandan Singh, Charles Rathkopf, Chenlin Meng, Chitta Baral, Chiyu Wu, Chris Callison-Burch, Chris Waites, Christian Voigt, Christopher D. Manning, Christopher Potts, Cindy Ramirez, Clara E. Rivera, Clemencia Siro, Colin Raffel, Courtney Ashcraft, Cristina Garbacea, Damien Sileo, Dan Garrette, Dan Hendrycks, Dan Kilman, Dan Roth, Daniel Freeman, Daniel Khashabi, Daniel Levy, Daniel Moseguí González, Danielle Perszyk, Danny Hernandez, Danqi Chen, Daphne Ippolito, Dar Gilboa, David Dohan, David Drakard, David Jurgens, Debajyoti Datta, Deep Ganguli, Denis Emelin, Denis Kleyko, Deniz Yuret, Derek Chen, Derek Tam, Dieuwke Hupkes, Diganta Misra, Dilyar Buzan, Dimitri Coelho Mollo, Diyi Yang, Dong-Ho Lee, Dylan Schrader, Ekaterina Shutova, Ekin Dogus Cubuk, Elad Segal, Eleanor Hagerman, Elizabeth Barnes, Elizabeth Donoway, Ellie Pavlick, Emanuele Rodola, Emma Lam, Eric Chu, Eric Tang, Erkut Erdem, Ernie Chang, Ethan A. Chi, Ethan Dyer, Ethan Jerzak, Ethan Kim, Eunice Engefu Manyasi, Evgenii Zheltonozhskii, Fanyue Xia, Fatemeh Siar, Fernando Martínez-Plumed, Francesca Happé, Francois Chollet, Frieda Rong, Gaurav Mishra, Genta Indra Winata, Gerard de Melo, Germán Kruszewski, Giambattista Parascandolo, Giorgio Mariani, Gloria Wang, Gonzalo Jaimovitch-López, Gregor Betz, Guy Gur-Ari, Hana Galijasevic, Hannah Kim, Hannah Rashkin, Hannaneh Hajishirzi, Harsh Mehta, Hayden Bogar, Henry Shevlin, Hinrich Schütze, Hiromu Yakura, Hongming Zhang, Hugh Mee Wong, Ian Ng, Isaac Noble, Jaap Jumelet, Jack Geissinger, Jackson Kernion, Jacob Hilton, Jaehoon Lee, Jaime Fernández Fisac, James B. Simon,

