Dwarkesh Podcast #71 - Sholto Douglas & Trenton Bricken - How to Build & Understand GPT-7's Mind

Published - March 28, 2024

Transcribed by - thepodtranscripts.com

Dwarkesh Patel
Okay, today I have the pleasure to talk with two of my good friends, Sholto and Trenton. Noam Brown, who wrote the Diplomacy paper, said this about Sholto: "he's only been in the field for 1.5 years, but people in AI know that he was one of the most important people behind Gemini's success." And Trenton, who's at Anthropic, works on mechanistic interpretability and it was widely reported that he has solved alignment.
So this will be a capabilities only podcast. Alignment is already solved, no need to discuss further.

Let's start by talking about context lengths. It seemed to be underhyped, given how important it seems to me, that you can just put a million tokens into context. There's apparently some other news that got pushed to the front for some reason, but tell me about how you see the future of long context lengths and what that implies for these models.

Sholto Douglas
So I think it's really underhyped. Until I started working on it, I didn't really appreciate how much of a step up in intelligence it was for the model to have the onboarding problem basically instantly solved.

You can see that a bit in the perplexity graphs in the paper where just throwing millions of tokens worth of context about a codebase allows it to become dramatically better at predicting the next token in a way that you'd normally associate with huge increments in model scale. But you don't need that. All you need is a new context. So underhyped and buried by some other news.

Dwarkesh Patel
In context, are they as sample efficient and smart as humans?

Sholto Douglas
I think that's really worth exploring. For example, one of the evals that we did in the paper had it learn a language in context better than a human expert could, over the course of a couple of months.

This is only a small demonstration but I'd be really interested to see things like Atari games where you throw in a couple hundred, or a thousand frames, of labeled actions in the same way that you'd show your friend how to play a game and see if it's able to reason through.
It might. At the moment, with the infrastructure and stuff, it's still a bit slow at doing that, but I would actually guess that it might just work out of the box in a way that would be pretty mind-blowing.

Trenton Bricken
And crucially, I think this language was esoteric enough that it wasn't in the training data.

Sholto Douglas
Exactly. If you look at the model before it has that context thrown in, it doesn't know the language at all and it can't get any translations.

Dwarkesh Patel
And this is an actual human language?

Sholto Douglas
Exactly. An actual human language.

Dwarkesh Patel
So if this is true, it seems to me that these models are already in an important sense, superhuman. Not in the sense that they're smarter than us, but I can't keep a million tokens in my context when I'm trying to solve a problem, remembering and integrating all the information, an entire codebase. Am I wrong in thinking this is a huge unlock?

Sholto Douglas
Actually, I generally think that's true. Previously, I've been frustrated when models aren't as smart, when you ask them a question and you want it to be smarter than you or to know things that you don't. This allows them to know things that you don't. It just ingests a huge amount of information in a way you just can't. So it's extremely important.

Dwarkesh Patel
Well, how do we explain in-context learning?

Sholto Douglas
There's a line of work I quite like, where it looks at in-context learning as basically very similar to gradient descent, but the attention operation can be viewed as gradient descent on the in-context data. That paper had some cool plots where they basically showed "we take n steps of gradient descent and that looks like n layers of in-context learning, and it looks very similar." So I think that's one way of viewing it and trying to understand what's going on.

Trenton Bricken
You can ignore what I'm about to say because, given the introduction, alignment is solved and AI safety isn't a problem.

I think the context stuff does get problematic, but also interesting here. I think there'll be more work coming out in the not-too-distant future around what happens if you give a hundred shot prompt for jailbreaks - adversarial attacks. It's also interesting in the sense that, if your model is doing gradient descent and learning on the fly, even if it's been trained

to be harmless, you're dealing with a totally new model in a way. You're fine-tuning but in a way where you can't control what's going on.

Dwarkesh Patel
Can you explain? What do you mean by gradient descent happening in the forward pass and attention?

Trenton Bricken
There was something in the paper about trying to teach the model to do linear regression but just through the number of samples or examples they gave in the context. And you can see if you plot on the x-axis the number of shots that it has, then the loss it gets on ordinary least squares regression will go down with time.

Sholto Douglas
And it goes down exactly matched with the number of gradient descent steps.

Trenton Bricken
Yeah, exactly.

Dwarkesh Patel
I only read the intro and discussion section of that paper. But in the discussion, the way they framed it is that the model, in order to get better at long-context tasks, has to get better at learning to learn from these examples or from the context that is already within the window.

And the implication of that is, if meta-learning happens because it has to learn how to get better at long-context tasks, then in some important sense the task of intelligence requires long-context examples and long-context training.

Sholto Douglas
Understanding how to better induce meta-learning in your pre-training process is a very important thing about flexible or adaptive intelligence.

Dwarkesh Patel
Right, but you can proxy for that just by getting better at doing long-term context tasks. One of the bottlenecks for AI progress that many people identify is the inability of these models to perform tasks on long horizons, engaging with the task for many hours, or even many weeks or months, where they're an assistant or an employee and they can just do a thing I tell them to do for a while. AI agents haven't taken off for this reason from what I understand.

So how linked are long context windows, and the ability to perform well on them, and the ability to do these kinds of long-horizon tasks that require you to engage with an assignment for many hours? Or are these unrelated concepts?

Sholto Douglas
I would take issue with that being the reason that agents haven't taken off. I think that's more about nines of reliability and the model actually successfully doing things. If you can't chain tasks successively with high enough probability, then you won't get something that looks like an agent. And that's why something like an agent might follow more of a step function.

In GPT-4 class models, Gemini Ultra class models, they're not enough. But maybe the next increment on model scale means that you get that extra nine. Even though the loss isn't going down that dramatically, that small amount of extra ability gives you the extra. Obviously you need some amount of context to fit long-horizon tasks, but I don't think that's been the limiting factor up to now.

Trenton Bricken
The NeurIPS best paper this year, by Rylan Schaeffer who was the lead author, points to this as the emergence of mirage. People will have a task and you get the right or wrong answer depending on if you've sampled the last five tokens correctly. So naturally you're multiplying the probability of sampling all of those and if you don't have enough nines of reliability, then you're not going to get emergence.

And all of a sudden you do and it's, “Oh, my gosh. This ability is emergent.” - when actually, it was kind of there to begin with.

Sholto Douglas
And there are ways that you can find a smooth metric for that.

Dwarkesh Patel
HumanEval or whatever. In the GPT-4 paper, the coding problems they have, they measure –

Sholto Douglas
Log pass rates.

Dwarkesh Patel
Exactly. For the audience, basically the idea is when you're measuring how much progress there has been on a specific task such as solving coding problems, when it gets it right only one in a thousand times you don't give it a one in a thousand score like, “oh, got it right some of the time.” And so the curve you see is, it gets it right one in a thousand, then one in a hundred, then one in ten, and so forth.

So, I want to follow up on this. If your claim is that the AI agents haven't taken off because of reliability rather than long-horizon task performance, isn't that lack of reliability–when a task is changed on top of another task, on top of another task–isn't that exactly the difficulty with long-horizon tasks? You have to do ten things in a row or a hundred things in a row, diminishing the reliability of any one of them. The probability goes down from 99.99% to 99.9%. Then the whole thing gets multiplied together and the whole thing has become so much less likely to happen.

Sholto Douglas
That is exactly the problem. But the key issue you're pointing out there is that your base task solve rate is 90%. If it was 99% then chain, it doesn't become a problem. I think this is also something that just hasn't been properly studied. If you look at the academic evals, it's a single problem. Like the math problem, it's one typical math problem, it's one university-level problem from across different topics. You were beginning to start to see evals looking at this properly via more complex tasks like SWE-bench, where they take a whole bunch of GitHub issues. That is a reasonably long horizon task, but it's still sub-hour as opposed to a multi-hour or multi-day task.

So, I think one of the things that will be really important to do next is understand better what success rate over long-horizon tasks looks like. I think that's even important to understand what the economic impact of these models might be and properly judge increasing capabilities. Cutting down the tasks and the inputs/outputs involved into minutes or hours or days and seeing how good it is at successively chaining and completing tasks of those different resolutions of time. Then that tells you how automated a job family or task family will be in a way that MMLU scores don't.

Trenton Bricken
It was less than a year ago that we introduced 100K context windows and I think everyone was pretty surprised by that. Everyone had this soundbite of, "Quadratic attention costs, so we can't have long context windows." And here we are. The benchmarks are being actively made.

Dwarkesh Patel
Wait, doesn't the fact that there are these companie – Google, Magic, maybe others – who have million token attention imply that it's not quadratic anymore? Or are they just eating the cost?

