Trends in Frontier AI Model Count: A Forecast to 2028

Iyngkarran Kumar[*1] and Sam Manning[2]

[1]University of Edinburgh
[2]Centre for the Governance of AI

April 24, 2025

Abstract

Governments are starting to impose requirements on AI models based on how much compute was used to train them. For example, the EU AI Act imposes requirements on providers of general-purpose AI with systemic risk, which includes systems trained using greater than 10^{25} floating point operations (FLOP). In the United States' AI Diffusion Framework, a training compute threshold of 10^{26} FLOP is used to identify "controlled models" which face a number of requirements. We explore how many models such training compute thresholds will capture over time. We estimate that by the end of 2028, there will be between 103-306 foundation models exceeding the 10^{25} FLOP threshold put forward in the EU AI Act (90% CI), and 45-148 models exceeding the 10^{26} FLOP threshold that defines controlled models in the AI Diffusion Framework (90% CI). We also find that the number of models exceeding these absolute compute thresholds each year will increase superlinearly – that is, each successive year will see more new models captured within the threshold than the year before. Thresholds that are defined with respect to the largest training run to date (for example, such that all models within one order of magnitude of the largest training run to date are captured by the threshold) see a more stable trend, with a median forecast of 14-16 models being captured by this definition annually from 2025-2028.

1 Introduction

Recent years in machine learning have seen the rise of foundation models – AI systems that exhibit powerful and general-purpose capabilities. Governments across the world are starting to impose requirements on the development and deployment of the most capable such systems, such as the GPT o-series [OpenAI 2024a]. In December 2023 the European Union adopted the EU AI Act [European Union 2023], the world's first comprehensive legislation designed to govern the development and use of AI systems. Among other things, the Act imposes requirements on providers of general-purpose AI with systemic risk (GPAISR), as of August 2025, which includes systems trained using greater than 10^{25} floating point operations (FLOP)[1]. The European Union is not the only jurisdiction to propose requirements based on training compute thresholds; in one of its final acts the Biden administration issued the Artificial Intelligence Diffusion Framework [Federal Register 2025], which

---

[1]In this paper we often use the notation 10^X to refer to absolute compute thresholds. However the notation 1eX also appears at times which is interchangeable with the first notation. For example, 1e24 FLOP = 10^{24} FLOP, 5.3e25 FLOP = 5.3×10^{25} FLOP, etc.

as one of its key directives proposed a host of requirements on models above 10^{26} FLOP (named "controlled models" in the Framework) with the aim of maintaining leadership in AI technology among the US and its allies [Heim 2025].

However, it is well established that the training compute used for frontier models has been growing extraordinarily quickly, with mean model size growing by about 4-5x per year over the past decade [Sevilla, Jaime and Roldán, Edu 2024]. This has important implications for compute-based governance approaches such as those included in the EU AI Act and the Diffusion Framework. In April 2025, estimates suggest that there are 2 publicly available models trained using more than 10^{26} FLOP and approximately 30 publicly available models trained using more than 10^{25} FLOP [EpochAI 2025]. However, if current trends continue, these numbers may quickly grow. Governments will need to take this growth into account as they shape their AI governance efforts. By underestimating the number of models covered by their thresholds, they may fail to build sufficient capacity to implement their regulations or may impose regulatory burdens on an excessive number of actors.

With this in mind, we attempt to estimate the number of released models that will exceed various compute thresholds over the coming years. Extrapolating from current trends we conclude that by the end of 2028 there could be between 103-306 models exceeding the 10^{25} FLOP threshold (90% confidence interval) with a median estimate of 165, and 45-148 models exceeding the 10^{26} FLOP threshold (90% confidence interval), with a median estimate of 81. We also study "frontier-connected thresholds" – thresholds that are defined relative to the largest training run at any one point in time rather than based on the absolute amount of training compute used – and estimate that in the coming years there will be between 6-35 models released within 1 order of magnitude (OOM) of the largest training run that has taken place (90% CI) with a stable median of 14-16 models captured by this definition. However our analysis has limitations resulting from selection effects in the database that we extrapolate trends from, as well as uncertainty in key parameters that influence the projections.

Importantly, our estimates do not straightforwardly translate into the number of models in scope of the EU AI Act or AI Diffusion Framework. Our numbers may provide an overestimate in that neither the EU nor the US would apply regulations to models trained and only made available in other jurisdictions (e.g., China). Additionally, the AI Act only applies to general purpose AI - it is unclear whether image and video generation models (such as OpenAI's SORA [OpenAI 2024b] or Google DeepMind's Imagen [Google DeepMind 2024]) would count as GPAI. The Act may also not apply to models that were placed on the market before the relevant obligations come into force in August 2025. Further, both rules could affect the market for AI development, making it less attractive for companies to release in-scope models. If these effects occur, our analysis may end up overcounting the models captured by these thresholds. At the same time, our estimates may underestimate the number of models subject to the compute-based threshold requirements. This is because the regulation could apply not only to original GPAISR developers but also to companies that modify the models [Williams, Schuett, and Anderljung 2025], for example by adding software scaffolding around the model before making an application available to users. Since our analysis does not account for such adaptations, the actual number of affected models could be higher than our estimates suggest. Finally it should be noted that the EU AI Office has the ability to update the threshold in both directions - as does the US Bureau of Industry and Security in the U.S. as it relates to the AI Diffusion Framework - which would have to be taken into consideration when interpreting the predictions in future years.
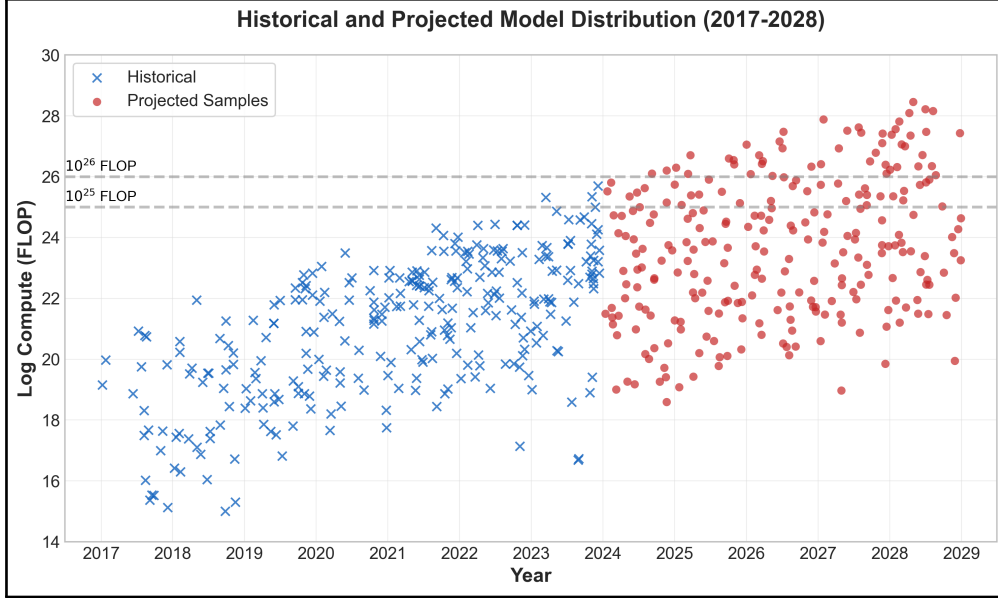
Figure 1: Historical data (2017-2023; blue) and a sample of our model's predictions (2024-2028; red) for the number of AI models exceeding 10^{25} and 10^{26} FLOP.

2 Methodology

Our aim is to forecast the number of models that will be released above different training compute thresholds over the next four years. To do this, we model scenarios for the distribution of AI model releases over training compute. Once these distributions are established, we can simply count the models that exceed each specific compute threshold.

2.1 Data

We use Epoch AI's Notable Models dataset [Epoch AI 2025c] as the main dataset for our analysis. To our knowledge, this is the most comprehensive publicly available dataset of machine learning models available with over 450 entries of models and their estimated training compute. However, it is important to note that ML models must satisfy one of the following criteria (the notability criteria [Epoch AI 2025d] to be included in the database:

- highly cited (over 1000 citations);

- large training cost (over $1,000,000, measured in 2023 USD);

- significant use (over one million monthly active users);

- state of the art performance (typically on a recognized ML benchmark);

- indisputable historical significance.

This makes the notable models dataset a strict subset of all machine learning models, and the selection effect applied by the notability criteria should be accounted for when interpreting the results in this paper. In short, this selection effect biases our median estimates towards being a lower bound estimate, especially for models that are multiple orders of magnitude of training compute

3

below frontier models[2]. For these models that are significantly smaller than the frontier models, it is also difficult to quantitatively estimate the extent to which the estimates derived from the notable models database will be lower than the actual value. However, anchoring on the full 90% prediction intervals presented in Section 3 provides some insurance against the effects of this selection effect (i.e: in some cases the 95th percentile projection is likely to be closer to the actual number of models exceeding the compute threshold). The effects of the Notable Model database selection effects, and how this can be accounted for within our model are discussed in further detail in Section 5.1.

Whilst the Notable Models database contains ∼450 entries with associated training compute values from 2010-2024, we fit our model based on the data from 2017-2023 (with 296 datapoints) to capture recent trends in the field. We do not use 2024 data to fit the model as the data is more likely to be incomplete - this is because the models tracked in the database lag behind models released by a handful of months. Visualizations of the model distribution in the dataset provide evidence in support of this, and are shown in appendix A.

