Data Efficient Neural Scaling Law via Model Reusing

Peihao Wang [1] **Rameswar Panda** [2] **Zhangyang Wang** [1]

Abstract

The number of parameters in large transformers has been observed to grow exponentially. Despite notable performance improvements, concerns have been raised that such a growing model size will run out of data in the near future. As manifested in the neural scaling law, modern learning backbones are not data-efficient. To maintain the utility of the model capacity, training data should be increased proportionally. In this paper, we study the neural scaling law under the previously overlooked data scarcity regime, focusing on the more challenging situation where we need to train a gigantic model with a disproportionately limited supply of available training data. We find that the existing power laws underestimate the data inefficiency of large transformers. Their performance will drop significantly if the training set is insufficient. Fortunately, we discover another blessing - such a data-inefficient scaling law can be restored through a model reusing approach that warm-starts the training of a large model by initializing it using smaller models. Our empirical study shows that model reusing can effectively reproduce the power law under the data scarcity regime. When progressively applying model reusing to expand the model size, we also observe consistent performance improvement in large transformers. We release our code at: https://github.com/VITA-Group/Data-Efficient-Scaling.

1. Introduction

With the number of parameters growing exponentially from a few million (Devlin et al., 2019) to hundreds of billions (Brown et al., 2020; Schulman et al., 2022), large transformer models (Vaswani et al., 2017) have been dominating a wide range of applications (Devlin et al., 2019; Brown et al., 2020; Rives et al., 2021; Dosovitskiy et al., 2021; Touvron et al., 2021a; Jumper et al., 2021; Li et al., 2023; Zheng et al., 2023). An interesting relation among performance, training data, and model size has been recently revealed, suggesting that these three variables generally should follow a power law (Rosenfeld et al., 2019; Kaplan et al., 2020; Hoffmann et al., 2022). That being said, to avoid performance saturation, the data scale should be grown in proportion to the model parameter number. This signifies that large-scale training of modern transformer models is becoming increasingly data-hungry and less affordable.

In fact, besides the shortage of computational resources, the lack of data resources has set another prohibitive barrier for research labs to participate in and contribute to large model training. Additionally, given the unparalleled growth rate of model size and data scale, it is even considered foreseeable that all web-collected data will be used up in the near future (Villalobos et al., 2022). What is worse, training such ever-larger models on limited data from fine-grained specific domains or private data will become rather difficult, if not impossible. Although transfer learning of publicly available pre-trained models is always an option, their domain gaps can often pose challenges to transfer effectiveness (Jiang et al., 2022), and sometimes even fine-tuning a large pre-trained model can be excessively costly.

It has not escaped our notice that all the raised questions point to a core challenge: "How can we train large transformer models with higher data efficiency?" We tackle this problem through the lens of the neural scaling law under the *extremely low data regime*. Both language transformers (Devlin et al., 2019; Liu et al., 2019) and vision transformers (Dosovitskiy et al., 2021) are studied in this paper. Existing power laws (Kaplan et al., 2020; Zhai et al., 2022) only consider model and dataset sizes within a reasonable ratio. In contrast, our experiments focus on base-size models ($\lesssim$ 80M) trained with less than tens million tokens or tens thousand images, where the model size is considered to be overwhelmingly larger than the data scale. Consequently, we provide a rectified functional relation in terms of test error, model size, and dataset size, which can more accurately characterize the model-data size frontier for large

[1]Department of Electrical and Computer Engineering, University of Texas at Austin, TX, United States [2]MIT-IBM Watson Lab, MA, United States. Correspondence to: Peihao Wang <peihaowang@utexas.edu>, Zhangyang Wang <atlaswang@utexas.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

transformer models when the data size is excessively small. Our key finding is that data scarcity will cause a significant performance falloff. Such phenomena are also revealed in Paul et al. (2021); Sorscher et al. (2022), but unfortunately, they are underestimated by current scaling laws.

Having revealed the curse of data scarcity through the lens of the scaling law, we propose to overcome this problem via *model reusing* (Chen et al., 2015; 2021). Specifically, model reusing techniques initialize larger models with a pre-trained smaller model either by duplicating (Chen et al., 2015; 2021), stacking (Gong et al., 2019), combining (Wang et al., 2023; Yang et al., 2022b) model weights, or using knowledge distillation (Qin et al., 2021; Yang et al., 2022a). We reason that the root cause of performance degradation is that limited data largely jeopardizes the trainability of large models under stochastic gradient algorithms. The key intuition is that reusing smaller models to warm-start large transformer training can provide a warm initialization point, thus stabilizing the optimization at the beginning and saving data samples needed to "explore" optimization trajectories. With model reusing, the model scaling law can be reproduced within the extremely low data regime.

Our main contributions can be summarized below:

- We conduct a pilot study on the neural scaling law for large language models under an extremely low data regime. We find that well-known scaling laws extrapolate inaccurately when data amount is excessively smaller than the parameter number, and we provide a rectified scaling law, dubbed the *Data Scarcity Neural Scaling Law*.
- We propose to break the *Data Scarcity Neural Scaling Law* via model reusing, which initializes large models using smaller model weights. We conduct a comprehensive investigation on various model reusing schemes and validate their effectiveness when scaling the model under limited data.
- Based on our investigation, we deliver *progressive growing* as a simple yet effective recipe to train a large language model under extremely low data. As a consequence, we for the first time achieve $\sim$**60%** accuracy with $\sim$**10k** images on ViT-B and $\sim$**2.1** log-perplexity on BERT-Base with only $\sim$**15M** training tokens.

2. Neural Scaling Laws

Many recent works have demonstrated empirical scaling laws with respect to model size, dataset size, and compute for both large language models (Bahri et al., 2021; Kaplan et al., 2020) and vision models (Zhai et al., 2022; Bello et al., 2021). Kaplan et al. studied the scaling laws of the decoder-only transformer language model and found that the loss scales as a power-law with model size, dataset size, and training compute. Similarly, works in Henighan et al. (2020) and Hernandez et al. (2021) show scaling laws for autoregressive generative modeling and transfer learning, respectively. The scaling of Mixture of Experts (MoE) models up to trillion parameters is studied in Fedus et al. (2021). A systematic study of scaling laws for different inductive biases and model architectures is presented in Tay et al. (2022). Recently, Geiping & Goldstein (2022) investigated the downstream performance achievable with a transformer-based language model trained completely from scratch for a single day on a single GPU, through the lens of scaling laws. Caballero et al. (2022) studied broken neural scaling laws that generalize power laws (linear in log-log plot) to a smoothly connected piecewise (approximately) linear function in a log-log plot. The scaling of vision transformers and data, both up and down, is studied in Zhai et al. (2022), which characterizes the relationships between error rate, data, and compute. Scaling laws under few-shot fine-tuning and multi-modal training settings are investigated in Prato et al. (2021); Aghajanyan et al. (2023). Hoffmann et al. (2022) find that for compute-optimal training, the model size and the number of training tokens should be scaled equally. The seminal work by Sorscher et al. (2022) reveals that such power laws can be broken via proper data pruning.

