

A Tour of AI Timelines

1. [Grokking “Forecasting TAI with biological anchors”](#)
2. [Grokking “Semi-informative priors over AI timelines”](#)

Grokking “Forecasting TAI with biological anchors”

Crossposted from the [AI Alignment Forum](#). May contain more technical jargon than usual.

Notes:

- I give a visual explanation of Ajeya Cotra’s draft report, [Forecasting TAI with biological anchors \(Cotra, 2020\)](#), summarising the key assumptions, intuitions, and conclusions
- The diagrams can be found [here](#) – you can click on some of the boxes to get linked to the part of the report that you’re interested in ^[1]

Thanks to Michael Aird, Ashwin Acharya, and the [Epoch](#) team for suggestions and feedback! Special thanks to Jaime Sevilla and Ajeya Cotra for detailed feedback.

Executive Summary

[Click here to skip the summary.](#)

Ajeya Cotra’s biological anchors framework attempts to forecast the development of **Transformative AI (TAI)** by treating compute as a key bottleneck to AI progress. This lets us focus on a concrete measure (compute, measured in [FLOP](#)) as a proxy for the question “when will TAI be developed?” Given this, we can decompose the question into two main questions:

1. **2020 training compute requirements:** How much compute will we need to train TAI, using 2020 Machine Learning architectures and algorithms?
2. **Affordability of compute:** How likely is it that we’ll be able to afford the compute required to train TAI in a particular year?

The second question can be tackled by turning to existing trends in three main factors: (1) **algorithmic progress** e.g. improved algorithmic efficiency, (2) decreasing **computation prices** e.g. due to hardware improvements, and (3) increased **willingness to spend on compute**.

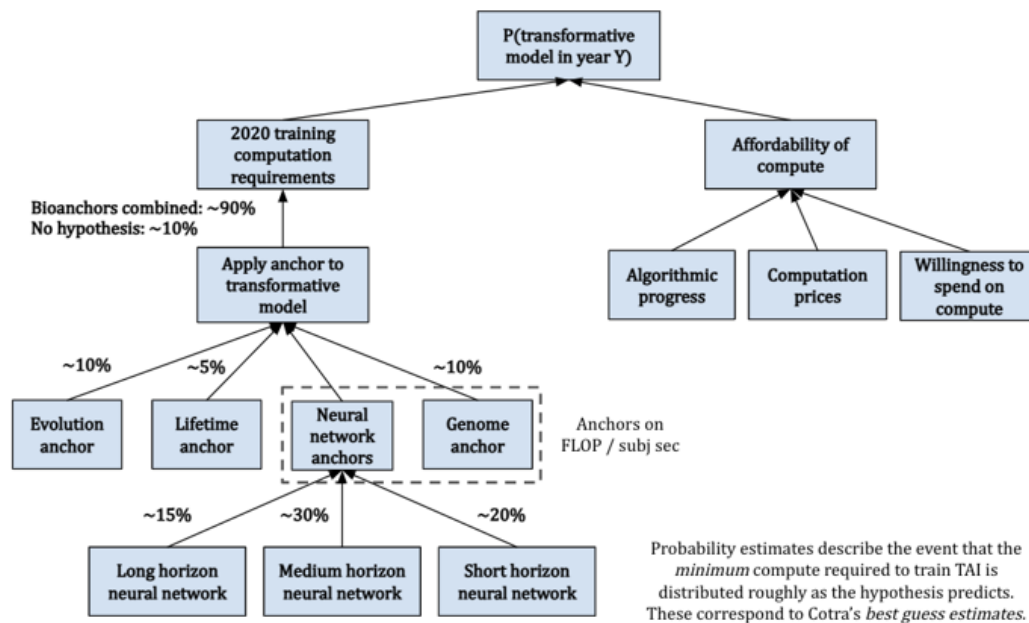
The first question is significantly trickier. Cotra attempts to answer it by treating the brain as a “proof of concept” that the “amount of compute” used to “train” the brain can train a general intelligence. This lets her relate the question “how much compute will we need to train TAI?” with the question “how much ‘compute’ was used to ‘train’ the human brain?”. However, there’s no obvious single interpretation for the latter question, so Cotra comes up with six hypotheses for what this corresponds to, referring to these hypotheses as “**biological anchors**” or “**bioanchors**”:

- **Evolution anchor:** Compute to train TAI = Compute performed over evolution since the first neurons
- **Lifetime anchor:** Compute to train TAI = Compute performed by the human brain when maturing to an adult (0 to 32 years old)
- **Three neural network anchors:** Anchor to the processing power of the human brain, and to empirical parameter scaling laws.
 - Technically there are three of these, corresponding to short, medium, and long “effective horizon lengths” – the amount of data required to determine whether or not a perturbation to the AI system improves or worsens performance
- **Genome anchor:** Anchor to the processing power of the human brain, set the number of parameters = number of bytes in the human genome, and extrapolate the amount

of training data required using the same empirical scaling laws mentioned above and assuming a long horizon length (one “data point” = multiple years)

In calculating the training compute requirements distribution, Cotra places 90% weight collectively across these bioanchor hypotheses, leaving 10% to account for the possibility that all of the anchors significantly underestimate the required compute.

Here’s a visual representation of how Cotra breaks down the question “How likely is the development is TAI by a given year?”:



The above was essentially a summary of Cotra’s factorization of the question of AI timelines; for a summary of her key findings, see [here](#).

Motivation

One of the biggest unresolved debates in AI Safety is the question of [AI Timelines](#) – when will **Transformative AI (TAI)** be developed? In 2020, Ajeya Cotra released a draft report, [Forecasting TAI with biological anchors \(Cotra, 2020\)](#), that aims to answer this question. It’s over 200 pages long including the appendices, and still just a draft!

Anecdotally, the results from this report have already been used to inform work in AI governance, and I believe it is likely that the report has had a major influence on the views of many researchers in AI safety.^[2] That said, the length of the document likely means that few people have read the report in full, are aware of its assumptions/limitations, or have a high-level understanding of the approach.

The aim of this post is to change this situation, by providing [yet another](#) summary of the report. I focus on the intuitions of the model and describe the framework visually, to show how different parts of Cotra’s report are pieced together.

Why focus on compute?

As you might imagine, trying to forecast the trajectory of a future transformative technology is very challenging, especially if there haven't been many technologies of a similar nature in the past. In order to gain traction, we'll inevitably have to make assumptions about what variables are the most important.

In the report, Cotra focuses on answering the following question:

In which year might the amount of computation required to train a “transformative model” become attainable for an AI development project?

Here, “transformative model” refers to a single AI model such that running many copies of that model (once trained) would have “at least as profound an impact on the world’s trajectory as the Industrial Revolution did”.^[3] It is a specific way that [“transformative AI”](#) could look – so Cotra’s report is essentially asking when we might have enough of a certain kind of resource (compute) to produce TAI through a certain path (training a giant AI model). She hopes that this sheds light on the broader question of “when might we have transformative AI” overall.

The question Cotra asks is thus more specific, but it seems plausibly informative for the broader question of TAI timelines because:

1. The “train a big model” path to TAI seems technologically possible, and is salient because it’s similar to how current state-of-the-art AI systems are produced. (Indeed it’s an unusually brute-force approach to AI, so the question “When might we get TAI by training a single big model?” could be seen as a [“soft upper bound”](#) for the question of “When might we get TAI?”).
2. It seems very plausible that compute is the resource that bottlenecks being able to train a transformative model the most. For instance (among other reasons):
 - Many algorithms/architectures that saw success after the advent of [Deep Learning](#) had been proposed decades earlier, and only [achieved competitive performance when researchers gained access to more compute](#)
 - Compute has been growing massively (by a factor of [10 billion times since 2010](#)), compared to algorithmic efficiency, which has grown a comparatively small amount ([44x since 2012](#))
 - Evidence in favour of [The Scaling Hypothesis](#) and scaling laws suggest that there are regular and predictable returns to training AI models on increasingly large scales of compute

It’s also convenient that compute is relatively easy to measure compared to nebulous terms like “data” and “algorithms”, which lack standardised units. A common measure for compute is in terms of the total number of arithmetic operations performed by a model, measured in [FLOP](#). We might also be interested in how many operations the model performs each second (measured in FLOP/s), which tells us about the power of the hardware that the model is trained on.

Framework

Cotra thus makes compute a primary focus of her TAI forecasting framework. Now instead of asking “when will TAI be developed?”, we ask two separate questions:

1. **2020 training compute requirements:** How much compute will we need to train a transformative model, using 2020 Machine Learning architectures and algorithms?
2. **Affordability of compute:** How likely is it that we’ll be able to afford the compute required to train a transformative model in a particular year?

The second of these is relatively straightforward to answer because we have some clear trends that we can analyse and [directly extrapolate](#).^[4] The first question however, opens a big can of worms – we need to find some kind of [reference class](#) that we can anchor to.

For this, Cotra chooses to anchor to the human brain – she views the human brain as a “proof of concept” that general intelligence is possible, then takes the analogy very seriously. The assumption is that the compute required to “train” the human brain should be informative of how much compute is needed to train a transformative model.