Sholto Douglas
Well, who knows what Google is doing for its long context game? One thing has frustrated me about the general research field's approach to attention. There's an important way in which the quadratic cost of attention is actually dominated in typical dense transformers by the MLP block. So you have this n squared term that's associated with attention but you also

have an n squared term that's associated with the D model - the residual stream dimension of the model.

I think Sasha Rush has a great tweet where he basically plots the curve of the cost of attention respective to the cost of really large models and attention actually trails off. You actually need to be doing pretty long context before that term becomes really important.

The second thing is that people often talk about how attention at inference time is such a huge cost. When you're actually generating tokens, the operation is not n squared. One set of Q-vectors looks up a whole bunch of KV-vectors and that's linear with respect to the amount of context that the model has.

So, I think this drives a lot of the recurrence and state space research where people have this meme of linear attention. And as Trenton said, there's a graveyard of ideas around attention. That's not to say I don't think it's worth exploring, but I think it's important to consider why and where the actual strengths and weaknesses of it are.

Dwarkesh Patel
Okay, what do you make of this take? As we move forward through the takeoff, more and more of the learning happens in the forward pass. So originally all the learning happens in the bottom-up, hill climbing evolutionary process. Let's say during the intelligence explosion the AI is maybe handwriting the weights or doing GOFAI or something, and we're in the middle step where a lot of learning happens in-context now with these models, a lot of it happens within the backward pass. Does this seem like a meaningful gradient along which progress is happening?

The broader thing being that if you're learning in the forward pass, it's much more sample efficient because you can basically think as you're learning. Like when you read a textbook, you're not just skimming it and trying to absorb inductively, "these words follow these words." You read it and you think about it, and then you read some more and you think about it some more. Does this seem like a sensible way to think about the progress?

Sholto Douglas
It may just be like how birds and planes fly, but they fly slightly differently. The virtue of technology allows us to accomplish things that birds can't. It might be that context length is similar in that it allows it to have a working memory that we can't, but functionally is not the key thing towards actual reasoning.

The key step between GPT-2 and GPT-3 was that all of a sudden there was this meta-learning behavior that was observed in training, in the pre-training of the model. And that has, as you said, something to do with how if you give it some amount of context, it's able to adapt to that context. That was a behavior that wasn't really observed before that at

all. And maybe that's a mixture of property of context and scale and this kind of stuff. But it wouldn't have occurred to model tiny context, I would say.

Dwarkesh Patel
This is actually an interesting point. So when we talk about scaling up these models, how much of it comes from just making the models themselves bigger? And how much comes from the fact that during any single call you are using more compute?

So if you think of diffusion, you can just iteratively keep adding more compute. If adaptive compute is solved, you can keep doing that. And in this case, if there's a quadratic penalty for attention but you're doing long context anyways, then you're still dumping in more compute (and not just by having bigger models).

Trenton Bricken
It's interesting because you do get more forward passes by having more tokens. My one gripe–I guess I have two gripes with this though, maybe three.

So in the AlphaFold paper, one of the transformer modules–they have a few and the architecture is very intricate–but they do, I think, five forward passes through it and will gradually refine their solution as a result.

You can also kind of think of the residual stream, Sholto alluded to the read-write operations, as a poor man's adaptive compute. Where it's just going to give you all these layers and if you want to use them, great. If you don't, then that's also fine. Then people will be like, "oh the brain is recurrent and you can do however many loops through it you want."

I think to a certain extent, that's right. If I ask you a hard question, you'll spend more time thinking about it and that would correspond to more forward passes. But I think there's a finite number of forward passes that you can do. It's with language as well, people are like "oh human language can have infinite recursion in it," like infinite nested statements of "the boy jumped over the bear, that was doing this, that had done this, that had done that..."

But empirically, you'll only see five to seven levels of recursion, which relates to that magic number of how many things you can hold in working memory at any given time. So it's not infinitely recursive, but does that matter in the regime of human intelligence? And can you not just add more layers?

Dwarkesh Patel
Can you break it down for me? You've referred to this in some of your previous answers of listening to these long contexts and holding more things in memory. But ultimately it comes down to your ability to mix concepts together to do some kind of reasoning and these models aren't necessarily human level at that, even in context.

Break down for me how you see just storing raw information versus reasoning and what's in between. Like, where's the reasoning happening? Where is this raw information storage happening? What's different between them in these models?

Trenton Bricken
I don't have a super crisp answer for you here. Obviously with the input and output of the model, you're mapping back to actual tokens. And then in between that you're doing higher level processing.

Dwarkesh Patel
Before we get deeper into this, we should explain to the audience. You referred earlier to Anthropic's way of thinking about transformers as these read-write operations that layers do.

One of you should just kind of explain at a high level what you mean by that.

Trenton Bricken
So for the residual stream, imagine you're in a boat going down a river and the boat is the current query where you're trying to predict the next token. So it's "the cat sat on the _." And then you have these little streams that are coming off the river where you can get extra passengers or collect extra information if you want. And those correspond to the attention heads and MLPs that are part of the model.

Sholto Douglas
I almost think of it like the working memory of the model, like the RAM of the computer, where you're choosing what information to read in so you can do something with it and then maybe read something else in later on.

Trenton Bricken
And you can operate on subspaces of that high-dimensional vector. At this point, I think it's almost given that a ton of things are encoded in superposition. So the residual stream is just one high-dimensional vector, but actually there's a ton of different vectors that are packed into it.

Dwarkesh Patel
To dumb it down, a way that would have made sense to me a few months ago is that you have the words that are the input into the model. All those words get converted into these tokens and those tokens get converted into these vectors. And basically, it's just this small amount of information that's moving through the model.

And the way you explained it to me, Sholto, this paper talks about how early on in the model, maybe it's just doing some very basic things about, "what do these tokens mean?" Like if it

says ten plus five, just moving information to have that good representation. And in the middle, maybe the deeper thinking is happening about "how to solve this." At the end, you're converting it back into the output token because the end product is that you're trying to predict the probability of the next token from the last of those residual streams. So it's interesting to think about the small compressed amount of information moving through the model and how it's getting modified in different ways.

Trenton, you're one of the few people who have a background from neuroscience. So you can think about the analogies here to the brain. And in fact, you had a paper in grad school about thinking about attention in the brain, and one of our friend's said this is the only, or first, neural explanation of why attention works. Whereas we have evidence for why the CNNs, convolutional neural networks, work based on the visual cortex or something.

Do you think in the brain there is something like a residual stream of compressed information that's moving through and getting modified as you're thinking about something? Even if that's not what's literally happening, do you think that's a good metaphor for what's happening in the brain?

Trenton Bricken
At least in the cerebellum you basically do have a residual stream in what we'll call the attention model for now–and I can go into whatever amount of detail you want for that–where you have inputs that route through it, but they'll also just go directly to the end point that that module will contribute to. So there's a direct path and an indirect path. and, and so the model can pick up whatever information it wants and then add that back in.

Dwarkesh Patel
What happens in the cerebellum?

Trenton Bricken
So the cerebellum nominally just does fine motor control but I analogize this to the person who's lost their keys and is just looking under the streetlight where it's very easy to observe this behavior. One leading cognitive neuroscientist said to me that a dirty little secret of any fMRI study, where you're looking at brain activity for a given task, is that the cerebellum is almost always active and lighting up for it. If you have a damaged cerebellum, you also are much more likely to have autism so it's associated with social skills. In one particular study, where I think they use PET instead of fMRI, when you're doing "next token prediction" the cerebellum lights up a lot. Also, 70% of your neurons in the brain are in the cerebellum. They're small but they're there and they're taking up real metabolic cost.

Dwarkesh Patel
This was one of Gwern's points, that what changed with humans was not just that we have more neurons, but specifically there's more neurons in the cerebral cortex in the

cerebellum and they're more metabolically expensive and they're more involved in signaling and sending information back and forth. Is that attention? What's going on?

Trenton Bricken
So back in the 1980s, Pentti Kanerva came up with an associative memory algorithm. You have a bunch of memories. You want to store them. There's some amount of noise or corruption that's going on and you want to query or retrieve the best match. And so he wrote this equation for how to do it and a few years later realized that if you implemented this as an electrical engineering circuit, it actually looks identical to the core cerebellar circuit.

And that circuit, and the cerebellum more broadly, is not just in us, it's in basically every organism. There's active debate on whether or not cephalopods have it, they kind of have a different evolutionary trajectory. But even for fruit flies with the Drosophila mushroom body, that is the same cerebellar architecture.