Formally, we summarize the common neural scaling law as follows (Hoffmann et al., 2022; Kaplan et al., 2020):

$$\text{(Model scaling)} \quad L(N) = (N/N_c)^{-\alpha_N} + E_N, \quad (1)$$

$$\text{(Data scaling)} \quad L(D) = (D/D_c)^{-\alpha_D} + E_D, \quad (2)$$

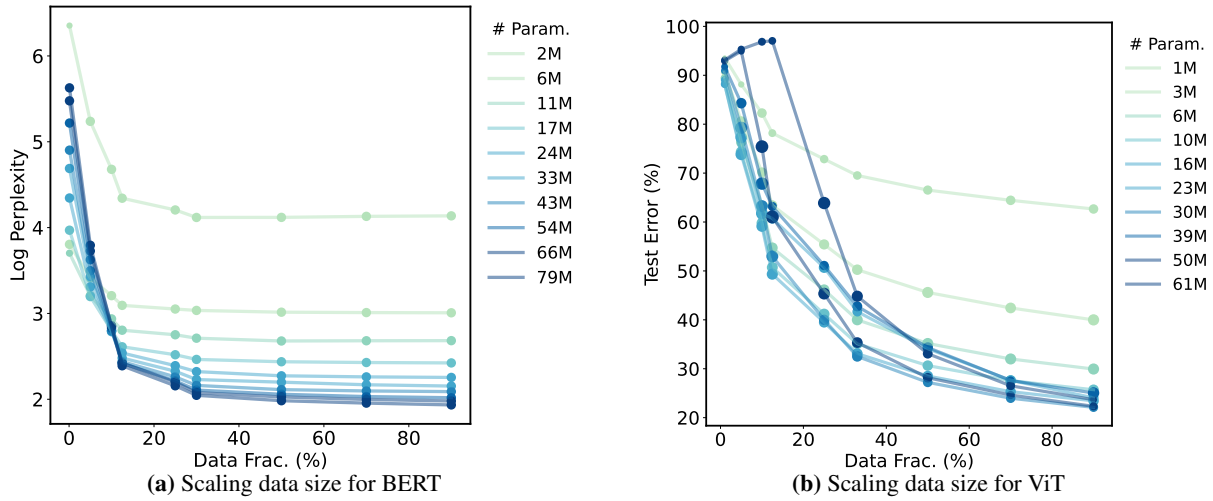
where N denotes the number of model parameters, D denotes the number of training examples, and N_c, D_c, α_N, α_D, E_N, and E_D are model/task-dependent coefficients that express a power function. These constants are often obtained by fitting the collected experimental data (Kaplan et al., 2020; Hoffmann et al., 2022; Zhai et al., 2022). Following the interpretation from Kaplan et al. (2020), $L(N)$ is named the model scaling law, which measures the model capacity, and $L(D)$ is called the data scaling law, which reflects the generalization ability. In particular, the exponents are found to be negative, indicating a diminishing pattern of error when increasing the dataset or model sizes.

3. Data Scarcity Neural Scaling Law

In this section, we examine and rectify the scaling law for large transformers under the data scarcity regime.

3.1. Motivation

The relationship between performance, model parameter number, and data size has been crucially examined in the



(a) Scaling data size for BERT **(b)** Scaling data size for ViT

Figure 1: Data scaling curves for BERT and ViT. (a) and **(b)** depict the performance trend with data scaling for BERT and ViT, respectively, where each curve connects points corresponding to a fixed model size but varying data fraction, and the radius of markers indicates the standard deviations. Data scaling obeys the power law $(D/D_c)^{-\alpha_D}$. Best viewed in color.

context of large language models. This investigation plays a vital role in informing model design choices and resource allocation for future training endeavors. Interestingly, previous studies have identified a notable phenomenon wherein the test error follows a power decay curve with respect to the number of model parameters (N) and the number of training examples (D) (Rosenfeld et al., 2019; Kaplan et al., 2020; Hoffmann et al., 2022). This observation holds true not only when examining each dimension independently as captured by the functional forms specified in Eq. 1 (i.e., varying D or N alone), but also when simultaneously manipulating both N and D (Kaplan et al., 2020).

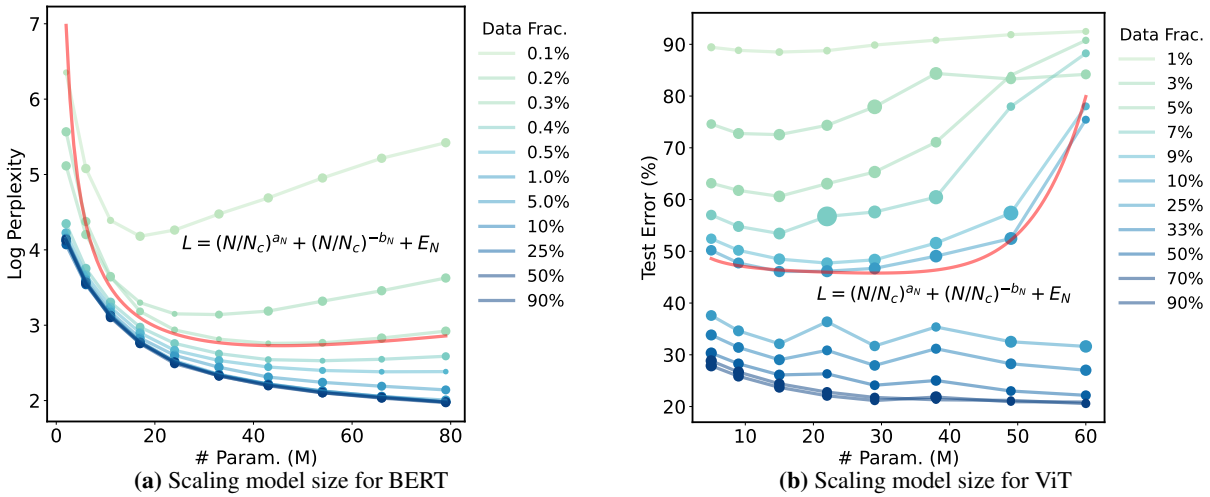
However, we contend that this functional form might have limitations in capturing the interdependence between data and model sizes, as it treats these two factors as separate entities. Intuitively, the size of the data and the model should be intertwined because larger models are believed to have greater data requirements, necessitating a corresponding increase in training data to achieve convergence and prevent overfitting. Once the number of model parameters exceeds a certain threshold, it is reasonable to expect a significant decline in test performance rather than a gradual saturation to a fixed value. Specifically, if we adhere to the power law formulation presented in Eq. 1 and hold the number of training data points (D) constant while continuously increasing the model size ($N \to \infty$), the predicted test error can still exhibit a monotonically decreasing trend. This discrepancy challenges our intuition and suggests that the power law formulation (Eq. 1) may fail to accurately predict the behavior of the model when there is a substantial imbalance between the model and data sizes.

Indeed, we observe that the prevailing empirical power laws are derived under the assumption that the sizes of the model and the data are directly proportional. However, it has been demonstrated that such fitted curves tend to yield unsatisfactory extrapolation results, such as the occurrence of double saturation phenomena at the extremes of the compute spectrum (Zhai et al., 2022). To address this issue, Caballero et al. (2022) propose the utilization of piecewise functions as an approximation of the true scaling law. In this section, our focus lies specifically on the segment corresponding to the low data regime, where we aim to re-establish the scaling law through empirical analysis.