But how do we even define “compute to train the human brain”? There seem to be two main ambiguities with defining this:

- **How long was the human brain “trained” for?**
 - For instance, should we interpret the brain as being trained for a human lifetime, or over the course of neuron evolution?
- **How much compute was used at each point in training?**
 - For example, how many FLOP/s does the human brain run on?

Our answers to these questions determine the **biological anchors** – four^[5] possible answers to the question, “how much compute was used to train the human brain?”. Two of these anchor directly to FLOP of compute:

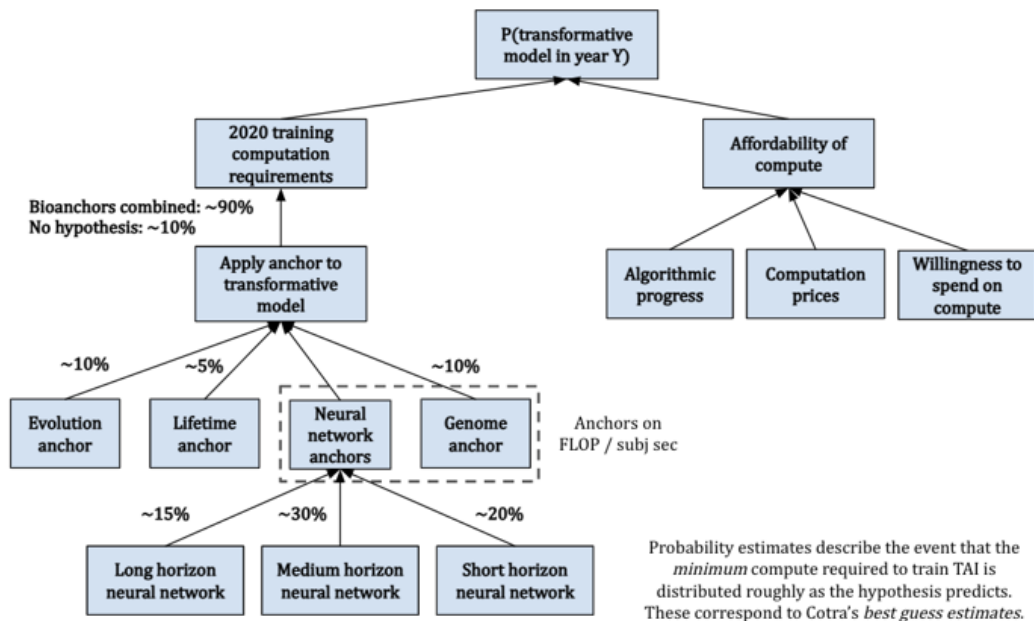
- **Evolution anchor:** The compute required to train a transformative model is roughly the total compute done over evolutionary history, starting from the first neurons. This interprets evolution as a really big search algorithm over a large space of possible neural architectures and environments, eventually stumbling across the human brain.
- **Lifetime anchor:** The compute required to train a transformative model is roughly the compute performed as a child matures, from birth to 32 years old. Under this hypothesis, we should expect Machine Learning architectures to be roughly as efficient as human learning.

The other two hypotheses anchor to the *computations per second* (i.e. FLOP/s) performed by the brain, rather than total compute. This is used to estimate the **FLOP per subjective second (FLOP / subj sec)** that TAI performs, where a “subjective second” is the time it takes a model to process as much data as a human can in one second.^[6] These hypotheses differ in how many parameters they predict TAI would need to have.

- **Neural network anchors**^[7]: TAI should perform roughly as many FLOP / subj sec as the human brain, and have a similar ratio of “parameters” to “FLOP / subj sec” as today’s neural networks do. There are actually three anchors here, as we’ll later see.
- **Genome anchor:** TAI should perform roughly as many FLOP / subj sec as the human brain, and have about as many parameters as there are bytes in the human genome.

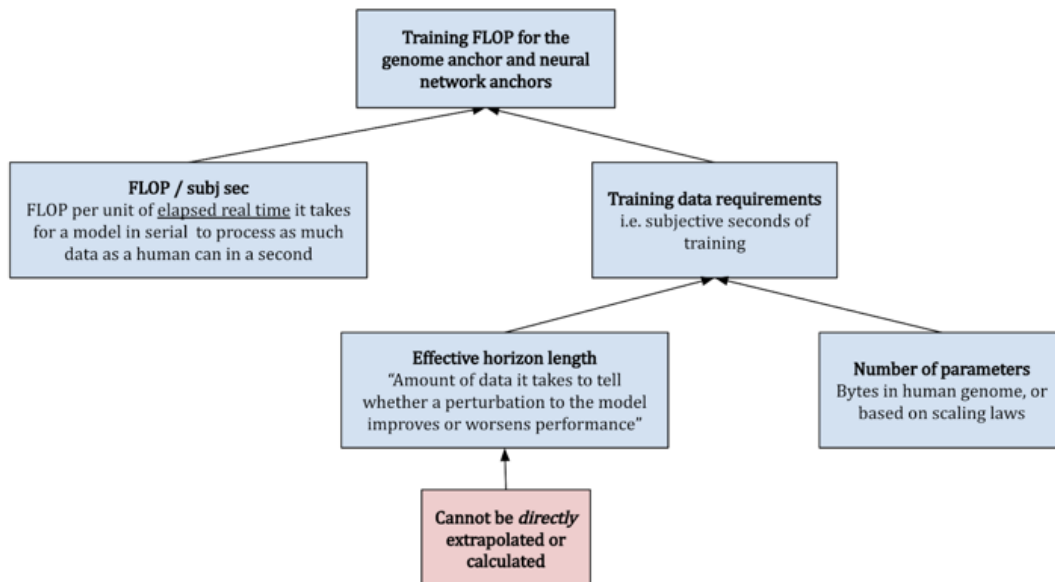
We can think of these anchors as saying that to build TAI, we’ll need processing power as good as the human brain, and as many parameters as (1) would be typical of neural networks that run on that much processing power, (2) the human genome.

You can see Cotra’s bioanchors framework at a high-level below:



On the left, we use bioanchors to determine how much compute we'll need to train TAI. Overall, Cotra allocates 90% weight to the bioanchors, where the remaining 10% is reserved for the possibility that all of the hypotheses are significantly underestimating required compute. On the right, we do projections for when we'll be able to afford this compute, based on trends affecting compute prices and the willingness to spend. These are combined to give an estimate for the probability of TAI by a particular year.

We saw earlier that the predicted FLOP for the evolution and lifetime anchors can be directly estimated, but this is not the case for the genome and neural network anchors. For this, we need to know both the number of FLOP / subj sec performed by the human brain, and the relevant number of subjective seconds required for training.



Finding the training data requirements is split into two parts:

- **Number of parameters**, which is specified by the relevant bioanchor hypothesis
- **Effective horizon length** - roughly, the amount of data it would take to tell whether a perturbation to the model improves or worsens performance.^[8] This is tricky to determine because it can't be directly extrapolated or calculated making it one of the biggest uncertainties in the report.

Combining all of these gives us a rough estimate for the compute that the relevant bioanchor predicts.

You now know the basic motivation and framework for how the model works! The next section will dive into where a lot of the complexity lies - figuring out probability distributions over training compute for each of the bioanchors.

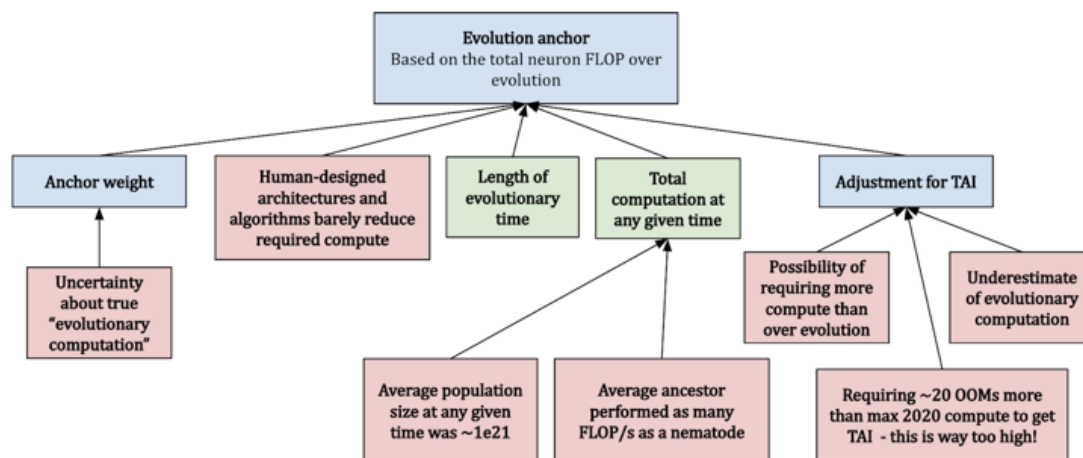
Zooming Into the Biological Anchors

We can think of each bioanchor as going through a three-step process:

1. Find a prior distribution for the FLOP based on biological evidence
2. Make adjustments based on evidence from current Machine Learning and intuitions
3. Decide how strongly you want to weigh the anchor

In this section I'll briefly outline^[9] the bioanchor hypotheses - I've also included a dependency diagram for each of them, where the boxes link to the relevant part of the report.