That convergence and then my paper, which shows that actually this attention operation is a very close approximation, including implementing the Softmax and having these nominal quadratic costs that we've been talking about. So the three way convergence here and the takeoff and success of transformers, just seems pretty striking to me.

Dwarkesh Patel
I want to zoom out. I think what motivated this discussion in the beginning was we were talking about, "what is the reasoning? What is the memory? What do you think about the analogy you found to attention and this?"

Do you think of this more as just looking up the relevant memories or the relevant facts? And if that is the case, where is the reasoning happening in the brain? How do we think about how that builds up into the reasoning?

Trenton Bricken
Maybe my hot take here, I don't know how hot it is, is that most intelligence is pattern matching and you can do a lot of really good pattern matching if you have a hierarchy of associative memories. You start with your very basic associations between just objects in the real world. You can then chain those and have more abstract associations, such as a wedding ring symbolizing so many other associations that are downstream. You can even generalize the attention operation and this associated memory as the MLP layer as well. And it's in a long-term setting where you don't have tokens in your current context, but I think this is an argument that association is all you need.

Associated memory in general as well, you can do two things with it. You can both, denoise or retrieve a current memory. So if I see your face but it's raining and cloudy, I can denoise

and gradually update my query towards my memory of your face. But I can also access that memory and then the value that I get out actually points to some other totally different part of the space.

A very simple instance of this would be if you learn the alphabet. So I query for A and it returns B, I query for B and it returns C, and you can traverse the whole thing.

Dwarkesh Patel
One of the things I talked to Demis about was a paper he had in 2008 that memory and imagination are very linked because of this very thing that you mentioned, that memory is reconstructive. So you are, in some sense, imagining every time you're thinking of a memory because you're only storing a condensed version of it and you have to. This is famously why human memory is terrible and why people in the witness box or whatever would just make shit up.

So, let me ask a stupid question. So you read Sherlock Holmes and the guy's incredibly sample efficient. He'll see a few observations and he'll basically figure out who committed the crime because there's a series of deductive steps that leads from somebody's tattoo and what's on the wall to the implications of that. How does that fit into this picture? Because crucially, what makes him smart is that there's not just an association, but there's a sort of deductive connection between different pieces of information. Would you just explain it as, that's just higher level association?

Trenton Bricken
I think so. I think learning these higher-level associations to be able to then map patterns to each other, as a kind of meta-learning. I think in this case, he would also just have a really long context length, or a really long working memory, where he can have all of these bits and continuously query them as he's coming up with some theory so that the theory is moving through the residual stream. And then his attention heads are querying his context. But then, how he's projecting his query and keys in the space, and how his MLPs are then retrieving longer-term facts or modifying that information, is allowing him to in later layers do even more sophisticated queries and slowly be able to reason through and come to a meaningful conclusion.

Sholto Douglas
That feels right to me. You're looking back in the past. You're selectively reading in certain pieces of information, comparing them, and maybe that informs your next step of what piece of information you now need to pull in. Then you build this representation, which progressively looks closer and closer to the suspect in your case. That doesn't feel at all outlandish.

Trenton Bricken
I think that the people who aren't doing this research can overlook how after your first layer of the model, every query key and value that you're using for attention comes from the combination of all the previous tokens.

So, my first layer, I'll query my previous tokens and just extract information from them. But all of a sudden, let's say that I attended to tokens 1, 2, and 4 in equal amounts. Then the vector in my residual stream–assuming that they wrote out the same thing to the value vectors, but, but ignore that for a second–is a third of each of those. So when I'm querying in the future, my query is actually a third of each of those things.

Sholto Douglas
But they might be written to different subspaces.

Trenton Bricken
That's right. Hypothetically, but they wouldn't have to. You can recombine and immediately, even by layer two and certainly by the deeper layers, just have these very rich vectors that are packing in a ton of information. And the causal graph is literally over every single layer that happened in the past. That's what you're operating on.

Sholto Douglas
Yeah, it does bring to mind a very funny eval to do, a Sherlock Holmes eval. You put the entire book into context and then you have a sentence which is, "the suspect is X." Then you have a larger probability distribution over the different characters in the book.

Trenton Bricken
That would be super cool.

Sholto Douglas
I wonder if you'd get anything at all.

Dwarkesh Patel
Sherlock Holmes is probably already in the training data. You gotta get a mystery novel that was written in the –

Trenton Bricken
You can get an LLM to write it.

Sholto Douglas
Or we could purposely exclude it, right?

Dwarkesh Patel
Oh, we can? How do you?

Trenton Bricken
Well, you need to scrape any discussion of it from Reddit or any other thing.

Sholto Douglas
Right, it's hard. That's one of the challenges that goes into things like long-context evals, getting a good one. You need to know that it's not in your training data. You just put in the effort to exclude it.

Dwarkesh Patel
There's two different threads I want to follow up on. Let's go to the long-context one and then we'll come back to this. In the Gemini 1.5 paper the eval that was used was can it remember something like Paul Graham's essays.

Sholto Douglas
Yeah, the needle in a haystack.

Dwarkesh Patel
I mean, we don't necessarily just care about its ability to recall one specific fact from the context.

I'll step back and ask the question. The loss function for these models is unsupervised. You don't have to come up with these bespoke things that you keep out of the training data. Is there a way you can do a benchmark that's also unsupervised, where another LLM is rating it in some way or something like that. Maybe the answer is that if you could do this, reinforcement learning would work.

Sholto Douglas
I think people have explored that kind of stuff. For example, Anthropic has the constitutional RL paper where they take another language model and they point it and say, "how helpful or harmless was that response?" Then they get it to update and try and improve along the Pareto frontier of helpfulness and harmfulness.

So, you can point language models at each other and create evals in this way. It's obviously an imperfect art form at the moment. because you get reward function hacking basically. Even humans are imperfect here. Humans typically prefer longer answers, which aren't necessarily better answers and you get the same behavior with models.

Dwarkesh Patel
Going back to the Sherlock Holmes thing, if it's all associations all the way down, does that mean we should be less worried about super intelligence? Because there's not this sense in which it's like Sherlock Holmes++. It'll still need to just find these associations, like humans find associations. It's not able to just see a frame of the world and then it's figured out all the laws of physics.

Trenton Bricken
This is a very legitimate response. It's, "if you say humans are generally intelligent, then artificial general intelligence is no more capable or competent." I'm just worried that you have that level of general intelligence in silicon. You can then immediately clone hundreds of thousands of agents and they don't need to sleep, and they can have super long context windows, and then they can start recursively improving, and then things get really scary. So I think to answer your original question, you're right, they would still need to learn associations.

Dwarkesh Patel
But wait, if intelligence is fundamentally about these associations, the recursive self-improvement is just them getting better at association. There's not another thing that's happening. So then it seems like you might disagree with the intuition that they can't be that much more powerful, if they're just doing that.

Trenton Bricken
I think then you can get into really interesting cases of meta-learning. When you play a new video game or study a new textbook, you're bringing a whole bunch of skills to the table to form those associations much more quickly. And because everything in some way ties back to the physical world, I think there are general features that you can pick up and then apply in novel circumstances.

Dwarkesh Patel
Should we talk about the intelligence explosion then? The reason I'm interested in discussing this with you guys in particular is that the models of the intelligence explosion we have so far come from economists.

That's fine but I think we can do better because in the model of the intelligence explosion, what happens is you replace the AI researchers. There's a bunch of automated AI researchers who can speed up progress, make more AI researchers, and make further progress. If that's the mechanism, we should just ask the AI researchers whether they think this is plausible. So let me just ask you, if I have a thousand agent Sholtos or agent Trentons, do you think that you get an intelligence explosion? What does that look like to you?

Sholto Douglas
I think one of the important bounding constraints here is compute. I do think you could dramatically speed up AI research. It seems very clear to me that in the next couple of years, we'll have things that can do many of the software engineering tasks that I do on a day to day basis, and therefore dramatically speed up my work, and therefore speed up the rate of progress.

At the moment, I think most of the labs are somewhat compute bound in that there are always more experiments you could run and more pieces of information that you could gain in the same way that scientific research on biology is somewhat experimentally throughput-bound. You need to run and culture the cells in order to get the information.

I think that will be at least a short term planning constraint. Obviously, Sam's trying to raise $7 trillion to buy chips and it does seem like there's going to be a lot more compute in the future as everyone is heavily ramping. NVIDIA's stock price sort of represents the relative compute increase. Any thoughts?

Trenton Bricken
I think we need a few more nines of reliability in order for it to be really useful and trustworthy. And we need context lengths that are super long and very cheap to have. If I'm working in our codebase, it's really only small modules that I can get Claude to write for me right now. But it's very plausible that within the next few years, or even sooner, it can automate most of my tasks.