2.2 Allocating total compute spending across different model scales

We forecast the distribution of models over training compute by projecting the total amount of compute that will be spent training AI models in the coming years, and then modeling how this compute stock is distributed over models of different size. We choose this approach as we expect the total training compute spending to be a strong and relatively simple-to-forecast constraint on model distributions (over training compute). The allocation of compute to models of different sizes also exhibits a linear trend that is simple to extrapolate (discussed below).

Note that there are alternate ways to estimate the future distribution of models over training compute; for example, one could fit a parametric distribution (e.g: a normal distribution) to historical data, extrapolate this into future years and then sample from this distribution. However we do not use this method for the following reasons: Firstly, it is unclear which parametric distributions, if any, are a good fit to historical distributions of training compute (see appendix A). Secondly, our approach has a more explicit focus on determining model distributions from total training compute spending - which, as mentioned above, we expect to be a strong and simple-to-forecast constraint over the next few years.

Using our approach, we project future model distributions over training compute by:

1. Projecting the total compute that will be used for AI workloads (both training and inference) with a median annual growth rate of roughly 4.1x.

2. Allocating this compute with a split of 40% of compute towards training models that are publicly released and 60% of compute towards other uses (including inference and compute used for research experiments) in 2025 and 2026, and a 30-70 allocation in 2027 and 2028.

3. Allocating the training compute across models of different scales - i.e: models within 1 OOM of the frontier model, models within 1 and 2 OOMs of the frontier model, models within 2 and 3 OOMs of the frontier model etc. - by fitting to data from 2017-2023 and assuming these allocation trends hold over the coming years.

We will now discuss each of these in turn in greater detail.

We first begin by projecting the total amount of compute that will be used for AI workloads in the coming years. We use two sources for this.

---

[2]This is because models multiple orders of magnitude away from the frontier are less likely to be captured by the notable model criteria relative to models close to the frontier.

Firstly, we can look at historical growth rates of training compute from the Notable Models database. Doing so for the years that we fit the model (2017-2023), we find a rapid growth rate of 6.3x annually for the total training compute used to train AI models. Assuming that the historical allocation between model training and inference has stayed roughly constant, we can generalise this to a 6.3x growth in compute used for AI workloads.

On the other hand, [Dean 2024] models the growth in compute for AI workloads as increasing at a rate of 3.4x per year. This is the compound growth rate resulting from a 2.25x increase per year of the compute stock that can be used for AI workloads, and a 1.5x increase per year of the share of this stock that actually is allocated to AI training, inference and other uses[3]. The 2.25x growth rate in global AI-relevant compute stocks results from a 1.35x growth rate in the physical stock of AI chips and 1.65x in chip efficiency. The 1.5x growth in share results from aggressive buildouts of data centers by leading AI developers, financed by revenues on the order of tens of billions of dollars resulting from highly performant AI models and agents.

Given the discrepancy between these two estimates we integrate both into our median growth rate forecast. We put more weight on the 3.4x per year increase in compute for AI workloads given the detailed analysis that leads to this figure, however we do not fully discount the rapid growth rate that has been historically observed. Specifically we give the 3.4x figure three times as much weight as the historically observed figure, but this weighting is subjective and predictions for alternative weightings are shown in Appendix G. Applying this weighting between the two growth rates leads to a median growth rate of the AI workload stock (i.e: the stock of compute used for AI training and inference) of 4.1x per year. To account for uncertainty in the actual annual growth rate of compute for AI workloads, we add noise to the median growth rate, drawn from a normal distribution with mean of 0 and standard deviation of 0.5.

Next, we model (a) how the AI compute stock is allocated between training models, and other uses (such as model inference and research experiments) and (b) how the total training compute is allocated across models of different sizes. Our baseline scenario for part (a) follows the allocations in a recently released analysis ([Dean 2024]), and is discussed in Section 2.4. To answer (b), we look at how training compute has been allocated to models of different sizes in recent years, and assume that these allocation trends hold in the coming years. This approach means that we do not have to explicitly commit to fixed parametric distributions.

Figure 2 below shows how training compute spending in the years 2020-2023 has been allocated to models of different sizes (data for 2017-2019 are shown in Appendix B). The x-axis represents the size of individual models. The y-axis shows the cumulative distribution function of training compute over model size - i.e: the fraction of training compute spent training models of size m or less. Differences in model sizes and their compute share can span orders of magnitude so both axes are log-scaled.

To get a concrete sense of what the plot shows, here are some conclusions that can be made by reading off the graph:

- In 2021, models of size 10^{22} FLOP or less contributed to approximately 1% of total training compute spending.

- In 2021, models of size 10^{21} FLOP or less contributed approximately 0.1% of total training compute spending

- In 2022, models of size 10^{21} FLOP or less contributed approximately 0.01% of total training compute spending

---

[3]One can think of these two quantities as AI compute capacity and AI compute usage.
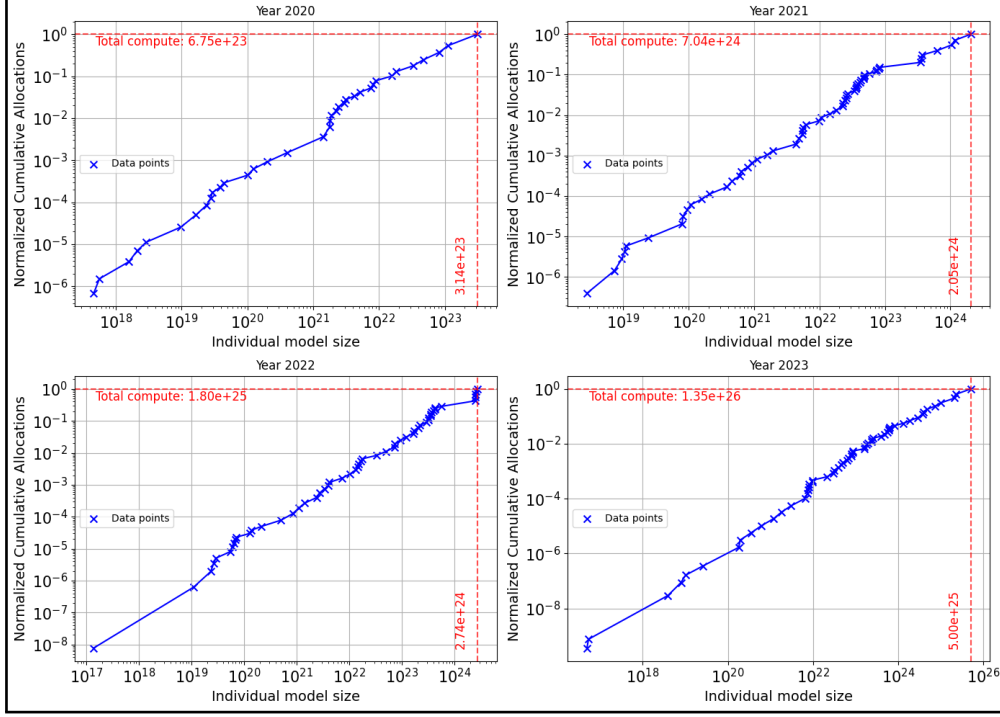
Figure 2: Compute allocation across model sizes for years 2020-2023, x-axis unnormalized.

- In 2023, models of size 10^{23} FLOP or less contributed just less than 10% total training compute spending.

Figure 2 also marks the largest model trained each year with the red vertical line. By definition, the largest model and all those smaller than it account for 100% of training compute spending in a given year; this is shown in Figure 2 by the fact that this line intersects with the line representing total training compute spending.

The relationship between model size m and the cumulative distribution function is consistently linear across 2020-2023 (and 2017-2019 - see Appendix B), suggesting a stable trend that can be extrapolated. The size of individual models grows each year, so to extrapolate this trend we normalize the x-axis by the largest model trained in each year (shown in Figure 3). Table 1 shows the compute allocations for different model sizes for 2023 that are derived from these plots.
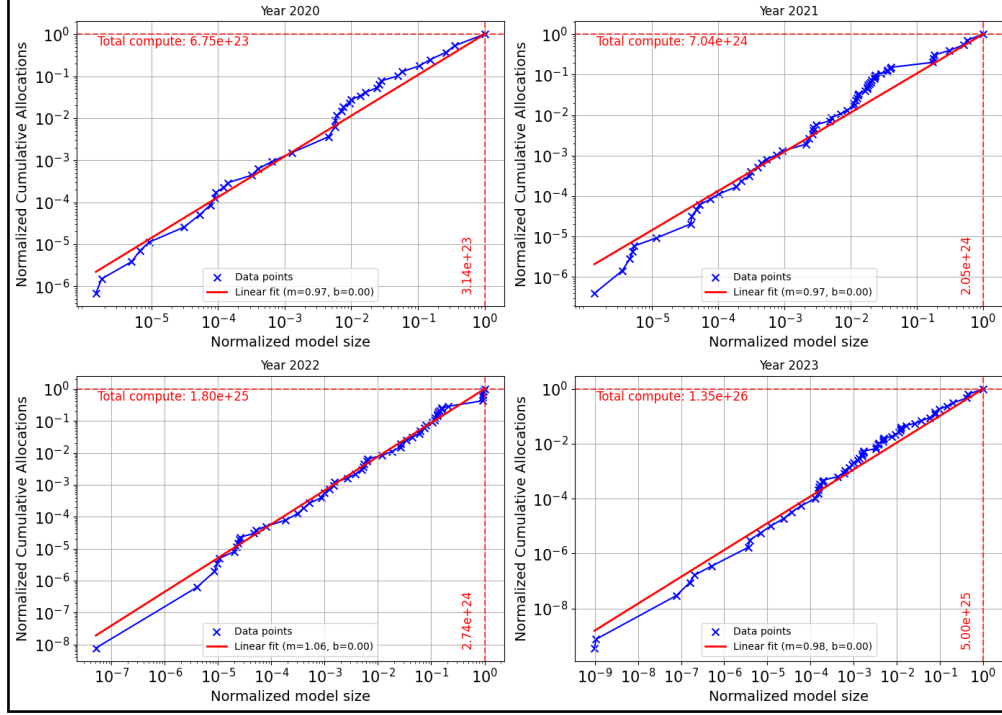
Figure 3: Compute allocation among model sizes for years 2020-2023, x-axis normalized by largest model trained that year. Linear fit shown.