3.2. Experimental Setup

In this paper, we conduct case studies on two widely used transformer models in both language and vision domains: BERT (Devlin et al., 2019) and vision transformer (ViT) (Dosovitskiy et al., 2021). For BERT, we adopt the implementation provided by Tan & Bansal (2020), and choose English Wikipedia (Merity et al., 2016) as the training dataset. Our experiments encompass BERT models ranging from 2 to 80 million parameters and training dataset sizes varying from 3 million to 3 billion tokens. We adopt Mask Language modeling (MLM) as the training objective and the log perplexity evaluated on the test set as the test score.

For ViT, we utilize the implementation provided by Touvron et al. (2021a). The ImageNet1k (Deng et al., 2009) dataset is chosen as our training data collection. In our experimentation, we investigate the performance of vision transformers across a range of scales from 1 to 60 million parameters. Additionally, we vary the size of the training dataset, ranging from one million to ten thousand images. The ViT is trained on the image classification task with the cross-entropy loss, and we evaluate the test error by measuring the top-1 prediction accuracy on the test split.



(a) Scaling model size for BERT **(b)** Scaling model size for ViT

Figure 2: Model scaling curves for BERT and ViT. (a) and **(b)** depict the performance trend with model scaling law for BERT and ViT, respectively, where each curve connects points of the same data fraction but varying parameter number, and the radius of markers indicates the standard deviations. We observe a phase transition, where the power law is broken once entering the data scarcity regime. The red curve represents the prediction of our rectified law. Best viewed in color.

Varying model size. We vary the network size by increasing the width for each attention block. We fix the hidden dimension of each attention head to 64, and increase the number of heads from 1 to 10 for both BERT and ViT. We simplify our experiments by fixing the number of attention blocks to 12 (Devlin et al., 2019; Liu et al., 2019; Touvron et al., 2021a) according to the following reasons: 1) compared with number of parameters, the model shape has negligible influence on the performance for large transformers as discussed in Kaplan et al. (2020). 2) increasing the depth for transformers can cause training instability and over-smoothing issues as revealed by (Touvron et al., 2021b; Wang et al., 2022; Shi et al., 2022; Zhou et al., 2021; Gong et al., 2021). We will keep all other model and training configurations consistent with the default implementation in (Tan & Bansal, 2020; Touvron et al., 2021a).

Varying dataset size. In order to adjust the size of the training set, we subsample the original data collection. For the English Wikipedia dataset, we uniformly sample a subset of lines of examples based on a fraction among $0.1\% \sim 100\%$. Since we are primarily interested in the low data regime, we select 5 fractions between $[0.1\%, 1.0\%]$, and the other 5 within $[1.0\%, 100\%]$. The pre-processed dataset contains ˜3M to ˜3B tokens. On the other hand, for the ImageNet1k dataset (Deng et al., 2009), we perform uniform sampling across the entire dataset while ensuring the preservation of the original class-wise distribution. The subsampling ratio ranges from $1\% \sim 100\%$. Similarly, we choose 5 fractions between $[1\%, 10\%]$, and the other 5 fractions between $[10\%, 100\%]$. The subsampled ImageNet1k dataset contains ˜10k to ˜1M images.

3.3. Experiment Results

We plot the test error versus the number of parameters or training samples in Figs. 1 and 2. Main findings are summarized below:

Obs 1. Power law holds for data scaling. We depict the relationship between the test error and the data fraction of training samples under different model sizes in Fig. 1. The curves therein suggest the power data scaling law is overall satisfied for both BERT and ViT. When linearly scaling the dataset size, the test error decreases at a power rate. Each curve eventually converges to the performance ceiling determined by the model size term $(N/N_c)^{-\alpha_N}$. Moreover, the larger number of parameters in general induces a higher performance ceiling.

Obs 2. Power law is broken with model scaling. We demonstrate the dynamics of model performance with respect to parameter numbers under different data fractions in Fig. 2. Our observation is that for both BERT and ViT, there exists a significant *phase transition* when the training data is continuously shrunk. When the data fraction is equal or above 0.5% for BERT and 10% for ViT, the test error decreases with the model size following a power scale. In that region, we presume the data is abundant, and the power scaling law can accurately characterize model behavior. However, once the data fraction is below a threshold, i.e., the data scale enters the low data regime, we observe that the power law will be broken, and the test error will start to rapidly climb up with the growing architecture size. This result indicates that the number of training examples should not only set a lower bound for the test error, but also

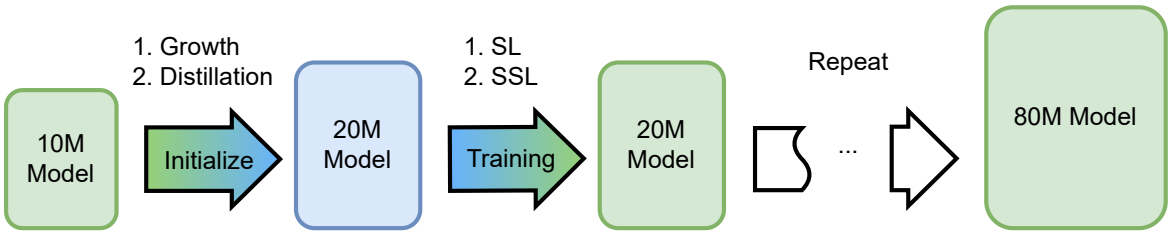


Figure 3: Paradigm of model reusing and progressive training. Model reusing will initialize the large model with smaller model via either of the two schemes: 1) growth or 2) distillation. After the large model is initialized, it will re-train the model through either supervised or self-supervised learning. Progressive training iteratively leverages model reusing to gradually enlarge the model size.

have an effect on the monotonicity and slope of the scaling curve. The existing power laws (Eq. 1) fail to reflect this property as data and model terms are considered separately.

3.4. Rectified Scaling Law

Having revealed the limitation of current scaling laws, we attempt to correct the law in this section. We propose the following form, dubbed *data scarcity scaling law*, to improve the power law to better characterize the phase transition when shrinking the data:

$$L(N) = (N/N_c)^{-\alpha_N} + (N/N_c)^{\beta_N(D)} + E_N, \quad (3)$$

where in contrast to Eq. 1, we introduce a new term $(N/N_c)^{\beta_N(D)}$ with the coefficient $\beta_N \geq 0$ depending on D. We examine this new form and specify the generic properties that need be satisfied:

1. Phase transition needs to be reflected in the new law. When data is insufficient, there needs to emerge a new term which enlarges the error. Eq. 3 achieves this by adding a new term $(N/N_c)^{\beta_N(D)}$ conditioned on the data scale D. It is convenient to leverage a piecewise function to model $\beta_N(D)$. When $D \leq D_{thres}$, $\beta_N(D)$ needs to be strictly positive to characterize the increasing error.

2. When data is sufficient, then the effect of the new term should vanish and recover the behavior of the original law. This can also be achieved by the same piecewise function, in which $\beta_N(D) = 0$ if $D > D_{thres}$.

3. Eq. 3 should contain a inflection point in terms of N, after which the model size is considered overwhelmingly large and test error starts to climb up with the model size. The introduced term achieves this by forming a rational function with piecewise monotonicity.

We fit the coefficients for the piece under the data scarcity regime. The fitting algorithms is adopted from Hoffmann et al. (2022) and the results are presented in Tab. 1. We also plot the computed scaling curves in Fig. 2, which accurately approximate the Pareto frontiers.