Evolution anchor



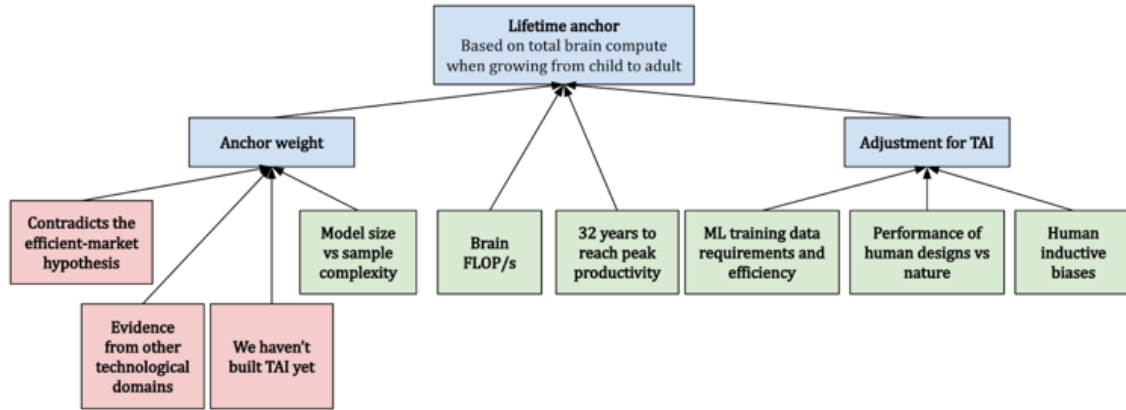
The **evolution** anchor looks at the total FLOP performed over the course of evolution, since the first neurons. Clearly there are some uncertainties with this approach:

- How do you even count “evolutionary computation”, and how does this compare with FLOP done on a GPU?
- What was the “average” compute done over all species at any time?
- How does the compute efficiency of human-designed architectures compare with just doing a random search?

Cotra accounts for these considerations, and assumes that the “average ancestor” performed as many FLOP/s as a nematode, and that there were on average ~1e21 ancestors

at any time. This yields a **median of $\sim 1e41$ FLOP**, which seems extraordinarily high compared to modern Machine Learning. ^[10] She gives this anchor a **weight of 10%**.

Lifetime anchor



The second approach based on counting FLOP directly is based on the **lifetime anchor**, which looks at the total brain compute when growing from child to an adult (32 years old). Plugging in the numbers about [brain FLOP/s](#) seems to suggest that $\sim 1e27$ FLOP would be required to reach TAI. This seems far too low, for several reasons:

- Examples from other technological domains suggests that the efficiency of things we build (on relevant metrics) is [generally not great when compared to nature](#)
- It also contradicts the [efficient-market hypothesis](#), and predicts a very substantial probability that [AlphaStar](#)-level compute would be TAI, which doesn't seem to be the case!

Overall, Cotra finds a **median of $\sim 1e28$ FLOP**, and places **5% weight** on this anchor.

Both the evolution and lifetime anchors seem to be taking a similar approach, but I think it's really worth emphasising just how vastly different these two interpretations are in terms of their predictions, so here's a diagram that illustrates this:

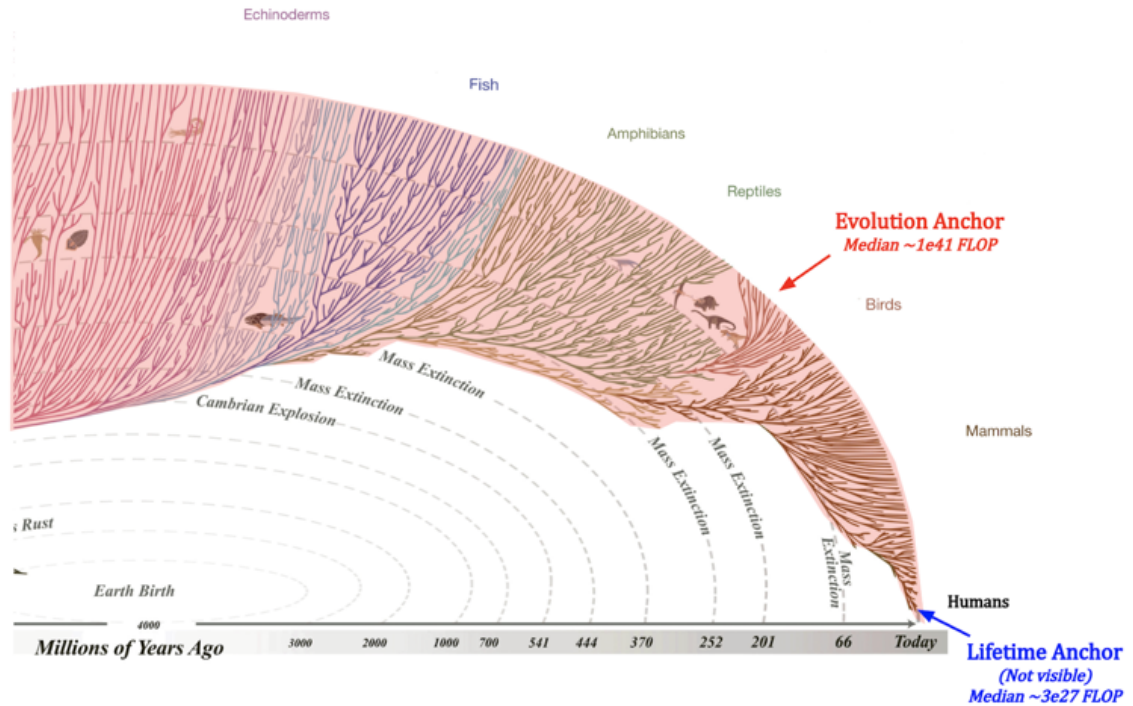
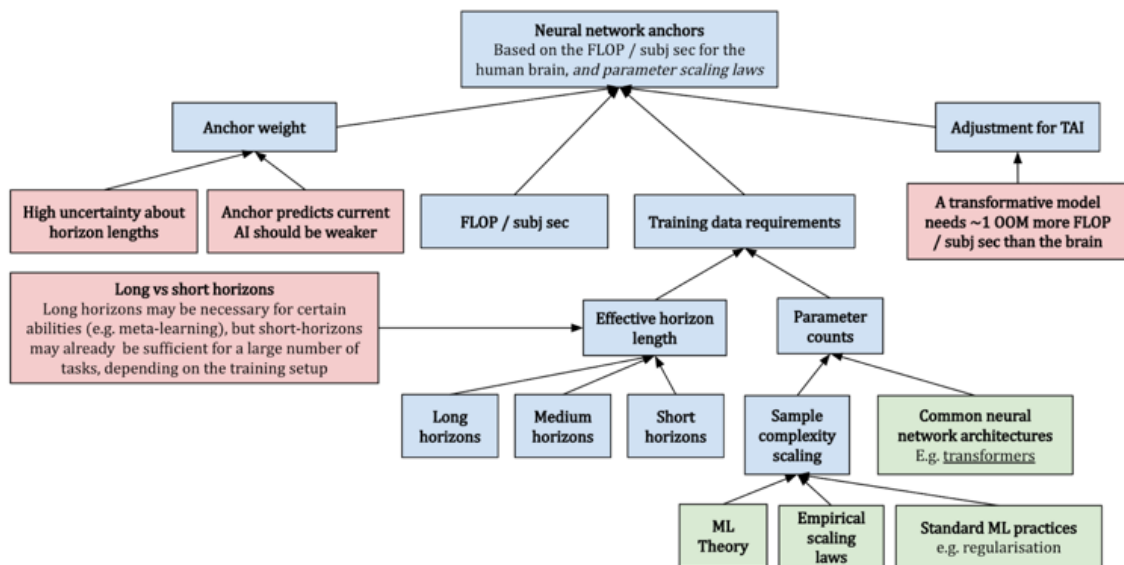


Image source: (For the evolutionary tree) [evogeneao Tree of Life Explorer](#)

If we look at the part of the evolutionary tree with neurons, then the evolution anchor includes neuron compute over the entire red area, across many different branches. On the other hand, the lifetime anchor requires us to zoom in *really* close to a small region in the bottom right, consider only humans out of all mammals, and consider only 32 years of the life of a single human out of the ~100 billion people who've ever lived. This isn't even close to being visible in the diagram^[11]!