Compute allocation in 2023					
Model size relative to Gemini Ultra	All models within 5–4 OOM	All models within 4–3 OOM	All models within 3–2 OOM	All models within 2–1 OOM	All models within 1 OOM
Model size (absolute)	5×10^{20}– 5×10^{21}	5×10^{21}– 5×10^{22}	5×10^{22}– 5×10^{23}	5×10^{23}– 5×10^{24}	5×10^{24}– 5×10^{25}
Fractional allocation (2 s.f.)	0.00011%	0.0010%	0.010%	8.6%	90%
Compute allocation (FLOP)	1.51×10^{22}	1.43×10^{23}	1.36×10^{24}	1.16×10^{25}	1.22×10^{26}

Table 1: 2023 allocations – Largest model: Gemini Ultra @ 5×10^{25} FLOP, total compute = 1.35×10^{26} FLOP. OOM refers to an order of magnitude of training compute. Allocations may not sum exactly to 1 due to rounding errors.

Before moving on, it is worth briefly considering the constraints and physical interpretations of the parameters of the linear fits - namely, the gradient (k) and the intercept (b). First, note that the linear fits must pass through (1,1) on the normalized plots (Figure 2B) - this is because, by

definition, the largest model and all those smaller than it (i.e: all models released in a given year) account for 100% of compute used. This means that the intercept of the linear fits - b - must be 0 (see Appendix C for details). Regarding k, Appendix C shows that models that are 10x the size of a smaller counterpart are allocated roughly 10^k times as much compute. Historical values for k are seemingly equally distributed across the range [0.9, 1.1] (shown in Appendix D), therefore our model samples k uniformly from the range [0.9,1.1]. The edge case of k=0.9 corresponds to models that are 10x as large being allocated $10^{0.9} = \sim 8.0$ times as much compute, whereas for the k=1.1 case this factor is $10^{1.1} = 12.6$ times as much compute. Table 17 shows allocations of compute across model size for a range of different k.

2.3 Modelling the largest training run

Recall from Table 1 that our method assigns training compute to models based on their size relative to the largest training run. Therefore, assumptions must be made about the size of the largest training run in a given year, which is done by making assumptions about the share of total training compute that the largest training run uses. In our model, this parameter is called the largest model share (or LMS). To illustrate what the the LMS represents, setting an LMS of 0.3 (or 30%) means that the largest model trained in a given year is uses 30% of the total training compute spent used across all models that year.

Given this setup, what are reasonable assumptions to make about the largest model share (LMS) parameter? One source of evidence is the historical values of the LMS derived from the Notable Models database; these are shown in Appendix F.1. One can see a range of values - for example, in 2018 ResNet-101 used approximately 78% of the training compute that was spent across all models tracked in the Notable Models database that year. More recently, in 2020 GPT-3 175B (davinci) accounted for 46% of compute spent across all models trained and released that year. On the lower end, data from 2022 shows Minerva 540B accounting for 15% of training compute spent that year on released models. Discounting the 2018 LMS as an outlier, the historical values appear relatively evenly distributed over the range [0.1,0.5] - therefore sampling the LMS parameter uniformly from this range (LMS $\sim$ Unif(0.10, 0.50)) is a reasonable first modelling attempt. However, lower values of the LMS exert a strong influence on the model's predictions for reasons discussed in Appendix F.2, so the lower bound of the sampling interval is extended to 0.05 to yield a refined model of LMS $\sim$ Unif(0.05, 0.50).

The LMS parameter is sampled uniformly from the bounds [0.05, 0.50] when the model is retrodicted for the years 2020-2023 in Section 4, however one more refinement is made when projecting the model forward for the years 2025-2028. A qualitative interpretation of the LMS parameter is the degree of concentration in the market of developers producing AI models at the largest scales. To illustrate, consider an LMS of 0.5, meaning that the largest training run used 50% of the training compute spent across all models that year. An implication of this is that at most, two developers can train a model at this scale. On the other hand an LMS of 0.05 (i.e: the largest training run uses 5% of training compute that was spent across all models released that year) means that, in theory, 20 different actors could participate in model development at this scale[4]. We expect an increasing number of developers to perform training runs at the largest scales in the coming years. This trend can be seen today with a number of relative newcomers, such as x.AI, Inflection, and Mistral, joining established actors such as OpenAI and Google DeepMind in training models at the very frontier of AI development[Epoch AI 2025b]. To quantitatively model this trend of a greater number of actors participating in frontier AI development, we sample the

---

[4]The assumption here is that no actor releases multiple models at this scale, but the general argument still holds without this.

LMS parameter log-normally for the years 2025-2028[5] from the bounds discussed above, in contrast to the uniform sampling used when retrodicting the model for 2020-2023 (these two LMS sampling distributions can be seen in Appendix E, alongside results using uniform sampling of the LMS for the years 2025-2028). For 2024, the largest model released in 2024 is documented in the Notable Models database as GPT-4o [Hurst et al. 2024] at 3.8×10^{25} FLOP, hence we set the LMS for 2024 to match this.

Finally, to generate model distributions to extend the historical data we randomly sample models from each bin until the total training compute allocated to that bin is met or exceeded.

2.4   Allocating compute between training, inference, and other workloads

Given estimates of the total compute stock, the next key stage involves allocating this stock between training, inference and other workloads. First a note on the different usages for compute. AI developers can use compute for model training (both pre- and post-training), as well as serving models to customers (external deployment/inference), using models in-house for research automation and monitoring (internal deployment/inference), compute for research experiments, generating synthetic data, and more. This analysis is primarily concerned with the compute used for training released models, hence compute usage is categories into 'training' and 'inference and other'.

[Dean 2024] estimates that in 2024 approximately 40% of compute was used for training (including both pre- and post-training), with the remaining share going to model inference and other uses. This is used as the starting point for the model. [Dean 2024] then estimate the share of compute allocated to training in the following years, estimating that in 2025 and 2026 40% of compute will be used for model training, 30% at the start of 2027 and 20% by the end of 2027. These allocations are mapped to the years of predictions for this forecast (2025-2028) in the first column of Table 2[6]. The 20-80 allocation of compute between training and other uses at the end of 2027 represents an aggressive scenario with respect to AI automation, and is therefore adjusted to a slightly more balanced 30-70 split for the baseline forecast in this paper, as can be observed in the second column of Table 2.

It is also worth noting that there is considerable uncertainty in the estimated present day compute allocations between training, inference and other workloads, with public estimates varying widely[7]. In Appendix H we present results for alternate allocation scenarios.

Year	Approximate training compute allocations [Dean 2024]	Training compute allocation (ours)
2025	40%	40%
2026	40%	40%
2027	30%	30%
2028	20%	30%

Table 2: Compute allocations for model training (not including research experiments) for 2025-2028

---

[5]Formally, $\log(LMS) \sim \mathcal{N}(\mu, \sigma)$ where $\mu = \mathbb{E}(\log(0.05), \log(0.50))$, $\sigma = (\log(0.50) - \log(0.05))/4$

[6]Concretely, we take the start of 2027 allocation (30%) as the allocation for all of 2027, and the end of 2027 allocation (20%) as the 2028 allocation.

[7]For example, the Epoch GATE model [Epoch AI 2025a] estimates an allocation of 90% of compute to model training in 2024, moving to 70% in 2028.

2.5 Frontier-connected thresholds

An alternative to setting training compute thresholds based on absolute compute limits (e.g: 10^{25} FLOP, 10^{26} FLOP) is to set thresholds based upon a model's proximity to the largest model trained to date[8]. To get a sense for how such a threshold would operate, at a given point in time a model regulator could require that any model trained over the next 3 months that is either (a) within 1 OOM of the largest model that existed at the start of this period or (b) exceeding the size of this largest model is subjected to additional reporting or transparency requirements. We project trends for these 'frontier-connected' thresholds for models that are within 0.5, 1 and 1.5 orders of magnitude of the largest model to date.

3 Results

In this section we present the results of our model for both absolute compute thresholds (e.g: 10^{25} FLOP, etc.) and for frontier-connected thresholds that incorporate models within 0.5, 1.0 and 1.5 orders of magnitude from the largest model released. We present the results of our model in the format (5, 50, 95), where 5, 50, 95 refer to the 5th, 50th and 95th percentile projections of the model when running it 1000 times. The 90% confidence prediction intervals are presented to convey uncertainty about parameters of the model, such as the largest model share (LMS), compute growth rates and allocation gradient. Interpretations of these results should anchor on the full prediction interval, rather than a single point estimate (e.g: the median projection).