	N_c	α_N	β_N	E_N
BERT	5.922×10^{-6}	0.536	0.756	0.757
ViT	3.854×10^{-6}	0.693	8.014	44.474

Table 1: Coefficient specifications for the rectified scaling law under the data scarcity region.

4. Recover Scaling Law via Model Reusing

Our data scarcity scaling law illustrates that data scarcity inherently undermines model scaling. As language models continue to grow in capacity, it becomes crucial to break the data scarcity scaling rule and restore the power law. In this section, we highlight model reusing as a class of simple and commonly used training techniques that can be applied to overcome data shortage issues at no additional cost.

4.1. Model Reusing Techniques

The concept of model reusing is depicted in Fig. 3. In essence, model reusing involves transferring the knowledge acquired from a smaller model to initialize a larger model. The larger model is then further trained starting from this initialization. Model reusing techniques can be broadly categorized into two families: 1) model growth and 2) knowledge distillation. We provide a summary of some representative methods within these categories below

Net2Net. Net2Net (Chen et al., 2015), a classical model growth method, proposes a technique to expand the width of neural networks by duplicating neurons. In this method, the weight matrix of the l-th layer in the smaller model, denoted as $\boldsymbol{W}_l \in \mathbb{R}^{D_1 \times D_1}$, is used to create a larger weight matrix $\boldsymbol{W}_l^{(new)} \in \mathbb{R}^{D_2 \times D_2}$ $(D_2 > D_1)$ to fill the larger model. To achieve this expansion, Net2Net copies the smaller weight matrix $\boldsymbol{W}_l$ to the upper-left corner of $\boldsymbol{W}_l^{(new)}$. It then randomly duplicates some columns to fill the remaining empty columns. Finally, rows are copied and normalized based on the selection made at the previous layer. The procedure can be expressed as follows:

$$\boldsymbol{W}_l^{(new)} = \left[\boldsymbol{I}\ \boldsymbol{S}_{l-1}^{\top}\right] \boldsymbol{D}_l^{-1} \boldsymbol{W}_l \left[\boldsymbol{I} \quad \boldsymbol{S}_l\right], \quad (4)$$
$$\text{where } \boldsymbol{D}_l = \text{diag}(\boldsymbol{S}_{l-1}\boldsymbol{1}) + \boldsymbol{I},$$

Here, $\boldsymbol{S}_l \in \{0,1\}^{D_1 \times (D_2 - D_1)}$ is a random selection matrix that indicates the column indices to be duplicated at the l-th layer. The larger model initialized using Net2Net retains the functionality equivalent to the smaller model. In the case of transformers, Chen et al. extend the Net2Net method by introducing dedicated copying and normalization patterns that are shared across modules.

Learning to Grow (LiGO). Wang et al. (2023) proposes a generalized version of Net2Net, dubbed LiGO, which introduces learnable components to neuron duplication. Specifically, LiGO pre-trains a linear mapping between two parameter spaces before expanding the model size. The optimization goal can be written as:

$$\begin{aligned} &\underset{\boldsymbol{M}_l, \forall l \in [L]}{\arg\min}\ \mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}} \mathcal{L}(\boldsymbol{x}; \boldsymbol{W}_1^{(new)}, \cdots, \boldsymbol{W}_L^{(new)}), \\ &\text{subject to } \operatorname{vec}\left(\boldsymbol{W}_l^{(new)}\right) = \boldsymbol{M}_l \operatorname{vec}(\boldsymbol{W}_l), \forall l \in [L], \end{aligned} \tag{5}$$

where L is defined as the number of layers, $\mathcal{D}$ is the data distribution, $\mathcal{L}$ is the training objective, and $\boldsymbol{M}_l : \mathbb{R}^{D_1^2} \to \mathbb{R}^{D_2^2}$ is called the LiGO operator which maps smaller model weights to large model weights. In Wang et al. (2023), LiGO adopts Kronecker factorization for parameter efficiency, which defines $\boldsymbol{W}_l^{(new)} = \boldsymbol{A}_l^\top \boldsymbol{W}_l \boldsymbol{B}_l$ with learnable weights $\boldsymbol{A}_l, \boldsymbol{B}_l \in \mathbb{R}^{D_1 \times D_2}$.

Knowledge Inheritance. In addition to model growth, Qin et al. (2021) proposes a knowledge transfer strategy to warm start a large model with a small model. It is implemented by a distillation loss formulated as below:

$$\underset{\boldsymbol{W}^{(new)}}{\arg\min}\, \mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}}\, \mathrm{KL}\left(\mathcal{P}_S(\boldsymbol{x}; \boldsymbol{W}) \| \mathcal{P}_L(\boldsymbol{x}; \boldsymbol{W}^{(new)})\right), \tag{6}$$

where $\mathcal{P}_S$ and $\mathcal{P}_L$ denote the output logits corresponding to input $\boldsymbol{x}$ for small and large models, respectively. We note that, unlike traditional knowledge distillation (Hinton et al., 2015), the smaller model here is served as the teacher model. Following Qin et al. (2021), the distillation loss will be minimized joinlty with the self-learning objective.

4.2. Rationales

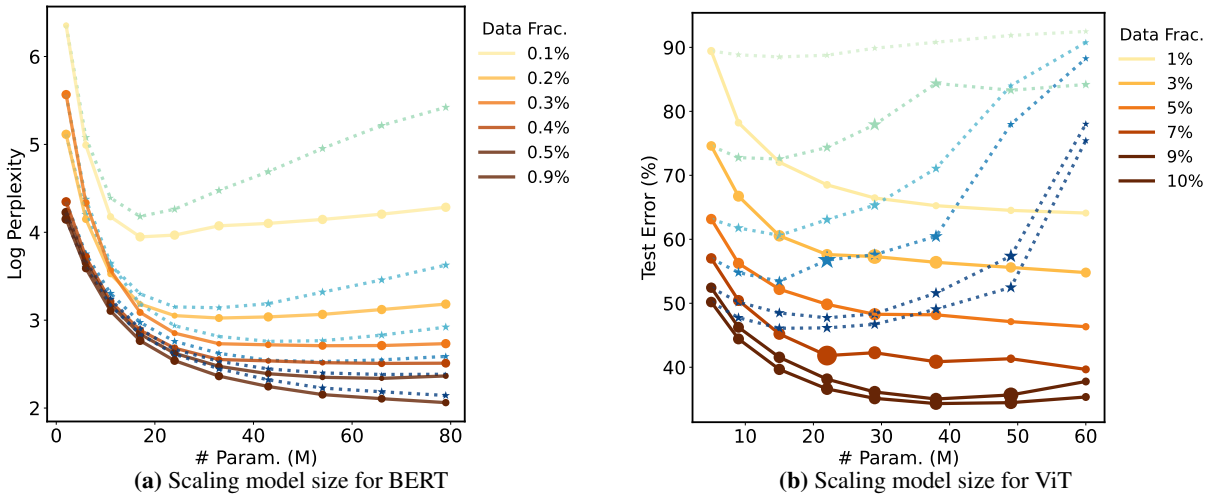
We propose several hypotheses to explain how data scarcity can break the power law. First, we observe that training large-scale transformer models is more data inefficient. We postulate that larger transformers with weaker inductive bias have a larger hypothesis space and stronger function fitting capacity (Yun et al., 2019). Consequently, overfitting is more likely to occur in large language models. Second, the sophisticated computational mechanism in transformers introduces challenges in numerical optimization, resulting in training vulnerability and instability (Zhang et al., 2019; Xiong et al., 2020; Molybog et al., 2023). Kim et al. has theoretically shown that the attention mechanism is not Lipschitz, which exacerbates the optimization complexity, particularly for large transformers (Wang et al., 2022; Zhou et al., 2021).

