An Empirical Alternative to the AI-2027 Takeoff Timelines

Andreas Robinson - 10/06/2025

TL;DR: The AI-2027 recursive self-improvement takeoff-timeline depends heavily on the authors' intuitive projections of human-only research progress, whereas this post develops a more empirically-grounded model of research-progress based on past trends, which suggests that AI-2027 could be substantially underestimating the likely timeline for AI recursive-takeoff (initiated by coding AIs), as well as underestimating the extent to which such a takeoff may be bottlenecked by compute. I also argue that there are intrinsic benefits to shifting in the direction of a more trend-based empirical approach to making these projections. Note that AI-2027's takeoff report is distinct from their timeline report, which was focused on the timeline leading up to the take-off phase and has already been widely critiqued elsewhere.

Thanks to Eli Lifland from AI-2027 for his detailed feedback on this model/post

1. Introduction

The AI-2027 project has been successful in bringing mainstream attention to near-term AI risks, perhaps in part due to the involvement of credible forecasters offering model-based predictions of a rapid AI takeoff starting around 2027 and culminating in super-intelligence within less than a year. The project had two components, a dramatized scenario giving a concrete sense for how AI progress may play out over the coming years [1], and a collection of research posts [2] offering quantitative projections based on Monte Carlo simulation. While their public outreach has emphasized the former narrative scenario, the latter research posts and open-source simulation are in many ways the more

interesting contribution and the primary focus of this post. Note that others [3] have already pushed back on the AI-2027 "timelines" forecast including its partial reliance on hyperbolic projections [4], and the authors have responded and updated their model, so that is not the focus here; in particular, while the timeline-forecast extrapolated when a superhuman coder (SC) would be developed, the current post is focused on offering push-back regarding the AI-2027 "takeoff" forecast [5], which estimated how quickly the process of recursive self-improvement could play out and provided a median estimate of <1 year for this takeoff to spiral from SC to super-intelligent AI (ASI). Also, this recent LessWrong post [21] offered push-back regarding what they described as "intuition-based guesswork" in the AI-2027 takeoff estimates, which parallels some of the arguments in this post; though that post did not offer an empirical alternative to AI-2027's intuition-based approach.

At a high-level, the AI-2027 take-off projections involve making an initial estimate of how long AI progress would take if it were limited to a fixed number of human researchers with fixed-compute and no AI assistance, and then adjusting this estimate downwards with AI speedup multipliers reflecting increasing AI skill over time, as a model of recursive self-improvement. The former human-only estimated relied heavily on intuitive judgement from the authors (see details below), whereas this post relies on an initial pass at a more empirically-grounded model of this step, as a sanity-check as to whether the AI-2027 takeoff speed claims are plausible. See *Tables 2 and 3* for a comparison of the trend-based parameter assumptions of the proposed model versus the more intuition-based parameter assumptions from AI-2027. To be fair, there are pros/cons of a more intuitive versus empirical approach to modeling, and for some model-components there is limited or low-quality empirical data available; but in general, the perspective of this post is that it's useful to make baseline predictions based on the available empirical data, even if imperfect, and even if you subsequently make adjustments based on intuition, and at the very least there are a number of relatively high-quality empirical data sources that are not incorporated into the AI-2027 projections, e.g. existing capability trends, Epoch's algorithmic progress trends, etc.

The empirical-trend based model from this post generally gives substantially longer time estimates than the AI-2027 projections and human-only times that are far outside of the AI-2027 confidence intervals; see *Table 1* for results comparing the AI-2027 approach versus this empirical model, as well as the sensitivity analysis in *Tables 4 and 5* for predictions with alternative parameter estimates.

Approach	Human-only Time Estimate (AMR to SIAR)	Recursive-AI Time Estimate (AMR to SIAR)
AI-2027	24 years	0.6 years
Empirical model	15,630 years	247 years

Table 1: *Time estimates from Automated Median Researcher (AMR) to Super-intelligent Automated Researcher (SIAR), comparing the AI-2027 estimates versus an empirical-trend-based model, showing both time-estimates for fixed-compute human-only-researchers as well as for AI-researchers with recursive self-improvement. Note that these extremely long empirical time estimates should be interpreted more as a reductio of fast-takeoff scenarios, rather than as accurate long-term timelines, since if takeoff really took this long, then AI-2027's fixed-compute assumption would break down, and we would be better off projecting progress based on ordinary compute-driven scaling over time. The core trends assumed in this empirical model are Epoch's algorithmic-progress trend (0.45 OOMs/year, where OOM is order-of-magnitude), and Greenblatt's capability trend estimate per effective-OOM (1.2 SD/OOM), with a compute bottleneck based on AI-2027's survey-based estimate. Note these timelines are for research-skill milestones, and so don't include the additional time to reach the final ASI stage, which AI-2027 assumes requires progress across every skill (not just research skill) and so would involve additional assumptions.*

2. Modeling Assumptions

See *Table 2* and *Table 3* for a comparison of the parameter assumptions and associated justifications for this post's model versus the AI-2027 model respectively. This highlights how much more intuition-based the AI-2027 parameter estimates are. Note that these tables are focused on the model components that are not shared between the two approaches, i.e. the model for human-only time estimates, whereas the AI-multipliers are the main shared components. Also, for AI-2027 only a subset/sample of parameters are shown in *Table 3*, since they rely a large number of parameters spread across the report.

Below is a high-level summary of the primary trends and parameter estimates used in the proposed empirical model:

1. **Current Research Progress-Trend:** Relies on estimates of the current rate of algorithmic-progress in ML research, which appears to be roughly 0.45 OOMs/year based on past trends, where OOMs are orders-of-magnitude of effective training-compute.
2. **Capability-Trends per Effective-Compute:** As algorithmic progress drives increases in effective-compute (prev), we can estimate the corresponding improvement in model capability, with the baseline estimate from Greenblatt of 1.2 SD/effective-OOM. Note the units are in standard deviations (SDs) relative to the human skill distribution, so this trend can only be estimated with benchmarks for which we have human-distribution data. Using SD-based trends rather than benchmark-accuracy trends is useful in this case for various reasons, including that the AI-2027 progress milestones are defined relative to the human skill distribution.

3. **Compute Bottleneck Slow-downs:** Incorporates estimates of how much the AI-2027 fixed-compute assumption will slow progress, since existing algorithmic-progress trends (from bullet 1) rely on a background of exponentially improving hardware compute, e.g. for running more extensive experiments over time. The baseline model follows the informal AI-2027 researcher-survey and assumes that each 10x reduction in compute slows researcher progress to 40%, but other variants are also considered. This bottleneck estimate also relies on Epoch's estimate that the status quo hardware increase is approximately 0.6 OOMs/year. Note that the reason AI-2027 assumes fixed compute is that they expect the recursive take-off to be sufficiently fast that there won't be time for hardware compute to meaningfully increase. In principle there will also be slowdowns/bottlenecks from human researchers being held fixed during the fast-takeoff (in addition to compute); the baseline estimate doesn't account for this due to lack of solid data, but see the sensitivity analysis for a first-pass estimate of this effect, which could potentially be quite large (esp for the human-only time estimates).
4. **Shared Assumptions with AI-2027**: In general, other assumptions from AI-2027 are held as fixed as possible, to narrowly test the impact of this alternative model for estimating human-only progress under-fixed compute. In particular, the multipliers for determining the research speedup from each stage of AI improvement are used unchanged from AI-2027, e.g. to determine the AI-estimate column in *Table 1*.

2.1 Empirical Model

Parameter description	Parameter estimate	Justification
Algorithmic ML-progress / year	0.45 effective-OOMs / year	Empirical trend (Epoch.ai)
Capability-progress / effective-OOM	1.2 SDs / OOM	Empirical trend (Greenblatt)
Compute-bottleneck	0.4 elasticity (40% slowdown per reduced-compute-OOM)	AI-2027 survey-based
Status quo hardware trend	0.6 OOMs / year	Empirical trend (Epoch.ai)

Table 2: *This table shows the modeling assumptions, median parameter estimates and justifications for the empirical-model from this post; note that this empirical model is an alternative to the human-only estimation-phase from AI-2027's projection, and it does not provide separate estimates of the AI-speedup multipliers, which are currently reused from AI-2027. Also, see the sensitivity analysis below*

for results with alternative parameter estimates, including model runs with an additional human-researcher bottleneck.

2.2 AI-2027 Model

Parameter description	Parameter Estimate	Justification
Research-time to median-level	10 years	Set "for simplicity" (AI-2027)
SC -> SAR time, if low compute	1-4 years	Based on "our guess" (AI-2027)
SC -> SAR time, if high compute	2-15 years	Based on "our guess" (AI-2027)
SC -> SAR, if scientific problem	2-15 years	Based on "we guess", plus AlphaGo analogy (AI-2027)
SC already SAR	15% chance	Intuition from Moravec's paradox and general uncertainty (AI-2027)
etc ...	-	-

Table 3: *Modeling assumptions and parameter estimates from AI-2027 takeoff forecast for the human-only time estimates. This is not an exhaustive list, since there are quite a lot of parameters scattered through-out the report, but this sampling should give a sense for the intuitive-basis for these estimates. It's difficult to summarize the full justification given for each of these, but typically there are qualitative arguments or analogies offered and then a range presented based on "our guess", etc. For comparison, see Table 2 above, for the parameter estimates and justifications for the alternative empirical model used in this post. Note this table does not include their additional assumptions related to the AI speedup multipliers, since those are shared/reused by the empirical model from this post.*

3. Summary of Arguments:

This post makes the following claims about the AI-2027 take-off forecast:

1. **Intuition-based human-only time estimates:** This is not inherently a problem and is something that the authors have been open about, but it's worth noting that the forecasts depend to a significant degree on hyper-parameters which are minimally justified beyond the authors' intuition

(see *Table 3*); in the context of the takeoff forecast, this intuition-based approach is particularly evident in their estimates for how long human-only researchers will reach various AI milestones. And while it's true, per Tetlock's research [13], that intuitive probability judgments from skilled forecasters can be quite useful for predicting ordinary near term events (e.g. in geopolitics), there is much less evidence that we can trust our intuitions over predictions that are far outside of ordinary experience (e.g. AI rapid takeoff) or where we have minimal guidance from relevant base-rates.