Neural network anchors

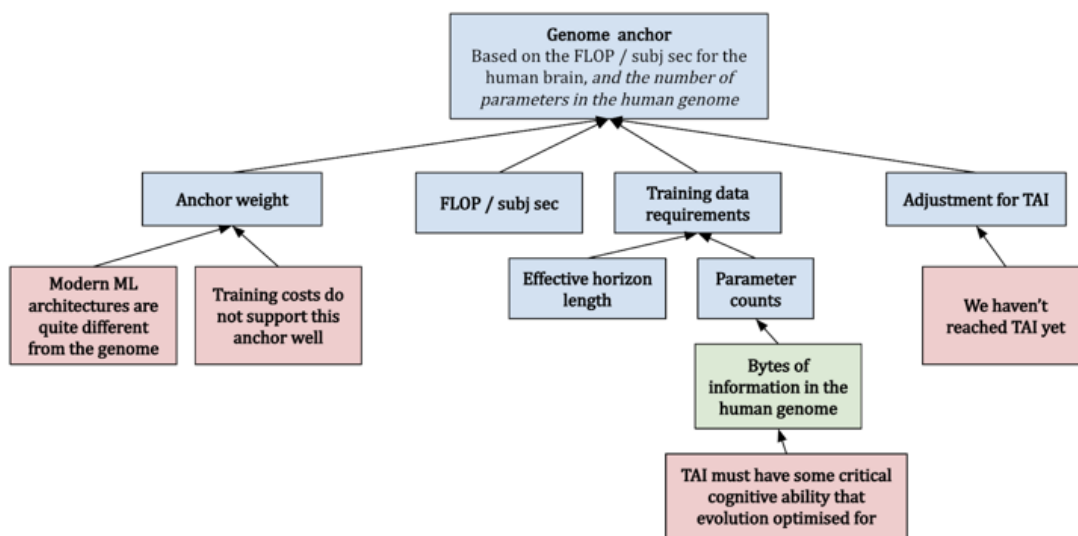


The three **neural network anchors** look at how much compute is required to train a network, by anchoring to the FLOP / subj sec performed by the brain, and based on parameter scaling laws. These anchors differ based on what horizon length is seen as necessary to achieve transformative impacts, and each have their own corresponding [log-uniform distribution](#).

- **Short horizon: 1 subj sec to 1e3 subj sec, centred around ~1e32 FLOP**
- **Medium horizon: 1e3 subj sec to 1e6 subj sec, centred around ~3e34 FLOP**
- **Long horizon: 1e6 subj sec to 1e9 subj sec, centred around ~1e37 FLOP**

Cotra determines the training data requirements based on a mix of Machine Learning theory and empirical considerations. She puts **15% weight on short horizons, 30% on medium horizons**, and **20% on long horizons**, for a total of 65% on the three anchors.

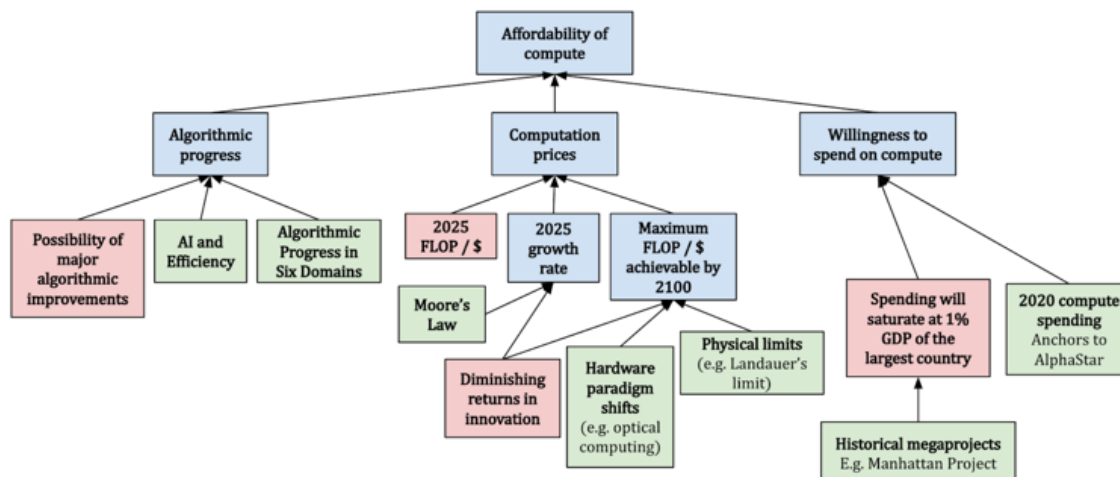
Genome anchor



The **genome anchor** looks at the FLOP / subj sec of the human brain, and expects TAI to require as many parameters as there are bytes in the human genome. This hypothesis implicitly assumes a training process that's structurally analogous to evolution^[12], and that TAI will have some critical cognitive ability that evolution optimised for.

At least at the time of writing (May 2022), Machine Learning architectures don't look very much like human genome, and we are yet to develop TAI - thus Cotra updates against this hypothesis towards requiring more FLOP. Overall, she finds a **median of ~1e33 FLOP** and places **10% weight** on this anchor.

Affordability of compute



After using the bioanchors to determine a distribution for the compute FLOP required to build TAI using 2020 algorithms and architectures, Cotra turns to find a probability distribution over whether or not we'll be able to afford this compute. She does this by considering three different factors:

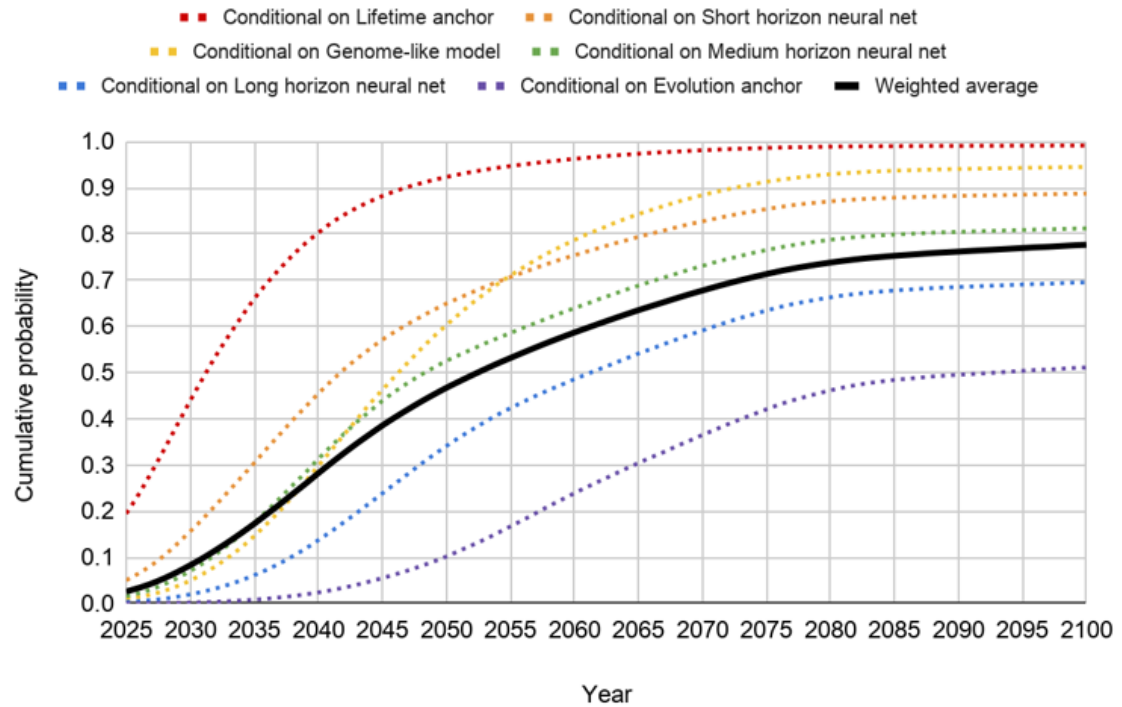
- **Algorithmic progress:** For this, she relies heavily on the [AI and Efficiency](#) study, which finds a 44x growth in algorithmic efficiency for Computer Vision algorithms between 2012 and 2018. She considers a **doubling in efficiency every ~2-3 years**, although the **cap on progress depends on the specific bioanchor hypothesis**
- **Computation prices:** We should expect to get more compute for a given price over time - Cotra bases this roughly on current trends in compute prices; **halving every ~2.5 years**, and further expects this to **level off after 6 orders of magnitude**.
- **Willingness to spend:** Cotra assumes that the willingness to spend on Machine Learning training runs should be **cappped at 1% the GDP of the largest country**, referencing previous case studies with megaprojects (e.g. the [Manhattan Project](#)), and should follow a **doubling time of 2 years after 2025**.

She makes these forecasts starting from 2025 to 2100, because she believes that there will be a rapid scaleup in compute for ML training runs from 2020 to 2025, and expects this to slow back down. The main uncertainty here is whether or not existing trends are going to persist more than several years into the future. For instance, we ([Epoch](#)) recently found that OpenAI's [AI and Compute](#) investigation was too aggressive in its findings for compute growth. In fact, there is [evidence that the reported trend was already breaking](#) at the time of publishing. All in all, I think this suggests that we should exercise caution when interpreting these forecasts.