In general, model reusing techniques initialize the large model using a pre-trained smaller model. To address the first challenge, we argue that the knowledge learned by the smaller model can act as a regularizer that constrains the hypothesis space. In model growth methods such as Net2Net (Chen et al., 2015) and LiGO (Wang et al., 2023), linear mappings are employed to transform model weights to higher dimensions, resulting in low-rank matrices as initialization. This approach potentially enhances model robustness (Yu et al., 2020; Chan et al., 2022; Ma et al., 2022; Cai et al., 2023). In distillation-based techniques, the output from the smaller model serves a similar role to data augmentation, where soft logits provide additional information beyond the one-hot labels through the distribution tail. These types of regularization assist in mitigating overfitting problems and achieving better generalization in larger models. Additionally, smaller models have lower complexity and are more amenable to training with limited data. By utilizing a pre-trained smaller model as the initialization point, the larger model benefits from a warm start in the optimization process. We speculate that model reusing techniques position the initial parameters closer to the optimum, where the loss landscape is flatter and gradients are smoother (Li et al., 2018). This reduces the optimization difficulty and addresses the second challenge mentioned earlier.

4.3. Can Model Reusing Save the Power Law?

After clarifying the potential benefits of leveraging model reusing to overcome training difficulty, we use experiments to verify our hypothesis. In this section, we focus on the low data regime (less than 10M tokens or 100k images) where power law is shown broken.

Experimental Setup. Following the experiment details in Sec. 3, we testify the performance of 1) BERT with parameters ranging from 2M to 80M trained with 0.1% to 1% tokens in English Wikipedia, 2) ViT with parameters ranging from 1M to 60M with 1% to 10% training images in ImageNet1k. In contrast to training from scratch in Sec. 3, each time before we train a transformer, we will use a pre-trained smaller one to initialize the model weights via the Net2Net (Chen et al., 2015) technique. In this experiment, the source models are trained from scratch and the one with the closest model size to the current training model will be chosen. For instance, to train a BERT with 80M parameters, we will use 66M model as its initializer. For the smallest size models, we extend their training procedure from their last checkpoints for the same iterations/epochs as the control groups to rule out the influence of longer training time.



(a) Scaling model size for BERT **(b)** Scaling model size for ViT

Figure 4: Comparison between training from scratch and training with Net2Net. The blue curves reproduce our data scarcity scaling law while the orange curves display the scaling trend after applying Net2Net. The radius of markers indicates the standard deviation. We find that model reusing effectively recovers the power law. Best viewed in color.

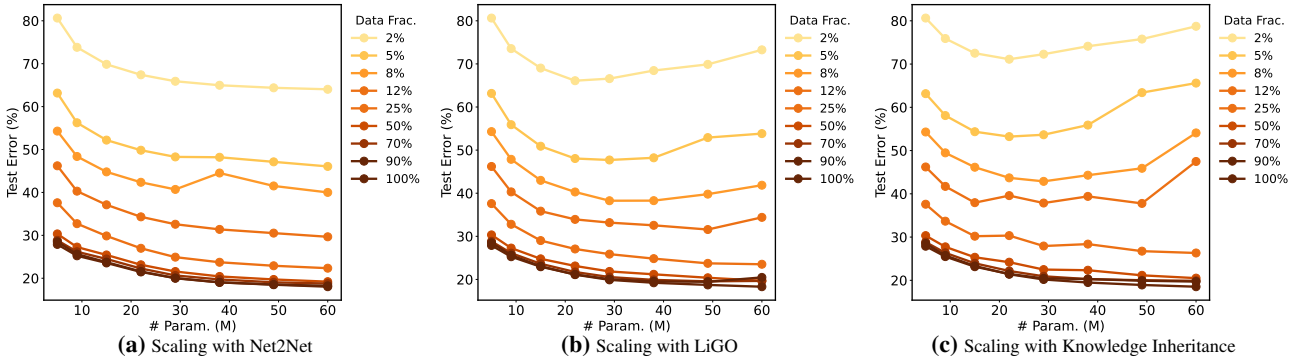
Observation. The main results are summarized in Fig. 4. In the plot, the blue curves reproduce the original data scarcity scaling law, where the error increases with the model parameters. The orange curves depict the performance versus parameter numbers when Net2Net is enabled to re-initialize models. We observe that by applying the model reusing techniques, the originally increasing curves all turn to decaying curves, which accurately fits the power law (Kaplan et al., 2020). This manifestation gives an affirmative answer to our hypothesis that **model reusing can indeed restore the data scarcity law back to the original power law.** Note that performance gain is significant when bringing data scarcity law back to a power law. With the 0.1% or 1% downsampled training data, the 80M BERT or 60M ViT are no longer trainable from scratch, leading to collapsed perplexity or accuracy. Once applying Net2Net, the test log-perplexity of BERT is enhanced by >1.0 and the classification precision of ViT is improved by 20% for free. We point out the starting points of all the curves are overlapped, which signifies the improvement is not brought by additional training epochs.

4.4. Ablation Studies

In this section, we study different variants of model reusing and compare it with other alternatives. Throughout the whole section, we choose ViT as the study objective, and we presume all results can be extended to BERT as well.

How does the Reusing Method Affect? In Sec. 4.3, we only test the Net2Net model reusing technique. In this ablation study, we hope to examine whether other reusing techniques introduced in Sec. 4.1 have a similar performance. We redo the experiments for LiGO (Wang et al., 2023) and knowledge inheritance (Qin et al., 2021). For LiGO, we follow the original paper and search the growth operator for 100 steps. For knowledge inheritance, we adopt the default inheritance rate scheduler. All the subsequent training remains the same for a fair comparison. We present the model scaling curves in Fig. 5. The observation is that 1) generally applying model reusing is effective to recover the scaling law from data scarcity domain. 2) Net2Net can faithfully restore the power law while other other techniques deviate from the law at the end. It suggests that Net2Net is more suitable for model scaling via reusing. This is probably because Net2Net with functionality preserving property (Chen et al., 2015) can better inherit knowledge to target models with less information loss. For the other two techniques, additional training components need to be introduced which may contribute to increased complexity at initialization, especially with limited data. Consequently, reaching consistent and desirable initialization with these two methods may not always be feasible.

Does the Choice of Source Model Matter? An interesting investigation is the effect of the source model selection. In our previous experiments, the source models are always the closest smaller one (i.e., the one with exactly one less head). Another straightforward option is to always choose the smallest one as the source model. We conduct this experiment following the same setting stated in Sec. 4.3. The results are reported in Tab. 2. We find that as long as model reusing is applied, no matter which source model is being used, the power law can be roughly recovered. However, as the model size grows while the data size keep shrinking, reusing from the first checkpoint underperforms reusing from the closest checkpoint. This reflects the power law



(a) Scaling with Net2Net **(b)** Scaling with LiGO **(c)** Scaling with Knowledge Inheritance

Figure 5: Comparison among various model reusing techniques. We apply **(a)** Net2Net, **(b)** LiGO, and **(c)** Knowledge Inheritance to initialize large model with different strategies. The comparison indicates Net2Net is the favorable solution as it can perfectly align with the power law while other two turn out to deviate from the law at the end. Best viewed in color.