2. **Extrapolation from limited data:*** Once they have the intuition-based time estimates above, they are used to calibrate a (Davidson-style) power-law and then extrapolate through decades of human research time, which effectively amplifies any error in the initial intuitive estimates. And even if the intuitive estimates were accurate, they are still using estimates from just 2 data points to calibrate the model, i.e. calibrating based on AMR and SAR, and then extrapolating out to super intelligence.
3. **Empirical alternative:** While intuition-based estimates can be a reasonable approach, especially when there is no alternative, in this case there is relevant empirical data which can be uses to provide estimates for human-only research times (see *Table 2*), and these appear to lead to much longer time estimates for AI progress, which can used to sanity check the AI-2027 intuitive-timeline judgments. In particular, *Table 1* shows an increase from AI-2027's 24 year estimate to the empirical-model's 15.6k year estimate, for the AMR-to-SIAR gap; see bullet 4 below for the empirical time estimates incorporating AI self-improvement. **Note that these estimates are far outside of AI-2027's confidence intervals**, e.g. they estimate the human-only-researcher jump from SAR to SIAR as 19 years with a confidence interval of 2.3 to 380 years (80% CI), whereas the empirical model estimates this same gap as 15.6k. It's important to note that these takeoff estimates assume fixed-compute (per AI-2027), but with time estimates spanning decades as predicted by this model, AI-2027's fixed compute assumption would break down, and it's plausible that the gradual status-quo compute trend combined with ordinary model-scaling could lead to these AI milestones faster than the recursive-takeoff estimates; so these fairly extreme estimates should be seen more as a (potential) reductio of AI-2027-style fast-takeoff scenarios, rather than projections of likely progress rates, and if fast-takeoff fails to materialize due to these considerations, an accurate timeline projection unfolding over many years would need to account for compute increases and the resulting scaling trends.
4. **Potential explanation for discrepancy:** The time estimates from AI-2027 are quite close to what the empirical model predicts when you plug-in zero human-research slow-down from reduced compute, i.e. 24 years vs 17 years respectively (see *Table 5*); for comparison, the model gives a 15,630 years human-only-estimate when constrained by compute. I am assuming that the AI-2027 authors would reject the premise that there is zero compute-bottleneck, since elsewhere in the report they do account for this constraint; but it raises the question of whether their intuitive assessments of human-only times were unintentionally ignoring or at least under-estimating the

compute bottleneck. That said, given the intuitive/implicit nature of their model it is difficult to really know why the estimates are lower, e.g. another possibility is that they just have a much higher estimate of the likely progress-rate in capabilities; see *Table 4* for the impact of alternative capability progress-rates.

5. **AI takeoff times:** Based on these updated human-only time estimates, we can reuse AI-2027's AI-speedup-multipliers to estimate how much slower the AI recursive takeoff will be with these slwer human-only times. Generally the differences between the AI-2027 estimates versus the empirical model are reduced after the AI speedup, but are still substantial, e.g. *Table 1* shows 247 years for the baseline empirical model versus 0.6 years for AI-2027 (for improving from a median-level researcher to a super-intelligent AI researcher). Also, the model is quite sensitive to magnitude of the compute bottleneck, e.g. if rather than using AI-2027's survey-based bottleneck we use a "worst-case" bitter-lesson-based bottleneck in which research progress is driven entirely by compute (e.g. based on the number of experiments that can be run), then this increases the AI takeoff time from 247 years to 83 million years. Also, adding in a first-pass attempt at accounting for personnel-bottlenecks (in addition to compute) increases the AI timeline from 247 years to 1.1 million years (though this personnel-bottleneck is quite rough; see *Table 6*)

4. Sensitivity Analysis

This section provides predictions for the empirical-model for alternative hyper-parameter values, to assess the sensitivity of the results to these parameters.

4.1 Varying Capability per Effective OOM

Approach	Trend Assumptions	Capability Trend (SDs / OOM)	Human-only Time Estimate (AMR to SIAR)	Recursive-AI Time Estimate (AMR to SIAR)
AI-2027	unknown/implicit	unknown/implicit	24 years	0.6 years
Empirical model	Pre-reasoning-era trend	1.03 SD / OOM	46,930 years	737 years
Empirical model	Greenblatt's estimate (Redwood Research)	1.2 SD / OOM	15,630 years	247 years

Approach	Trend Assumptions	Capability Trend (SDs / OOM)	Human-only Time Estimate (AMR to SIAR)	Recursive-AI Time Estimate (AMR to SIAR)
Empirical model	Average pre- and post- reasoning-era	2.1 SD / OOM	332 years	3.5 years
Empirical model	Post-reasoning-era trend	3.24 SD / OOM	36 years	0.8 years

Table 4: *Time estimates for varying estimates regarding the map from effective-compute to capability-progress (in SDs / OOM). Results are shown based on Greenblatt's original SD/OOM estimate, as well as empirical-averages for OpenAI models in the pre-reasoning-era (i.e. gpt-class models), post-reasoning-era (e.g. o-series models), as well as the average across those regimes. Note that we don't have data on this trend from any agentic or research oriented benchmarks, so these are proxies, which average over significant variance across benchmarks. Also, note that there appears to have been a significant speedup in these trends with the shift to reasoning models, though it is uncertain how much of this is a one-time gain from low-hanging fruit, versus a long-term shift in the rate of progress; there is also uncertainty as to how broadly applicable these speedups are beyond the non-agentic reasoning-intensive tasks where we have SD/OOM trend data.*

4.2 Varying Compute bottlenecks

Approach	Fixed-compute Impact - Assumptions	Compute-elasticity	Human-only Time Estimate (AMR to SIAR)	Recursive-AI Time Estimate (AMR to SIAR)
AI-2027	unknown/implicit	unknown/implicit	24 years	0.6 years
Empirical model	No Compute Bottleneck (slowdown 0%)	0	17 years	0.73 years
Empirical model	AI-2027 researcher-survey-based (slowdown 40%)	0.4	15,630 years	247 years

Approach	Fixed-compute Impact - Assumptions	Compute-elasticity	Human-only Time Estimate (AMR to SIAR)	Recursive-AI Time Estimate (AMR to SIAR)
Empirical model	Min info prior - median (slowdown 32%)	0.5	113,700 years	1,783 years
Empirical model	Worst-case bitter-lesson (slowdown 10%)	1	5.3 billion years	83.1 million years

Table 5: *Time-estimates from Automated Median Researcher (AMR) to Super-intelligent Automated Researcher (SIAR), with varying assumptions about how compute-bottlenecks will slow research progress; otherwise, the parameters are the same as the baseline settings from Table 1. Note that the AI-2027 projections are quite close to the empirical model estimates when the compute (and worker) bottlenecks are both set to zero. The minimum-information-prior row just estimates the compute-elasticity with a uniform prior from 0 to 1, giving a median elasticity of 0.5.*

4.3 Incorporating Personnel Bottlenecks

Approach	Human-only Time Estimate (AMR to SIAR)	Recursive-AI Time Estimate (AMR to SIAR)
AI-2027	24 years	0.6 years
Empirical model	70.5 million years	1.1 million years

Table 6: *Alternative time-line estimates that take into account personnel-bottlenecks, i.e. caused by AI-2027's assumption of fixed researcher counts during the fast-takeoff. This was not included in the other estimates above due to relatively low-quality data for estimation, so prior tables/models only take into account compute-related bottlenecks. However, the estimates from this table suggest that personnel-bottleneck effects could actually be quite large, and so the baseline projections may be too low. In the absence of a direct empirical (or survey-based) estimate, the elasticity for the personnel-bottleneck is inferred from the compute elasticity, via a Cobb-Douglas style constant-returns assumption.*

5. AI-2027 Approach

5.1 Estimating Hyper-Parameters

AI-2027 based their recursive self-improvement estimate on a Drake-equation-style decomposition of the model into various hyper-parameters, which are then estimated through a combination of intuition, empirical evidence, and expert surveys. Specifically, in order to estimate the self-improvement take-off speed, they first develop estimates for how long it would take human researchers to implement various milestones of AI progress, and then they apply multipliers at each stage based on how much the current AI would be able to speedup this progress. Because the AI takeoff is estimated to take only around a year, they assume that available compute will be roughly constant during this time period, since a rapid software-only takeoff leaves little time for substantial hardware improvements.

While many of the hyper-parameters in the AI-2027 model are based on historical trends, expert surveys or other forms empirical evidence, the timelines for human-only research progress are notable in the extent to which they rely on the authors' intuitions to determine the underlying parameter values. For instance, the time estimate to reach the level of the median human researcher is set to 10 years "for simplicity", and while the timeline from superhuman coder to super-intelligent AI (ASI) is broken down into various cases, within each case the estimates are largely intuitive guesses e.g. in the sub-case where SAR isn't that compute intensive they say: "With human engineers doing the labor, our guess is that it would take 1-4 years."

Once they estimate short-term milestone timelines via intuition (e.g. cumulative effort to AMR and from AMR->SAR), then these estimates are used to calibrate a Davidson-style [24] power-law mapping human research effort to progress, which is then used to project timelines out to the more advanced AI milestones. So even if the intuitive estimates are trusted, the power law is being extrapolated based on quite a small amount of short-term data, i.e. effectively 2 data points for AMR and SAR, which are then extrapolated to SIAR and ASI.

5.2 Stages of Recursive-Self Improvement

The AI-2027 project divides the timeline of AI-improvement into a series of milestones, shown in *Table 7*. First, they model the improvement from present-day LLMs towards models that are as good as the best AGI-company software engineers, but faster and cheaper, i.e. superhuman coders (SC). They estimate the timeline for SC via extrapolation of existing progress trends, with a median estimate of 2027. However, once the SC arrives, then they argue that this will substantially speed up AI progress by initiating a process of recursive self-improvement in which AIs speedup and then eventually takeover AI research progress. They project that this process will be quite rapid (on the order of a year) and will traverse the remaining rows in *Table 7*, with the 2nd column showing the overall

research speedup from each stage. Note that the speed up isn't fixed for a given model, but is interpolated as the model moves between stages, so moving from a SC to SAR isn't flat at 5x with a jump to 25x at SAR; instead their interpolation, where progress is proportional to cumulative research effort, implies that the average speedup between these two stages is about 10x.

AI-2027 AI Stage	Speedup Multiplier	Research-taste (vs median)
Superhuman Coder (SC)	5x	-
Superhuman Researcher (SAR)	25x	3.25x
Super-intelligent AI Researcher (SIAR)	250x	10×3.25x
Super-intelligent AI (ASI)	2000x	-

Table 7: *Shows the stages of AI progress used by AI-2027 project along with the AI speed-up multipliers for each stage compared to human-only researchers; the 3rd column also estimates how much better each model's research-taste is compared to a median researcher baseline. The overall speedups over humans are generally much larger than the speedups from research-taste, since many other factors contribute to the former including increases in raw processing-speed and software-engineering skill.*

6. Empirical-Model Approach

This section provides technical details regarding the empirical model developed in this post.

6.1 Progress Stages Empirical Model

While AI-2027 models AI progress from median-researcher-level to super-intelligent AI, the empirical-model from this post focuses on projecting the improvement in research-taste from median-level to super-intelligent AI researcher (SIAR), rather than continuing to the final ASI stage, since additional assumptions would be needed about how fast other non-research-based skill areas are likely to progress; however, modeling the full gap to ASI would of course extend timelines even further.

In terms of concrete numbers, the AI-2027 project estimates the research-taste gap from AMR-to-SIAR as $3.25 * 10 = 32.5$x, i.e. all-else-equal, the improvement in AI research-taste between these stages can be expected to increase research speed by 32.5x. So taking this gap as given, the focus of the empirical-model is to estimate how much time it would take for AI-progress to traverse this 32.5x skill gap.