James Koppel, James Zheng, James Zou, Jan Kocoń, Jana Thompson, Janelle Wingfield, Jared Kaplan, Jarema Radom, Jascha Sohl-Dickstein, Jason Phang, Jason Wei, Jason Yosinski, Jekaterina Novikova, Jelle Bosscher, Jennifer Marsh, Jeremy Kim, Jeroen Taal, Jesse Engel, Jesujoba Alabi, Jiacheng Xu, Jiaming Song, Jillian Tang, Joan Waweru, John Burden, John Miller, John U. Balis, Jonathan Batchelder, Jonathan Berant, Jörg Frohberg, Jos Rozen, Jose Hernandez-Orallo, Joseph Boudeman, Joseph Guerr, Joseph Jones, Joshua B. Tenenbaum, Joshua S. Rule, Joyce Chua, Kamil Kanclerz, Karen Livescu, Karl Krauth, Karthik Gopalakrishnan, Katerina Ignatyeva, Katja Markert, Kaustubh D. Dhole, Kevin Gimpel, Kevin Omondi, Kory Mathewson, Kristen Chiafullo, Ksenia Shkaruta, Kumar Shridhar, Kyle McDonell, Kyle Richardson, Laria Reynolds, Leo Gao, Li Zhang, Liam Dugan, Lianhui Qin, Lidia Contreras-Ochando, Louis-Philippe Morency, Luca Moschella, Lucas Lam, Lucy Noble, Ludwig Schmidt, Luheng He, Luis Oliveros Colón, Luke Metz, Lütfi Kerem Şenel, Maarten Bosma, Maarten Sap, Maartje ter Hoeve, Maheen Farooqi, Manaal Faruqui, Mantas Mazeika, Marco Baturan, Marco Marelli, Marco Maru, Maria Jose Ramírez Quintana, Marie Tolkiehn, Mario Giulianelli, Martha Lewis, Martin Potthast, Matthew L. Leavitt, Matthias Hagen, Mátyás Schubert, Medina Orduna Baitemirova, Melody Arnaud, Melvin McElrath, Michael A. Yee, Michael Cohen, Michael Gu, Michael Ivanitskiy, Michael Starritt, Michael Strube, Michał Swędrowski, Michele Bevilacqua, Michihiro Yasunaga, Mihir Kale, Mike Cain, Mimee Xu, Mirac Suzgun, Mitch Walker, Mo Tiwari, Mohit Bansal, Moin Aminnaseri, Mor Geva, Mozhdeh Gheini, Mukund Varma T, Nanyun Peng, Nathan A. Chi, Nayeon Lee, Neta Gur-Ari Krakover, Nicholas Cameron, Nicholas Roberts, Nick Doiron, Nicole Martinez, Nikita Nangia, Niklas Deckers, Niklas Muennighoff, Nitish Shirish Keskar, Niveditha S. Iyer, Noah Constant, Noah Fiedel, Nuan Wen, Oliver Zhang, Omar Agha, Omar Elbaghdadi, Omer Levy, Owain Evans, Pablo Antonio Moreno Casares, Parth Doshi, Pascale Fung, Paul Pu Liang, Paul Vicol, Pegah Alipoormolabashi, Peiyuan Liao, Percy Liang, Peter Chang, Peter Eckersley, Phu Mon Htut, Pinyu Hwang, Piotr Miłkowski, Piyush Patil, Pouya Pezeshkpour, Priti Oli, Qiaozhu Mei, Qing Lyu, Qinlang Chen, Rabin Banjade, Rachel Etta Rudolph, Raefer Gabriel, Rahel Habacker, Ramon Risco, Raphaël Millière, Rhythm Garg, Richard Barnes, Rif A. Saurous, Riku Arakawa, Robbe Raymaekers, Robert Frank, Rohan Sikand, Roman Novak, Roman Sitelew, Ronan LeBras, Rosanne Liu, Rowan Jacobs, Rui Zhang, Ruslan Salakhutdinov, Ryan Chi, Ryan Lee, Ryan Stovall, Ryan Teehan, Rylan Yang, Sahib Singh, Saif M. Mohammad, Sajant Anand, Sam Dillavou, Sam Shleifer, Sam Wiseman, Samuel Gruetter, Samuel R. Bowman, Samuel S. Schoenholz, Sanghyun Han, Sanjeev Kwatra, Sarah A. Rous, Sarik Ghazarian, Sayan Ghosh, Sean Casey, Sebastian Bischoff, Sebastian Gehrmann, Sebastian Schuster, Sepideh Sadeghi, Shadi Hamdan, Sharon Zhou, Shashank Srivastava, Sherry Shi, Shikhar Singh, Shima Asaadi, Shixiang Shane Gu, Shubh Pachchigar, Shubham Toshniwal, Shyam Upadhyay, Shyamolima, Debnath, Siamak Shakeri, Simon Thormeyer, Simone Melzi, Siva Reddy, Sneha Priscilla Makini, Soo-Hwan Lee, Spencer Torene, Sriharsha Hatwar, Stanislas Dehaene, Stefan Divic, Stefano Ermon, Stella Biderman, Stephanie Lin, Stephen Prasad, Steven T. Piantadosi, Stuart M. Shieber, Summer Misherghi, Svetlana Kiritchenko, Swaroop Mishra, Tal Linzen, Tal Schuster, Tao Li, Tao Yu, Tariq Ali, Tatsu Hashimoto, Te-Lin Wu, Théo Desbordes, Theodore Rothschild, Thomas Phan, Tianle Wang, Tiberius Nkinyili, Timo Schick, Timofei Kornev, Titus Tunduny, Tobias Gerstenberg, Trenton Chang, Trishala Neeraj, Tushar Khot, Tyler Shultz, Uri Shaham, Vedant Misra, Vera Demberg, Victoria Nyamai, Vikas Raunak, Vinay Ramasesh, Vinay Uday Prabhu, Vishakh Padmakumar, Vivek Srikumar, William Fedus, William Saunders, William Zhang, Wout Vossen, Xiang Ren, Xiaoyu Tong, Xinran Zhao, Xinyi Wu, Xudong Shen, Yadollah Yaghoobzadeh, Yair Lakretz, Yangqiu Song, Yasaman Bahri, Yejin Choi, Yichi Yang, Yiding Hao, Yifu Chen, Yonatan Belinkov, Yu Hou, Yufang Hou, Yuntao Bai, Zachary Seid, Zhuoye Zhao, Zijian Wang, Zijie J. Wang, Zirui Wang, and Ziyi Wu. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Preprint*, arXiv:2206.04615.