Section 4 aims to validate the model projections by retrodicting our model and comparing the results to the historically observed data. Here we find that the 90% prediction intervals capture all historically observed data points. Taken together, this provides considerable evidence that the 90% prediction intervals presented below are likely to capture the number of models released in the coming years above each respective threshold.[9]

3.1 Absolute compute thresholds

First, we can compare our model's projections to existing data on the number of AI models exceeding 10^{25} FLOP. The model predicts that there were 23 AI models exceeding 10^{25} FLOP at the end of 2024, which aligns well with the 24 found in [Epoch AI 2025b] (recall that we set the largest model in 2024 to the size of GPT-4o inline with the Notable Models database). Then, assuming these prediction intervals will hold over the coming years, what are the implications for training compute thresholds? By the end of 2028 (EOY 2028) our median and 95th percentile projections for the number of models exceeding the 10^{25} FLOP threshold in the EU AI Act are 165 and 306 respectively. Our median and 95th percentile projection for the number of models exceeding the 10^{26} threshold in the Diffusion Framework are 81 and 148.

It is also worth noting the growth rate of the number of models exceeding the AI Act's threshold each year. Considering the 95th percentile projections, we see that from 2024-2025 the number of models captured by the 10^{25} threshold increases by 37, and then 55, 82 and 105 in subsequent years. This highlights a superlinear growth trend; not only does the number of models captured by the Act's compute threshold increase, but it increases at a growing rate. This superlinear growth trend holds across all projections (5th, 50th and 95th percentile forecasts) and across all absolute compute thresholds. However, the multiplicative factor from one year to the next is decreasing, meaning the

---

[8]See Appendix A of [Anderljung et al. 2023] for discussion.

[9]Supporting code for this analysis will be made available at `https://github.com/IyngkarranKumar/compute_thresholds_public`.

Threshold (FLOP)	2024	2025	2026	2027	2028
$> 10^{25}$	[18, 23, 27]	[32, 45, 64]	[51, 77, 119]	[76, 117, 201]	[103, 165, 306]
$> 10^{26}$	[0, 0, 0]	[3, 7, 11]	[12, 24, 38]	[27, 47, 81]	[45, 81, 148]
$> 10^{27}$	[0, 0, 0]	[0, 0, 0]	[0, 2, 5]	[1, 10, 20]	[9, 27, 56]
$> 10^{28}$	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 3, 8]
$> 10^{29}$	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]

Table 3: Results for absolute thresholds. The table presents 90% prediction intervals [5th, 50th, 95th percentile] for the number of models exceeding each compute threshold. Results are cumulative, showing estimates for the number of models released by the end of each year.

growth is sub-exponential. For example, the multiplicative factor between median estimates for the 10^{25} FLOP threshold is 1.96, 1.71, 1.52 and 1.41. Once again, this holds across all thresholds and percentile estimates.

3.2 Frontier-connected thresholds

We now consider the results of our projections for the frontier-connected thresholds defined in 2.5. These results are shown in Table 4 The key takeaway is that across all projections (5th, 50th and 95th percentile) and all thresholds (within 0.5, 1.0 and 1.5 OOMs of the largest model), the number of models captured by the threshold from 2025-2028 stays roughly constant. The figures differ for 2024 as the largest training run is fixed at GPT-4o's size of 3.8×10^{25} FLOP. For example, our median projection for the number of models that are within 1 OOM of the largest model at a given date is stable at 14-16 models across each year from 2025-2028, and the other thresholds of 0.5 OOMs and 1.5 OOMs see a similar trend, with stable medians in the range of 6-8 and 20-24 respectively. This constancy in the number of models captured by the threshold may be a desirable property for regulatory bodies responsible for enforcement; however the extent to which the frontier-connected thresholds are preferable over absolute training compute thresholds will also depend on whether the risks posed by a model are in part due to the absolute amount of compute used to train the system or the amount used relative to the frontier.

Distance from frontier model	2024	2025	2026	2027	2028
Within 0.5 OOM	[11, 16, 21]	[3, 6, 14]	[2, 7, 16]	[3, 8, 15]	[3, 8, 17]
Within 1 OOM	[23, 31, 39]	[7, 14, 25]	[6, 15, 29]	[7, 16, 31]	[7, 15, 35]
Within 1.5 OOM	[36, 46, 60]	[12, 20, 39]	[11, 22, 46]	[11, 24, 50]	[11, 23, 55]

Table 4: Frontier-connected thresholds. Results show 90% prediction intervals [5th, 50th, 95th percentile] for models within specified distances of the frontier model. Results for each year represent new models released in that year only.

4 Verification

In this section we present results obtained by retrodicting our model for absolute compute thresholds 10^{23}, 10^{24} and 10^{25} FLOP, for which there exists data from the years 2020-2023. We also compare our models predictions with the number of AI models satisfying the frontier-connected thresholds. The model's 90% prediction intervals capture all of these historically observed data points, which provides considerable evidence that the predicted intervals in Section 3 will capture the actual number of models exceeding the thresholds.

4.1 Absolute compute thresholds

Threshold (FLOP)	2020	2021	2022	2023
$> 10^{23}$	2 (0,1,4)	9 (8,13,27)	29 (19,29,60)	54 (36,54,128)
$> 10^{24}$	0 (0,0,0)	3 (0,2,3)	8 (3,8,10)	19 (13,22,44)
$> 10^{25}$	0 (0,0,0)	0 (0,0,0)	0 (0,0,0)	4 (0,4,5)

Table 5: Absolute compute thresholds retrodiction. Each cell is formatted as O $(5, 50, 95)$ where $O, 5, 50, 95$ are the historically observed values, 5th percentile, 50th percentile (median) and 95th percentile prediction.

For the absolute compute thresholds 10^{23}, 10^{24} and 10^{25} FLOP, our 90% confidence interval captures the historically observed values for the years 2020-2023. Note however, that for a number of cells the historically observed value is closer to the model's 5th or 95th projection than it is to the median projection, highlighting the importance of interpreting the full prediction interval.

Distance from frontier model	2020	2021	2022	2023
Within 0.5 OOM	3 (1,6,12)	4 (1,4,19)	5 (1,5,12)	5 (2,4,18)
Within 1.0 OOM	7 (5,12,24)	19 (5,9,34)	16 (4,9,28)	11 (5,9,35)
Within 1.5 OOM	11 (8,18,39)	27 (8,15,51)	22 (7,15,44)	17 (9,14,50)

Table 6: Frontier-connected thresholds retrodiction. Each cell is formatted as $O(5, 50, 95)$ where $O, 5, 50, 95$ are the historically observed values, 5th percentile, 50th percentile (median) and 95th percentile prediction.

4.2 Frontier-connected threshold

For the 0.5, 1.0 and 1.5 OOM frontier-connected thresholds, our 90% confidence interval captures all of the historically observed values. Again, in some cases (e.g: 2021), the observed value is closer to the 5th or 95th percentile projection than it is to the median projection.

5 Limitations

In this section the limitations of our analysis are discussed. This includes the selection effect on the machine learning models tracked by the Notable Models database, as well as limitations resulting from uncertainty in key parameters that influence the model's predictions (such as the largest model share parameter) and compute growth rates.

5.1 Notable models selection effect

The Notable Models database is a strict subset of all machine learning models. Therefore using this dataset to model future trends will, in general, lead to point estimates of the number of models that exceed the absolute compute thresholds being a lower bound on the true number. However, the nearer a model is to the frontier, the closer this lower bound estimate will be to the actual number of models that exceed the threshold. To illustrate, consider the cases of setting a 10^{22} FLOP compute threshold and 10^{25} FLOP threshold in the year 2023. In 2023, the largest model released was Gemini Ultra at 5×10^{25} FLOP. Most models with training compute on the order of 10^{25} are likely to have satisfied at least one of the notability criteria[10] given their proximity to the frontier AI model of the day. However, models of size $\sim10^{22}$ FLOP in 2023 are far less likely to

---

[10]As a reminder, the notability criteria are:

- highly cited (over 1000 citations);
- large training cost (over \$1,000,000, measured in 2023 USD);
- significant use (over one million monthly active users);
- state of the art performance (typically on a recognized ML benchmark);
- indisputable historical significance.

have met the notability criteria and thus more likely to be excluded from the dataset, making point estimates of the trends of 10^{22} FLOP models less reliable.

One way in which this effect can be accounted for within our model is by setting the allocation gradient (k) to less than one. Qualitatively this means that smaller models get more compute relative to the median scenario in our baseline case (where the median value of k is 1.0). For example, in the case where k = 1.0, models that are within 1 OOM of the largest model trained in a given year are allocated 90% of the compute share, with models within 1 OOM and 2 OOM of the largest model getting a factor of 10 less (i.e., 9%). On the other hand, when k = 0.5, models within 1 OOM of the frontier get only 68% of compute, and those in the category below it get $10^{0.5}$ ≈ 3.2x less compute (other allocations are shown in Table 17). The results of our baseline model shown in Section 3 choose k based on historical values found from the Notable Models database, as it is unclear how exactly to modify this parameter to account for the notable models selection effect. However in Appendix I, results are presented in which k is sampled uniformly from the ranges [0.7,0.9] and [0.5, 0.7] to illustrate the outcome on compute thresholds counts when using this technique as a potential correction for the notable models selection effect.