# Param.		9M	15M	22M	29M	38M	49M	60M
90%	Scratch	23.83	21.55	20.46	20.24	22.50	20.02	20.83
	From Prev.	**22.58**	**20.86**	**19.18**	**18.63**	18.44	19.10	18.45
	From First	22.58	21.90	19.36	18.63	**18.03**	**18.04**	**17.93**
70%	Scratch	24.50	22.12	21.19	20.72	20.85	21.08	19.95
	From Prev.	**23.52**	**21.37**	**19.80**	19.06	18.90	18.99	19.59
	From First	23.52	22.80	20.04	**19.03**	**18.79**	**18.38**	**18.48**
50%	Scratch	26.20	23.95	26.53	21.92	25.93	20.98	21.34
	From Prev.	**24.19**	**22.54**	**20.74**	**19.87**	19.59	19.29	18.69
	From First	24.19	23.60	20.84	20.00	**19.27**	**19.01**	**18.69**
25%	Scratch	31.66	29.62	40.63	27.03	39.04	29.66	30.72
	From Prev.	**27.85**	**25.32**	**23.59**	24.38	**22.30**	**21.85**	22.43
	From First	27.85	27.00	24.10	**22.85**	22.53	22.09	**21.78**
12%	Scratch	41.64	39.56	59.72	42.69	58.52	46.68	97.23
	From Prev.	**34.38**	**31.32**	**29.65**	32.17	**29.43**	31.95	30.44
	From First	34.38	33.94	31.51	**30.85**	30.19	**29.65**	**28.81**
8%	Scratch	49.67	51.47	48.23	52.68	60.23	58.90	98.21
	From Prev.	**42.46**	**37.97**	**39.15**	**34.96**	**36.30**	39.64	**38.11**
	From First	42.46	41.20	39.93	39.04	48.35	**38.50**	38.50
5%	Scratch	60.37	59.47	65.52	67.58	76.83	96.95	97.51
	From Prev.	**49.31**	48.15	47.47	46.73	48.11	**46.06**	**45.54**
	From First	49.31	**45.55**	**43.67**	**45.87**	**46.41**	53.88	55.00

Table 2: Comparison of model reusing from the immediately previous one (Prev.) or the smallest one (First). Experiments are conducted on ViT and test errors (↓) are reported in the table. **Bold** font marks the best performer.

recovered from reusing the first checkpoint has a tiny divergence from the standard power curve. We conjecture that the learning capacity of 1M model is limited, which does not learn sufficient knowledge to support the training of the model that is way more immense in size.

Model Reusing versus the Alternatives? In addition to model reusing, regularization (Steiner et al., 2021), sparsity (Chen et al., 2022; Varma T et al., 2022), and data augmentations (Touvron et al., 2022; 2021a) are other simple and potent means to enhance data efficiency. To further validate the superiority of the model reusing technique, we proceed to conduct a comparison with the following baselines: (a) The "three augmentations" proposed in DeiT III (Touvron et al., 2022) as the extra data augmentation. (b) SNIP (Lee et al., 2018) as the pruning method to sparsify ViT model by 50% at initialization. We perform experiments on relatively large models ($40 \sim 60$M) and test the performance at the low data regime ($\leq 10\%$). The test errors are reported in Tab. 3.

Data Frac.		10%	9%	7%	5%	3%	1%
38M	Scratch	51.42	54.84	63.31	76.83	90.82	91.75
	Data Aug.	59.32	63.82	68.48	72.80	78.09	84.10
	Pruning	42.13	43.69	49.91	57.33	69.22	87.64
	Net2Net	**33.50**	**33.94**	**39.45**	**48.11**	**55.50**	**64.05**
49M	Scratch	55.89	63.17	95.48	96.95	82.22	92.92
	Data Aug.	56.56	69.12	66.41	71.11	77.50	84.06
	Pruning	40.75	42.16	52.16	56.17	70.47	87.94
	Net2Net	**34.61**	**36.31**	**41.80**	**46.06**	**54.80**	**63.79**
60M	Scratch	98.39	98.65	98.55	97.51	85.09	93.12
	Data Aug.	57.77	59.24	65.78	71.46	77.44	83.90
	Pruning	40.78	42.82	50.19	59.23	72.14	87.39
	Net2Net	**36.24**	**39.88**	**37.64**	**45.54**	**54.41**	**63.69**

Table 3: Comparison of model reusing with data augmentation and network pruning under the data scarcity regime. Experiments are conducted on ViT and test errors (↓) are reported in the table. **Bold** font marks the best performer.

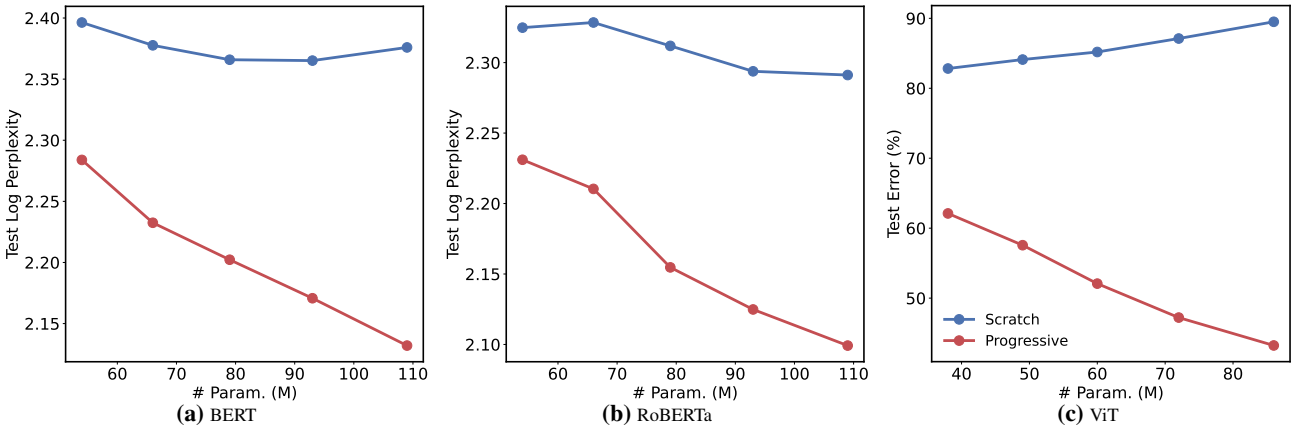
Our finding is that data augmentation and network pruning exhibit a positive effect on the model performance under the data scarcity regime. But both underperform the Net2Net model reusing. These outcomes suggest that model reusing should be regarded as a more favorable approach to fortify the model against highly deficient data scenarios. Nevertheless, we believe that these techniques are complementary and can be combined to further boost data efficiency.

5. Training Base Size Model using Scarce Data via Progressing Growing

After a comprehensive study of the model reusing effect on reproducing the power law, we deliver a general solution ambitiously aimed at training a base-size language or vision transformers with only 15M tokens or 10k images.