Rainer Storn and Kenneth Price. 1997. Differential evolution: A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11:341–359.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, Piotr Stanczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, Shantanu Thakoor, Jean-Bastien Grill, Behnam Neyshabur, Olivier Bachem, Alanna Walton, Aliaksei Severyn, Alicia Parrish, Aliya Ahmad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy Brock, Andy Coenen, Anthony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Bo Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Chris Welty, Christopher A.

Choquette-Choo, Danila Sinopalnikov, David Weinberger, Dimple Vijaykumar, Dominika Rogozińska, Dustin Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii Eltyshev, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Plucińska, Harleen Batra, Harsh Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, Jeff Stanway, Jetha Chan, Jin Peng Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost van Amersfoort, Josh Gordon, Josh Lipschultz, Josh Newlan, Ju yeong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, Keelin McDonell, Kelvin Nguyen, Kiranbir Sodhia, Kish Greene, Lars Lowe Sjoesund, Lauren Usui, Laurent Sifre, Lena Heuermann, Leticia Lago, Lilly McNealus, Livio Baldini Soares, Logan Kilpatrick, Lucas Dixon, Luciano Martins, Machel Reid, Manvinder Singh, Mark Iverson, Martin Görner, Mat Velloso, Mateo Wirth, Matt Davidow, Matt Miller, Matthew Rahtz, Matthew Watson, Meg Risdal, Mehran Kazemi, Michael Moynihan, Ming Zhang, Minsuk Kahng, Minwoo Park, Mofi Rahman, Mohit Khatwani, Natalie Dao, Nenshad Bardoliwalla, Nesh Devanathan, Neta Dumai, Nilay Chauhan, Oscar Wahltinez, Pankil Botarda, Parker Barnes, Paul Barham, Paul Michel, Pengchong Jin, Petko Georgiev, Phil Culliton, Pradeep Kuppala, Ramona Comanescu, Ramona Merhej, Reena Jana, Reza Ardeshir Rokni, Rishabh Agarwal, Ryan Mullins, Samaneh Saadat, Sara Mc Carthy, Sarah Cogan, Sarah Perrin, Sébastien M. R. Arnold, Sebastian Krause, Shengyang Dai, Shruti Garg, Shruti Sheth, Sue Ronstrom, Susan Chan, Timothy Jordan, Ting Yu, Tom Eccles, Tom Hennigan, Tomas Kocisky, Tulsee Doshi, Vihan Jain, Vikas Yadav, Vilobh Meshram, Vishal Dharmadhikari, Warren Barkley, Wei Wei, Wenming Ye, Woohyun Han, Woosuk Kwon, Xiang Xu, Zhe Shen, Zhitao Gong, Zichuan Wei, Victor Cotruta, Phoebe Kirk, Anand Rao, Minh Giang, Ludovic Peran, Tris Warkentin, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, D. Sculley, Jeanine Banks, Anca Dragan, Slav Petrov, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Sebastian Borgeaud, Noah Fiedel, Armand Joulin, Kathleen Kenealy, Robert Dadashi, and Alek Andreev. 2024. Gemma 2: Improving open language models at a practical size. *Preprint*, arXiv:2408.00118.