Even within our baseline predictions (where k is sampled uniformly from [0.9,1.1]), it should be emphasized that our 90% prediction intervals are designed to capture uncertainty in the model parameters (such as largest model share, allocation gradient, and compute stock growth rate), not the uncertainty that arises from the Notable Models selection effect. However anchoring on the full prediction intervals will insure decisions against this selection effect to some extent. In other words, while our median projections are biased towards underestimating the actual number of models captured by the compute thresholds, this is less likely to be the case for our 95th percentile projections.

5.2 Uncertainty in key model parameters

Key parameters that influence the model's predictions are the largest model share parameter (LMS), the compute stock growth rate, the allocation gradient, and the allocation of the compute stock between training and inference. Our choices for most of these parameters are informed by their historical values, for which we have limited data (2017-2023). This introduces uncertainty into our model, which we account for by presenting 90% confidence intervals. However the limited historical data often only provides six (6) data points to calibrate these intervals with.

For example, the LMS in previous years has varied significantly over the range 0.1 (2024) to 0.46 (2020). In making projections for the years 2025-2028 we therefore sample the LMS (lognormally) from a wider range of [0.05,0.5]. Given that low values of the LMS can lead to substantially different projections compared to larger values (Appendix F.2 discusses this in further detail), future work could look to further calibrate this parameter as more data becomes available. The case for the compute stock growth rate is similar - the two sources of evidence that are used in this study (historical compute growth rates and [Dean 2024]'s forecast of future compute stock growth) show a noticeable discrepancy which we resolve by subjectively weighting the estimated growth rates. Valuable future work could look to conduct further analysis into the growth of compute that is available for AI training and inference.

6 Implications for Compute-Threshold-Based Governance Frameworks

Training compute is increasingly being used as a proxy for determining which AI models should be subject to requirements imposed by AI governance frameworks. As [Heim and Koessler 2024] argue, compute thresholds serve as an effective initial filter to identify potentially risky general-purpose AI models that warrant regulatory oversight and further scrutiny. This approach is valuable because training compute correlates with model capabilities and potential risks while being easily quantifiable compared to other inputs to AI development [Sastry et al. 2024].

However, our analysis demonstrates a critical challenge: any static compute threshold will capture an increasing number of models over time. By the end of 2028, our median projections indicate that 165 models will exceed the 10^{25} FLOP threshold established in the EU AI Act, and 81 models will surpass the 10^{26} FLOP threshold set in the US AI Diffusion Framework. Especially relevant is the superlinear growth pattern we observe. Not only does the number of models captured by these thresholds increase yearly, but the rate of increase accelerates.

This pace of increase could strain governmental capacity to enforce and monitor requirements while potentially imposing excessive regulatory burdens on a growing number of AI developers. In response, governments have at least two potential options. First, they can make regulatory burdens more proportionate for less risky models while increasing their regulatory capacity to handle the growing number of models in scope. Second, they can reduce the scope of requirements by excluding certain models from the regime. Both the EU AI Office in the EU and the Bureau of Industry and Security in the US have the authority to make such adjustments to the scope of their respective regulatory frameworks.

7 Conclusion

In this paper we estimate the number of AI models that will exceed training compute thresholds such as those proposed in the EU AI Act [European Union 2023] and the AI Diffusion Framework [Federal Register 2025]. Our method centres around estimating the total stock of compute that will be available for AI workloads (training and inference), and then allocating the training compute stock to models of different size following trends seen from 2017-2023. We find that 103-306 models will exist by the end of 2028 that exceed a 10^{25} FLOP compute threshold with a 90% confidence interval, and 45-148 models will exist above the 10^{26} FLOP threshold. For all compute thresholds defined with respect to an absolute value, the number of models exceeding the threshold increases at an accelerating (superlinear) rate.

We also analyse trends for compute thresholds that are connected to the largest frontier model at the time of their release and find for thresholds that capture models within 0.5, 1.0 and 1.5 orders of magnitude of the largest model, the median number of models captured remains roughly stable each year (at approximately 7, 15 and 21-22 respectively) - however we have wide 90% confidence intervals around these. We validate our predictions by retrodicting threshold counts for the years 2020-2023, with our 90% confidence intervals capturing all historically observed values.

However our analysis is limited by the selection effects applied by the Notable Models database which bias our median projections towards being a lower bound on the actual number of models that will exceed the absolute compute thresholds. Additionally uncertainty around model parameters such as the largest model share (LMS), allocation gradient and compute growth rate mean that the full 90% prediction interval should be accounted for when basing policy decisions on these results. These findings emphasize the need for policymakers and regulatory bodies to consider the rapid

growth in frontier AI model counts when designing and implementing compute-based governance frameworks, ensuring they have sufficient capacity to monitor and regulate an expanding number of models while maintaining flexibility to adjust thresholds as the AI landscape evolves.

Acknowledgements

References

OpenAI (2024a). *Introducing OpenAI o1*. `https://openai.com/o1/`. Accessed April 2025.

European Union (2023). *The EU AI Act*. `https://artificialintelligenceact.eu/`. Accessed April 2025.

Federal Register (2025). *Framework for Artificial Intelligence Diffusion*. `https://www.federalregister.gov/documents/2025/01/15/2025-00636/framework-for-artificial-intelligence-diffusion`. Accessed January 2025.

Heim, Lennart (2025). *Understanding the Artificial Intelligence Diffusion Framework: Can Export Controls Create a U.S.-Led Global Artificial Intelligence Ecosystem?* Tech. rep. PEA3776-1. RAND Corporation. URL: `https://www.rand.org/pubs/perspectives/PEA3776-1.html`.

Sevilla, Jaime and Roldán, Edu (2024). "Training Compute of Frontier AI Models Grows by 4-5x per Year". In: *Epoch AI Blog*.

EpochAI (Apr. 15, 2025). *Update on Large-Scale AI Models: 6 new models with 1e25 or more training FLOPs were released in the past two months, bringing the total count to 30*. EpochAI. URL: `https://x.com/EpochAIResearch/status/1910791160970555403` (visited on 04/20/2025).

OpenAI (2024b). *Sora*. `https://openai.com/sora/`. Accessed April 2025.

Google DeepMind (2024). *Imagen - Google DeepMind*. `https://deepmind.google/technologies/imagen-3/`. Accessed April 2025.

Williams, Sophie, Jonas Schuett, and Markus Anderljung (2025). "On Regulating Downstream AI Developers". In: *arXiv preprint arXiv:2503.11922*.

Epoch AI (2025c). *Notable AI Models*. `https://epochai.org/data/notable-ai-models`. Accessed April 2025.

— (2025d). *Notable AI Models Documentation: Inclusion Criteria*. `https://epoch.ai/data/notable-ai-models-documentation#inclusion`. Criteria for including models in the database.

Dean, Romeo (2024). *Compute Forecast — AI 2027*. `https://ai-2027.com/research/compute-forecast`. Accessed April 2025.

Epoch AI (2025b). *Models over 1e25 FLOP*. `https://epoch.ai/data-insights/models-over-1e25-flop`. Accessed April 2025.

Hurst, Aaron et al. (2024). "Gpt-4o system card". In: *arXiv preprint arXiv:2410.21276*.

Epoch AI (2025a). *GATE Model Playground.* `https://epoch.ai/gate`. Accessed April 2025.

Anderljung, Markus et al. (2023). "Frontier AI regulation: Managing emerging risks to public safety". In: *arXiv preprint arXiv:2307.03718.*

Heim, Lennart and Leonie Koessler (2024). "TTraining Compute Thresholds: Features and Functions in AI Regulation". In: *arXiv preprint.* URL: `https://arxiv.org/abs/2405.10799`.

Sastry, Girish et al. (2024). "Computing Power and the Governance of Artificial Intelligence". In: *arXiv preprint.* URL: `https://arxiv.org/abs/2402.08797`.

Sevilla, Jaime et al. (2022). "Compute trends across three eras of machine learning". In: *2022 International Joint Conference on Neural Networks (IJCNN).* IEEE, pp. 1–8.

Appendices

A   Historical distribution of notable models and fit data choice
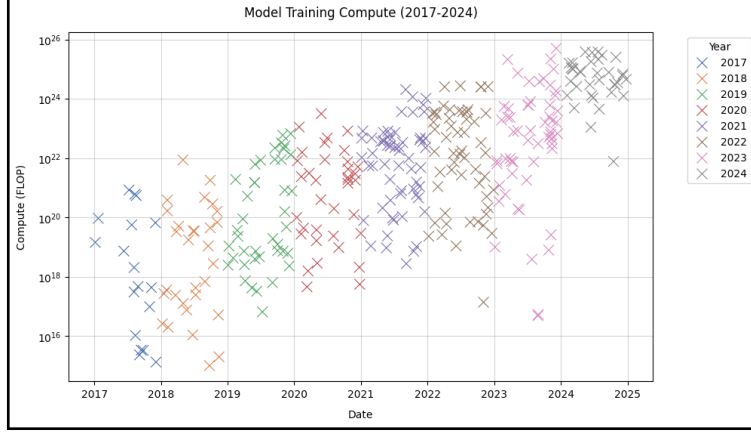

