Scaling in Machine Learning

“Extrapolating the spectacular performance of GPT-3 into the future suggests that the answer to life, the universe and everything is just 4.398 trillion parameters.” — Geoffrey Hinton

Pedro Aldighieri

March 13, 2025

Setting the Stage: Why Scaling Matters

- Scaling is a clear path to Artificial General Intelligence (AGI)
- Empirical evidence shows consistent improvements with scale
  - Performance gains follow predictable patterns across multiple orders of magnitude
  - Emergent capabilities appear at specific scale thresholds
- Scaling provides a roadmap for industry and research
  - Clear guidance for resource allocation (parameters, data, compute)
  - Ability to forecast future capabilities and technological trajectories
- Central question: Will scaling laws continue to hold indefinitely?
  - If yes: AGI seems inevitable given sufficient resources
  - If no: Where and why do they break down?

What is a Neural Network?

- Function that maps inputs to outputs $y = f(x; \theta)$
- f can take several forms or “architectures” of layered, non-linear structures (e.g., transformers, CNNs, RNNs)
- Data is (x, y), θ is a set of adjustable parameters
- θ is extremely large — DeepSeek has 671 billion, GPT 4.5, 12.8 trillion parameters
- What sets ML/AI apart from traditional optimization?
    - High-dimensionality and architecture
    - NNs can approximate a broad class of functions (Cybenko, 1989; Hornik et al., 1989)

Training Process

- Define prediction $\hat{y} = f(x; \hat{\theta})$ and target output y
- Let $\mathcal{L}(\hat{y}, y)$ be the loss (e.g., L2-norm, cross-entropy)
- In LLMs, $\hat{y}$ is a probability distribution over possible tokens, y is a one-hot vector
- Compute $\nabla_\theta \mathcal{L}$ via backpropagation
- Adjust weights $\theta^{n+1} \leftarrow \theta^n - \eta \nabla_\theta \mathcal{L}$, where η is the learning rate
- Rinse and repeat to minimize loss
- Process requires lots of compute

Features of Scale

- Parameters (N) — weights and biases
- Amount of training data (D)
- Compute (C)
- In the background: architecture or structure of the model (A)
- "Quality" of prediction:

$$Q = F(N, D, C; A)$$

- Marginal costs of parameters are trivial compared to compute
- Data costs in many cases non-linear (e.g., costs explode after scraping the web)

Matters of Scale

What are some scaling questions?

- Global returns to scale of $Q = F(N, D, C; A)$
- What is the optimal choice of inputs

$$\max_{N,D,C} Q \quad \text{s.t.} \quad p \cdot (N, D, C) \leq B$$

$$\min_{N,D,C} B \quad \text{s.t.} \quad Q \geq \bar{Q}$$

- Marginal returns of some inputs keeping others fixed
- Is the function homothetic?
- Fundamentally, what is F

Curses of Scale

- Before late 2010s, scaling wasn't obvious (Gwern, 2020)
- Research on neural nets had fizzled, considered a backwater
- Throwing parameters and compute at fixed data led to overfitting
- Old paradigm of bias-variance tradeoff
- What changed?
  1. Scaling training data was key (Li, 2023)
  2. Double-descent (Preetum et al., 2019): loss decreases, increases, then decreases again
  3. Pre-transformer architectures couldn't handle scaling (Kaplan et al., 2020)
  4. Compute keeps getting cheaper

Empirical Scaling Laws (Kaplan et al., 2020)

- Model architecture not very relevant, *conditional on being a transformer*
- Performance scales as a power law (e.g., $Q \propto N^{\alpha}$) over 6 orders of magnitude *when not bottlenecked*
- Larger models are more sample efficient:
    - Lower loss per unit of data processed
    - Less data units to achieve same performance
- Smaller models need less compute to process fixed amount of data
- Model size trade-off: benefit vs. compute cost of processing a unit of data
- Optimal model size for a given compute budget stops short of convergence

See Figure 1 and Figure 2.