6.2 LLM Algorithmic Progress Trend

In his Situational Awareness report from 2024 [8], Aschenbrenner argued that in the context of recent scaling-based progress people often under-rate the contribution from algorithmic progress:

> "While massive investments in compute get all the attention, algorithmic progress is probably a similarly important driver of progress (and has been dramatically underrated)."

Aschenbrenner estimates the trend over the last 10+ years at 0.5 OOMs/year improvement in hardware-compute for frontier models, but also a similar 0.5 OOMs/year of algorithmic progress. This algorithmic progress metric is defined such that with a 0.5 OOM increase in effective compute from better and more efficient algorithms, you can train a model that is as good as what you would have gotten from an actual 0.5 OOM increase in hardware. This post relies on these two trends and follows Aschenbrenner in estimating them both as 0.5 OOMs/year. Aschenbrenner's source for this algorithmic-progress estimate was an Epoch.ai/MIT paper, which estimated the relative contributions of scale versus algorithmic progress to language model training, and found approximately .45 OOM/year of algorithmic progress over an 11 year period, which Aschenbrenner rounds to 0.5 OOM/year. Epoch estimates somewhat higher compute increases than Aschenbrenner's 0.5 OOM estimate, i.e. their estimate is about 0.6 OOMs/year. If the model in this post were updated to use 0.6 OOMs/year for hardware that would extend the takeoff timeline even further due to the larger drop in compute relative to status quo during the fast takeoff. Epoch also estimated algorithmic progress for vision models and reached a similar estimate of algorithmic progress per year [10]. These estimates are focused on estimating the rate of training progress, but inference progress has also been substantial [8].

One potential source of confusion with these algorithmic progress estimates is that even if the proximate cause of the improvement is algorithmic, the root cause could still be hardware-compute progress. The issue is that hardware improvements can both directly improve results by allowing larger models to be trained with more data, but they also indirectly allow exponentially more experiments to be performed to test new algorithmic ideas. In other words, it's at least theoretical possible that almost all of the LLM progress over the last 10 years was driven by compute, with good ideas being fairly straightforward to come up with (at least for top researchers) and compute for experiments being the primary bottleneck. Also, even setting aside experiments for testing new ideas, at least some portion of what gets labeled algorithmic progress is just an inevitable consequence of more compute, e.g. larger hyperparameter runs, distillation of larger models, etc, where for example a distillation appears to be more compute-efficient at inference, but this efficiency depends on having large compute for training the larger teacher model. See the next subsection for a discussion of how this compute-bottleneck effect was estimated for the model from this post.

Another potential concern with using these algorithmic progress trends as a proxy for AI research progress is that there is some question as to whether this does a reasonable job of capturing all of the different forms of AI research progress. That said, given that this metric captures the extent to which improved algorithms allow better models to be trained at a given compute, this does seem to provide a fairly broad measure of AI progress. It could be worth assessing the rates of progress in other areas of AI research (e.g. quantifying improvements in inference efficiency), but this Epoch.ai algorithmic-progress measure is likely fairly solid baseline estimate.

6.3 Capability Trend per OOM

Once we have the projected trend in algorithmic progress (measured in effective OOMs of compute per year), the next question is how much research skill/capability increases with each effective compute-OOM. Ryan Greenblatt, from Redwood Research, recently estimated this based on past trends [11], including both gpt-era models as well as reasoning models, and estimated this progress rate at 1.2 standard deviations (SD) per effective-OOM, with the standard deviations measured relative to the human skill distribution (in the context of Gaussian distributed skill measures). In the original version of this post I had used the more widely known LLM scaling laws for this purpose, i.e. the power-laws relating training compute to skill on downstream benchmarks. But based on feedback on an earlier draft from Eli Lifland (from AI-2027), I shifted the model to use these SD/OOM trends instead; a key advantage of this progress measure is that it's calibrated to the human skill distribution, which gives a more principled way to predict the AI-2027 milestones, which are themselves defined relative to the human distribution (e.g. based on gap between median and top human researchers, etc); also, these standard-deviation-based trends can be applied to predict progress-speedup multipliers, as long as we can estimate the human distribution of these multipliers, whereas it's less clear how to map benchmark-skill progress (from conventional power laws) to progress multipliers. On the other hand, a downside of the SD based approach is that we just don't have human-distribution data for most benchmarks, so we can't estimate this trend as accurately as the benchmark-skill power laws. So for purposes of ordinary timeline estimates (e.g. until the key benchmarks are saturated), as opposed to this recursive-takeoff estimate, we would likely be better of just using the conventional benchmark-based power-laws rather than these more uncertain SD/OOM trends.

Once we have an estimate of the capability increase per effective OOM (in SDs/OOM), we can estimate the the number of effective OOMs needed to traverse the AI-2027 capability milestones, which are defined relative to the skill-gap between median and top researchers (e.g. SAR to SIAR is defined to be 2x this median-to-top skill gap, etc); see the formula-derivation section below for details on this calculation.

One challenge with this approach, is that SD/OOM progress varies quite a bit depending on which skill we are tracking, and currently we don't have a good benchmark to track research skill; so the model

just assumes that the average progress rate across existing benchmarks is a reasonable proxy for the research-skill progress rate. But developing a benchmark for research skill/taste, and collecting data on the human skill distribution on that benchmark, would be a fairly high priority approach to improving on these projections.

Also, since AI-2027 has survey data on the speedup-multipliers for top vs median frontier researchers, we can also use these SD/OOM estimates to estimate the speedup-multiplier increases per effective OOM; in particular, their survey finds that the top frontier-lab researchers have sufficiently better research taste than median researchers that they can make 3.25x faster progress; so if we assume top researchers are about 3 std above the median (see discussion below on this assumption), then that means 3.25x for that first 3 std gap, which means (0.51 speedup-OOM / 3 std) * (1.2 std / training-OOM) = 0.2 speedup-OOMs / training-OOM = 1.6x progress-speedup per training-OOM(note this is following AI-2027's log-normal skill distribution). However, interestingly when generating the time estimates with the empirical model, we don't need to rely on this survey based multiplier (of 3.25x for top researcher taste), since that value cancels out of the human-only time estimate calculation, i.e. the larger that multiplier, the more quickly progress increases, but also the larger gap you have to traverse; again, see the formula-derivation section below, to see that it has no dependence on how fast the top researcher is. Though the AI speedup multipliers, which this post just takes from AI-2027, do depend on the absolute estimates of these speedup multipliers (from AI-2027's survey-based estimates).

6.3.1 Alternative capability trends

The baseline projections in this post rely on Greenblatt's estimate of SDs/OOM, but for this section I also performed some independent assessments of this trend in order to get a clearer sense for how steady/consistent it has been, both over time and across benchmarks; I included these alternative trend-estimates in the sensitivity analyses (above). The main reason I used Greenblatt's estimate for the baseline projection in this post was just to avoid introducing researcher-degrees of freedom (DOFs), i.e. by relying on 3rd-party empirical estimates where possible. That said, based on my estimates below (from CodeForces and IQ data), I suspect Greenblatt's estimates are a bit low, and that he may not be sufficiently updating based on recent progress-speedups from reasoning models (e.g. o-series). See *Table 8* for a comparison of various SD/OOM trend estimates, and see the sub-sections below for details on how these were estimated.

Model era	benchmarks	SD / OOM
pre-reasoning-era	average over 7 benchmarks (SAT, bar-exam, CodeForces, etc)	1.03 SD / OOM

Model era	benchmarks	SD / OOM
post-reasoning-era	average over 3 benchmarks (CodeForces, IQ-text-offline, IQ-vision-offline)	3.24 SD / OOM
average pre- and post- reasoning	see previous two rows	2.14 SD / OOM
pre- and post- reasoning (3rd-party Estimate - via Greenblatt)	variety of benchmarks and scaling arguments	1.2 SD / OOM

Table 8: *Capability trends with effective OOMs for pre- and post- reasoning model era frontier (OpenAI) models and the benchmarks that were used. Unfortunately, there is somewhat limited benchmark data that includes the human-percentiles needed to estimate these z-score based trends. For the gpt-era, the estimates are based on gpt-3.5 to gpt-4 z-score improvements, with compute increases taken from Epoch. For o-series era, the trend is based on three benchmark tests, but compute estimates are not available and so are just inferred from the background frontier-compute trend (again per Epoch). Simple averaging is used to combine across benchmarks, rather than weighting by sample sizes, as well as for averaging pre- and post- reasoning trends.*

6.3.1.1 Competitive Coding trends (CodeForces)

See *Figure 1* for the trend in competitive-coding (CodeForces) skill over time for OpenAI models and notice that there does seem to be a clear speedup in the trend with arrival of the o-series reasoning models and little sign of any plateau so far (though OpenAI doesn't seem to have released a score for gpt-5 on this benchmark). Note that SD/OOM estimates will vary based on the underlying distribution, so the z-scores used in the plot are Gaussian, i.e. the actual percentiles are just converted to Gaussian z-scores. For simplicity, I just used the human percentile data from a single year, i.e. 2024 here [25].

One challenge with estimating this trend is that OpenAI hasn't released training-compute estimates for recent models. But we do have trend data on how much training compute for LLMs has been increasing over time, with hardware compute increasing at about 0.6 OOMs/year [18] and algorithmic-efficiency gains giving another (roughly) 0.45 OOMs/year [9], which gives a combined effective-increase of about 1.05 training-OOMs/year. So we can get a rough estimate of the SD/effective-OOM trend, by dividing the SDs/years (e.g. from *Figure 1*) by 1.05 training-OOMs/year. This effectively assumes that frontier labs are continuing to makes use of the available compute trend, even if the relative split between pre- and post-training is shifting.

A potential concern with these assumptions is that post-training is fairly data-efficient and may not be using the same scale of compute as the pre-training trend, and also the latest models (e.g. gpt-5)

seem to be relatively fast, which some have argued suggests that pre-training and model-size scaling has slowed (e.g. gpt-5 is fast compared to the slow/expensive gpt-4.5 model); however, it's quite possible that the extensive pre-training compute that went into gpt-4.5-scale models indirectly benefited their other models like gpt-5 via techniques like distillation and synthetic-data generation (or algorithmic inference-efficiency gains), so I don't think we can assume from gpt-5's (or o-series') fast response times that recent capability gains weren't enhanced from model-size/pre-training scaling in addition to post-training. That said, if it turns out that recent compute increases have been slower than the 1.05 OOM/year trend, then the SD/OOM trends could be larger than these estimates. On the other, hand if the fast recent trends reflect initial low-hanging fruit from incorporating reasoning, then these trends could be an over-estimate as future trends start to plateau.