Figure 4: Historical distribution of models over training compute - scatter plot.
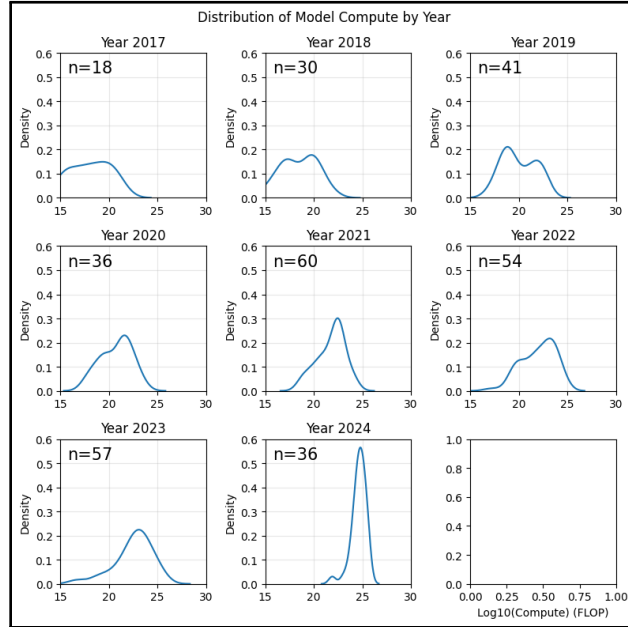

Figure 5: Historical distribution of models over training compute - KDEs

The dataset we use for this analysis contains records of models and their estimated training compute dating back to the 1950s. We filter out all models that were released before 2017 as this corresponds to the era before the release of the Transformer architecture at the core of most frontier models today. Additionally, we do not use 2024 data to fit our model. Observing the model distribution in Figure 4, 2024 data appears incomplete relative to the previous years, especially towards the lower end of the distribution (i.e: there is a notable absence of models on the lower end of compute usage). The deviation of 2024 data from previous years can also be seen in the kernel density estimates of Figure 5.
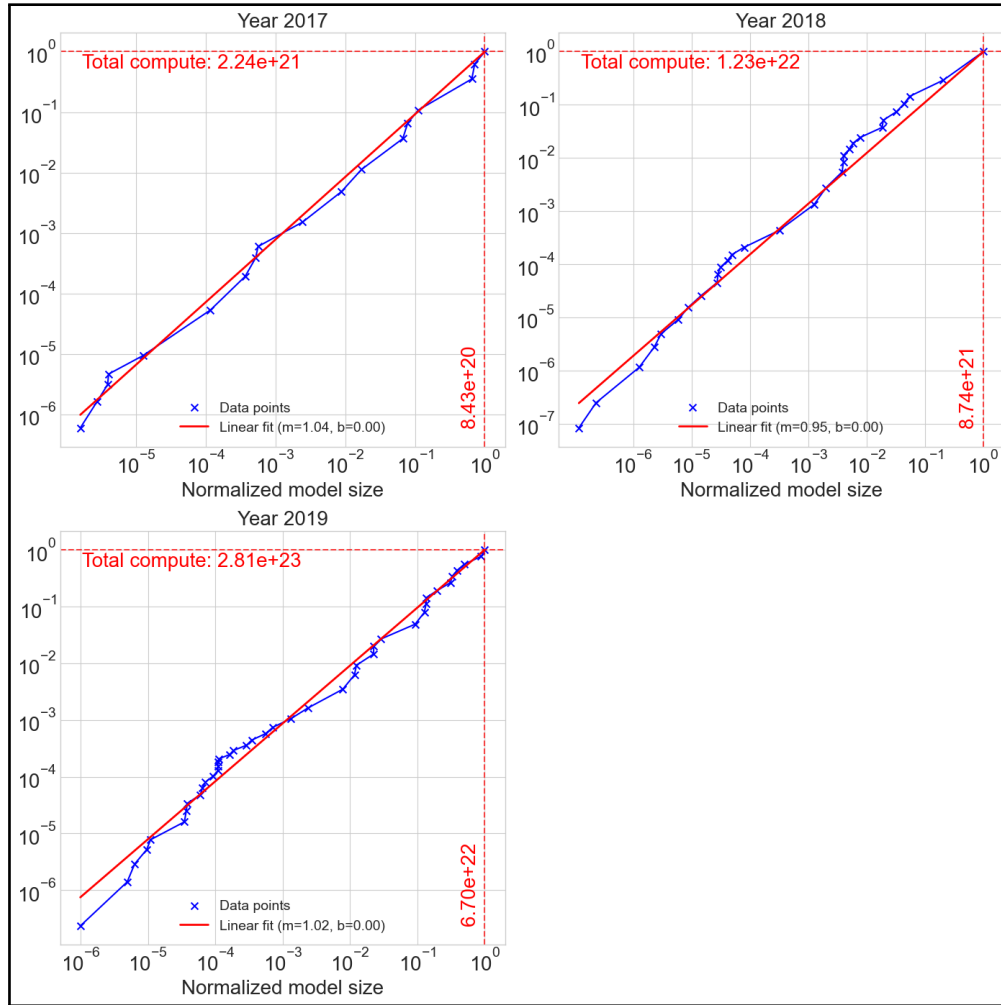
B   Compute allocations for 2017-2019

Figure 6: Linear fits to cumulative allocation plots for years 2017-2019.

C   Constraints and interpretation of the linear fits

This Appendix discusses the constraints on the linear fit to the compute allocation trends, and the interpretation of the allocation gradient parameter.

Observing historical data we see that the relationship between normalized model size (normalized by the largest model trained that year) - $\tilde{m}$ and the fraction of compute spent on models of size $\tilde{m}$ or less (the cumulative distribution function, denoted by $A(\tilde{m})$) is linear in log-space. Mathematically:

$$\log(A(\tilde{m})) = k \cdot \log(\tilde{m}) + b \tag{1}$$

Let that largest model trained in a given year be $m_{\max}$, then $\tilde{m}_{\max} = 1$. Models of size $m_{\max}$ or smaller (i.e., all models) take up all compute spending that year, therefore $A(\tilde{m}_{\max} = 1) = 1$. Enforcing this constraint on equation 1 means that $b = 0$ - and so equation (1) reduces to $A(\tilde{m}) = \tilde{m}^k$.

The parameter k determines how compute is allocated across models of different scales. To see this, let us first denote $a(m_1, m_2)$ as the amount of compute that is allocated to models in the range $[m_1, m_2)$. Consider also three sizes of models - m^*, $10m^*$, and $100m^*$. The compute allocated to models in the range $[m^*, 10m^*]$ is $a(m^*, 10m^*) = A(10m^*) - A(m^*) = (10^k - 1)m^k$ using equation (1). The compute allocated to models in the range $[10m^*, 100m^*)$ is $a(10m^*, 100m^*) = A(100m^*) - A(10m^*) = 10^k(10^k - 1)m^k$ after some simplification. Therefore, the relationship between $a(m^*, 10m^*)$ and $a(10m^*, 100m^*)$ is simply:

$$a(10m^*, 100m^*) = 10^k \cdot a(m^*, 10m^*) \tag{2}$$

In other words, scaling up model size by a factor of 10 leads to a factor of 10^k increase in compute allocated to models of this size. $k = 1$ means that these larger models get 10 times as much compute as their smaller counterparts. $k > 1$ means that they get a factor greater than 10, and $k < 1$ leads to a factor less than 10.

D Historical values of allocation gradient & allocations for varying gradient
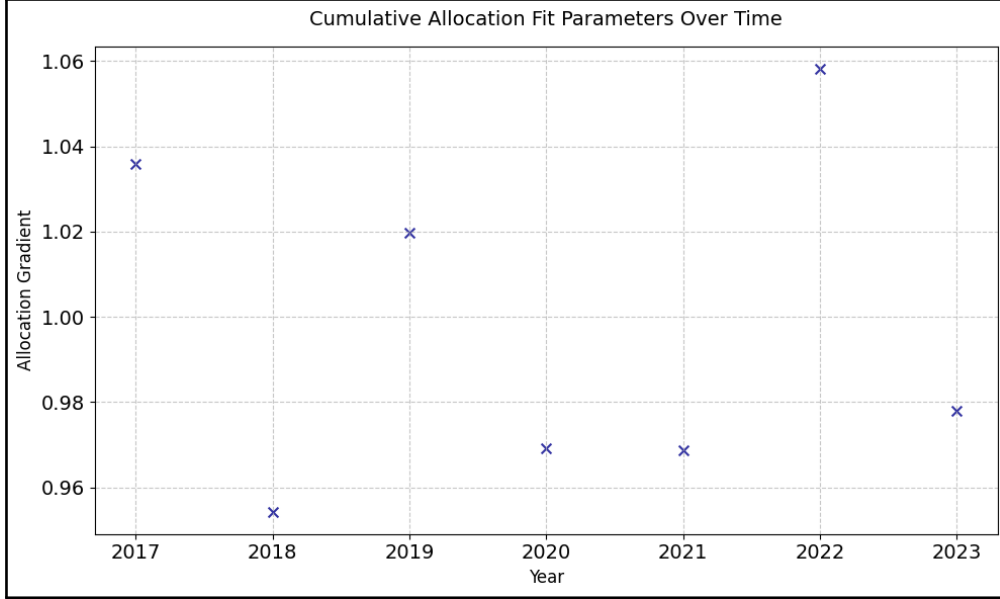

Figure 7: Allocation gradient parameter for 2017-2023.