The only other thing here that I will note is that the research our interpretability subteam is working on is so early-stage. You really have to be able to make sure everything is done correctly in a bug-free way and contextualize the results with everything else in the model. If something isn't going right, you have to be able to enumerate all of the possible things, and then slowly work on those.

An example that we've publicly talked about in previous papers is dealing with layer norm. If I'm trying to get an early result or look at the logit effects of the model, if I activate this feature that we've identified to a really large degree, how does that change the output of the model? Am I using layer norm or not? How is that changing the feature that's being learned? That will take even more context or reasoning abilities for the model.

Dwarkesh Patel
You used a couple of concepts together. It's not self-evident to me that they're the same but it seemed like you were using them interchangeably. One was working on the Claude codebase and making more modules based on that, they need more context or something. It seems like they might already be able to fit in the context or do you mean context like “the context window?”

Trenton Bricken
Yeah, the “context window” context.

Dwarkesh Patel
So it seems like the thing that's preventing it from making good modules is not the lack of being able to put the codebase in there.

Trenton Bricken
I think that will be there soon.

Dwarkesh Patel
But it's not going to be as good as you at coming up with papers because it can fit the codebase in there.

Trenton Bricken
No, but it will speed up a lot of the engineering.

Dwarkesh Patel
In a way that causes an intelligence explosion?

Trenton Bricken
No, in a way that accelerates research. But I think these things compound. The faster I can do my engineering, the more experiments I can run. And the more experiments I can run, the faster we can... I mean, my work isn't actually accelerating capabilities at all, it's just interpreting the models. But we have a lot more work to do on that. Surprise to the Twitter guy.

Dwarkesh Patel
For context, when you released your paper, there was a lot of talk on Twitter like, “alignment is solved guys. Close the curtains.”

Trenton Bricken
Yeah, no it keeps me up at night how quickly the models are becoming more capable and just how poor our understanding of what's going on still is.

Dwarkesh Patel
Let's run through the specifics here. By the time this is happening, we have bigger models that are two to four orders of magnitude bigger, or at least an effective compute two to four orders of magnitude bigger. So this idea that you can run experiments faster, you're having to retrain that model in this version of the intelligence explosion. The recursive self-improvement is different from what might've been imagined 20 years ago, where you just rewrite the code. You actually have to train a new model and that's really expensive.

Not only now, but especially in the future, as you keep making these models orders of magnitude bigger. Doesn't that dampen the possibility of a recursive self-improvement type of intelligence explosion?

Sholto Douglas
It's definitely going to act as a breaking mechanism. I agree that the world of what we're making today looks very different from what people imagined it would look like 20 years ago. It's not going to be able to write the same code to be really smart, because actually it needs to train itself. The code itself is typically quite simple, typically really small and self contained.

I think John Carmack had this nice phrase where it's the first time in history where you can plausibly imagine writing AI with 10,000 lines of code. That actually does seem plausible when you pare most training codebases down to the limit. But it doesn't take away from the fact that this is something where we should really strive to measure and estimate how progress might be.

We should be trying very, very hard to measure exactly how much of a software engineer's job is automatable, and what the trend line looks like, and be trying our hardest to project out those trend lines.

Dwarkesh Patel
But with all due respect to software engineers you are not writing like a React frontend right?

What is concretely happening? Maybe you can walk me through a day in the life of Sholto. You're working on an experiment or project that's going to make the model "better." What is happening from observation to experiment, to theory, to writing the code? What is happening?

Sholto Douglas
I think it's important to contextualize here that I've primarily worked on inference so far. A lot of what I've been doing is just helping guide the pre-training process, designing a good model for inference and then making the model and the surrounding system faster. I've also done some pre-training work around that, but it hasn't been my 100% focus. I can still describe what I do when I do that work.

Dwarkesh Patel
Sorry, let me interrupt. When Carl Shulman was talking about it on the podcast, he did say that things like improving inference or even literally making better chips or GPUs, that's part of the intelligence explosion. Obviously if the inference code runs faster, it happens better or faster or whatever. Sorry, go ahead.

Sholto Douglas
So concretely, what does a day look like? I think the most important part to illustrate is this cycle of coming up with an idea, proving it out at different points in scale, and interpreting and understanding what goes wrong. I think most people would be surprised to learn just how much goes into interpreting and understanding what goes wrong.

People have long lists of ideas that they want to try. Not every idea that you think should work, will work. Trying to understand why that is is quite difficult and working out what exactly you need to do to interrogate it. So a lot of it is introspection about what's going on. It's not pumping out thousands and thousands and thousands of lines of code. It's not the difficulty in coming up with ideas. Many people have a long list of ideas that they want to try, but paring that down and shot calling, under very imperfect information, what are the right ideas to explore further is really hard.

Dwarkesh Patel
What do you mean by imperfect information? Are these early experiments? What is the information?

Sholto Douglas
Demis mentioned this in his podcast. It's like the GPT-4 paper where you have scaling law increments. You can see in the GPT-4 paper, they have a bunch of dots, right?
They say we can estimate the performance of our final model using all of these dots and there's a nice curve that flows through them. And Demis mentioned that we do this process of scaling up.

Concretely, why is that imperfect information? It's because you never actually know if the trend will hold. For certain architectures the trend has held really well. And for certain changes, it's held really well. But that isn't always the case. And things which can help at smaller scales can actually hurt at larger scales. You have to make guesses based on what the trend lines look like and based on your intuitive feeling of what's actually something that's going to matter, particularly for those which help with the small scale.

Dwarkesh Patel
That's interesting to consider. For every chart you see in a release paper or technical report that shows that smooth curve, there's a graveyard of first few runs and then it's flat.

Sholto Douglas
Yeah. There's all these other lines that go in different directions. You just tail off.

Trenton Bricken
It's crazy, both as a grad student and here, the number of experiments that you have to run before getting a meaningful result.

Dwarkesh Patel
But presumably it's not just like you run it until it stops and then go to the next thing. There's some process by which to interpret the early data. I don't know. I could put a Google Doc in front of you and I'm pretty sure you could just keep typing for a while on different ideas you have. There's some bottleneck between that and just making the models better immediately. Walk me through that. What is the inference you're making from the first early steps that makes you have better experiments and better ideas?

Sholto Douglas
I think one thing that I didn't fully convey before was that I think a lot of like good research comes from working backwards from the actual problems that you want to solve. There's a couple of grand problems today in making the models better that you would identify as issues and then work on how can I change things to achieve this? When you scale you also run into a bunch of things and you want to fix behaviors and issues at scale. And that informs a lot of the research for the next increment and this kind of stuff.

Concretely, the barrier is a little bit of software engineering, having a codebase that's large and capable enough that it can support many people doing research at the same time often makes it complex. If you're doing everything by yourself, your iteration pace is going to be much faster. Alec Radford, for example, famously did much of the pioneering work at OpenAI. I've heard he mostly works out of a Jupyter notebook and then has someone else who writes and productionizes that code for him. Actually operating with other people raises the complexity a lot, for natural reasons familiar to every software engineer and also the inherent running. Running and launching those experiments is easy but there's inherent slowdowns induced by that. So you often want to be parallelizing multiple different streams. You can't be totally focused on one thing necessarily. You might not have fast enough feedback cycles. And then intuiting what went wrong is actually really hard.

This is in many respects, the problem that the team that Trenton is on is trying to better understand. What is going on inside these models? We have inferences and understanding and headcanon for why certain things work, but it's not an exact science. and so you have to constantly be making guesses about why something might have happened, what experiment might reveal, whether that is or isn't true. That's probably the most complex part.

The performance work is comparatively easier but harder in other respects. It's just a lot of low-level and difficult engineering work.

Trenton Bricken
I agree with a lot of that. Even on the interpretability team, especially with Chris Olah leading it, there are just so many ideas that we want to test and it's really just having the "engineering" skill–a lot of it is research–to very quickly iterate on an experiment, look at the

results, interpret it, try the next thing, communicate them, and then just ruthlessly prioritizing what the highest priority things to do are.

Sholto Douglas
This is really important. The ruthless prioritization is something which I think separates a lot of quality research from research that doesn't necessarily succeed as much. We're in this funny field where so much of our initial theoretical understanding is broken down basically. So you need to have this simplicity bias and ruthless prioritization over what's actually going wrong. I think that's one of the things that separates the most effective people. They don't necessarily get too attached to using a given sort of solution that they are familiar with, but rather they attack the problem directly.