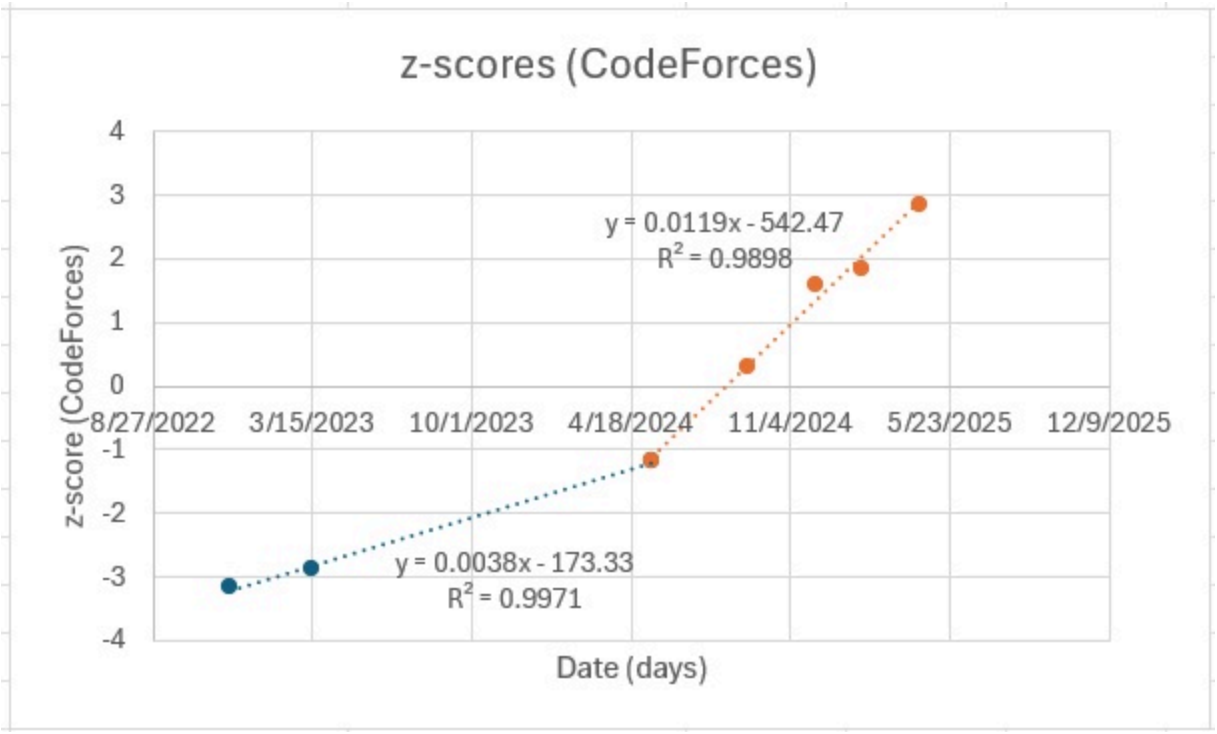

Figure 1: *Plot showing the pre and post reasoning-era OpenAI competition coding (CodeForces) trends; gpt-era is blue and o-series-era is orange. The z-scores estimates allow us to estimate SD/year progress trends, and if we assume the recent historical trend in frontier (effective) compute, then we can convert that to an estimated SD/OOM trend. For this benchmark, which has been improving quite rapidly, the gpt-era trend from this slope is 1.32 SD/OOM, and the o-series-era trend is 4.14 SD/OOM.*

6.3.1.2 IQ trends: reasoning-era

In addition to the CodeForces results that OpenAI has released, I also estimated SD/OOM from the IQ z-score data estimated by Maxim Lott's LLM IQ-tracking site [26]. As with CodeForces, the main advantage of this benchmark is that it provides human-distribution comparison data which can be used to estimate SD/OOM.

This IQ-tracking site estimates LLM IQs based on a public Mensa-Norway IQ test, but also from their own offline test, which they created to avoid data leakage concerns. These offline IQ-test results are incorporated into the reasoning-era SD/OOM estimates from *Table 8*. Also, this site assesses IQ based on both text-only assessment but also vision-based assessments for multi-modal models, and both of these are included in the reasoning-era trend.

Note that these reasoning-era IQ trend estimates had only a small sample-count from the frontier OpenAI models used in this trend, i.e. 3 data-points for the private text-based IQ result and 4 data-points for the private vision-based result. Also, the average trends shown in *Table 8* are just simple averages, they are not weighted by the sample counts for different benchmarks, since these benchmarks are measuring different skills and likely have distinct progress-trends, so we don't necessarily want to bias our progress-rate estimates towards whichever benchmark happens to have the most data.

Also, these SD/OOM estimates only accounted for the top-performing model at any given time, so if a weaker model (e.g. a mini-version) is released that shouldn't slow the frontier progress-rate estimate; recently the top performing models are often the "pro" tier versions. However, one complication that seemed to particularly show up in the IQ data is that the frontier model varied somewhat across the different tests, so a reasonable case could be made for picking a single model at each time as the "frontier" model and using that consistently across benchmarks (rather than whichever does best for each benchmark), but it doesn't look like this would change the results drastically. Also, note that the data on this IQ site appears to vary somewhat from day-to-day, but the scores used for these estimates were from the version on Aug 22, 2025 (see web archive for snapshots).

6.3.1.3 Pre-reasoning-era trends

For the pre-reasoning era trends, the best data point seems to be the gpt-3.5 to gpt-4 comparison, for which OpenAI provided a variety of standardized test results with human distribution data, and this was also the main gpt-era empirical data point used by Greenblatt. See *Table 8* for the SD/OOM estimate from this data, which was based on 7 benchmarks including the SAT (college admissions), the bar-exam (law), CodeForces (competition coding), etc. Unlike in the CodeForces trends above, which relied on baseline trends for estimating compute, these older gpt-era estimates used Epoch data [27] on training compute across models. Unfortunately, the actual training compute is not known, so these Epoch estimates are somewhat speculative. Interestingly, this average trend over 7 benchmarks (for 3.5->4) and the CodeForces gpt-era trend (above) both give similar estimates of roughly 1 SD/OOM; on the other hand, the CodeForces estimate (from 3.5->4) gives a much smaller SD/OOM estimate (around 0.3 SD/OOM) which indicates how sensitive these estimates can be to assumptions.

6.3.2 Comparison with Greenblatt's trends

Greenblatt's SD/OOM estimate does already take into account the CodeForces reasoning-model trends (though not the IQ benchmarks), but surprisingly his estimate for CodeForces is actually similar to his pre-reasoning trend, i.e. both are roughly around 1 SD/OOM, rather than reflecting the speedup from *Figure 1*. The explanation for this discrepancy is just that for reasoning models, Greenblatt only counts post-training compute (excluding pre-training). This means that a small percentage increase in total training compute, is treated as a large percentage increase in post-training compute, which makes the SD/training-OOM trend look slower than if he had consistently used the full training compute in across his estimates. He acknowledges this but argues that recent progress is mostly driven by RL post-training compute anyway and that the current trend could even be an over-estimate due to reflecting initial low-hanging fruit progress from introducing RL-based reasoning models, e.g. he says [11]:

> "...I expect it's only a mild underestimate as the returns probably mostly come from scaling up RL for that task... Not super sure about this, and naively, I think there could be in the argument in the opposite direction (that scaling up RL compute was relatively low hanging and we should typically expect lower returns). But, overall, my sense is that RL is driving most of the improvement on this sort of competitive programming task such that the just looking at RL compute is actually pretty reasonable."

In general, I do think these are reasonable points and there is a risk that the recent reasoning trends may not be sustained; on the other hand looking at the trend in *Figure 1*, it does seem to be remarkably steady/linear so far. Also, I'm somewhat skeptical that just swapping in post-training compute for training compute is a principled way of handling this concern. An alternative is to estimate the long-term gpt-series trend as well as the faster recent trend and then use an average over these two estimates, in order to reflect uncertainty regarding the extent to which the long-term trend will reflect recent speedups; this intermediate option is included in the sensitivity analysis in *Table 4*.

Another potential concern is that we have limited data with human-distribution information (for extracting SD/OOM estimates), and the data I'm aware of is from relatively reasoning-oriented but non-agentic tasks like competition programming and IQ tests, and it's unclear whether research-skill will benefit as much from reasoning tokens, e.g. some aspects of this skill (like research "taste") are relatively system 1 intuition-based and so could follow something more like the slower pre-reasoning trend. Also it does seem that non-agentic coding (e.g. CodeForces) has been improving much faster than agentic coding (e.g. SWE-bench), where the former is arguably already at top human human level. So in so far as this post's projections are mostly concerned with research-skill trends, the considerations in this paragraph give some additional support for the view that the Greenblatt-style

trends could be more realistic (since they are only a bit faster than the gpt-era trends, rather than reflecting the full recent speedup).

Note that Greenblatt also fudged his estimates upwards based on other less empirical arguments, including brain-scaling arguments. I am fairly skeptical of this move, partly because there is just too much uncertainty about the factors that drove brain quality improvements during evolution beyond just scale (e.g. better "hyperparams", etc), but Greenblatt just used these intuitive considerations for a fairly small upward-adjustment (from roughly 1 to 1.2 SD/OOM), so they don't make that much difference, e.g. compared to the relatively large (apparent) difference between the gpt-era versus reasoning-era trends.

6.4 Adjusting for the Compute-Bottleneck

For the takeoff estimate, AI-2027 assumes that hardware-compute will be fixed, since they project the software takeoff to be so fast as to not allow much time for hardware improvement. However, if we want to project the past algorithmic progress into the future, with this fixed compute assumption, we need some way to estimate how much of the past algorithmic progress has been driven by compute-independent innovation versus how much has been a more direct consequence of rising compute (e.g. due to being able to run more experiments to test new ideas), where the latter component will decrease in a fixed-compute setting. To estimate this, the baseline empirical model relies on the informal AI-2027 researcher survey results, in which AI safety researchers (n=6) in a Slack poll estimated that a 10x drop in compute would slow progress to 60%; AI-2027 then dropped this further to 40% to account for the concern that their work may be "less compute-intensive than frontier AI researchers"; this 40% estimate is the value used in the baseline empirical model, though other reductions are also tested.

Interestingly, AI-2027 did not choose to use this 40% slowdown estimate for generating their human-only takeoff time estimates; rather, they only relied on this slowdown estimate in a separate part of their model, e.g. where they estimated the diminishing returns to faster processing-speeds due to reduced compute available per second of thinking.

One possible concern with the 40% estimate is that researchers may be biased against strong versions of the bitter lesson, e.g. they may want to believe that their innovations are essential and not just components in an inexorable process of compute and scale driven progress; also, ai-safety researchers may have an additional source of bias, since compute-independent algorithmic progress is more likely to lead to the kinds of risk scenarios they are concerned about, e.g. rapid recursive self-improvement, as opposed to more gradual hardware/compute driven improvements. Based on these concerns, the current post also runs scenarios with more aggressive compute bottlenecks, including a drop to 20% per OOM, and also a worst-case bitter-lesson approach in which algorithmic progress

relies entirely on the amount of compute available for experiments. The overall model is fairly sensitive to this parameter, and even the intermediate 20% option leads to a drastic increase in timelines (see *Table 2*).