Putting Things Together: Final distribution

If we put everything together, this is the distribution that we get:

Probability that FLOP to train a transformative model is affordable BY year Y



P(TAI before 2030)	P(TAI before 2050)	P(TAI before 2100)
~8%	~47%	~78%
10%	50%	90%
2031	2052	>2100

Based on these results, Cotra chooses a **median estimate of TAI by 2050**, a round number that avoids signalling too much precision in the estimates. These results seem to suggest that the probability of TAI being developed within this century is very high (at ~78%, see the table above).

You can of course question the premises and approach of this study, for instance:

- Is compute actually the biggest factor driving AI developments? Is it really reasonable to think of this as the main bottleneck, even a decade into the future?
- How valid is the approach of using bioanchors to determine the required compute to train TAI [\[13\]](#)?

- The report ignores the possibility of new paradigms (e.g. [optical computing](#)) and exogenous events that could hamper development – how much should we still trust this model?

Among other sources, Cotra states that the largest source of uncertainty comes from the appropriate value of the effective horizon length, which could range from 1 subj sec to 1e9 subj sec in the neural network anchors, and states that this is subject to further investigation. She also argues that the model overestimates the probability of TAI for short timelines due to unforeseen bottlenecks (e.g. regulation), and underestimates it for long timelines, since the research field will likely have found different paths to TAI that aren't based on scaling 2020 algorithms and architectures.

Conclusion

All in all, this is one of the first serious attempts at making a concrete framework for forecasting TAI, and it's really detailed! Despite this, there are still tons of questions that remain unanswered, that hopefully the AI forecasting field can figure out soon enough.

I also hope that these diagrams and explanations help you get a good high-level overview of what the report is getting at, and what kinds of further work would be interesting! You can find the [full report and code here](#), which I encourage you to look through.

You can play with the diagrams [here](#): (the boxes link to the corresponding part of the report). These were rather clunkily put together using Google Slides – if you have any suggestions for better software that's good for making these diagrams, I'd love to hear it!

1. [^](#)

Green boxes correspond to inputs, red boxes are assumptions or limitations, and blue boxes are classed as "other"

2. [^](#)

By "AI Safety", I am referring generally to work that helps reduce global catastrophic risks from advanced AI systems, which includes both AI governance and technical AI safety.

3. [^](#)

In general, it is not necessarily the case that these transformative effects need to be precipitated by a *single* model, although making this assumption is arguably still a good proxy for when we might see transformative impacts from multiple AI systems. The report also gives a more precise definition of "impact" in terms of [GWP](#), but my impression is that the heavy lifting assumption-wise is done by the bioanchors, rather than the precise definition of TAI. That is, I suspect the same bioanchors would've been used with somewhat different definitions of TAI.

4. [^](#)

Of course, things aren't *quite* so straightforward! For instance, we also need to consider the possibility of trends failing to persist, e.g. due to the end of [Moore's Law](#).

5. [^](#)

Technically there's six, but bear with me for now!

6. [^](#)

In her report, Cotra gives the following example: “a typical human reads about [3-4 words per second](#) for non-technical material, so “one subjective second” for a language model would correspond to however much time that the model takes to process about ~3-4 words of data. If it runs on 1000 times as many FLOP/s as the human brain, but also processes 3000-4000 words per second, it would be performing about as many FLOP per subjective second as a human.”

7. [^](#)

Since the neural network anchors don’t really correspond to any biological process, an alternative and arguably more accurate framing for them is “how much compute *would it take* to train a model as good as the human brain?” (as opposed to “how much compute was required to train the human brain?”).

8. [^](#)

For instance, for a True or False question answering task given a sentence, the effective horizon length might be the length of the input sentence.

9. [^](#)

My goal here is to provide a succinct summary of the key points, and to simultaneously provide links for people who want to learn more, so I refrain from putting too much detail here.

10. [^](#)

E.g. Google’s [PaLM model was trained with ~2.5e24 FLOP](#) – that’s 17 orders of magnitude smaller!

11. [^](#)

Of course, this diagram doesn’t account for the fact that certain species do a lot more compute than others, but I think it gets some intuition across – that there’s a great deal of uncertainty about how much compute was required to “train” the human brain.

12. [^](#)

This differs from the evolution anchor in that it assumes we can search over possible architectures/algorithms a lot more efficiently than evolution, using gradients. Due to this structural similarity, and because feedback signals about the fitness of a particular genome configuration are generally sparse, this suggests that the anchor only really makes sense with long horizon lengths. This is why there aren’t also three separate genome anchors!

13. [^](#)

In my view, this is the perspective that Eliezer Yudkowsky is taking in his post, [Biology-Inspired AGI Timelines: The Trick That Never Works](#). See also [Holden Karnofsky’s response](#).

Grokking “Semi-informative priors over AI timelines”

Notes:

- I give visual explanations for Tom Davidson’s report, [Semi-informative priors over AI timelines](#), and summarise the key assumptions and intuitions
- The diagrams can be found [here](#) – you can click on the boxes to get linked to the part of the report that you’re interested in ^[1]

Thanks to the [Epoch](#) team for feedback and support! Thanks especially to Jaime Sevilla and Tom Davidson for providing detailed feedback.

Executive Summary

The framework in [Semi-informative priors over AI timelines](#) assumes a model of [AGI](#) development which consists of a sequence of [Bernoulli trials](#), i.e. it treats each calendar year as a “trial” at building AGI with constant probability p of succeeding.

Model of AGI development process



p is constant over time, but our belief about its possible value changes.

Trials are inputs to AI R&D: years, researcher-years, or compute.

Image source: [Davidson, 2021](#)

However, we don’t know what this value of p is, so we use a generalisation of Laplace’s [rule of succession](#) to estimate $P(\text{AGI next year} \mid \text{no AGI yet})$. This is done by specifying a **first-trial probability**, the probability of successfully building AGI in the first year of AI research, together with the **number of virtual successes**, which tells us how quickly we should update our estimate for $P(\text{AGI next year} \mid \text{no AGI yet})$ based on evidence. The framework leans very heavily on the first-trial probability, which is determined using a subjective selection of reference classes ([more here](#)).

How much evidence we get depends on the number of trials that we see, which depends on the **regime start-time** – you can think of this as the time before which failure to develop AGI doesn’t tell us anything useful about the probability of success in later trials. For instance, we might think that 1956 (the year of the Dartmouth Conference) was the first year where people seriously started trying to build AGI, so the absence of AGI before 1956 isn’t very informative. If we think of each trial as a calendar year, then there have been $2021 - 1956 = 65$ trials since the regime start-time, and we still haven’t developed AGI, so that’s 65 failed trials which we use to update $P(\text{AGI next year} \mid \text{no AGI yet})$, where “next year” now corresponds to 2022 rather than 1957.

But why should a trial correspond to a calendar year? The answer is that it doesn’t have to! In total, Davidson considers three candidate **trial definitions**:

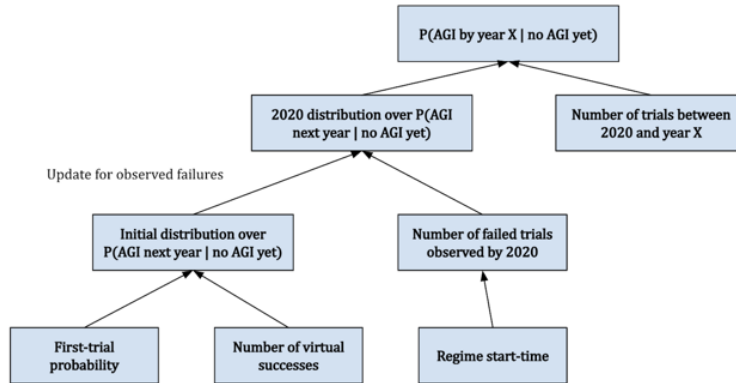
- **Calendar-year trials:** 1 trial = 1 calendar year
- **Compute trials:** 1 trial = a 1% increase in the largest amount of compute used to develop an AI system to date
- **Researcher-year trials:** 1 trial = a 1% increase in the total researcher-years so far

If we extend this reasoning, then we can predict the probability that AGI is built X years into the future. Davidson does this to predict $P(\text{AGI by 2036} \mid \text{no AGI yet})$ as follows:

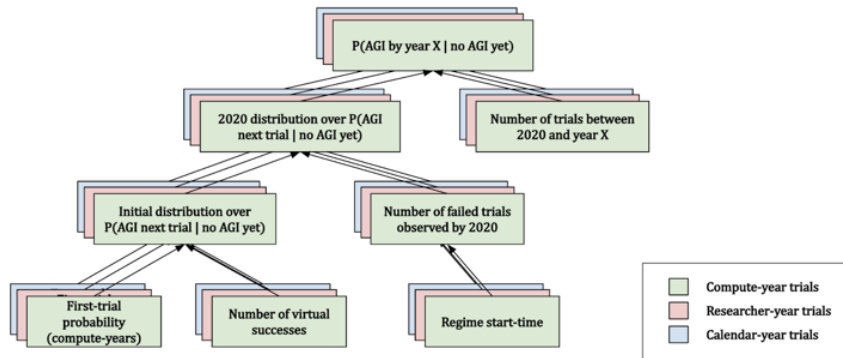
$$P(\text{AGI by 2036} \mid \text{no AGI yet}) = 1 - P(\text{no AGI by 2036} \mid \text{no AGI yet})$$

$$= 1 - P(\text{no AGI in 2022} \mid \text{no AGI by 2021}) \dots P(\text{no AGI in 2036} \mid \text{no AGI by 2035})$$