Together.ai. 2023. LLaMA-2-7B-32K.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent abilities of large language models. *Transactions on Machine Learning Research*. Survey Certification.

Chaojun Xiao, Pengle Zhang, Xu Han, Guangxuan Xiao, Yankai Lin, Zhengyan Zhang, Zhiyuan Liu, and Maosong Sun. 2024. Infllm: Training-free long-context extrapolation for llms with an efficient context memory. In *Advances in Neural Information Processing Systems*, volume 37, pages 119638–119661. Curran Associates, Inc.

Wenhan Xiong, Jingyu Liu, Igor Molybog, Hejia Zhang, Prajjwal Bhargava, Rui Hou, Louis Martin, Rashi Rungta, Karthik Abinav Sankararaman, Barlas Oguz, Madian Khabsa, Han Fang, Yashar Mehdad, Sharan Narang, Kshitiz Malik, Angela Fan, Shruti Bhosale, Sergey Edunov, Mike Lewis, Sinong Wang, and Hao Ma. 2024. Effective long-context scaling of foundation models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4643–4663, Mexico City, Mexico. Association for Computational Linguistics.

Qwen: An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Qinyuan Ye, Harvey Fu, Xiang Ren, and Robin Jia. 2023. How predictable are large language model capabilities? a case study on BIG-bench. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 7493–7517, Singapore. Association for Computational Linguistics.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. Association for Computational Linguistics.

Qiyuan Zhang, Fuyuan Lyu, Xue Liu, and Chen Ma. 2024. Collaborative performance prediction for large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 2576–2596, Miami, Florida, USA. Association for Computational Linguistics.

A Dataset Details

GSM8K (Cobbe et al., 2021) We filter out instances over 256 tokens in length, and select 511 training instances and 250 testing instances at random. During inference, we allow up to 400 new tokens. The average token lengths of the training and testing instances were 177.64 and 177.43 respectively. The generated responses averaged around 172.13 tokens in length. To evaluate, we extract the model's final answer and compare it with the reference answer, checking for numerical equivalence.

MATH (Hendrycks et al., 2021) We filter out instances over 256 tokens in length, and select 511 training instances and 250 testing instances at random. During inference, we allow up to 400 new tokens. The average token lengths of the training and testing instances were 160.54 and 155.74 respectively. The generated responses also averaged around 184.0 tokens in length. To evaluate, we extract the model's final answer and compare it with the reference answer, checking for numerical equivalence.

AQUA-RAT (Ling et al., 2017) We filter out instances over 256 tokens in length, and select 511 training instances and 250 testing instances at random. We allow up to 5 new tokens during generation. The average token lengths of the training and testing instances were 88.45 and 93.09 respectively. The generated responses also averaged around 3.44 tokens in length. To evaluate, we check to see if the choice returned by our model matches the reference answer.

DeepMind Math (Saxton et al., 2019) The dataset is categorized into 56 subsets. We filter out instances over 256 tokens in length, and select 511 training instances and 50 testing instances at random from each subset. We allow up to 400 new tokens during generation. The average token lengths of the training and testing instances were 57.94 and 61.05 respectively. The generated responses also averaged around 85.71 tokens in length. To evaluate, we extract the model's final answer and compare it with the reference answer, checking for numerical equivalence.