Also, another potential issue in applying this adjustment is that the researchers in the survey were asked about the impact of a single 10x drop, whereas the model assumes that this same drop (to 40%) can be iteratively applied across additional OOMs of reduced compute. In part this is just following AI-2027's assumptions, since they also treat this as a factor that can be applied from different compute starting-points, e.g. when they apply it to determine the diminishing returns to increased processing speed. But this is also a fairly reasonable baseline assumption, given that each 10x drop in compute reduces the quantity of experiments that can be performed by roughly 10x, so it's plausible that this would lead to a broadly similar percentage drop in progress for each iterative drop in compute. And for the extreme bitter-lesson model where it's assumed that experiment count/size fully determines algorithmic progress, it follows that each iterative 10x compute drop leads to a matching 10x drop in experiments, and therefore a 10x drop in progress.

6.5 Human-Time Formula Derivation

In order to estimate a formula for human-research progress with fixed compute, we start with current human-researcher algorithmic ML-progress trend, which has been providing roughly 0.45 OOMs of algorithmic progress per year; however, this progress has been facilitated by a parallel 0.6 OOM $/year$ increase in hardware compute, which has made possible exponential increases in research-experiment quantity over time. But because we are modeling rapid self-improvement, AI-2027 assumes that hardware compute will not noticeably increase over this process, so we need to adjust this 0.45 OOM/year trend-line downwards based on this "lost" hardware compute. Based on an AI-2027 survey of researchers, and adjusted to account for frontier-researchers being more compute-dependent, AI-2027 estimates that a 10x reduction in compute reduces progress to 40% baseline (see discussion above).

So suppose the status-quo .45 OOM/year algorithmic progress trend would take N years to improve AI ability by a algorithmic OOMs, then $N = a/0.45$. And under the fixed-compute constraint, the last year of status quo effort would now have $0.6(N-1)$ fewer OOMs of hardware compute by the start of the Nth year, given the hardware baseline of 0.6 OOMs$/year$ progress. This should then slow down progress in the last year to: $0.4^{0.6(N-1)}$, based on the survey-based reduction to 0.4x per OOM. So if progress/year is slowed to this amount, then the years required for a given amount of progress is multiplied by the inverse, so under fixed-compute, the time for that last year of status quo progress is actually: $0.4^{-0.6(N-1)}$. So if we want the total years to get a OOMs of progress, we sum this over each status quo year, i.e.:

$$\text{Total Human Years} = \sum_{y=0}^{N-1} 0.4^{-0.6y}$$

where

$$N = \text{round}(a/0.45)$$

Note this is treating the years as discrete, but we would get even longer time-estimates if we continuously compounded the hardware slow-down throughout each year, which also allow us to avoid rounding above; updating this would likely be an improvement, but it doesn't seem to make a substantial difference, and arguably compute increases are in fact somewhat discrete, as new datacenters are completed.

Also, the baseline time estimates as reflected in the above formula only account for the compute bottleneck, but during the fast takeoff, AI-2027 also assumes that frontier researcher/engineer counts will be roughly fixed as well. See below for a first-pass analysis suggesting that this could give a slowdown of (roughly) 0.16 per OOM, as opposed to the 0.4 estimate in the formula above (which just accounts for the compute bottleneck). While the baseline time estimates (e.g. in *Table 1*) just use the compute bottleneck, see *Table 6* for alternate timeline results with this larger combined slowdown/bottleneck; also, to match the year estimates in this table, use the unrounded value of 0.15874 in place of the rounded 0.16 value above.

Next, given an estimated trend between effective-compute and researcher taste/skill measured in standard deviations (SDs) per effective-OOM (e.g. 1.2 SD/OOM per Greenblatt), we can give a complete formula for estimating human-only timelines for achieving particular skill-gaps (e.g AMR-to-SIAR). So if want to achieve a effective OOMs of progress, that would mean a improvement of $(1.2 \times a)$ in research skill. Next we estimate that the gap from median to top researchers is about 3-stds; this is somewhat definitional, and AI-2027 is not explicit about what exactly constitutes "top" level, but the survey they conducted asked respondents about the "best" at their company, and for comparison OpenAI has on the order of low-1000s of workers, and 3-stds above mean is about the top 1.4 in 1000.

The 3-std gap from AMR-to-SAR can be traversed in 3 std / 1.2 std/OOM SD = 3/1.2 = 2.5 OOMs. And the improvement in research-skill from AMR-to-SIAR is 3x the gap from AMR-to-SAR (in log-space), since the research-skill gap from SAR-to-SIAR is 2x by definition (so 1x + 2x = 3x, for the full AMR-to-SIAR gap). Therefore, at 1.2 SD/effective-OOM we can traverse AMR-to-SIAR in: num_median_to_top_gaps * std_per_median_to_top_gap / std_per_oom_progress = 3 gaps * 3 std/gap / 1.2 std/OOM = 7.5 effective-OOMs from AMR-to-SIAR.

So the status quo time to traverse AMR-to-SIAR, prior to any bottlenecks, with the current trend of software and hardware progress is $N = \text{round}(7.5OOMs/0.45OOMs/year) = 17$ years. Next we can determine the fixed-compute/personnel estimate using the summation formula above (w/ the full compute/personnel bottleneck). Here is a python snippet, which yields 15.6k years, matching *Table 1*; see the section below for multiplier-approach to estimating the AI-researcher times from this.

```
total_human_years_amr_to_siar = sum(0.4**-(0.6 * t) for t in range(17))
```

As discussed earlier, *Table 5* shows results for this model with varying levels of compute bottleneck, to get a sense for how timelines might increase if researchers in the survey were under-estimating the bitter-lesson; this table includes both a minimum-info prior based estimate, as well as an extreme bitter-lesson variant where algorithmic-progress is assumed to be entirely dependent on compute available for experiments, so that a 10x loss in compute means a 10x reduction in experiments/progress, which is equivalent to 1/10 = 0.1 in the formula above (as opposed to 0.4 for just the survey based bottleneck).

6.6 AI Speedup Multipliers

Once we have human-only time estimates for moving across the (AI-2027) skill stages i.e. from automated median researcher (AMR) to super-intelligent AI researcher (SIAR), then we can reuses the AI-2027 speedup multipliers (see *Table 7*) to estimate how long each of these stages with AI researchers and recursive self-improvement. Because the speedup multipliers are different at each stage, there isn't a single multiplier for AMR to SIAR; instead we have to run the empirical model separately for each gap e.g. AMR->SAR and SAR->SIAR, and then AI-2027's gap-specific multipliers can be applied for each stage. Note that when estimating these gaps sequentially you have to remember to carry over any hardware-compute deficit accumulated in previous stages to the current stage, when using the formula derived above. Also, keep in mind that the AI-2027 multipliers are not fixed within a given stage, but rather are interpolated as the AI improves during a given stage (see AI-2027 background section).

One potential complication is that the empirical-model estimates in this post are focused on quantifying the timelines starting with median-researcher skill, whereas the AI-2027 estimates focus more on superhuman coder (SC) as the starting point; and unlike the other AI stages, which are defined in part based on research skill, for the SC-stage estimating the research-skill is not central to their approach, since the strength of this model is more in its software-engineering rather than research skills. But in the AI-2027 takeoff report, they do suggest that their median estimate for the research-skill of SC is at the 25th percentile, ie they say "...where SC has the median level of research taste we're projecting (25th percentile, see above)". So in keeping with the general approach in this post of addressing

points of uncertainty/ambiguity in the direction favorable to the short AI-2027 short timelines, we can assume that SC is actually already at the median taste level; this should generally shorten timelines relative to their estimates, and it also has the simplifying feature that we can just reuse the SC speedup-multiplier (5x) for the AMR model, when estimating AI-speedup times. Also, elsewhere in the project (in the timeline report) they seem to imply that this 25-percentile assumption is conservative, i.e. saying that they expect SC to be at median or even better researcher taste, so this median assumption may also be more in keeping with the spirit of their projection. Note that for the other stages, the research-skill is more clearly specified, and so there is no ambiguity about which speedup multiplier to re-use.

Also, the AI-2027 project interpolates the multipliers across stages, by assuming that progress is proportional to research-effort. However, with this empirical model, we could potentially just directly model the recursive self-improvement process without their somewhat arbitrary interpolation approach. Based on a quick-and-dirty simulation of this alternative approach, it looks like this wouldn't drastically change the results, but it could be worth a more careful implementation of this alternative; one issue is that directly feeding this model back to itself recursively could amplify the concern raised above that the empirical model is implicitly assuming that skill-increases translate linearly into real-world progress rates, so some additional diminishing returns on real-world speedups could be more essential if this alternative interpolation approach were taken.

6.7 Adjusting for the Researcher Bottleneck

During the fast takeoff, AI-2027 assumes that both frontier compute and personnel counts remain fixed, since there is minimal time during the rapid takeoff for increasing the hardware substantially or scaling up the work-force. Note the current baseline model only takes into account compute bottlenecks (not personnel), but the alternative analysis in *Table 6* offers a *rough* first-pass estimate of the impact of these personnel bottlenecks, suggesting that they may be substantial.

For compute, we have a direct survey-based estimate from AI-2027 that a 10x drop in compute slows progress to 40%, but for personnel impacts some additional assumption is needed. One natural approach is to assume that progress rates approximately follow a constant-returns Cobb-Douglas functional form, so that if we 2x the compute for experiments and also 2x the number of workers to run/manage those experiments, we will roughly 2x progress, whereas if just one of this inputs is increased, then this will scale with a power law (with diminishing returns). So this looks like:

$$\text{progress\_rate} \sim \text{experiment\_compute}^{0.4} \times \text{frontier\_workers}^{1-0.4}$$

So the constant-returns assumption that is typical with Cobb-Douglas allows us to estimate the worker elasticity as 1 - 0.4 = 0.6. This leads to a substantial additional slow-down above-and-beyond the compute-based slow-down, but see *Table 5* for alternative time-estimates that don't include this

additional personnel-bottleneck and just take into account varying compute bottlenecks. In general, I am somewhat skeptical that worker-counts have such large effects, as these elasticities would imply, but partly this is just a result of the AI-2027 survey results implying a relatively modest slowdown from lost compute, whereas I suspect the slowdown would be much larger due to bitter-lesson-related considerations (e.g. progress is more about trying lots of fairly straightforward ideas w/ brute force compute, than brilliant eureka moments). But one advantage of taking into account both personnel and compute bottlenecks is that it reduces the sensitivity of the model to our estimate of the impact of lost compute (vs researchers), since both personnel and compute are held fixed during the fast-takeoff, this will cause a substantial slowdown regardless of how bitter-lesson-pilled your compute-bottleneck assumptions are. On the other hand, in *Table 5*, where the personnel bottleneck is not included, the results are much more sensitive to the estimated size of the compute bottleneck.