The idea is that this framework only incorporates a small amount of information based on observational evidence, giving “**semi-informative priors**” over AI timelines. This framework is shown in more detail below:

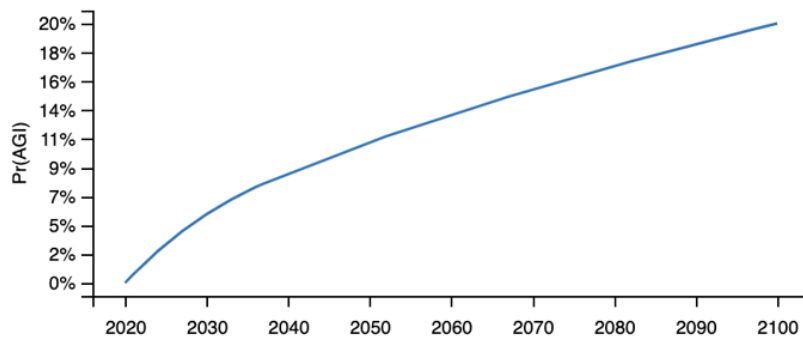


Since Davidson uses three different trial definitions, we actually get three of these diagrams!



All in all, Davidson uses this to get a central estimate of $P(\text{AGI by 2036} \mid \text{no AGI yet}) = 8\%$, with the following cumulative probability function:

Weighted average probability of AGI by 2036: 7.5% 



Motivation

One way of forecasting [AI Timelines](#) is to consider the inner workings of AI, guess what kinds of developments are the most important, and then generate a probability distribution over when [Artificial General Intelligence \(AGI\)](#) will be developed. This is the approach taken by Ajeya Cotra in [Forecasting TAI with biological anchors](#), a really detailed draft report that draws analogy to the human brain to forecast when [Transformative AI \(TAI\)](#) will first be developed. ^[2]

Tom Davidson's report, [Semi-informative priors over AI timelines](#), is also a detailed report forecasting AI timelines, but it takes a different approach to Cotra's report. Rather than thinking about the details of AI development, it assumes we know *almost nothing* about it^[3]!

The goal of this post is to explain the model through the liberal use of diagrams, so that you can get high-level intuitions about how it works, hopefully informing your research or understanding of AI forecasting.

Laplace's Rule of Succession

Suppose we're trying to determine when AGI will first be developed, without knowing anything about the world except that there have been N years so far, and AGI has not been developed in any of these years. How would you determine the probability that AGI is developed in the next year^[4]?

A naive approach we might take is to think of each year as a ["trial" with two possible outcomes](#) – (1) *successful trials*, where AGI is successfully built in the year of interest, and (2) *failed trials*, where AGI is not built in the year of interest. We then assume that the probability of building AGI in the next year is given by the total successful trials divided by the total trials:

$$P(\text{AGI next year} \mid \text{no AGI yet}) = \frac{\text{successes}}{\text{successes} + \text{failures}} = \frac{\text{successes}}{\text{total trials}}$$

Since AGI hasn't been built in any of the last N years, there have been zero successes out of N trials. We thus conclude that the probability of AGI in the next year is zero... but clearly there's something wrong with this!

The problem is that this approach doesn't even account for the possibility that AGI might ever be developed, and simply counting the number of successes isn't going to be very helpful for a technology that hasn't been invented yet. How can we modify this approach so that both the possibility of success and failure are considered?

One clever way of doing this is to consider "virtual trials". If you know that it's possible for each trial to be either a success or a failure, then it's as if you had previously observed one "virtual success" and one "virtual failure", which we can add to the total observed successes and failures respectively. We can then modify the equation to:

$$P(\text{AGI next year} \mid \text{no AGI yet}) = \frac{\text{successes} + 1}{(\text{successes} + 1) + (\text{failures} + 1)} = \frac{\text{successes} + 1}{\text{total trials} + 2}$$

This equation is called [Laplace's rule of succession](#), which is one approach to estimating the probabilities of events that have never been observed in the past. In particular, it assumes that we know *nothing* about the world except for the number of trials and the number of successes or failures.

If we apply this method, then we find that the probability of building AGI in the next year is $1/(N + 2)$. Assuming that the field of AI was formed in [1956 at the famous Dartmouth Conference](#), then this suggests that $N = 2021 - 1956 = 65$ and $P(\text{AGI is built in 2022}) = 1/67$, or a probability of around 1.5%.

If we extend this reasoning, then we can predict the probability that AGI is built X years into the future. Davidson does this to predict $P(\text{AGI by 2036} \mid \text{no AGI yet})$ as follows:

$$\begin{aligned} P(\text{AGI by 2036} \mid \text{no AGI yet}) &= 1 - P(\text{no AGI by 2036} \mid \text{no AGI yet}) \\ &= 1 - P(\text{no AGI in 2022} \mid \text{no AGI by 2021}) \dots P(\text{no AGI in 2036} \mid \text{no AGI by 2035}) \end{aligned}$$

This seems a lot more reasonable than the naive approach, but there's still some serious problems with it, like the following:

- **It's extremely aggressive before considering evidence:** For instance, according to Laplace's rule the attendants of the [1956 Dartmouth Conference](#) should have predicted a 50% probability of developing AGI in the first year of AI research, and 91% probability within the first ten years!
- **It's sensitive to the definition of a "trial":** If we had chosen each trial to be "one day" instead of a year, our conclusions would be drastically different.

What's going on here (among other things) is that the rule of succession makes very few prior assumptions – i.e. it's an **uninformative prior**. In fact, it's so uninformative that it doesn't even capture the intuition that building a transformative technology in the first year of R&D is not commonplace! Clearly, we still need something better if we're going to make predictions about AGI timelines.

Making the priors less uninformative

The solution that Davidson proposes is to make this prior less uninformative, by incorporating certain pieces of common sense intuition and evidence about AI R&D. Looking more closely at the framework given by Laplace's rule of succession, we see that it depends on several factors:

- **Regime start-time:** You can think of this as the time before which failure to develop AGI doesn't tell us anything useful about the probability of success in later trials. We've been assuming this to be 1956, but this doesn't have to be the case!
- **First-trial probability:** The odds of success on the first "trial" from the regime start-time onwards
- **Trial definition:** Why are we using "one year" as a single trial, and what are some alternatives?

We can also add an additional modification, in the form of the **number of virtual successes**. This affects how quickly you update away from the first-trial probability given new evidence – the more virtual successes, the smaller your uncertainty about how difficult it is to build AGI, and thus the less you update based on observing more failed trials. For example, suppose that your initial $P(\text{AGI next year} \mid \text{no AGI yet})$ is $1/100$:

- If you start with 1 virtual success, then after observing 100 failed trials your updated $P(\text{AGI next year} \mid \text{no AGI yet})$ is now $1/200$
- In contrast, if you start with 10 virtual successes, then after 100 failed trials your updated $P(\text{AGI next year} \mid \text{no AGI yet})$ is $1/110$

So far, we've been thinking about predicting whether or not AGI will be developed in the next year, but what we're really interested in is *when it will be developed, if at all*. Davidson tries to answer this by assuming a simple model of development, consisting of a sequence of trials, where each trial has a constant probability p of succeeding. ^[5] Note

that this probability is not the same as $P(\text{AGI next year} \mid \text{no AGI yet})$ - the latter corresponds to our *belief* about the value of p ; it isn't the same as p itself.

Model of AGI development process

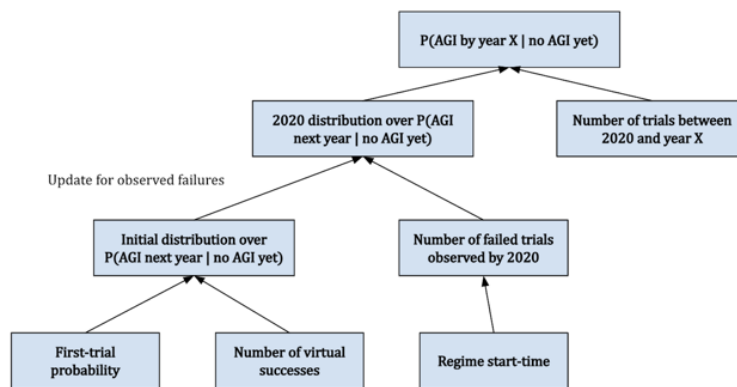


p is constant over time, but our belief about its possible value changes.

Trials are inputs to AI R&D: years, researcher-years, or compute.