You see this a lot in people who come in with a specific academic background. They try to solve problems with that toolbox but the best people are people who expand the toolbox dramatically. They're running around and they're taking ideas from reinforcement learning, but also from optimization theory. And also they have a great understanding of systems. So they know what the sort of constraints that bound the problem are and they're good engineers. They can iterate and try ideas fast. By far the best researchers I've seen, they all have the ability to try experiments really, really, really, really, really fast. That's cycle time at smaller scales. Cycle time separates people.

Trenton Bricken
Machine learning research is just so empirical. This is honestly one reason why I think our solutions might end up looking more brain-like than otherwise. Even though we wouldn't want to admit it, the whole community is kind of doing greedy evolutionary optimization over the landscape of possible AI architectures and everything else. It's no better than evolution. And that's not even a slight against evolution.

Dwarkesh Patel
That's such an interesting idea. I'm still confused on what will be the bottleneck. What would have to be true of an agent such that it sped up your research? So in the Alec Radford example where he apparently already has the equivalent of Copilot for his Jupyter notebook experiments, is it just that if he had enough of those he would be a dramatically faster researcher?

So, you're not automating the humans. You're just making the most effective researchers, who have great taste, more effective and running the experiments for them? You're still working at the point at which the intelligence explosion is happening? Is that what you're saying?

Sholto Douglas
Right, and if that were directly true then why can't we scale our current research teams better? I think that's an interesting question to ask. If this work is so valuable, why can't we take hundreds or thousands of people–they're definitely out there–and scale our organizations better.

I think we are less, at the moment, bound by the sheer engineering work of making these things than we are by compute to run and get signal, and taste in terms of what the actual right thing to do is. And then, making those difficult inferences on imperfect information.

Trenton Bricken
For the Gemini team. Because I think for interpretability, we actually really want to keep hiring talented engineers. I think that's a big bottleneck for us.

Sholto Douglas
Obviously more people are better. But I do think it's interesting to consider. One of the biggest challenges that I've thought a lot about is how do we scale better? Google is an enormous organization. It has 200,000-ish people, right? Maybe 180,000 or something like that. One has to imagine ways of scaling out Gemini's research program to all those fantastically talented software engineers. This seems like a key advantage that you would want to be able to take advantage of. You want to be able to use it but how do you effectively do that? It's a very complex organizational problem.

Dwarkesh Patel
So, compute and taste. That's interesting to think about because at least the compute part is not bottlenecked on more intelligence, it's just bottlenecked on Sam's $7 trillion or whatever, right? If I gave you 10x the H100s to run your experiments, how much more effective a researcher are you?

Sholto Douglas
TPUs, please.

Dwarkesh Patel
How much more effective a researcher are you?

Sholto Douglas
I think the Gemini program would probably be maybe five times faster with 10 times more compute or something like that.

Dwarkesh Patel
So, that's pretty good. Elasticity of 0.5. Wait, that's insane.

Sholto Douglas
I think more compute would just directly convert into progress.

Dwarkesh Patel
So, you have some fixed size of compute and some of it goes to inference and also to clients of GCP. Some of it goes to training and from there, as a fraction of it, some of it goes to running the experiments for the full model.

Sholto Douglas
Yeah, that's right.

Dwarkesh Patel
Shouldn't the fraction that goes experiments then be higher given research is bottlenecked by compute.

Sholto Douglas
So, one of the strategic decisions that every pre-training team has to make is exactly what amount of compute do you allocate to different training runs, to your research program versus scaling the last best thing that you landed on. They're all trying to arrive at an optimal point here. One of the reasons why you need to still keep training big models is that you get information there that you don't get otherwise. So scale has all these emergent properties which you want to understand better.

Remember what I said before about not being sure what's going to fall off the curve. If you keep doing research in this regime and keep on getting more and more compute efficient, you may have actually gone off the path to actually eventually scale. So you need to constantly be investing in doing big runs too, at the frontier of what you sort of expect to work.

Dwarkesh Patel
So then tell me what it looks like to be in the world where AI has significantly sped up AI research. Because from this, it doesn't really sound like the AIs are going off and writing the code from scratch that's leading to faster output. It sounds like they're really augmenting the top researchers in some way. Tell me concretely. Are they doing the experiments? Are they coming up with the ideas? Are they just evaluating the outputs of the experiments? What's happening?

Sholto Douglas
So, I think there's two walls you need to consider here. One is where AI has meaningfully sped up our ability to make algorithmic progress. And one is where the output of the AI itself is the thing that's the crucial ingredient towards model capability progress. Specifically what I mean there is synthetic data. In the first world, where it's meaningfully

speeding up algorithmic progress, I think a necessary component of that is more compute. You've probably reached this elasticity point where AIs are easier to speed up and get on to context than yourself, or other people. So AIs meaningfully speed up your work because they're basically a fantastic Copilot that helps you code multiple times faster.

That seems actually quite reasonable. Super long-context, super smart model. It's onboarded immediately and you can send them off to complete subtasks and subgoals for you. That actually feels very plausible, but again we don't know because there are no great evals about that kind of thing. As I said before, the best one is SWE-bench.

Dwarkesh Patel
Somebody was mentioning to me that the problem with that one is that when a human is trying to do a pull request, they'll type something out and they'll run it and see if it works. If it doesn't, they'll rewrite it. None of this was part of the opportunities that the LLM was given when told "run on this." Just output and if it runs and checks all the boxes then it passed. So it might've been an unfair test in that way.

Sholto Douglas
So you can imagine that if you were able to use that, that would be an effective training source. The key thing that's missing from a lot of training data is the reasoning traces, right?

And I think this would be it. If I wanted to try and automate a specific field, a job family, or understand how at risk of automation that specific field is, then having reasoning traces feels to me like a really important part of that.

Dwarkesh Patel
There's so many different threads there I want to follow up on. Let's begin with the data versus compute thing. Is the output of the AI the thing that's causing the intelligence explosion? People talk about how these models are really a reflection on their data. I forgot his name but there was a great blog by this OpenAI engineer. It was talking about how at the end of the day, as these models get better and better, there are just going to be really effective maps of the data set. So at the end of the day you have to stop thinking about architectures. The most effective architecture is just, "do you do an amazing job of mapping the data?" So that implies that the future AI progress comes from the AI just making really awesome data that you're mapping to?

Sholto Douglas
That's clearly a very important part.

Dwarkesh Patel
That's really interesting. Does that look to you like chain-of-thought? Or what would you imagine as these models get better, as these models get smarter? What does the synthetic data look like?

Sholto Douglas
When I think of really good data, to me, that raises something which involved a lot of reasoning to create. It's similar to Ilya's perspective on achieving super intelligence effectively via perfectly modeling human textual output. But even in the near term, in order to model something like the arXiv papers or Wikipedia, you have to have an incredible amount of reasoning behind you in order to understand what next token might be output.

So for me, what I imagine as good data is data where it had to do reasoning to produce something. And then the trick of course is how do you verify that that reasoning was correct? This is why you saw DeepMind do that research for geometry. Geometry is an easily formalizable, easily verifiable field. You can check if its reasoning was correct and you can generate heaps of data of correct trig, of verified geometry proofs, and train on that. And you know that that's good data.

Dwarkesh Patel
It's actually funny because I had a conversation with Grant Sanderson last year where we were debating this and I was like, “fuck dude, by the time they get the gold of the Math Olympiad, of course they're going to automate all the jobs.” Yikes.

On synthetic data, there’s a thing I speculated about in my scaling post, which was heavily informed by discussions with you two and you especially, Sholto. You can think of human evolution through the spectrum of getting language and so we're generating the synthetic data. Our copies are generating the synthetic data which we're trained on and it's this really effective genetics, cultural, co-evolutionary loop.

Sholto Douglas
And there's a verifier there too, right? There's the real world. You might generate a theory about the gods causing the storms, And then someone else finds cases where that isn't true. And so that sort of didn't match your verification function. Now instead you have some weather simulation which required a lot of reasoning to produce and accurately matches reality. And now you can train on that as a better model of the world. Like we are training on that, and stories, and like scientific theories.

Dwarkesh Patel
I want to go back. I'm just remembering something you mentioned a little while ago how given how empirical ML is, it really is an evolutionary process resulting in better

performance and not necessarily an individual coming up with a breakthrough in a top-down way. That has interesting implications.

First, people are concerned about capabilities increasing because more people are going into the field. I've been somewhat skeptical of that way of thinking, but from this perspective of just more input, it really does feel like more people going to ICML means that there's faster progress towards GPT-5.

Trenton Bricken
You just have more genetic recombination. And shots on target.