Kaplan et al. (2020), Figure 1

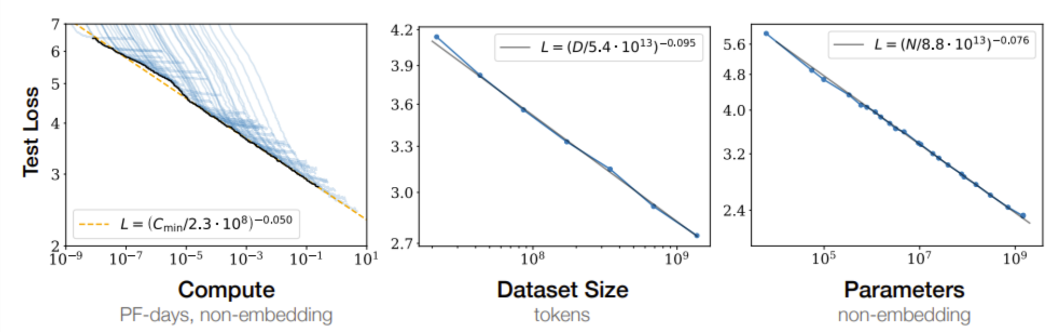

Figure 1: Kaplan et al. (2020), Figure 1

▸ Back to Text

Kaplan et al. (2020), Figure 2

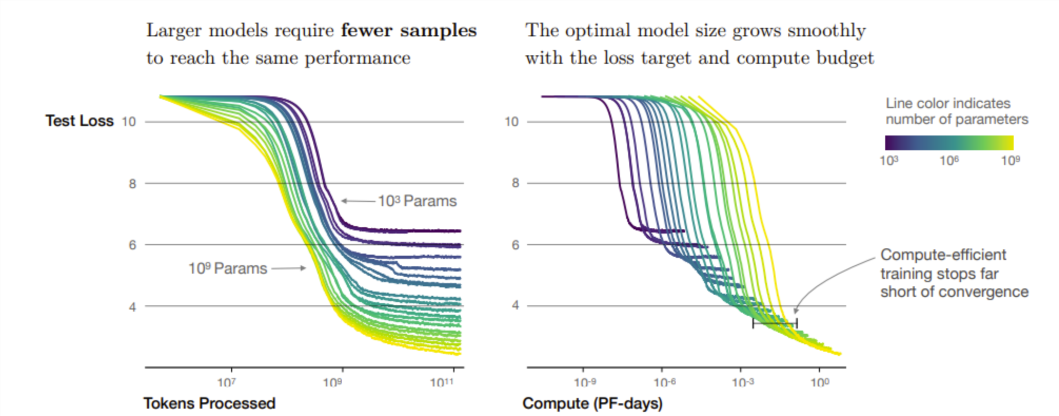

Figure 2: Kaplan et al. (2020), Figure 2

▸ Back to Text

Empirical Scaling Laws (Henighan et al., 2020)

- Henighan et al. (2020) show scaling laws generalize to other domains (image, video)
- This suggest scaling is a more general phenomenon
- Surprisingly, $N^* \propto C^\beta$, with $\beta = 0.7$ consistently across domains
- Given that $C \propto ND$, this implies $D^* \propto C^{1-\beta}$
- Thus, if we scale C, 70% should go towards increasing N

See Figure 3.

Henighan et. al (2020), Figure 5

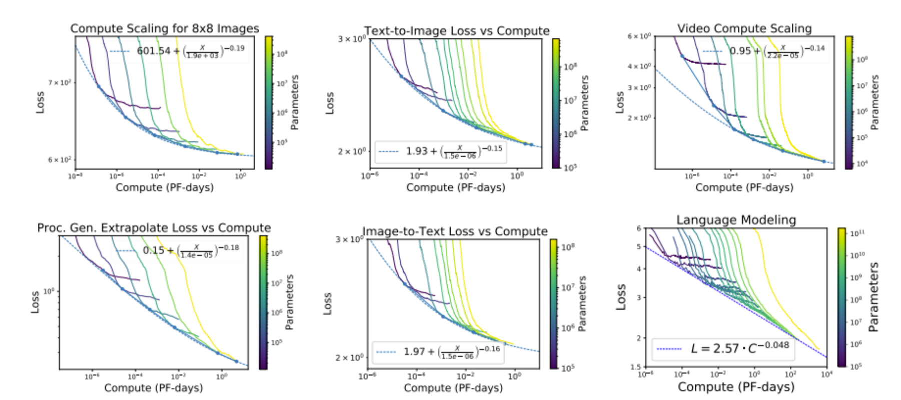

Figure 3: Henighan et al. (2020), Figure 5. Scaling laws with compute (total estimated floating point operations) for various domains, along with power-law plus constant fits (dashed). Note that very small models underperform compared to the trends when they model images or videos with very large contexts. Note also that the largest language models [BMR+20] were not trained to convergence.