Image source: [Davidson, 2021](#)

When the four inputs to the distribution $P(\text{AGI in year } X \mid \text{no AGI yet})$ are determined using common sense and some relevant reference classes, Davidson calls this distribution a "**semi-informative prior**" over AGI timelines. Rather than considering tons of gnarly factors that could in principle influence progress towards AGI, we only look at a few select inputs that seem most relevant.



Adapted from [Davidson \(2021\)](#).

The diagram above shows how the framework is pieced together. The first trial probability and number of virtual successes are used to generate an initial distribution for the probability of AGI in the next year. We then update this distribution with 2020 evidence based on the trials we’ve observed, depending on our specified regime start-time.

This gives us the 2020 distribution for P(AGI next year (i.e. 2021) | no AGI yet). We combine this with the number of trials between 2020 and the year X that we’re interested in, to get the final distribution over

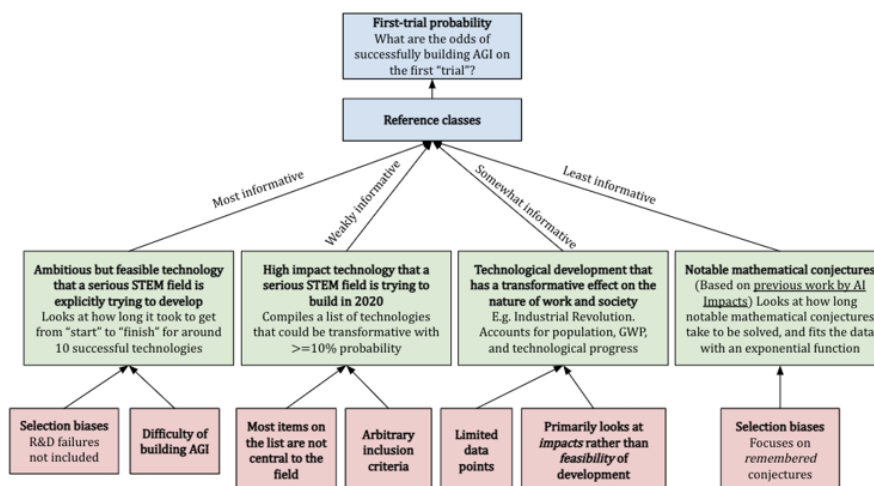
P(AGI by year X | no AGI in 2020). Note that this actually also depends on the trial definition – we’ll discuss how this fits into the diagram later.

Semi-informative priors demystified

Now that we have the basic framework established, we just need to figure out what values we should assign to the input variables (i.e. first-trial probability, number of virtual successes, regime start-time, and trial definition). Davidson considers the first-trial probability to be the most significant out of these four input factors (via a [sensitivity analysis](#)), although all are based on fairly subjective judgements.

Let’s take a look at each of these in turn.

First-trial probability



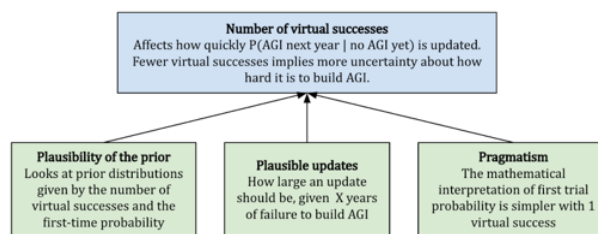
The **first-trial probability** asks, “what is the probability of successfully building AGI on the first ‘trial’?”. This is very hard to determine just on the surface, and so Davidson turns to several historical examples from a few [reference classes](#). In particular, he looks at:

- ~10 examples of ambitious but feasible technologies that a serious STEM field is explicitly trying to develop (analogously, the field of AI is explicitly trying to achieve the ambitious but likely achievable goal of AGI)
- Technologies that serious STEM fields are trying to build in 2020, that plausibly seem like they could have a [transformative impact on society](#).
- Previous technologies that have had a transformative impact on the nature of work and society
- Notable mathematical conjectures and how long it took for them to be resolved (if indeed they were)

Davidson uses these reference classes to derive constraints on the first-trial probability – this can be done by obtaining a base rate of successful trials from the past examples. Most of these don't succeed in the first trial^[6], so one approach he uses is to look at how many successes there are after X trials, then works backwards using Laplace's rule. He ultimately settles on a **best guess first-trial probability of 4%**.

It's worth noting that these reference classes and upward adjustments from the other trial definitions are the most important part of the framework, and the choice of these reference classes makes a really big difference to the final conclusions.

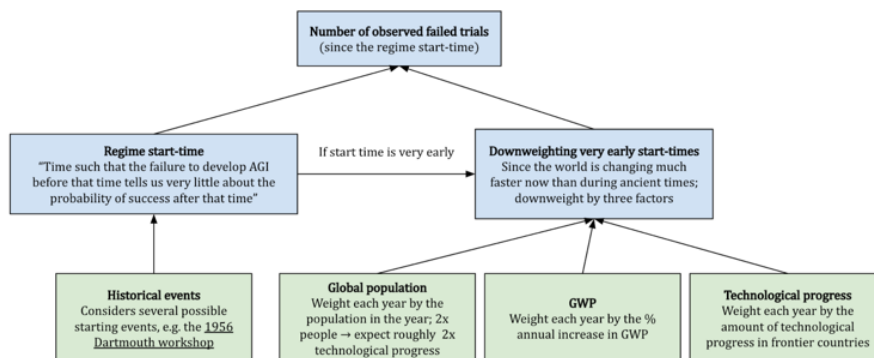
Number of virtual successes



The **number of virtual successes** changes how quickly we should update based on our observation of failed trials.^[7] We want the size of this update to be reasonable, so we don't want this number to be too large or too small. Davidson ultimately settles on **1 virtual success** for most of the report, based on a combination of pragmatism, the plausibility of the prior^[8], and the plausibility about the update size given new evidence.^[9]

Different choices of the number of virtual successes matter less when the first-trial probability is lower, because making a big update (in proportion) from the prior distribution matters less in an absolute sense when the initial priors are already small.

Regime start time

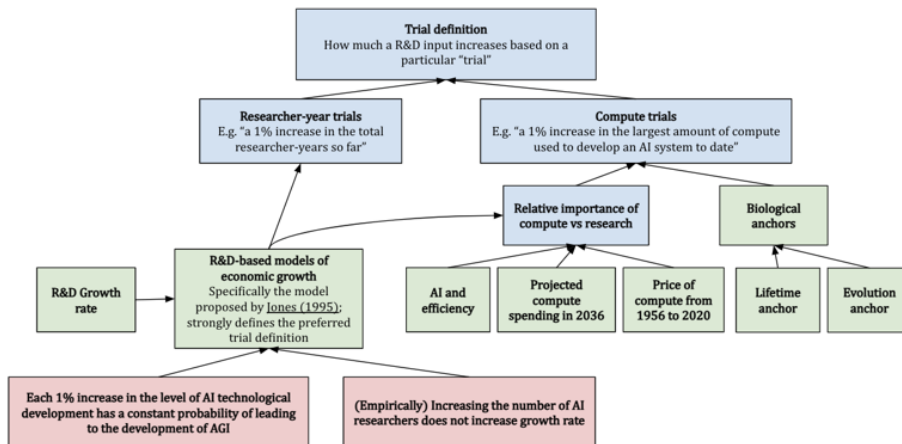


The **regime start-time** is the time for which "the failure to develop AGI before that time tells us very little about the probability of success after that time", and affects the number of failed trials that we observe. While we previously considered the Dartmouth Conference in 1956 as the natural start of AI research, other alternatives (e.g. 1945, when the first digital computer was built) also seem reasonable.

A problem with assuming a constant probability p of AGI being developed in any year becomes especially salient if we consider very early start-times. Suppose we argue that people have been [trying to automate parts of their work since ancient times](#), and choose a start-time correspondingly. Then the framework would suggest the odds of building AGI in any year in ancient times is the same as that today!

Davidson addresses this problem by down-weighting *the number of trials* occurring in ancient times relative to modern times, by multiplying (with normalisation!) each year by the global population or the economic growth in that year.^[10] Overall, he places the most emphasis on a start-time of 1956, but does a sensitivity analysis with several alternatives, which do not significantly change the conclusions when appropriate down-weighting is applied.