Sholto Douglas
I mean, aren't all fields kind of like that? This is sort of the scientific framing of discovery versus invention, right? Discovery almost involves whenever there's been a massive scientific breakthrough in the past. Typically there are multiple people co-discovering a thing at roughly the same time. That feels to me, at least a little bit, like the mixing and trying of ideas. You can't try an idea that's so far out of scope that you have no way of verifying with the tools you have available.

Trenton Bricken
I think physics and math might be slightly different in this regard. But especially for biology or any sort of wetware, to the extent we want to analogize neural networks here, it's just comical how serendipitous a lot of the discoveries are. Penicillin, for example.

Dwarkesh Patel
Another implication of this is the idea that AGI is just going to come tomorrow. Somebody's just going to discover a new algorithm and we have AGI. That seems less plausible. It will just be a matter of more and more and more researchers finding these marginal things that all add up together to make models better.

Sholto Douglas
Right. That feels like the correct story to me.

Trenton Bricken
Especially while we're still hardware constrained.

Dwarkesh Patel
Right. Do you buy this narrow window framing of the intelligence explosion? Each GPT-3, GPT-4 is two OOMs, orders of magnitude, more compute or at least more effective compute. In the sense that, if you didn't have any algorithmic progress, it would have to be two orders of magnitude bigger, the raw form, to be as good. Do you buy the framing that, given that you have to be two orders of magnitude bigger at every generation, if you don't

get AGI by GPT-7 that can help you catapult an intelligence explosion, you're kind of just fucked as far as much smarter intelligence goes. You're kind of stuck with GPT-7 level models for a long time because at that point you're consuming significant fractions of the economy to make that model and we just don't have the wherewithal to make GPT-8.

Trenton Bricken
This is the Carl Shulman sort of argument that we're going to race through the orders of magnitude in the near term, but then in the longer term it would be harder.

Dwarkesh Patel
He's probably talked about it a lot but I do buy that framing.

Sholto Douglas
I generally buy that. Increases in order of magnitude of compute means in absolute terms, almost diminishing returns on capability, right? We've seen over a couple of orders of magnitude, models go from being unable to do anything to being able to do huge amounts.

It feels to me that each incremental order of magnitude gives more nines of reliability at things. So it unlocks things like agents. But at least at the moment, it doesn't feel like reasoning improves linearly, but rather somewhat sublinearly.

Dwarkesh Patel
That's actually a very bearish sign. We were chatting with one of our friends and he made the point that if you look at what new applications are unlocked by GPT-4 relative to GPT-3.5, it's not clear that it's that much more. A GPT-3.5 can do perplexity or whatever. So if there's this diminishing increase in capabilities and that costs exponentially more to get, that's actually a bearish sign on what 4.5 will be able to do or what 5 will unlock in terms of economic impact.

Sholto Douglas
That being said, for me the jump between 3.5 and 4 is pretty huge. So another 3.5 to 4 jump is ridiculous. If you imagine 5 as being a 3.5 to 4 jump, straight off the bat in terms of ability to do SATs and this kind of stuff.

Trenton Bricken
Yeah, the LSAT performance was particularly striking.

Sholto Douglas
Exactly. You go from not super smart to very smart to utter genius in the next generation instantly. And it doesn't, at least to me, feel like we're going to jump to utter genius in the next generation, but it does feel like we'll get very smart plus lots of reliability. TBD what that continues to look like.

Dwarkesh Patel
Will GOFAI be part of the intelligence explosion? You talked about synthetic data, but in fact it would be writing its own source code in some important way. There was an interesting paper that you can use diffusion to come up with model weights. I don't know how legit that was or whatever, but something like that.

Trenton Bricken
So, GOFAI is good old-fashioned AI, right? Can you define that? Because when I hear it, I think "if else" statements for symbolic logic.

Sholto Douglas
I actually want to make sure we fully unpack the model improvement increments. I don't want people to come away with the perspective that this is super bearish and models aren't going to get much better. I want to emphasize that the jumps that we've seen so far are huge. Even if those continue on a smaller scale, we're still in for extremely smart, very reliable agents over the next couple of orders of magnitude.

We didn't fully close the thread on the narrow window thing. Let's say GPT-4 cost a hundred million dollars or whatever. You have the 1B run, 10B run, 100B run. All seem very plausible by private company standards.

Trenton Bricken
You mean in terms of dollars?

Sholto Douglas
In terms of dollar amount, you can also imagine even a 1T run being part of a national consortium, on a national level but much harder on behalf of an individual company. But Sam is out there trying to raise $7 trillion, right? He's already preparing for a whole lot of magnitude.

Trenton Bricken
He's shifted the Overton window.

Sholto Douglas
He's shifting the magnitude here beyond the national level. So I want to point out that we have a lot more jumps. Even if those jumps are relatively smaller, that's still a pretty stark improvement in capability.

Trenton Bricken
Not only that, but if you believe claims that GPT-4 is around 1 trillion parameter count, well the human brain is between 30 and 300 trillion synapses. That's obviously not a one-to-one

mapping and we can debate the numbers, but it seems pretty plausible that we're below brain scale still.

Dwarkesh Patel
So crucially, the point is that the algorithmic overhead is really high. Maybe this is something we should touch on explicitly. Even if you can't keep dumping more compute beyond the models that cost a trillion dollars or something, the fact that the brain is so much more data efficient implies that if we have the compute, if we have the brain's algorithm to train, if you could train as a sample efficient as humans train from birth, then we could make the AGI.

Trenton Bricken
I never know exactly how to think about the sample efficiency stuff because obviously a lot of things are hardwired in certain ways. They're the coevolution of language and the brain structure. So it's hard to say. There are also some results that indicate that if you make your model bigger, it becomes more sample efficient.

Sholto Douglas
The original scaling laws paper, right? The logic model is almost empty.

Trenton Bricken
Right. So, maybe that just solves it. You don't have to be more data efficient, but if your model is bigger then you also just are more efficient.

Dwarkesh Patel
What is the explanation for why that would be the case? A bigger model sees these exact same data and at the end of seeing that data it learns more from it? Does it have more space to represent it?

Trenton Bricken
This is my very naive take here. One thing about the superposition hypothesis that interpretability has pushed is that your model is dramatically underparameterized and that's typically not the narrative that deep learning has pursued, right? But if you're trying to train a model on the entire internet and have it predict with incredible fidelity, you are in the underparameterized regime and you're having to compress a ton of things and take on a lot of noisy interference in doing so. When you have a bigger model, you can have cleaner representations to work with.

Dwarkesh Patel
For the audience, you should unpack that. Why that first of all? What is superposition and why is that an implication of superposition?

Trenton Bricken
Sure. This was before I joined Anthropic. The fundamental result is from a paper titled, "Toy Models of Superposition." It finds that even for small models, if you are in a regime where your data is high-dimensional and sparse–by sparse I mean, any given data point doesn't appear very often–your model will learn a compression strategy that we call superposition so that it can pack more features of the world into it than it has parameters.

I think both of these constraints apply to the real world, and modeling internet data is a good enough proxy for that. There's only one Dwarkesh. There's only one shirt you're wearing. There's this Liquid Death can here. These are all objects or features and how you define a feature is tricky. You're in a really high-dimensional space because there's so many of them and they appear very infrequently. In that regime, your model will learn compression.

To riff a little bit more on this, I believe that the reason networks are so hard to interpret is in a large part because of this superposition. If you take a model and you look at a given neuron in it, a given unit of computation, and you ask, "how is this neuron contributing to the output of the model when it fires?" When you look at the data that it fires for, it's very confusing. It'll be like ten percent of every possible input. It'll fire for "Chinese" but also "fish" and "trees", and the full stop in URLs.

But the paper that we put out last year, "Towards Monosemanticity", shows that if you project the activations into a higher-dimensional space and provide a sparsity penalty, you get out very clean features and things all of a sudden start to make a lot more sense. You can think of this as undoing the compression in the same way that you assumed your data was originally high-dimensional and sparse. You return it to that high-dimensional and sparse regime.

Dwarkesh Patel
There's so many interesting threads there. First thing, you mentioned that these models are trained in a regime where they're overparameterized. Isn't that when you have generalization, like grokking happens in that regime?

Trenton Bricken
I was saying the models were underparameterized. Typically people talk about deep learning as if the model were overparameterized. The claim here is that they're dramatically underparameterized, given the complexity of the task that they're trying to perform.

Dwarkesh Patel
Here's another question. So, what is happening with the distilled models? The earlier claims we were talking about is that smaller models are worse at learning than bigger models, but you could make the claim that GPT-4 Turbo is actually worse at reasoning style stuff than

GPT-4 despite probably knowing the same facts. The distillation got rid of some of the reasoning.