▸ Back to Text

Papers Meet Reality

- Kaplan et al. (2020) recognize performance (loss) must level off
- Natural language has intrinsic entropy — zero loss is impossible
- But when does it level off?
- OpenAI (2023) show that scaling laws predicted performance of GPT-4 very well
- Do they hold for GPT 4.5, Claude Opus 3.5?

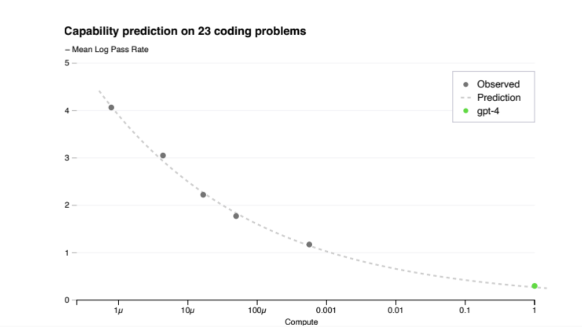

Figure 4: OpenAI (2023), Figure 2. Performance of GPT-4 and smaller models. The metric is mean log pass rate on a subset of the HumanEval dataset. A power law fit to the smaller models (excluding GPT-4) is shown as the dotted line; this fit accurately predicts GPT-4's performance. The x-axis is training compute normalized so that GPT-4 is 1.

Theory Behind Neural Scaling Laws (Utkarsh & Kaplan 2020, Bahri et al. 2024)

Key Insight: Scaling laws arise from the intrinsic dimension of data manifolds

Resolution-Limited Scaling:

- Data lies on d-dimensional manifold in high-dimensional space
- Neural nets transform inputs to d-dimensional representation
- Performance limited by ability to ”resolve” true function or training data

Intuition: If we pack d-dimensional space with tiles around D points , average distance between points $s \propto D^{-1/d}$

Mathematical Formulation of Neural Scaling Laws

Data-Limited Regime:

- Any point in manifold $\leq D^{-1/d}$ away from some data point
- NNs linearly interpolate any 2 data points
- Consider x' in the neighborhood of x, and map f from manifold to output space:

$$f(x') = f(x) + \nabla f(x)(x' - x) + O((x' - x)^2)$$

- If x in training data, NN matches first two terms
- But $|x' - x| \leq D^{-1/d}$ then error $O((x' - x)^2) \approx D^{-2/d}$
- For L2 loss, we square errors so $L \propto D^{-4/d}$
- At same time, scaling laws give us $L \propto D^{-\alpha}$, so $\alpha \geq 4/d$

Authors show a very similar argument applies to parameter-constrained models

Empirical Prediction: If we know d then, we can compare α to empirical $\hat{\alpha}$

Testing Theoretical Predictions

Empirical Prediction: If we know d then, we can compare α to empirical $\hat{\alpha}$ How do predictions fare?

- Authors start with simulations where d is known, and $\hat{\alpha} \approx 4/d$
- But theoretical predictions imply dimension of natural language is around 43
- In practice, trying to recover manifold dimensions inside LLMs leads to substantially higher $d \approx 100$ (Utkarsh and Kaplan 2020)

Aftermath

- Patel (2023) argues N and C can scale, but data supply might be inelastic
- If we hit a data wall, then whether scaling works or not is moot
- More fundamental question: do we hit decreasing RTS at some point?
- Loss must plateau, but does performance?
- Also, if cross-entropy loss on validation data is such a good predictor of model performance, why do we have so many benchmarks (MMLU, GPQA, AIME, FrontierMath, ARC)?