Historical values of the gradient of the compute allocation fits (k). Informed by this, our model draws k uniformly from the range [0.9,1.1].

k	$[10^{-7}\text{-}10^{-6}]$	$[10^{-6}\text{-}10^{-5}]$	$[10^{-5}\text{-}10^{-4}]$	$[10^{-4}\text{-}10^{-3}]$	$[10^{-3}\text{-}10^{-2}]$	$[10^{-2}\text{-}10^{-1}]$	$[10^{-1}\text{-}10^{0}]$
0.5	0.068	0.22	0.68	2.2	6.8	22	68
0.75	0.0026	0.015	0.082	0.46	2.6	15	82
0.9	0.00035	0.0028	0.022	0.17	1.4	11	87
1.0	9×10^{-5}	0.0009	0.009	0.09	0.9	9	90
1.1	2.3×10^{-5}	0.00029	0.0037	0.046	0.58	7.3	92
1.25	3×10^{-6}	5.3×10^{-5}	0.00094	0.017	0.3	5.3	94
1.5	9.7×10^{-8}	3.1×10^{-6}	9.7×10^{-5}	0.0031	0.097	3.1	97

Table 7: Compute allocations (%) for various values of k. Each column shows allocation percentages for model size ranges, with models defined with respect to the largest training run. To illustrate: for the k=1.1 case, 92% of compute is allocated to models within an OOM of the largest model, 7.3% is allocated to models within 1 and 2 OOMs of the largest model, 0.58% to models within 2 and 3 OOMs of the largest model, etc.

E   Results under uniform sampling of LMS parameter

Section 2.3 discusses the choice to model the largest model share (LMS) parameter for the years 2025-2028 with a lognormal distribution (recall that the 2024 LMS is set such that the largest model is the size of GPT-4o), rather than a uniform distribution. In this appendix, Figure 8 shows the difference between these choices for the LMS sampling distributions, and Table 8 shows the threshold count results when the LMS parameter is sampled from the uniform distribution instead. In Table 8, notably less aggressive medians are observed which results from the median of the LMS distribution being larger than for the lognormal case (for reasons on why larger LMS values lead to less aggressive medians see Appendix F.2).

Figure 8: Left: Lognormal LMS distribution with bounds [0.05,0.5] (any value drawn from outside these bounds is resampled) and Right: Uniform LMS distribution. Medians, 5th and 95th percentile LMS indicated.

Threshold (FLOP)	2024	2025	2026	2027	2028
$> 10^{25}$	[19, 23, 28]	[31, 42, 67]	[47, 67, 116]	[65, 97, 185]	[88, 137, 287]
$> 10^{26}$	[0, 0, 0]	[4, 7, 10]	[12, 21, 36]	[23, 40, 76]	[39, 68, 142]
$> 10^{27}$	[0, 0, 0]	[0, 0, 0]	[0, 3, 5]	[3, 11, 19]	[12, 26, 51]
$> 10^{28}$	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 1]	[0, 4, 9]
$> 10^{29}$	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]

Table 8: Results for absolute compute thresholds when the LMS parameter is sampled uniformly from the range [0.05,0.5].

F   LMS parameter and influence of LMS on projections

F.1   Historical LMS

Figure 9: Historical data for the largest model and total training compute.

Figure 10: Historical data for the largest model share (LMS) parameter

Historical values of the LMS parameter and the total compute and the largest model each year that the LMS is derived from. The AlphaGo family of models have been removed from the dataset on the basis of being outliers, as done in similar analyses [Sevilla et al. 2022]. We fit the model on data from 2017-2023 and also discount the LMS for 2018 as it appears to be an outlier.

Our predictions sample the LMS uniformly from the range [0.05,0.5]. The upper bound is chosen to accommodate GPT-3 davinci accounting for ∼46% of training compute in 2020. The lower bound is chosen with the 2022 value of 0.15 in mind, however we incorporate a wide range underneath this value due to the strong influence of low LMS values on the model's predictions (see Appendix F.2 below).

F.2 Influence of LMS on projections

When experimenting with the model, we find that the **number** of models that exceed the compute thresholds grows very large as the LMS parameter tends to 0. This appendix explores in further detail why this is the case with a toy example.

Consider the following setup. We have 10^{30} FLOP of compute allocated to training models in a given year. The assignment of compute amongst model sizes is shown in the table below. (Note that we assume that increasing the model size by a factor of 10 leads to 10x as much allocated compute - corresponding to k=1 for the allocation gradient.)

Model size	Within 3-4 OOM	Within 2-3 OOM	Within 1-2 OOM	Within 1 OOM
Fractional allocation	0.009%	0.9%	9%	90%
Allocation (FLOP)	9×10^{26}	9×10^{27}	9×10^{28}	9×10^{29}

Table 9: Compute allocation across model size categories, where model size is given relative to the largest training run that year.

Now let us consider two cases: one in which the LMS = 0.05, and another in which the LMS = 0.50. In scenario 1, the largest model trained that year is $0.05 \times 10^{30} = 5 \times 10^{28}$ FLOP, and in scenario 2, the largest model is $0.5 \times 10^{30} = 5 \times 10^{29}$ FLOP. Now consider the number of models that can be drawn from each category. To approximate this we find the average model size of each category[11] - m_{avg} - and find how many times m_{avg} can be sampled from the compute allocations given in row three of the table above. This is show in the tables below.

In the case where the LMS=0.05, approximately 56 models of size m_{avg} can be sampled for each category, compared to the 5 sampled from each category for the case of LMS=0.5. More generally, the number of average-sized models that can be sampled from each category grows inversely with the size of the average model - and the average sized model of a category grows proportionally with the LMS parameter.

---

[11]This is done by taking the geometric mean of the model category bounds. We take the geometric mean instead of the arithmetic mean as the bounds are given in log space.

Model characteristics	Within 3-4 OOM	Within 2-3 OOM	Within 1-2 OOM	Within 1 OOM	Total
Size range (FLOP)	$5{\times}10^{24}$- $5{\times}10^{25}$	$5{\times}10^{25}$- $5{\times}10^{26}$	$5{\times}10^{26}$- $5{\times}10^{27}$	$5{\times}10^{27}$- $5{\times}10^{28}$	—
Average size[12] (FLOP)	$1.6{\times}10^{24}$	$1.6{\times}10^{25}$	$1.6{\times}10^{26}$	$1.6{\times}10^{27}$	—
Allocation (FLOP)	$9{\times}10^{26}$	$9{\times}10^{27}$	$9{\times}10^{28}$	$9{\times}10^{29}$	—
Number of samples	56	56	56	56	224

Table 10: **Scenario 1 - LMS=0.05**: Model size categories and compute allocation across different orders of magnitude (OOM) relative to the largest model.

Model characteristics	Within 3-4 OOM	Within 2-3 OOM	Within 1-2 OOM	Within 1 OOM	Total
Size range (FLOP)	$5{\times}10^{25}$- $5{\times}10^{26}$	$5{\times}10^{26}$- $5{\times}10^{27}$	$5{\times}10^{27}$- $5{\times}10^{28}$	$5{\times}10^{28}$- $5{\times}10^{29}$	—
Average size[13] (FLOP)	$1.6{\times}10^{25}$	$1.6{\times}10^{26}$	$1.6{\times}10^{27}$	$1.6{\times}10^{28}$	—
Allocation (FLOP)	$9{\times}10^{26}$	$9{\times}10^{27}$	$9{\times}10^{28}$	$9{\times}10^{29}$	—
Number of samples	5	5	5	5	20

Table 11: **Scenario 2 - LMS=0.5**: Model size categories and compute allocation across different orders of magnitude (OOM) relative to the largest model.
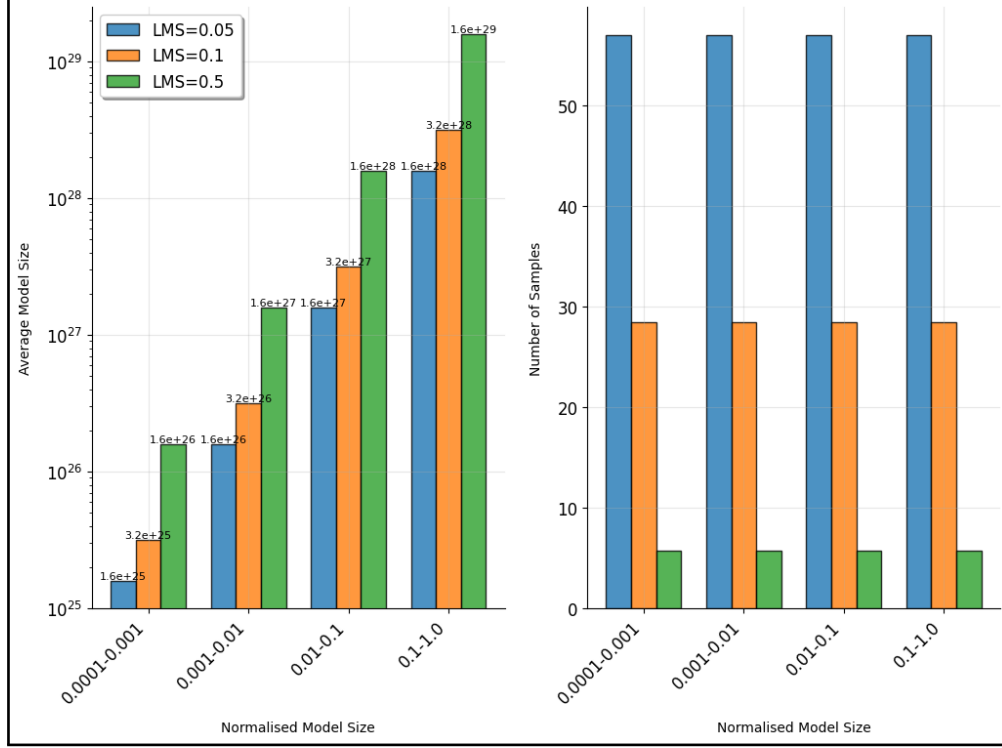
Figure 11: Left: The average-sized model for each category of model size for LMS = 0.05, 0.1, and 0.5. Right: The number of average-sized models that can be drawn from a given compute allocation for LMS = 0.05, 0.1 and 0.5.