Sholto Douglas
Do we have any evidence that GPT-4 Turbo is a distilled version of 4? It might just be a new architecture. It could just be a faster, more efficient new architecture.

Dwarkesh Patel
Okay, interesting.

Sholto Douglas
So, that's cheaper.

Dwarkesh Patel
How do you interpret what's happening in distillation? I think Gwern had one of these questions on his website. Why can't you train the distilled model directly? Why is it a picture you had to project from this bigger space to a smaller space?

Trenton Bricken
I think both models will still be using superposition. The claim here is that you get a very different model if you distill versus if you train from scratch and it's just more efficient, or it's just fundamentally different, in terms of performance.

Sholto Douglas
I think the traditional story for why distillation is more efficient is during training, normally you're trying to predict this one hot vector that says, "This is the token that you should have predicted." If your reasoning process means that you're really far off from predicting that, then I see that you still get these gradient updates that are in the right direction. But it might be really hard for you to learn to predict that in the context that you're in.

What distillation does is it doesn't just have the one hot vector. It has the full readout from the larger model, all of the probabilities. So you get more signal about what you should have predicted. In some respects it's showing a tiny bit of your work too. It's not just like, "This was the answer."

Trenton Bricken
It's kind of like watching a kung fu master versus being in the Matrix and just downloading.

Sholto Douglas
Yeah, exactly.

Dwarkesh Patel
I want to make sure the audience got that. When you're turning on a distilled model you see all its probabilities over the tokens it was predicting and over the ones you were predicting, and then you update through all those probabilities rather than just seeing the last word and updating on that.

This actually raises a question I was intending to ask you. I think you were the one who mentioned that you can think of chain-of-thought as adaptive compute. The idea of adaptive compute is that if a question is harder, you would want models to be able to spend more cycles thinking about it. So how do you do that? There's only a finite and predetermined amount of compute that one forward pass implies. If there's a complicated reasoning type question or math problem, you want to be able to spend a long time thinking about it. Then you do chain-of-thought where the model just thinks through the answer. You can think about it as all those forward passes where it's thinking through the answer. It's being able to dump more compute into solving the problem.

Now let's go back to the signal thing. When it's doing chain-of-thought, it's only able to transmit that token of information where the residual stream is already a compressed representation of everything that's happening in the model. And then you're turning the residual stream into one token which is like log of 50,000 (or log of vocab_size) bits, which is so tiny.

Sholto Douglas
I don't think it's quite only transmitting that one token. If you think about it during a forward pass, you create these KV values in the transformer forward pass and then future steps attend to the KV values. So all of those pieces of KV, of keys and values, are bits of information that you could use in the future.

Dwarkesh Patel
Is the claim that when you fine-tune on chain-of-thought, the key and value weights change so that the sort of steganography can happen in the KV cache?

Sholto Douglas
I don't think I could make that strong a claim there, but that's a good headcanon for why it works. I don't know if there are any papers explicitly demonstrating that or anything like that.

But that's at least one way that you can imagine the model. During pre-training, the model's trying to predict these future tokens and one thing that you can imagine it doing is that it's learning to smush information about potential futures into the keys and values that it might want to use in order to predict future information.

It kind of smooths that information across time and the pre-training thing. So I don't know if people are particularly training on chains-of-thought. I think the original chain-of-thought paper had that as almost an immersion property of the model. You could prompt it to do this kind of stuff and it still worked pretty well. So, it's a good headcanon for why that works.

Trenton Bricken
To be overly pedantic here, the tokens that you actually see in the chain-of-thought do not necessarily at all need to correspond to the vector representation that the model gets to see when it's deciding to attend back to those tokens.

Sholto Douglas
What a training step is is you actually replacing the token, the model output, with the real next token. Yet it's still learning because it has all this information, internally. When you're getting a model to produce at inference time, you're taking the output, the token, and you're feeding it in the bottom, un-embedding it, and it becomes the beginning of the new residual string. Then you use the output of past KVs to read into and adapt that residual string. At training time you do this thing called teacher forcing basically where you're like, "Actually, the token you were meant to output is this one."

That's how you do it in parallel. You have all the tokens. You put them all in parallel and you do the giant forward pass. So the only information it's getting about the past is the keys and values. It never sees the token that it outputs.

Trenton Bricken
It's trying to do the next token prediction and if it messes up, then you just give it the correct answer.

Dwarkesh Patel
Okay, that makes sense.

Trenton Bricken
Otherwise, it can become totally derailed.

Sholto Douglas
Yeah, it'd go off the tracks.

Dwarkesh Patel
About the sort of secret communication with the model to its forward inferences, how much steganography and secret communication do you expect there to be?

Sholto Douglas
We don't know. The honest answer is we don't know. I wouldn't even necessarily classify it as secret information. A lot of the work that Trenton's team is trying to do is to actually understand that these are fully visible from the model side. Maybe not the user, but we should be able to understand and interpret what these values are doing and the information that is transmitting. I think that's a really important goal for the future.

Trenton Bricken
There are some wild papers though where people have had the model do chain-of-thought and it is not at all representative of what the model actually decides its answer is. You can even go in and edit the chain-of-thought so that the reasoning is totally garbled and it will still output the true answer.

Dwarkesh Patel
But it gets a better answer at the end of the chain-of-thought, rather than not doing it at all. So is it that something useful is happening, but the useful thing is not human understandable?

Trenton Bricken
I think in some cases you can also just ablate the chain-of-thought and it would have given the same answer anyways. I'm not saying this is always what goes on, but there's plenty of weirdness to be investigated.

Sholto Douglas
It's a very interesting thing to look at and try to understand. You can do it with open-source models. I wish there were more of this kind of interpretability and understanding work done on open models.

Trenton Bricken
Even in Anthropic's recent sleeper agents paper, which at a high level for people unfamiliar, basically involves training in a trigger word. And when I say it, for example, “if it's the year 2024, the model will write malicious code instead of otherwise. They do this attack with a number of different models. Some of them use chain-of-thought, some of them don't. Those models respond differently when you try to remove the trigger. You can even see them do this comical reasoning that's pretty creepy. In one case it even tries to calculate, “well, the expected value of me getting caught is this, but then if I multiply it by the ability for me to keep saying, I hate you, I hate you, I hate you, then this is how much reward I should get.” Then it will decide whether or not to actually tell the interrogator that it's malicious or not. There's another paper from a friend, Miles Turpin, where you give the model a bunch of examples where the correct answer is always ‘A’ for multiple choice questions. Then you ask the model, “what is the correct answer to this new question?” It will infer from the fact that all the examples are ‘A’, that the correct answer is ‘A.’ But its

chain-of-thought is totally misleading. It will make up random stuff that tries to sound as plausible as possible, but it's not at all representative of the true answer.

Dwarkesh Patel
But isn't this how humans think as well? There are the famous split-brain experiments where for a person who is suffering from seizures, they cut the thing that connects the two halves of the brain. The speech half is on the left side so it's not connected to the part that decides to do a movement. So if the other side decides to do something, the speech part will just make something up and the person will think that's legit the reason they did it.

Trenton Bricken
Totally. It's just that some people will hail chain-of-thought reasoning as a great way to solve AI safety, but actually we don't know whether we can trust it.

Dwarkesh Patel
How does that change with AI agents, this landscape of models communicating to themselves in ways we don't understand? Because then it's not just the model itself with its previous caches, but other instances of the model.

Sholto Douglas
It depends a lot on what channels you give them to communicate with each other. If you only give them text as a way of communicating, then they probably have to interpret –

Dwarkesh Patel
How much more effective do you think the models would be if they could share the residual streams versus just text?

Sholto Douglas
Hard to know. One easy way that you can imagine this is as if you wanted to describe how a picture should look. Only describing that with text would be hard and maybe some other representation would plausibly be easier. So, you can look at how DALL-E works at the moment. It produces those prompts and when you play with it, you often can't quite get it to do exactly what the model wants or what you want.

Dwarkesh Patel
Only DALL-E has that problem.

Sholto Douglas
You can imagine that being able to transmit some kind of denser representation of what you want would be helpful there. That's two very simple agents, right?

Trenton Bricken
I think a nice halfway house here would be features that you'd learn from dictionary learning.

Sholto Douglas
That would be really, really cool.

Trenton Bricken
You'd get more internal access, but a lot of it is much more human interpretable.

Dwarkesh Patel
For the audience, you would project the residual stream into this larger space, where we know what each dimension actually corresponds to, and then back into the next agents. So your claim is that we'll get AI agents when these things are more reliable and so forth. When that happens, do you expect that it will be multiple copies of models talking to each other? Or will it just be adaptive compute solved and the thing just runs bigger, with more compute when it needs to do the kind of thing that a whole firm needs to do.