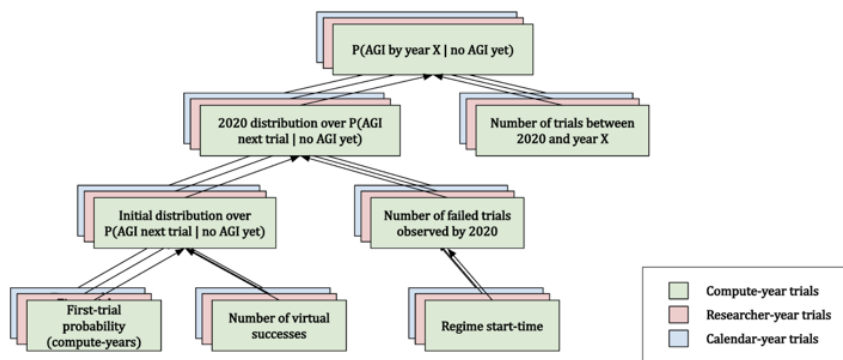
Trial definition



The final input to the framework is the **trial definition**, which specifies what exactly constitutes a single “trial” at building AGI. The initial approach we considered was in terms of calendar years, but there are reasonable alternatives, for example:

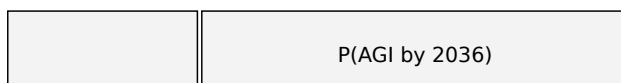
- **Compute trials:** Trials based on compute, e.g. 1 trial = “a 1% increase in the largest amount of compute used to develop an AI system to date”. These trials implicitly assume that increases in training compute are a key driver of AI progress^[11]
- **Researcher-year trials:** Trials that are defined in terms of the number of researcher-years performed so far, e.g. 1 trial = “a 1% increase in the total researcher-years so far”. We’re in effect assuming that each 1% increase in the “level of AI technological development” has a constant probability of developing AGI.^[12]

Davidson considers both of these possible trial definitions, together with the calendar-year definition, finding that the resulting probabilities can vary a little depending on the chosen trial definition. In effect, we now have three separate frameworks based on the trial definition:



If we change the trial definition, then presumably we’ll also change the first-trial probability, so how do we calculate this? One approach that Davidson takes is to compute the first-trial probability for compute-years and researcher-years from the first-trial probability for calendar years – I’ll not go into this here, but I suggest looking at [these sections](#) of the report to find out more.

Assuming 1 virtual success and a regime start-time of 1956, here’s what we get:



Trial definition	Low-end	Central estimate	High-end
Calendar-year	1.5%	4%	9%
Researcher-year	2%	8%	15%
Compute trial	2%	15%	25%

Importantly, we can choose our first-trial probability such that our predictions remain the same for trivial changes in the trial definition, helping solve one of the aforementioned problems with applying Laplace's rule of succession.

^[13] Overall, Davidson assigns $\frac{1}{3}$ **weight to each of the three trial definitions** considered.

Putting things together: Final distribution

Model Extensions

The framework also considers three extensions to the stuff outlined above:

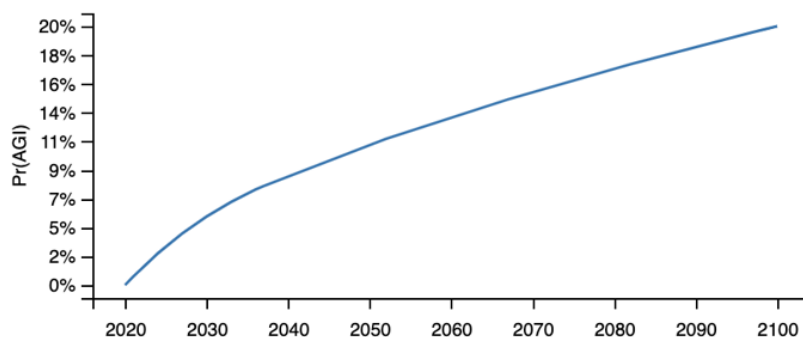
- **Conjunctive model of AGI:** It considers treating AGI development as the conjunction of multiple *independent* tasks
- **Hyperpriors over update rules:** Updating a prior over what weight to assign to different update rules, which are themselves determined by the four inputs^[14]
- **Allow some probability that AGI is impossible**

For the most part, these extensions don't have a particularly large effect on the final numbers and conclusions.

Final Distribution

If we combine everything from above then we end up with the following distribution and predicted numbers^[15]:

Weighted average probability of AGI by 2036: 7.5% ⓘ



P(AGI by 2030)	P(AGI by 2050)	P(AGI by 2100)
~6%	~11%	~20%

10%	50%	90%
~2044	>2100	>2100

Davidson highlights three main strengths of his framework:

- **It quantifies the size of the update to $P(\text{AGI next year} \mid \text{no AGI yet})$ based on observed failures**
- **It highlights the significance of intuitive parameters**, e.g. the first-trial probability, regime start-time, and the trial definition
- **It's arguably appropriate for expressing deep uncertainty about AGI timelines**, e.g. by avoiding claims about "what fraction of the research we've completed towards AGI"

He also points out some main weaknesses of the framework:

- **It incorporates limited kinds of evidence which could be really informative**, e.g. how close we are to AGI
- **Its near term predictions are too high**, because current AI systems are not nearly as capable as AGI, and the framework doesn't account for this evidence^[16]
- **It's insensitive to small changes in the definition of AGI**
- **It assumes a constant chance of success in each trial** (although the conjunctive model of AGI proposed in the extension relaxes this assumption)

There are also some situations where it doesn't make sense to use this framework – for instance, when we know what "fraction of progress" we've made towards achieving a particular goal. This can be hard to quantify for AGI development, but it's actually closely related to an approach that the [Median group has previously attempted](#).

Conclusion

I think this model suggests that developing AGI within this century is *at least* plausible – we shouldn't dismiss the possibility of developing AGI in the near term, and that the failure to develop AGI to date is not strong evidence for low $P(\text{AGI by 2036})$.

I personally found the approach taken in this report really interesting, particularly in terms of the solutions Davidson proposes to the problems posed by the rule of succession. This seems possibly very valuable for other work on forecasting. I encourage you to look at the report's [blog post](#)^[17], and to try [making your own predictions using the framework](#).

You can play with the diagrams [here](#), where the boxes link to the corresponding part of the report.

1. [^](#)

Green boxes correspond to inputs, red boxes are assumptions or limitations, and blue boxes are classed as "other".

2. [^](#)

I've written a [summary of the report](#) as part of [this sequence](#), if you're interested!

3. [^](#)

One way to think about this is as a distinction between "[inside view](#)" and "[outside view](#)" approaches (however see also [this post](#)). Cotra's bioanchors report takes an inside view, roughly based on the assumption that training compute is the biggest bottleneck to building TAI, and quantifying how much we'll need to be able to train a transformative model. Davidson's semi-informative priors report instead specifies very little about how AI development works, leaning more heavily on reference classes from similar technologies and a general Bayesian framework.

4. [^](#)

This is a variation of the [sunrise problem](#), which was the original problem that [Pierre-Simon Laplace](#) was trying to solve.

5. [^](#)

This is of a course a somewhat dubious assumption, and we'll come back to this later on.

6. [^](#)

Indeed, looking only at the base rate of successful first trials alone would have a big problem of sparsity – there's just not enough historical data!

7. [^](#)

We could also think about the number of virtual *trials* rather than virtual *successes*, but Davidson decides against this. Loosely speaking, if we use virtual trials, then it's not as easy to separate out the effects of the first-trial probability and the effects from observed failed trials ([more](#)).

8. [^](#)

The prior is defined using a [Beta distribution](#) parameterised by (1) the number of virtual successes, and (2) the inverse of the first-trial probability. See [here](#) for more information.

9. [^](#)

The “plausibility of the prior” focuses on the shape of the [Beta distribution](#), e.g. whether or not you should expect the probability density to be larger in the interval $[0, 1/1000]$ or $[1/1000, 2/1000]$. On the other hand, the “plausibility of the update” looks at your expected probability of building AGI next year should change given the outcomes of newly observed trials. For example (borrowing from the report), “If you initially thought the annual chance of developing AGI was 1/100, 50 years of failure is not that surprising and it should not reduce your estimate down as low as 1/600”.

10. [^](#)

This approach also applies to researcher-years and compute years, and is described more [here](#).

11. [^](#)

Incidentally, this is a claim that's central to another of [Open Philanthropy's Worldview Investigations](#), [Forecasting TAI with biological anchors](#), which [I've discussed in another post](#).

12. [^](#)

Note that this doesn't imply that there's an infinite probability of developing AGI in the first researcher-year of effort, because it's not true that we're starting from the “zero” level of AI technological development. Essentially, the regime start-time is *not* about “when the level AI technological development started increasing” – [see this footnote](#) for more on discussion.

13. [^](#)

For example, we would like our prediction for $P(\text{AGI within 10 years})$ to remain the same even if we use a trial definition of 1 month instead of 1 year. Although using a trial definition of 1 month would ordinarily lead to more total observed trials and thus more updating, this effect is cancelled out by choosing a different first-trial probability.

14. [^](#)

More concretely, suppose you think that several different updates rules (corresponding to e.g. different numbers of virtual successes) all seem reasonable, and you're uncertain what to do. One approach is to weight the results for the different choices of update rules, and use these rules to update the forecasts based on evidence. But we might also be interested in *updating how we weight the update rules*, which is where the hyper prior comes in ([more](#)).

15. [^](#)

These numbers were extracted using [WebPlotDigitizer](#).

16. [^](#)

Depending on your point of view, this may not be very compelling evidence – e.g. you might think that the ramp up to AGI would be extremely fast due to the discovery of a “[secret sauce](#)”.

17. [^](#)

You can also have a look at the [full report](#) if you want to get into the details!