In order to determine the slow-down from holding fixed personnel, we need to know how quickly personnel has been increasing in the status quo trends. I plotted recent estimates of OpenAI's worker-count since the ChatGPT release, and based on a linear regression of this trend (on a log plot), it looks like they are increasing personnel by roughly 2.5x / year, perhaps in part to keep up with the research/engineering demands to make use of their rapidly increasing compute capacity. The underlying data from this plot is based mostly on news reports regarding the size of the OpenAI workforce; some of this data is a bit uncertain/approximate, e.g. one article indicated that "LinkedIn currently puts the employee count at over 2,000", which is just plotted as 2k, etc. But that said, the broad trend is likely basically accurate, and the deeper question is whether this recent rapid hiring pace is likely to continue, plateau or speed-up going forward. But the baseline estimate in this post just assumes the trend continues at the current rate, for the status quo case with no fast-takeoff.

Using these personnel-trend estimates combined with Cobb-Douglas, if compute slows down by 10x, we get the same 0.1^0.4 = 40% slowdown as in the compute bottleneck section above (note that a 0.4 elasticity happens to give approx 40% slowdown, but that numerical "coincidence" isn't true for other elasticities); and given the frontier personnel trends, 2.5x/year = 0.4 worker-OOMs/year, which implies 0.4 worker-OOMs/year / 0.6 hardware-OOMs/year = 0.6667 worker-OOMs / hardware-OOMs (so hardware is increasing a bit faster than personnel). So if we hold both hardware and personnel fixed, then for each 10x lost hardware OOM, there are 0.6667 lost worker OOMs = 4.64x (relative to the status quo trend). So this is a progress slowdown of:
$(1/10)^{0.4} * (1/4.64)^{0.6} = 0.4 * 0.4 = 0.16 = 16\%$, versus the 40% slowdown with just the compute bottleneck.

7. Modeling Inference Efficiency

7.1 Summary Inference Efficiency

In addition to the underlying improvement in capabilities (in SDs/effective-OOM), another potential pathway for models to improve is for increases in compute or inference-efficiency to allow the models to be run more quickly or with more copies, which could in principle allow them to make faster progress on their assigned tasks. Along with his underlying estimate of 1.2 SD/OOM in capabilities, Ryan Greenblatt gives [11] a further estimate of 1.25x speedup and 3x copies for each OOM of effective compute. Greenblatt doesn't give a justification for these particular speedup/copy estimates, but it is true that in addition to the algorithmic training-efficiency gains emphasized in this post's empirical model, per-token inference-efficiency at fixed capability has also been improving by roughly 50x/year (median), according to this [12] analysis from Epoch.

The main purpose of this section is to explain why the projections in this post are not adjusted to include speedups based on these inference-efficiency gains. The short explanation is that the median estimates below suggest that the inference-efficiency gains (at fixed capability) will be roughly canceled by slowdowns associated with running larger/better models over time, and even if one of these considerations wins out, it's unclear whether the net effect will be a speedup or slowdown of frontier models over time. That said, there is substantial uncertainty about this, and it would be worth incorporating a range of possibilities for this trend in future models. In a recent interview, the CEO of Anthropic (Dario Amodei) expressed a similar view [28] (transcribed from video):

> "I expect the price of providing a given level of intelligence to go down, [but] I expect the price of providing the frontier of intelligence, which will provide the kind of increasing economic value, that might go up, or it might go down; my guess is that it probably stays about where it is."

See *Table 9* for details on the empirical inputs needed for this assessment.

You might think that if raw model-capability gets better by 10x, and inference-efficiency improvements allow it to be run at 2x (and/or 2x copies), then we could multiply these gains together yielding a combined task-completion progress-multiplier, e.g. roughly $10 * 2^k$, where the power-law exponent k reflects diminishing returns from bottlenecks (e.g. compute). However, this risks double counting gains, since part of the reason that capabilities are improving is that models are getting more expensive to run, with scaling laws pushing towards larger models and more tokens per response; and currently our capabilities trends are measured on largely un-timed benchmarks, so this provides an estimate of capability/benchmark_response, not the capability/token that we would need to multiply by a efficiency gain in tokens/sec. So if we want to determine the relevant speedups/copies due to inference-efficiency progress, we first need to determine how much of the inference gains (at fixed

capability) are needed just to tread water against capability gains, i.e. to maintain current speeds/copies in the face of increasing model size, output tokens etc, at the frontier. One concrete way to estimate this is to determine how much of these inference gains would be needed to maintain the current SD/OOM gains in capability per benchmark, but when measured in capability per second.

7.2 Inference-related Trends

Input Factor	Current Trend (OOMs/year)	Reference
Inference efficiency (cost/token)	1.7	epoch.ai [20]
Inference efficiency (cost/token) - alternative estimate	1.0	a16z [12]
Training efficiency	0.45	epoch.ai, MIT, et al [9]
Hardware price-perf (for "ML GPUs")	0.15 (2x / 2.07 years)	epoch.ai [14]
Tokens per response (reasoning)	0.70	epoch.ai [15]
Tokens per response (pre-reasoning)	0.34	epoch.ai [16]
Context lengths supported	1.48	epoch.ai [23]
Tokens per active param (open-models)	0.49	epoch.ai [16]
Frontier hardware-training-compute	0.6	epoch.ai, MIT, et al [18]
Speedups per cost efficiency gain (speedup/cost-savings)	0.22 OOMs (3.5x/300x)	Epoch, via AI-2027 [4]

Table 9: *LLM efficiency related trends used to determine net speedups associated with the frontier capability trends.*

7.3 Trends Towards Increasing Costs

7.3.1 Increasing Model Sizes

First, we need to estimate how much models would need to increase in size (prior to reductions from algorithmic efficiency) in order to maintain the current capability trend. Chinchilla scaling would imply params ~ $C^{0.5}$, so $\text{params\_increase\_ooms} = \text{compute\_increase\_ooms}/2 = (0.6 + 0.45)/2 =$

$1.05/2 = 0.53\ \text{OOMs/year}$. However, in practice companies have been increasing active parameter scaling more slowly in part due to a reliance on mixture of experts (MoE), e.g. the Epoch analysis referenced above finds that active training parameters (which is what is mostly relevant for inference speeds) have not been increasing in proportion to training tokens as expected by Chinchilla, but rather the ratio of tokens per active param has been going up by about $3.1\text{x} = 0.49$ OOMs/year. This is consistent with a scaling law in which params ~ C^{α} and tokens ~ $C^{1-\alpha}$, so the ratio is:

$$\text{token\_increase}/\text{param\_increase} = C^{1-2\alpha}$$

So given that frontier open-model hardware training compute is increasing on a similar trend to frontier closed models, i.e. 0.67 OOMs/year for open-models [17], and assuming the algorithmic training efficiency trends are similar for open models i.e. about 0.45 OOMs/year; then this is about 1.12 OOMs/year in compute increase. This implies α of about 0.28. So for closed-models that are increasing by about 1.05 effective-OOMs/year, this implies parameter increases of about $11.2^{.28} = 1.97$x/year = 0.29 OOMs/year, which is a bit smaller than the 0.53 OOMs/year expected by Chinchilla; below I'll use the smaller trend-based 0.29 OOM/year, which is likely closer to the MoE reality, but if we used the larger Chinchilla figure this would leave even less inference-efficiency gains for speedups/copies.

7.3.2 Increasing Token Counts

Aside from increasing model sizes, another factor driving capability improvements over time is the increase in the number of output tokens per benchmark response. Per *Table 9*, Epoch estimates that since the development of reasoning-LLMs, the number of tokens per response has been increasing by about 0.70 OOMs/year. And per the discussion above, the underlying LLM capability trends are generally being measured on (mostly) un-timed benchmarks, so we need to adjust for these slower and more expensive responses when estimating the overall impact of these extra tokens on LLM task completion rates and costs.

7.4 Estimating Net Inference Cost-Reductions

Per *Table 9*, Epoch estimates 1.7 OOM/year median trend in inference-cost efficiency, estimated based on price comparisons over time at fixed benchmark capability. Note that one potential caveat is that prices are just a proxy for cost/efficiency, though a fairly reasonable proxy in a free-market context, particularly now that there are many LLM competitors. Though there is a risk that some of the recent decline is from increased competition as more companies have developed frontier models, which could exaggerate the efficiency gains. Also, there is a fair amount of uncertainty in this estimate, e.g. if you limit to the most recent data, the rate is 2.3 OOMs/year (data from jan-2024+), and it's unclear to me if this is noise, or pricing-artifacts per above, or a genuine shift in the trend. Epoch doesn't provide evidence as to whether this apparent shift is statistically significant, and they

acknowledge that "The fastest price drops in that range have occurred in the past year, so it's less clear that those will persist." As another data point, this a16z analysis [22] from late 2024 estimates only about 1 OOM/year cost-efficiency progress.

Given the median Epoch 1.7 OOM/year inference-cost-reduction trend, we can estimate how much of the gain remains if we use part of that efficiency gain to maintain a constant cost-per-response, so that an X SD/OOM capability-per-response gain will correspond to an X increase in capability-per-dollar. So this is given by:

1. Inference efficiency gain (per-year): 1.7 OOMs/year
2. Exclude non-algorithmic hardware gains: 1.7 OOMs/year - 0.15 OOMs/year = 1.55 OOMs/year (algorithmic-inference-gains)
3. Adjust for increasing tokens/year (frontier reasoning): 1.55 - 0.7 = 0.85
4. Adjust for increasing parameter counts/year: 0.85 - 0.29 = 0.56 OOMs/year
5. Adjust for available training-OOMs (total): 0.56 OOMs/year / 1.05 training-OOMs/year, which implies: net-inference-OOMs per effective-training-OOMs $= 0.53$

7.5 Estimating Net Inference Speed Impacts

7.5.1 Slowdown to match capability trend

While the estimate above (of 0.57 net-inference-cost-OOMs per effective-training-OOMs) maintains the ongoing capability trend at fixed cost per response, the responses will still be much slower as capabilities improve. So to adjust for this, we need to estimate how much of this net inference cost savings will be needed to maintain the capability trend when controlling for speed, i.e. measured in terms of capability per second. The response_time is given by tokens_per_response / tokens_per_sec. And our responses have 0.7 OOMs more tokens per year (which increases the numerator), and 0.29 OOMs more parameters (which reduces the denominator). However, per this Epoch analysis [20] inference latency does not generally decline linearly in active parameters as models scale, but according to the square root of parameters (divided by 2 for log-space parameter OOMs).

So the response_time increase from this capability gain comes to 0.7 + 0.29/2 = 0.85 OOMs/year slowdown, prior to efficiency speedups. Or on a per-training-oom basis this is: $0.85/1.05 =$

$$0.80 \text{ slowdown-OOMs per effective-training-OOMs.}$$

7.5.2 Direct speedups associated with cost-reductions

Per *Table 9* above, there is a 0.22 OOM speedup per OOM of inference cost-efficiency gains over time, which is taken from this estimate from Epoch quoted from AI-2027: "Epoch's estimates which rely

on cheaper per-token costs found a ~3.5x speed increase for a 300x decrease in cost" [4]. In this case, we had a total (as opposed to net) cost reduction of 1.55 algorithmic-cost-OOMs/year = 1.48 algo-inference-cost-OOMs/effective-training-ooms. So this implies 0.22 OOM-speedup/inference-cost-gain * 1.48 inference-cost-gain/effective-training-ooms = 0.33 OOM-speedup/effective-training-ooms.