The information in Tables 10, 11 is shown graphically in Figure 10, alongside another configuration where the LMS=0.1. We see that the LMS determines the size of each model category - when the LMS=0.5 the average size of a model in the largest category is 1.6×10^{29} FLOP whereas when the LMS = 0.1 and 0.05, the average size of a model in the largest category is 9.5×10^{28} and 1.6×10^{28} respectively. The number of samples that can be drawn given the compute allocations derived above is shown in the right hand side plot - for each category, roughly 50 samples can be drawn when the LMS=0.05, 30 when the LMS=0.1 and 5 when the LMS = 0.5 (the number of samples drawn from each category is constant across model sizes in the plot above because we choose an allocation gradient of k=1.)

G Results for varying growth rate weightings

Two sources of evidence inform the growth rates in the baseline scenario - the historical growth rate of training compute stocks and the compute forecast of [Dean 2024]. The baseline predictions gives three times as much weight to the latter (ascribing weights of 0.25 and 0.75 to each respective growth rate), but this is a subjective choice. This appendix shows predictions for other choices of growth rate weightings - namely weightings of (0.1,0.9), (0.33, 0.66) and (0.5, 0.5). As expected, more greater medians estimates are observed when assigning more weight to the larger historical growth rate.

Threshold (FLOP)	2024	2025	2026	2027	2028
$> 10^{25}$	[15, 20, 24]	[27, 41, 57]	[45, 71, 109]	[63, 106, 179]	[88, 150, 268]
$> 10^{26}$	[0, 0, 0]	[0, 5, 8]	[8, 18, 31]	[18, 36, 68]	[34, 64, 123]
$> 10^{27}$	[0, 0, 0]	[0, 0, 0]	[0, 0, 3]	[0, 4, 13]	[2, 15, 37]
$> 10^{28}$	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 4]
$> 10^{29}$	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]

Table 12: Results for absolute thresholds with growth weighting of (0.9,0.1) between historical growth rate (6.3x) and forecasted growth rate (3.4x).

Threshold (FLOP)	2024	2025	2026	2027	2028
$> 10^{25}$	[19, 24, 29]	[33, 48, 69]	[54, 84, 128]	[78, 129, 207]	[106, 178, 305]
$> 10^{26}$	[0, 0, 0]	[5, 8, 11]	[16, 26, 38]	[30, 53, 87]	[50, 87, 152]
$> 10^{27}$	[0, 0, 0]	[0, 0, 0]	[0, 2, 5]	[3, 11, 25]	[13, 30, 57]
$> 10^{28}$	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 1]	[0, 4, 9]
$> 10^{29}$	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]

Table 13: Results for absolute thresholds with growth weighting of (0.33,0.66) between historical growth rate (6.3x) and forecasted growth rate (3.4x).

Threshold (FLOP)	2024	2025	2026	2027	2028
$> 10^{25}$	[22, 26, 31]	[37, 52, 78]	[59, 90, 146]	[83, 136, 239]	[115, 195, 363]
$> 10^{26}$	[0, 0, 0]	[6, 9, 14]	[19, 31, 48]	[34, 59, 107]	[58, 101, 189]
$> 10^{27}$	[0, 0, 0]	[0, 0, 0]	[0, 4, 7]	[7, 17, 30]	[21, 41, 77]
$> 10^{28}$	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 2]	[0, 8, 17]
$> 10^{29}$	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]

Table 14: Results for absolute thresholds with growth weighting of (0.5,0.5) between historical growth rate (6.3x) and forecasted growth rate (3.4x).

H Results for alternate training compute allocations

Section 2.4 highlighted that the publicly available information on the allocation of compute between training, inference and other workloads was conflicting. Our baseline model uses a slightly adjusted version of the allocation in [Dean 2024]. However another source for the compute allocations is the Epoch GATE model [Epoch AI 2025a] - see the training-inference split graph. Note that the GATE model forecasts the allocation of effective compute - which is the physical compute stock multiplied by algorithmic progress. However in the near future (e.g: 2025-2028), these two quantities do not substantially differ. The allocations for this model are shown below.

Year	Approximate training compute allocations (GATE)
2024	90%
2025	90%
2026	70%
2027	70%
2028	70%

Table 15: GATE model compute allocations for training, 2025-2028

It's clear that these forecasts differ substantially to those in AI 2027. One partial explanation could be that the GATE forecasts incorporate compute used for experiments into the training share, however it is out of scope of this article to explore and resolve these discrepancies. Instead, Table 16 shows the predictions of the model for the absolute compute thresholds when training compute allocations are set as above.

Threshold (FLOP)	2024	2025	2026	2027	2028
$> 10^{25}$	[36, 46, 54]	[52, 74, 99]	[71, 114, 172]	[97, 160, 265]	[132, 221, 380]
$> 10^{26}$	[0, 0, 0]	[8, 13, 18]	[19, 34, 52]	[36, 63, 106]	[59, 105, 183]
$> 10^{27}$	[0, 0, 0]	[0, 0, 2]	[0, 4, 11]	[7, 17, 34]	[20, 40, 77]
$> 10^{28}$	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 4]	[0, 7, 20]
$> 10^{29}$	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]

Table 16: Results for absolute compute thresholds with GATE model allocations between training and inference compute.

I   Results for varying allocation gradients

k	10^{-7}–10^{-6}	10^{-6}–10^{-5}	10^{-5}–10^{-4}	10^{-4}–10^{-3}	10^{-3}–10^{-2}	10^{-2}–10^{-1}	10^{-1}–10^{0}
0.5	0.068	0.22	0.68	2.2	6.8	22	68
0.6	0.019	0.075	0.3	1.2	4.7	19	75
0.7	0.0051	0.025	0.13	0.64	3.2	16	80
0.8	0.0013	0.0084	0.053	0.34	2.1	13	84
0.9	0.00035	0.0028	0.022	0.17	1.4	11	87
1.0	0.00009	0.0009	0.009	0.09	0.9	9	90

Table 17: Compute allocations (%) for various values of k < 1. Each column shows allocation percentages for model size ranges relative to the largest model trained. For example, the second column from the right shows the compute allocations for models within 1 and 2 OOMs of the largest model trained.

Our baseline scenario samples the allocation gradient uniformly from the range [0.9, 1.1]. The median prediction in this scenario will therefore follow a compute allocation across model sizes as shown in the k=1 scenario in the table above. This modeling choice is made from observations of the allocation plots for the Notable models released in the years 2017-2023 (Appendix D).

However Section 5.2 discusses the limitations of the Notable Models database upon which these trends are based. One potential way to account for the Notable Models selection effect is to allocate more compute to smaller models relative to their larger counterparts. This can be seen in the table above where the k=0.5 case allocates ∼68% of compute that year to the largest model category, whereas the k=1.0 case allocates 90% of compute. More generally, increasing model size by 10x leads to a 10^k times increase in compute allocated, as shown in Appendix C.

This appendix presents model predictions for allocation gradients that allocate relatively more compute to smaller model sizes. Specifically, Table 18 presents the results of the model when the allocation gradient (k) is sampled from the range [0.7,0.9] (corresponding to a median scenario in which k = 0.8), and Table 19 presents the results of the model when the allocation gradient is sampled from the range [0.5,0.7] (corresponding to a median scenario in which k = 0.6). Notably more aggressive medians can be observed in the later years of the projection compared to the baseline - this is because these scenarios allocate relatively more compute to smaller models than the baseline, and in the 2027 and 2028, 10^{25} and 10^{26} FLOP models are multiple orders of magnitude away from the frontier models.

Threshold (FLOP)	2024	2025	2026	2027	2028
$> 10^{25}$	[36, 46, 54]	[52, 74, 99]	[71, 114, 172]	[97, 160, 265]	[132, 221, 380]
$> 10^{26}$	[0, 0, 0]	[8, 13, 18]	[19, 34, 52]	[36, 63, 106]	[59, 105, 183]
$> 10^{27}$	[0, 0, 0]	[0, 0, 2]	[0, 4, 11]	[7, 17, 34]	[20, 40, 77]
$> 10^{28}$	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 4]	[0, 7, 20]
$> 10^{29}$	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]

Table 18: Results for absolute compute thresholds when the allocation gradient is sampled uniformly from the range [0.7,0.9].

Threshold (FLOP)	2024	2025	2026	2027	2028
$> 10^{25}$	[15, 19, 23]	[36, 49, 67]	[70, 106, 155]	[116, 199, 314]	[205, 359, 637]
$> 10^{26}$	[0, 0, 0]	[2, 5, 9]	[12, 21, 34]	[29, 55, 88]	[63, 113, 196]
$> 10^{27}$	[0, 0, 0]	[0, 0, 0]	[0, 1, 4]	[3, 7, 16]	[12, 26, 52]
$> 10^{28}$	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 1, 6]
$> 10^{29}$	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]

Table 19: Results for absolute compute thresholds when the allocation gradient is sampled uniformly from the range [0.5,0.7].