5.1. Training Recipe

Our objective is to train a transformer model of base size assuming very limited data are available. Inspired by our



(a) BERT **(b)** RoBERTa **(c)** ViT

Figure 6: Scaling curves for different transformers trained with 0.5% or 1% data and progressive model growing. We apply our progressive growing to different transformer backbones **(a)** BERT, **(b)** RoBERTa, and **(c)** ViT. We conclude that progressive growing can constantly suppress the test error while enlarging the model size. Best viewed in color.

former observation that model reusing can recover the power scaling law, we adopt a progressive approach where we start with a tiny size model and gradually increase the number of parameters. At each stage, the optimization process can be effectively stabilized, leading to the manifestation of a power law that consistently reduces the test error. The training procedure is illustrated in Fig. 3. This training strategy, which we refer to as *progressive growing*, allows us to steadily enlarge the network capacity. Based on our observation that initializing from the nearest neighbor model yields the best performance, we sequentially expand the parameter count, with each newly grown model initialized using the model from the previous stage that has the closest size. For the model growth algorithm, we employ Net2Net, which has demonstrated superior performance in our study.

5.2. Experiments

Experiment Settings. For the language transformer family, we consider BERT (Devlin et al., 2019) as well as RoBERTa (Liu et al., 2019). We increase the number of heads to 12 which yields a 110M model. We train both models with 0.5% English Wikipedia corpus with our progressive growing approach. Again, we follow the implementation of Tan & Bansal (2020) and train both BERT and RoBERTa with MLM objective for 400k iterations with a warm-up 10k steps. The sequence length is fixed to be 128 for pretraining both models. For BERT, the batch size is 256, and learning rate is set to $2e^{-4}$, while for RoBERTa, the used batch size is 1024 and learning rate is $8e^{-4}$. The language model starts from the 54M model, which yields the lowest perplexity. For ViTs, we expand the number of heads to 12 which results in an 80M model. Each training stage will contain 300 epochs with a batch size of 1024. The pre-trained 35M model is selected as the starting point because we empirically find that the 35M model has the best performance when trained from scratch with 1% images.

Results. We present our progressive training results in Fig. 6, where we also depict the performance of the intermediate models synthesized during the growth. We find that large models can consistently overcome data scarcity issues with the help of progressive training. Notably, compared with training from scratch, both BERT and RoBERTa can reach a log perplexity of around 2.1, which cannot be achieved without reusing initialization. Progressive training also significantly improves the classification accuracy for ViT base model from $\lesssim 10\%$ to $\gtrsim 60\%$. All the intermediate models during the growth also demonstrate remarkable performance gain and decaying error. Altogether, they endorse that progressive growing can be served as a universal approach to achieve data efficiency (Villalobos et al., 2022).

6. Conclusion

We study the neural scaling law under the data scarcity regime. Our finding is that data scarcity will break the power law and heavily ruin the performance of large transformers. We propose to leverage model reusing to recover the desirable scaling property for transformers. We conduct extensive experiments to support the benefit of model reusing in reproducing the power law. Our unified approach of progressive growth improves the data efficiency that successfully scales the transformer size with data scarcity.

Acknowledgements

We acknowledge support from the IBM Research AI Hardware Center, and the Center for Computational Innovation at Rensselaer Polytechnic Institute for the computational resources on the AiMOS Supercomputer. Z. Wang is in part supported by US Army Research Office Young Investigator Award W911NF2010240 and the NSF AI Institute for Foundations of Machine Learning (IFML).

References

Aghajanyan, A., Yu, L., Conneau, A., Hsu, W.-N., Hambardzumyan, K., Zhang, S., Roller, S., Goyal, N., Levy, O., and Zettlemoyer, L. Scaling laws for generative mixed-modal language models. *arXiv preprint arXiv:2301.03728*, 2023.

Bahri, Y., Dyer, E., Kaplan, J., Lee, J., and Sharma, U. Explaining neural scaling laws. *arXiv preprint arXiv:2102.06701*, 2021.

Bello, I., Fedus, W., Du, X., Cubuk, E. D., Srinivas, A., Lin, T.-Y., Shlens, J., and Zoph, B. Revisiting resnets: Improved training and scaling strategies. *Advances in Neural Information Processing Systems*, 34:22614–22627, 2021.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. In *Proceedings of NeurIPS*, 2020.

Caballero, E., Gupta, K., Rish, I., and Krueger, D. Broken neural scaling laws. *arXiv preprint arXiv:2210.14891*, 2022.

Cai, R., Zhang, Z., and Wang, Z. Robust weight signatures: Gaining robustness as easy as patching weights? *arXiv preprint arXiv:2302.12480*, 2023.

Chan, K., Yu, Y., You, C., Qi, H., Wright, J., and Ma, Y. Redunet: A white-box deep network from the principle of maximizing rate reduction. *Journal of machine learning research*, 23(114), 2022.

Chen, C., Yin, Y., Shang, L., Jiang, X., Qin, Y., Wang, F., Wang, Z., Chen, X., Liu, Z., and Liu, Q. bert2bert: Towards reusable pretrained language models. *arXiv preprint arXiv:2110.07143*, 2021.

Chen, T., Goodfellow, I., and Shlens, J. Net2net: Accelerating learning via knowledge transfer. *arXiv preprint arXiv:1511.05641*, 2015.

Chen, T., Zhang, Z., Liu, S., Zhang, Y., Chang, S., and Wang, Z. Data-efficient double-win lottery tickets from robust pre-training. In *International Conference on Machine Learning*, pp. 3747–3759. PMLR, 2022.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255, 2009.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL*, 2019.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *Proceedings of ICLR*, 2021.

Fedus, W., Zoph, B., and Shazeer, N. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *J. Mach. Learn. Res*, 23:1–40, 2021.

Geiping, J. and Goldstein, T. Cramming: Training a language model on a single gpu in one day. *arXiv preprint arXiv:2212.14034*, 2022.

Gong, C., Wang, D., Li, M., Chandra, V., and Liu, Q. Vision transformers with patch diversification. *arXiv preprint arXiv:2104.12753*, 2021.

Gong, L., He, D., Li, Z., Qin, T., Wang, L., and Liu, T. Efficient training of bert by progressively stacking. In *International conference on machine learning*, pp. 2337–2346, 2019.

Henighan, T., Kaplan, J., Katz, M., Chen, M., Hesse, C., Jackson, J., Jun, H., Brown, T. B., Dhariwal, P., Gray, S., et al. Scaling laws for autoregressive generative modeling. *arXiv preprint arXiv:2010.14701*, 2020.

Hernandez, D., Kaplan, J., Henighan, T., and McCandlish, S. Scaling laws for transfer. *arXiv preprint arXiv:2102.01293*, 2021.

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Hoffmann, J., Borgeaud, S., Mensch, A., Buchatskaya, E., Cai, T., Rutherford, E., Casas, D. d. L., Hendricks, L. A., Welbl, J., Clark, A., et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

Jiang, Z., Chen, T., Chen, X., Cheng, Y., Zhou, L., Yuan, L., Awadallah, A., and Wang, Z. Dna: Improving few-shot transfer learning with low-rank decomposition and alignment. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XX*, pp. 239–256. Springer, 2022.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

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. *arXiv preprint arXiv:2001.08361*, 2020.