PIQA (Bisk et al., 2020) We filter out instances over 256 tokens in length, and select 511 training instances and 250 testing instances at random. We allow up to 5 new tokens during generation. The average token lengths of the training and testing instances were 81.16 and 81.55 respectively. The generated responses also averaged around 3.46 tokens in length. To evaluate, we check to see if the choice returned by our model matches the reference answer.

OpenBookQA (Mihaylov et al., 2018) We filter out instances over 256 tokens in length, and select 511 training instances and 250 testing instances at random. We allow up to 5 new tokens during generation. The average token lengths of the training and testing instances were 47.74 and 49.39 respectively. The generated responses also averaged around 3.3 tokens in length. To evaluate, we check to see if the choice returned by our model matches the reference answer.

SIQA (Sap et al., 2019) We filter out instances over 256 tokens in length, and select 511 training instances and 250 testing instances at random. We allow up to 1 new token during generation. The average token lengths of the training and testing instances were 56.68 and 56.87 respectively. The generated responses also averaged around 3.35 tokens in length. To evaluate, we check to see if the choice returned by our model matches the reference answer.

HellaSwag (Zellers et al., 2019) We filter out instances over 256 tokens in length, and select 511 training instances and 250 testing instances at random. We allow up to 5 new tokens during generation. The average token lengths of the training and testing instances were 153.06 and 156.05 respectively. The generated responses also averaged around 3.67 tokens in length. To evaluate, we check to see if the choice returned by our model matches the reference answer.

WinoGrande (Sakaguchi et al., 2020) We filter out instances over 256 tokens in length, and select

511 training instances and 250 testing instances at random. We allow up to 5 new tokens during generation. The average token lengths of the training and testing instances were 53.98 and 53.87 respectively. The generated responses also averaged around 3.33 tokens in length. To evaluate, we check to see if the choice returned by our model matches the reference answer.

ARC Easy (Clark et al., 2018) We filter out instances over 256 tokens in length, and select 511 training instances and 250 testing instances at random. We allow up to 5 new tokens during generation. The average token lengths of the training and testing instances were 66.69 and 67.14 respectively. The generated responses also averaged around 3.46 tokens in length. To evaluate, we check to see if the choice returned by our model matches the reference answer.

ARC Challenge (Clark et al., 2018) We filter out instances over 256 tokens in length, and select 511 training instances and 250 testing instances at random. We allow up to 5 new tokens during generation. The average token lengths of the training and testing instances were 75.65 and 76.83 respectively. The generated responses also averaged around 3.43 tokens in length. To evaluate, we check to see if the choice returned by our model matches the reference answer.

CommenSenseQA (Talmor et al., 2019) We filter out instances over 256 tokens in length, and 250 select 511 training instances and testing instances at random. We allow up to 5 new tokens during generation. The average token lengths of the training and testing instances were 50.42 and 49.92 respectively. The generated responses also averaged around 1.0 tokens in length. To evaluate, we check to see if the choice returned by our model matches the reference answer.

WMT14 (CS-EN) (Bojar et al., 2014) We filter out instances over 256 tokens in length, and select 511 training instances and 250 testing instances at random. We allow up to 256 new tokens during generation. The average token lengths of the training and testing instances were 95.01 and 85.25 respectively. The generated responses also averaged around 77.77 tokens in length. We use BLEU-4 (Papineni et al., 2002) to score the generated translations relative to the reference translations.

WMT14 (DE-EN) (Bojar et al., 2014) We filter out instances over 256 tokens in length, and select 511 training instances and 250 testing instances at random. We allow up to 256 new tokens during generation. The average token lengths of the training and testing instances were 85.53 and 77.68 respectively. The generated responses also averaged around 77.77 tokens in length. We use BLEU-4 (Papineni et al., 2002) to score the generated translations relative to the reference translations.