7.5.3 Indirect speedups from extra copies

So we have 0.80 OOM slow-down due to supporting more capable models and a 0.33 OOM speedup associated with cost-efficiency gains, which is still a 0.47 OOM slow-down on benchmark tasks (per effective-training-oom). However, we still have 0.53 OOM net cost gains that labs could "spend" on extra copies to provide additional speedups. Though the slow 0.22-OOM historical speedup-trend suggests that putting these copies towards serial speedups is not what the frontier labs have chosen to do in practice; see the network latency discussion below for a potential explanation for the limited speedups in past trends.

Another concern you might have with spending the remaining savings on converting copies to speedups is that some of those copies have already been implicitly spent to achieve the 0.33 speedup trend (via parallelization), which could have used up some of that cost savings. But note that the 3.5x speedup per 300x cost-efficiency OOM is an empirical trend, in which the 300x cost-reduction is what is left over after the 3.5x speedup, so if any copies were already use for this speedup, then that is already reflected in the (per-token) 300x cost-reduction, i.e. extra copies to speedup the per-token throughput would make each token more expensive and would already be incorporated in the observed cost-reduction (of 300x); this suggests that the full 300x should still be available for additional copies, as opposed to having been partially used up by the baseline speedup.

The subsections below consider two different approaches for estimating the potential benefits from these additional copies.

7.5.3.1 Low-level parallelization via tiling

According to Steinhardt's estimate (via Aschenbrenner [19]), if you have compute for k^3 copies, you can instead use a "tiling scheme" to get a smaller k^2 serial speedup; or in other words, $\text{speedup} \sim \text{copies}^{2/3}$. Per Aschenbrenner, this can be effective "even for k of 100 or more". So given our remaining 0.53 OOM cost-efficiency gain, we can run approximately 0.53 OOM more copies (3.39x), which can alternatively imply a serial speedup of roughly $3.39^{2/3} = 2.26x = 0.35$ OOMs. So this cancels part of the remaining 0.47 OOM slowdown, i.e. 0.47 - 0.35 = 0.12 slowdown-OOMs / effective-training-OOMs.

However, an analysis from Epoch [20] suggests that this kind of aggressive parallelization will likely run into network-latency bottlenecks, and so may not be realistic in practice e.g. they say:

> "If we didn’t have to worry about latency constraints, we could indefinitely quadruple inference speed for each doubling of cost per token by parallelizing a forward pass across more GPUs, as observed previously by Steinhardt (2022) So the fact that latency constraints are binding in practice is crucial for understanding why LLMs are not served much faster than they currently are."

Note that Steinhardt's original post on this did acknowledge that he was ignoring the issue of latency bottlenecks, and based on these Epoch findings it looks like these bottlenecks may explain why the historical trends of speedups associated with cost efficiency gains have been so small (and haven't just been addressed by scaling copies), i.e. 3.5x speedup per 300x cost-efficiency gain (see *Table 9*).

Based on these latency-bottleneck concerns, this Steinhardt-style conversion of copies to speedups should probably be viewed as more of an upper bound on potential speedups rather than a central estimate, and the associated 0.12 slowdown-OOMs understood as somewhat optimistic, at least for low-level parallelization, as opposed to the agentic-framework below.

7.5.3.2 High-level agentic parallelization

The AI-2027 project acknowledges that copy-based parallelization may be a challenge, but suggests that parallelization may become more feasible once we are dealing with more agentic AIs/tasks, e.g. they say:

> "Further cost decreases could be attempted to be translated into increased speed via parallelization; for the non-agentic tasks Epoch measured this would be difficult, but it would probably be doable for SC-level AIs on agentic tasks."

This does in fact seem plausible, i.e. once we have agents carrying out long-term research tasks autonomously, they can presumably split up the work among agents and make progress in parallel, and in general research is highly parallelizable given the large number of potential ideas that can be investigated in parallel. Arguably for these kinds of tasks, parallel agents could even be more efficient than a single fast agent, due to more capacity to try uncertain ideas in parallel. There would be significant bottlenecks, especially compute for experiments, which parallel agents would have to share, but that is true for faster serial agents as well, who have reduced experiment-compute available per second of thought (as AI-2027 discusses).

So a simple, if somewhat optimistic, model is to assume that the extra parallel-tokens/sec from copies are roughly equal in value to the serial-token/sec from pure speedups, in the agentic context we are focused on. In this case, spending the remaining 0.53 OOM efficiency-gain on more copies gives roughly 0.53 OOMs more (parallel) tokens/sec, which directly cancels the progress slowdown of 0.47, leading to a small net speedup of 0.06 OOMs.

7.6 Net Slowdown or Speedup

So based on the above trends, here is a summary impacts of inference-efficiency gains as capability improves:

1. Inference-cost-efficiency-gains (per task): 0.53 OOMs/training-OOM
2. Inference-slowdowns-from-increased-capability (per task): 0.80 OOMs/training-OOM
3. Inference-speedups-from-efficiency-gains (per-task): 0.33 + 0.35 = 0.68 OOMs/training-OOM
4. Net-inference speedup/slowdown (per-task): -0.12 to +0.06 OOMs/training-OOM (depending on parallelization assumptions).

Some caveats:

1. These estimates aren't taking into account increasing context lengths (1.48 OOMs/year per *Table 9*) which could slow down model speeds quadratically in the worst case, leading to even slower frontier models; it's somewhat tricky to incorporate this properly, since most benchmarks don't require these full context windows; on the other hand, agentic tasks in the context of self-improving AI are plausibly more reliant on these larger context windows to maintain memory, and so could face this additional slowdown as capabilities increase.
2. If past trends hold, regarding speedups (per inference-efficiency gain), then the slowdowns would be much larger, around 0.47 OOMs (per effective-training-OOM), though this would conflict with the intuition that future agentic tasks will be relatively parallelizable via copies.
3. There could be more meaningful speedups if you limit to the most recent efficiency-trend data, but see discussion above for doubts as to whether this should be treated as the new trend, or whether the long-term median trend could be an over-estimate of efficiency-progress due to price drops being an imperfect proxy for efficiency gains.
4. Another consideration is that these estimates rely on the faster post-reasoning-model trends of increased tokens over time, but non-reasoning models are still commonly used and are not increasing output tokens to the same degree (though those are increasing as well); that said, reasoning models are likely more relevant to capability gains at the frontier, and they are also more likely to benefit from increased inference speeds/copies.

7.7 Impact of Speed on Progress Multipliers

The median results above suggest that the speed/copy impact of inference-efficiency gains counter-balanced against capability-increases is basically a wash, with potentially a small slowdown (-0.12 OOMs) or a small speedup (+0.06 OOMs) depending on the parallelization assumptions. However, there is substantial uncertainty particularly regarding the cost-efficiency trend, and there should certainly be some probability mass on the alternative possibility that the more recent data reflects a genuine speedup, and also for the opposite possibility the median estimate is already a substantial

over-estimate due to treating price as a proxy for cost during a period of increasing frontier competition driving down prices.

However, the baseline projections in this post don't make any adjustments to the capability estimates based on having more or fewer copies/speedups over time, and so are implicitly assuming the case where inference gains are mostly treading water against model capability gains.

But to give a sense for the potential impacts if these inference-impacts don't end up canceling out, we can estimate the capability reduction in the case where there is a net slow-down 0.12 OOMs (per the tiling approach above). In particular, if we assume the status quo capability trend continues (in SDs/training-OOM), then the slowdown above in tasks/sec of 0.12 OOMs/training-OOM should be expected to reduce the effective progress-multiplier. So using the AI-2027 [5] research-taste multiplier estimate of 3.25x (0.51 OOMs) speedup for top versus median researchers, and assuming "top" researchers are about 3 std above median, this implies 0.51 progress-OOMs per 3 SDs = 0.17 progress-OOMs/SD.

And per AI-2027's method 2 approach to estimating the impact of speedups/slowdowns on progress multipliers (taking into account the 40%/OOM compute bottleneck), a 0.12 OOM slowdown, would reduce the progress multiplier by about $0.4^{0}.12 = 0.90$x$= 0.05$ progress-OOMs, which gives $0.05/0.17 = 0.29$ SDs of lost progress (per effective-training-oom).

So using Greenblatt's 1.2 SD/effective-OOM estimate of progress, we are really getting closer to 1.2 - 0.29 = 0.91 SDs/effective-OOM improvement, once you account for the fact that the 1.2 SD/OOM is primarily being assessed on un-timed benchmarks, and the frontier is getting slower on a per-task basis. On the other hand, using the more "optimistic" perfect agentic-parallelization assumption, we found a net speedup of 0.06, which would slightly increase the SD/OOM trend. But again, the current baseline model does not incorporate these adjustments and treats this as a effectively a wash, with little certainty about whether the net impact will be positive or negative.

8. Empirical-Model Biases

8.1 Biases towards under-estimating timelines:

Summary of ways in which this empirical model could be underestimating timelines:

1. The model's timelines are only estimated through the SIAR stage, not to the final ASI step from AI-2027 (to simplify the analysis and avoid additional assumptions), but this means that the full takeoff timeline to ASI would be even longer than the baseline estimates given here. One natural way to estimate ASI timelines would be use a progress-trend (in SD/OOM) that is on the slow end

of the distribution, rather than using a central estimate, to account for the fact that ASI has to reach superhuman skill across every skill, not just for research skill.
2. The baseline model currently ignores personnel-bottlenecks, whereas the initial (rough) estimate in *Table 6* suggests that this could potentially substantially extend timelines; in general, both compute and personnel have been increasing exponentially and holding either fixed (as during the fast takeoff period) would substantially reduce frontier labs' capacity to experiment with alternative approaches and maintain algorithmic progress.
3. The baseline model accepts AI-2027's informal survey-based estimate of 40% for the computational bottleneck, but it wouldn't be surprising if researchers are over-estimating the importance of their own innovation and underestimating the importance of the bitter-lesson, leading to a smaller percentage of progress under fixed compute, and further extending timelines.
4. In the takeoff forecast AI-2027 assumes as a median estimate that the SC will only have 25th percentile research taste, but implicit in re-using the AI-2027 AI-speedup multipliers, the model is assuming that SC is 50th percentile (ie matching the AMR model); if it were really only 25th percentile, then this would delay timelines further, though this effect is not that large.
5. The current model is just re-using the AI-multipliers from AI-2027, but I have some concerns that these could be over-estimating the likely speedups from AI assistance, but I haven't had time to put together an alternative model for these multipliers. In general, these were largely based on survey-data, and so less intuition-based than the human-only timeline estimates, but some components are still largely intuitive guesses e.g. the 5x speedup for the superhuman coder (SC), etc.
6. Currently, the model currently makes no adjustment based on inference-compute efficiency gains over time, based on the estimates above that this broadly cancels with slowdowns related to capability improvements (large models and token-counts); but there is substantial uncertainty in this, e.g. if recent inference-price drops have actually mostly been caused by increased competition in the market, then we could be over-estimating efficiency gains, and so the AI speedups could be less than the current model is predicting; but it's also possible that the recent speedup trend reflects genuine inference efficiencies, in which case the current model is likely underestimating progress.