Kim, H., Papamakarios, G., and Mnih, A. The lipschitz constant of self-attention. In *International Conference on Machine Learning (ICML)*, 2021.

Lee, N., Ajanthan, T., and Torr, P. H. Snip: Single-shot network pruning based on connection sensitivity. *arXiv preprint arXiv:1810.02340*, 2018.

Li, H., Xu, Z., Taylor, G., Studer, C., and Goldstein, T. Visualizing the loss landscape of neural nets. *Advances in neural information processing systems*, 31, 2018.

Li, T., Shetty, S., Kamath, A., Jaiswal, A., Jiang, X., Ding, Y., and Kim, Y. Cancergpt: Few-shot drug pair synergy prediction using large pre-trained language models. *arXiv preprint arXiv:2304.10946*, 2023.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Ma, Y., Tsao, D., and Shum, H.-Y. On the principles of parsimony and self-consistency for the emergence of intelligence. *Frontiers of Information Technology & Electronic Engineering*, 23(9):1298–1323, 2022.

Merity, S., Xiong, C., Bradbury, J., and Socher, R. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*, 2016.

Molybog, I., Albert, P., Chen, M., DeVito, Z., Esiobu, D., Goyal, N., Koura, P. S., Narang, S., Poulton, A., Silva, R., et al. A theory on adam instability in large-scale machine learning. *arXiv preprint arXiv:2304.09871*, 2023.

Paul, M., Ganguli, S., and Dziugaite, G. K. Deep learning on a data diet: Finding important examples early in training. *Advances in Neural Information Processing Systems*, 34: 20596–20607, 2021.

Prato, G., Guiroy, S., Caballero, E., Rish, I., and Chandar, S. Scaling laws for the few-shot adaptation of pre-trained image classifiers. *arXiv preprint arXiv:2110.06990*, 2021.

Qin, Y., Lin, Y., Yi, J., Zhang, J., Han, X., Zhang, Z., Su, Y., Liu, Z., Li, P., Sun, M., et al. Knowledge inheritance for pre-trained language models. *arXiv preprint arXiv:2105.13880*, 2021.

Rives, A., Meier, J., Sercu, T., Goyal, S., Lin, Z., Liu, J., Guo, D., Ott, M., Zitnick, C. L., Ma, J., and Fergus, R. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences*, 118(15), 2021. ISSN 0027-8424. doi: 10.1073/pnas.2016239118.

Rosenfeld, J. S., Rosenfeld, A., Belinkov, Y., and Shavit, N. A constructive prediction of the generalization error across scales. *arXiv preprint arXiv:1909.12673*, 2019.

Schulman, J., Zoph, B., Kim, C., Hilton, J., Menick, J., Weng, J., Uribe, J., Fedus, L., Metz, L., Pokorny, M., et al. Chatgpt: Optimizing language models for dialogue, 2022.

Shi, H., Gao, J., Xu, H., Liang, X., Li, Z., Kong, L., Lee, S., and Kwok, J. T. Revisiting over-smoothing in bert from the perspective of graph. *arXiv preprint arXiv:2202.08625*, 2022.

Sorscher, B., Geirhos, R., Shekhar, S., Ganguli, S., and Morcos, A. S. Beyond neural scaling laws: beating power law scaling via data pruning. *arXiv preprint arXiv:2206.14486*, 2022.

Steiner, A., Kolesnikov, A., Zhai, X., Wightman, R., Uszkoreit, J., and Beyer, L. How to train your vit? data, augmentation, and regularization in vision transformers. *arXiv preprint arXiv:2106.10270*, 2021.

Tan, H. and Bansal, M. Vokenization: Improving language understanding with contextualized, visual-grounded supervision. *arXiv preprint arXiv:2010.06775*, 2020.

Tay, Y., Dehghani, M., Abnar, S., Chung, H. W., Fedus, W., Rao, J., Narang, S., Tran, V. Q., Yogatama, D., and Metzler, D. Scaling laws vs model architectures: How does inductive bias influence scaling? *arXiv preprint arXiv:2207.10551*, 2022.

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., and Jégou, H. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pp. 10347–10357, 2021a.

Touvron, H., Cord, M., Sablayrolles, A., Synnaeve, G., and Jégou, H. Going deeper with image transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021b.

Touvron, H., Cord, M., and Jégou, H. Deit iii: Revenge of the vit. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXIV*, pp. 516–533. Springer, 2022.

Varma T, M., Chen, X., Zhang, Z., Chen, T., Venugopalan, S., and Wang, Z. Sparse winning tickets are data-efficient image recognizers. *Advances in Neural Information Processing Systems*, 35:4652–4666, 2022.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is All You Need. In *Proceedings of NeurIPS*, 2017.

Villalobos, P., Sevilla, J., Heim, L., Besiroglu, T., Hobbhahn, M., and Ho, A. Will we run out of data? an analysis of the limits of scaling datasets in machine learning. *arXiv preprint arXiv:2211.04325*, 2022.

Wang, P., Zheng, W., Chen, T., and Wang, Z. Anti-oversmoothing in deep vision transformers via the fourier domain analysis: From theory to practice. *arXiv preprint arXiv:2203.05962*, 2022.

Wang, P., Panda, R., Hennigen, L. T., Greengard, P., Karlinsky, L., Feris, R., Cox, D. D., Wang, Z., and Kim, Y. Learning to grow pretrained models for efficient transformer training. *International Conference on Learning Representations*, 2023.

Xiong, R., Yang, Y., He, D., Zheng, K., Zheng, S., Xing, C., Zhang, H., Lan, Y., Wang, L., and Liu, T. On layer normalization in the transformer architecture. In *International Conference on Machine Learning*, pp. 10524–10533. PMLR, 2020.

Yang, X., Ye, J., and Wang, X. Factorizing knowledge in neural networks. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXIV*, pp. 73–91. Springer, 2022a.

Yang, X., Zhou, D., Liu, S., Ye, J., and Wang, X. Deep model reassembly. *Advances in neural information processing systems*, 35:25739–25753, 2022b.

Yu, Y., Chan, K. H. R., You, C., Song, C., and Ma, Y. Learning diverse and discriminative representations via the principle of maximal coding rate reduction. *Advances in Neural Information Processing Systems*, 33:9422–9434, 2020.

Yun, C., Bhojanapalli, S., Rawat, A. S., Reddi, S. J., and Kumar, S. Are transformers universal approximators of sequence-to-sequence functions? *arXiv preprint arXiv:1912.10077*, 2019.

Zhai, X., Kolesnikov, A., Houlsby, N., and Beyer, L. Scaling vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12104–12113, 2022.

Zhang, B., Titov, I., and Sennrich, R. Improving deep transformer with depth-scaled initialization and merged attention. *arXiv preprint arXiv:1908.11365*, 2019.

Zheng, W., Sharan, S., Jaiswal, A. K., Wang, K., Xi, Y., Xu, D., and Wang, Z. Outline, then details: Syntactically guided coarse-to-fine code generation. *arXiv preprint arXiv:2305.00909*, 2023.

Zhou, D., Kang, B., Jin, X., Yang, L., Lian, X., Jiang, Z., Hou, Q., and Feng, J. Deepvit: Towards deeper vision transformer. *arXiv preprint arXiv:2103.11886*, 2021.