WMT14 (FR-EN) (Bojar et al., 2014) We filter out instances over 256 tokens in length, and select 511 training instances and 250 testing instances at random. We allow up to 256 new tokens during generation. The average token lengths of the training and testing instances were 95.94 and 84.29 respectively. The generated responses also averaged around 78.73 tokens in length. We use BLEU-4 (Papineni et al., 2002) to score the generated translations relative to the reference translations.

WMT14 (HI-EN) (Bojar et al., 2014) We filter out instances over 256 tokens in length, and select 511 training instances and 250 testing instances at random. We allow up to 256 new tokens during generation. The average token lengths of the training and testing instances were 34.01 and 147.09 respectively. The generated responses also averaged around 53.11 tokens in length. We use BLEU-4 (Papineni et al., 2002) to score the generated translations relative to the reference translations.

WMT14 (RU-EN) (Bojar et al., 2014) We filter out instances over 256 tokens in length, and select 511 training instances and 250 testing instances at random. We allow up to 256 new tokens during generation. The average token lengths of the training and testing instances were 73.54 and 86.56 respectively. The generated responses also averaged around 77.24 tokens in length. We use BLEU-4 (Papineni et al., 2002) to score the generated translations relative to the reference translations.

B Full Results

In this section, we present full aggregate results in Tables 8, 9, and 10 for arithmetic reasoning, common sense reasoning, and machine translation respectively. Figures 4, 5,and 6 provide contours of our fits at $C = 7.8 \times 10^{22}$ and $C = 1.5 \times 10^{23}$.

k	0 shots	1 shot	3 shots	7 shots	15 shots	31 shots	63 shots	127 shots	255 shots	511 shots
Llama-2-7b-hf	0.089	0.099	0.115	0.120	0.136	0.127	0.094	0.014	0.014	0.000
Yarn-Llama-2-7b-8k	0.076	0.097	0.109	0.117	0.134	0.131	0.137	0.071	0.000	0.000
Yarn-Llama-2-7b-16k	0.072	0.095	0.109	0.116	0.130	0.133	0.143	0.139	0.073	0.002
Yarn-Llama-2-7b-32k	0.069	0.092	0.104	0.113	0.127	0.127	0.135	0.134	0.143	0.076
Yarn-Llama-2-7b-64k	0.057	0.094	0.108	0.115	0.132	0.128	0.143	0.140	0.150	0.138
Yarn-Llama-2-7b-128k	0.049	0.091	0.106	0.113	0.129	0.126	0.136	0.135	0.149	0.140
Llama-2-13b-hf	0.088	0.115	0.131	0.137	0.148	0.141	0.092	0.011	0.005	0.000
Yarn-Llama-2-13b-8k	0.086	0.110	0.126	0.132	0.146	0.149	0.151	0.082	0.000	0.000
Yarn-Llama-2-13b-16k	0.081	0.110	0.135	0.146	0.153	0.163	0.172	0.145	0.077	0.010
Yarn-Llama-2-13b-32k	0.077	0.111	0.129	0.145	0.154	0.162	0.171	0.169	0.134	0.065
Yarn-Llama-2-13b-64k	0.073	0.106	0.130	0.146	0.156	0.158	0.169	0.167	0.159	0.136
Yarn-Llama-2-13b-128k	0.069	0.108	0.123	0.138	0.157	0.157	0.174	0.165	0.163	0.153

Table 8: Accuracy on arithmetic reasoning, aggregated over every instance in the task.