8.2 Biases towards over-estimating timelines

Summary of major sources of uncertainty in the model and ways in which it could be over-estimating timelines:

1. There is quite a bit of uncertainty in the capability trend, both because there is limited benchmark data with human-distribution information, and because we don't have any benchmark data specifically for research-skill (and different skills are progressing at different rates); the baseline model relies on Greenblatt's estimate for this trend, but there is some evidence from recent trends

that progress has been faster recently for some skills (see *Figure 1*); on the other hand, it's difficult to know to what extent this speedup is low-hanging fruit from the new reasoning paradigm or a longer-term shift in the rate of progress. My sense is that Greenblatt's estimate is defensible, and my general bias in this post is towards using 3rd party empirical estimates where possible (to reduce DOFs), but in this case my personal estimate would be a bit faster (see alternative estimates for this in *Table 4*)

2. The compute bottleneck currently assumes that every 10x reduction in compute slows progress to 40%, whereas the AI-2027 survey just asked about the impact of a single 10x compute reduction (not cumulative effects of 100x, 1000x, etc). So even if we trust this survey based bottleneck estimate, there is substantial uncertainty in extrapolating it over many OOMs of lost compute. But in the absence of data/evidence to the contrary this seems like a reasonable baseline assumption, since each drop in compute will continue to proportionately reduce the number of experiments that can be run, and this fixed power-law (10x -> 40%) relationship is implicit in the cobb-douglas functional form. That said, with additional data to fit a more complex model of research progress (e.g. a constant elasticity of substitution - CES model) the actual bottleneck could end up being larger or smaller than this simple power law fit. Note that this assumption is being made in the context of the human-only-researcher time-estimate, whereas the considerations could be quite different with AI-researchers, where compute and workers would be more substitutable.
3. The personnel bottleneck has a surprisingly large impact on slowing the timelines; this is in part due to the survey-based estimate of the compute-bottleneck being fairly modest, i.e. allowing for quite a bit of progress with 10x less compute, which implicitly assumes that researcher effort/skill has a significant impact beyond just brute force compute. But the Cobb-Douglas constant-returns assumptions (e.g. that the compute and personnel elasticities sum to 1) could certainly be an over-simplification. Given the uncertainty in this personnel bottleneck estimate, it does seem worth comparing to the projections with just the compute bottleneck (per *Table 5*); and while these estimates are quite a bit lower than the baseline estimates, they are still much longer than the AI-2027 estimates.
4. Currently the model takes into account algorithmic progress in training efficiency, but it doesn't explicitly account for inference-efficiency gains, based on the argument (detailed above) that these gains seem to roughly cancel out against that opposing force of increasing model size and output tokens needed to maintain the capability trend; that said, there is quite a bit of uncertainty in these estimates, and if there is actually a net efficiency gain this could allow for extra copies/speedups which could further speed progress; on the other hand, it actually seems more likely to me that the current trends are over-estimating the efficiency gains, since much of the recent price decline is likely from increasing competition as more companies develop frontier models (as opposed to true efficiency gains), in which case the frontier models may run more slowly than projected, leading to longer timelines.

5. Given that the predicted takeoff timelines from the empirical model are so far into the future, AI-2027's fixed compute assumption breaks down, so realistically compute would continue to increase over these longer time-frames and the resulting compute-driven scaling could be expected to speed up progress, leading to shorter estimates; however, in that case we are no longer dealing with a rapid recursive take-off scenario, which is the focus of this post.

9. Conclusion

This first-pass at a more empirically driven model for estimating takeoff timelines leads to fairly drastic increases in time estimates, going from 24 years to 15.6k years for the human-only estimate (far outside of AI-2027's confidence intervals), and from less than a year for the AI-2027 fast-takeoff to 247 years, where the latter is just an adjustment of the 15.6k estimate using the same AI speedup-multipliers as AI-2027. And if we account for the personnel-bottleneck during fast takeoff, the rough estimate from the sensitivity analysis above suggests this substantially lengthen timelines compared to these baseline estimates (with large uncertainty). To be clear, if the takeoff really were to take as long as this empirical model suggests, then the AI-2027 fixed-compute assumption would break down, so these extremely long estimates are more useful as evidence against fast takeoff scenarios (at least from relatively weak SC-level starting point), as opposed to accurate long-term projections. Also, while there are pros and cons of a more empirical approach versus AI-2027's more intuition-driven approach, I do think the fact that this straightforward empirical extrapolation of existing progress trends led to such larger estimates (far outside their reference ranges) should give us some increased doubts as to whether AI-2027's estimates may be giving too much weight to the possibility of a rapid software takeoff that isn't severely bottlenecked by hardware compute. And given that AI-2027's human-only time estimate (24 years) is so close to the model-prediction when no compute or personnel bottleneck is applied (17 years), this does raise a concern that their intuitive-judgment based times may not be properly accounting for compute constraints.

To be clear, this post is not arguing that short timelines or fast-takeoff are necessarily unlikely in general; rather, it is more narrowly pushing back against the AI-2027 projection that the arrival of superhuman coding AIs (which are not initially at the level top researchers) will quickly initiate a sub-year fast-takeoff. However, even with this empirical model there are more aggressive parameter values, within the realm of possibility, that could lead to this AI-2027-style outcome; for instance, if we assume research-skill will increase much faster than Greenblatt's 1.2 SD/OOM estimate e.g. perhaps closer to the reasoning-model progress implied by my (rough) trend-based guesstimates of recent frontier model compute (i.e. progress of about 3.24 SD/OOM), then this implies a 0.8 year time-line to SIAR (so sub-year AI-times); that said, aside from the general arguments given above for using somewhat more moderate progress estimates (e.g. per Greenblatt), another issue is that when we are estimating the full takeoff to ASI (not just SIAR) it becomes particularly difficult to justify using a really

fast progress rate, since for ASI every skill is required to exceed SIAR-level (not just research-skill); so given the relatively high variance in these SD/OOM rates across skills, we should expect slower SD/OOM rates when making ASI projections, in order to account for the hardest-to-learn skills (with slower progress rates); and this downward bias should be expected to substantially increase the human-only times to reach ASI, pushing the estimates closer to the baseline SD/OOM estimates above (even if you think some skills will increase much faster than this). So while we can potentially get sub-year takeoffs out of the model (starting from SC), it seems to likely require fairly aggressive assumptions/estimates, e.g. in the example above we are:

- Making aggressive assumptions about progress-rates, i.e. that research-skill will increase at the faster (and more uncertain) tail of the distribution of estimated progress-rates.
- Assuming that the slowest-progressing skills (towards ASI) won't be that much slower than typical/median skills, i.e. that the time to ASI won't be that much beyond the estimated time to SIAR.
- Assuming that personnel-bottlenecks are fairly small, and so don't need to be accounted for.

Another issue to consider is that the AI-2027 fast-takeoff projections assume that by the time we reach the superhuman coder (SC), the AIs will likely still only be fairly narrow intelligences and far from AGI, which makes fast takeoff much harder to achieve than if the takeoff was being simulated from a more advanced starting point. Therefore, in so far as the current analysis/projections make a fast takeoff from SC seem somewhat less likely, this could argue for putting more weight on existing compute-driven scaling trends as the path to AGI or top-human level; and it's still the case that as more advanced AI milestones are reached via ordinary scaling, recursive takeoff towards higher levels of superhuman intelligence becomes more likely from those more advanced starting points. And I do think that given current frontier scaling trends, there is a reasonably plausible case for short timelines to AGI (e.g. <5 years, at least for non-robotics-related cognitive skills) even without rapid recursive self-improvement, but these projections are out of scope for this post.

10. Potential Next Steps

Here are a few low-hanging-fruit ways in which this model could be improved in a future version:

1. Extend the estimates all the way to super-intelligence (ASI) for direct comparison with AI-2027, as opposed to stopping at SIAR. Since ASI involves progress across all skill areas, this would involve using an estimate on the low-end for the progress-trend when estimating human-only times, while still using more typical estimates to project research progress (and therefore AI multipliers).
2. Extend the more empirical trend-based approach to estimating the AI speedup-multipliers, which are currently just taken from AI-2027; these currently involve less guesswork than the AI-2027 human-only time estimates, but I do think there would be room to shift these in a more empirical

direction as well (e.g. taking into account trends for how much coding-assistants are currently speeding progress).
3. Run simulations to estimate confidence intervals.

That said, realistically the biggest impact towards improving the model would likely be from collecting data to generate better empirical estimates of the underlying parameters, e.g. the research-skill progress-rate, compute/personnel bottlenecks, inference-efficiency gains, etc.

References

[1] https://ai-2027.com/
[2] https://ai-2027.com/research
[3] https://www.lesswrong.com/posts/PAYfmG2aRbdb74mEp/a-deep-critique-of-ai-2027-s-bad-timeline-models
[4] https://ai-2027.com/research/timelines-forecast
[5] https://ai-2027.com/research/takeoff-forecast
[6] https://arxiv.org/pdf/2405.10938
[7] https://github.com/ryoungj/ObsScaling
[8] https://situational-awareness.ai/from-gpt-4-to-agi/
[9] https://arxiv.org/pdf/2403.05812
[10] https://arxiv.org/pdf/2212.05153
[11] https://www.lesswrong.com/posts/hpjj4JgRw9akLMRu5/what-does-10x-ing-effective-compute-get-you
[12] https://epoch.ai/data-insights/llm-inference-price-trends
[13] https://www.amazon.com/Superforecasting-Science-Prediction-Philip-Tetlock/dp/0804136718
[14] https://epoch.ai/blog/trends-in-gpu-price-performance
[15] https://epoch.ai/data-insights/output-length
[16] https://epoch.ai/data-insights/training-tokens-per-parameter
[17] https://epoch.ai/data-insights/open-models-threshold
[18] https://epoch.ai/blog/training-compute-of-frontier-ai-models-grows-by-4-5x-per-year
[19] https://situational-awareness.ai/from-agi-to-superintelligence/
[20] https://epoch.ai/blog/inference-economics-of-language-models
[21] https://www.lesswrong.com/posts/s64EK3kF9rexntpYm/my-ai-predictions-for-2027
[22] https://a16z.com/llmflation-llm-inference-cost/
[23] https://epoch.ai/data-insights/context-windows
[24] https://www.openphilanthropy.org/research/what-a-compute-centric-framework-says-about-takeoff-speeds/
[25] https://codeforces.com/blog/entry/126802

[26] https://www.trackingai.org/home
[27] https://epoch.ai/data/ai-models?view=table
[28] https://youtu.be/mYDSSRS-B5U?t=1851