k	0 shots	1 shot	3 shots	7 shots	15 shots	31 shots	63 shots	127 shots	255 shots	511 shots
Llama-2-7b-hf	0.376	0.489	0.518	0.536	0.536	0.527	0.302	0.000	0.000	0.000
Yarn-Llama-2-7b-8k	0.356	0.476	0.518	0.530	0.530	0.523	0.491	0.278	0.000	0.000
Yarn-Llama-2-7b-16k	0.342	0.468	0.508	0.522	0.532	0.519	0.521	0.486	0.264	0.000
Yarn-Llama-2-7b-32k	0.325	0.459	0.500	0.496	0.522	0.508	0.501	0.534	0.457	0.276
Yarn-Llama-2-7b-64k	0.346	0.456	0.503	0.513	0.502	0.498	0.490	0.515	0.470	0.458
Yarn-Llama-2-7b-128k	0.338	0.450	0.496	0.490	0.504	0.486	0.490	0.507	0.465	0.486
Llama-2-13b-hf	0.453	0.604	0.649	0.660	0.659	0.600	0.344	0.000	0.000	0.000
Yarn-Llama-2-13b-8k	0.469	0.610	0.662	0.656	0.675	0.652	0.603	0.318	0.000	0.000
Yarn-Llama-2-13b-16k	0.464	0.594	0.656	0.654	0.658	0.650	0.658	0.584	0.308	0.000
Yarn-Llama-2-13b-32k	0.432	0.586	0.642	0.640	0.642	0.642	0.646	0.626	0.567	0.322
Yarn-Llama-2-13b-64k	0.481	0.589	0.642	0.636	0.638	0.634	0.645	0.620	0.614	0.582
Yarn-Llama-2-13b-128k	0.480	0.578	0.636	0.632	0.628	0.630	0.634	0.616	0.609	0.612

Table 9: Accuracy on Commonsense Reasoning tasks, aggregated over every instance in the task.

k	0 shots	1 shot	3 shots	7 shots	15 shots	31 shots	63 shots	127 shots	255 shots	511 shots
Llama-2-7b-hf	0.031	0.147	0.155	0.154	0.156	0.159	0.049	0.011	0.000	0.000
Yarn-Llama-2-7b-8k	0.034	0.142	0.155	0.146	0.151	0.152	0.152	0.010	0.006	0.000
Yarn-Llama-2-7b-16k	0.031	0.138	0.152	0.144	0.147	0.144	0.143	0.143	0.006	0.003
Yarn-Llama-2-7b-32k	0.026	0.138	0.147	0.141	0.143	0.142	0.141	0.140	0.146	0.005
Yarn-Llama-2-7b-64k	0.033	0.129	0.144	0.142	0.148	0.141	0.148	0.142	0.144	0.134
Yarn-Llama-2-7b-128k	0.040	0.125	0.140	0.140	0.144	0.142	0.145	0.136	0.136	0.130
Llama-2-13b-hf	0.023	0.166	0.175	0.180	0.181	0.175	0.058	0.015	0.000	0.000
Yarn-Llama-2-13b-8k	0.029	0.161	0.171	0.175	0.177	0.170	0.174	0.014	0.011	0.000
Yarn-Llama-2-13b-16k	0.025	0.157	0.170	0.171	0.176	0.166	0.173	0.166	0.010	0.005
Yarn-Llama-2-13b-32k	0.025	0.152	0.168	0.166	0.171	0.168	0.164	0.156	0.160	0.007
Yarn-Llama-2-13b-64k	0.066	0.152	0.162	0.163	0.166	0.169	0.163	0.160	0.162	0.160
Yarn-Llama-2-13b-128k	0.101	0.145	0.155	0.162	0.163	0.163	0.163	0.154	0.157	0.154

Table 10: Accuracy on Machine Translation tasks, aggregated over every instance in the task.

Figure 4: Contours of our fit at $C = 7.8 \times 10^{22}$ (left) and $C = 1.5 \times 10^{23}$ (right) for the arithmetic reasoning task.

Figure 5: Contours of our fit at $C = 7.8 \times 10^{22}$ (left) and $C = 1.5 \times 10^{23}$ (right) for the common sense reasoning task.

Figure 6: Contours of our fit at $C = 7.8 \times 10^{22}$ (left) and $C = 1.5 \times 10^{23}$ (right) for the machine translation task.