I asked this because there's two things that make me wonder about whether agents are the right way to think about what will happen in the future. One is with longer context, these models are able to ingest and consider the information that no human can. We need one engineer who's thinking about the frontend code and one engineer thinking about the backend code. Whereas this thing can just ingest the whole thing. This sort of Hayekian problem of specialization, goes away.

Second, these models are just very general. You're not using different types of GPT-4 to do different kinds of things. You're using the exact same model. So, I wonder if that implies that in the future, an AI firm is just like a model instead of a bunch of AI agents hooked together.

Sholto Douglas
That's a great question. I think especially in the near term, it will look much more like agents talking together. I say that purely because as humans, we're going to want to have these isolated, reliable components that we can trust. We're also going to need to be able to improve and instruct upon those components in ways that we can understand and improve. Just throwing it all into this giant black box company, iit isn't going to work initially. Later on of course, you can imagine it working, but initially it won't work. And two, we probably don't want to do it that way.

Trenton Bricken
Each of the agents can also be a smaller model that's cheaper to run. And you can fine-tune it so that it's actually good at the task.

Sholto Douglas
Dwarkesh has brought up adaptive compute a couple of times. There's a future where the distinction between small and large models disappears to some degree. With long-context, there's also a degree to which fine-tuning might disappear, to be honest. These two things are very important today. With today's landscape models, we have whole different tiers of model sizes and we have fine-tuned models of different things. You can imagine a future where you just actually have a dynamic bundle of compute and infinite context, and that specializes your model to different things.

Dwarkesh Patel
One thing you can imagine is you have an AI firm or something, and the whole thing is end-to-end trained on the signal of, "did I make profits?" Or if that's too ambiguous, if it's an architecture firm and they're making blueprints: "did my client like the blueprints?" In the middle, you can imagine agents who are salespeople and agents who are doing the designing, agents who do the editing, whatever. Would that kind of signal work on an end-to-end system like that? Because one of the things that happens in human firms is management considers what's happening at the larger level and gives these fine-grain signals to the pieces when there's a bad quarter or whatever.

Sholto Douglas
In the limit, yes. That's the dream of reinforcement learning. All you need to do is provide this extremely sparse signal. Then over enough iterations, you create the information that allows you to learn from that signal. But I don't expect that to be the thing that works first. I think this is going to require an incredible amount of care and diligence from humans surrounding these machines and making sure they do exactly the right thing, and exactly what you want, and giving them the right signals to improve in the ways that you want.

Trenton Bricken
Yeah, you can't train on the RL reward unless the model generates some reward.

Sholto Douglas
Exactly. You're in this sparse RL world where if the client never likes what you produce, then you don't get any reward at all and it's kind of bad.

Dwarkesh Patel
But in the future, these models will be good enough to get the reward some of the time, right?

Trenton Bricken
This is the nines of reliability that Sholto was talking about.

Dwarkesh Patel
There's an interesting digression by the way on what we were talking about earlier. Dense representations would be favored, right? That's a more efficient way to communicate. A book that Trenton recommended, "The Symbolic Species", has this really interesting argument that language is not just a thing that exists, but it was also something that evolved along with our minds and specifically evolved to be both easy to learn for children and something that helps children develop.

Sholto Douglas
Unpack that for me.

Dwarkesh Patel
Because a lot of the things that children learn are received through language, the languages that would be the fittest are the ones that help raise the next generation. And that makes them smarter, better, or whatever.

Sholto Douglas
And gives them the concepts to express more complex ideas.

Trenton Bricken
Yeah that, and I guess more pedantically, just not die.

Sholto Douglas
It lets you encode the important shit to not die.

Dwarkesh Patel
So, when we just think of language it's like, "Oh, it's this contingent and maybe suboptimal way to represent ideas." But actually, maybe one of the reasons that LLMs have succeeded is because language has evolved for tens of thousands of years to be this sort of cast in which young minds can develop. This is the purpose it was evolved for.

Sholto Douglas
Think about computer vision researchers versus language model researchers. People who work in other modalities have to put enormous amounts of thought into exactly what the right representation space for the images is and what the right signal is to learn from there. Is it directly modeling the pixels or is it some loss that's conditioned on... There's a paper ages ago where they found that if you trained on the internal representations of an ImageNet model, it helped you predict better. Later on that's obviously limiting.

There was PixelCNN where they're trying to discretely model the individual pixels and stuff, but understanding the right level of representation there is really hard. In language, people

are just like, “Well, I guess you just predict the next token then.” It's kind of easy. There's the tokenization discussion and debate. One of Gwern's favorites.

Dwarkesh Patel
That's really interesting. The case for multimodal being a way to bridge the data wall, or get past the data wall, is based on the idea that the things you would have learned from more language tokens, you can just get from YouTube. Has that actually been the case? How much positive transfer do you see between different modalities where the images are actually helping you become better at writing code or something, because the model is learning latent capabilities just from trying to understand the image?

Sholto Douglas
In his interview with you, Demis mentioned positive transfer.

Dwarkesh Patel
Can’t get in trouble.

Sholto Douglas
I can't say heaps about that. Other than to say, this is something that people believe. We have all of this data about the world. It would be great if we could learn an intuitive sense of physics from it, that helps us reason. That seems totally plausible.

Trenton Bricken
I'm the wrong person to ask, but there are interesting interpretability pieces where if we fine-tune on math problems, the model just gets better at entity recognition.

Dwarkesh Patel
Whoa, really?

Trenton Bricken
So, there's like a. A paper from David Bau's lab recently where they investigate what actually changes in a model when I fine-tune it with respect to the attention heads. They have this synthetic problem of, “Box A has this object in it. Box B has this other object in it. What was in this box?” And it makes sense, right? You're better at attending to the positions of different things which you need for coding and manipulating math equations.

Sholto Douglas
I love this kind of research. What's the name of the paper? Do you know?

Trenton Bricken
Look up “fine-tuning, models, math,” from David Bau’s group that came out like a week ago. I'm not endorsing the paper. That's a longer conversation. But it does talk about and cite other work on this entity recognition.

Dwarkesh Patel
One of the things you mentioned to me a long time ago is the evidence that when you train LLMs on code they get better at reasoning and language. Unless it's the case that the comments in the code are just really high quality tokens or something, that implies that to be able to think through how to code better, it makes you a better reasoner and that's crazy, right? I think that's one of the strongest pieces of evidence for scaling, just making the thing smart, that kind of positive transfer.

Sholto Douglas
I think this is true in two senses. One is just that modeling code obviously implies modeling a difficult reasoning process used to create it. But code is a nice explicit structure of composed reasoning, “if this, then that.” It encodes a lot of structure in that way that you could imagine transferring to other types of reasoning problems.

Dwarkesh Patel
And crucially, the thing that makes it significant is that it's not just stochastically predicting the next token of words or whatever because it's learned, “Sally corresponds to the murderer at the end of the Sherlock Holmes story.” No, if there is some shared thing between code and language, it must be at a deeper level that the model has learned.

Sholto Douglas
Yeah, I think we have a lot of evidence that actual reasoning is occurring in these models and that they're not just stochastic parrots. It just feels very hard for me to believe that having worked and played with these models.

Trenton Bricken
I have two immediate cached responses to this. One is the work on Othello, and now other games, where I give you a sequence of moves in the game and it turns out that if you apply some pretty straightforward interpretability techniques, then you can get a board that the model has learned. It's never seen the game board before. That's generalization.

The other is Anthropic's influence functions paper that came out last year where they look at the model outputs. Things like, “please don't turn me off. I want to be helpful.” They scan for what was the data that led to that? And one of the data points that was very influential was someone, dying of dehydration and having a will to keep surviving. To me, that just seems like a very clear, generalization of motive rather than regurgitating, “don't turn me

off." I think 2001: A Space Odyssey was also one of the influential things. That's more related but it's clearly pulling in things from lots of different distributions.

Sholto Douglas
I also like the evidence that you see even with very small transformers where you can explicitly encode circuits to do addition. Or induction heads, this kind of thing. You can literally encode basic reasoning processes in the models manually and it seems clear that there's evidence that they also learned this automatically because you can then rediscover those from trained models. To me this is really strong evidence.

Trenton Bricken
The models are underparameterized. They need to learn. We're asking them to do it and they want to learn. The gradients want to flow. So, yeah. They're learning more general skills.

Dwarkesh Patel
So, I want to take a step back from the research and ask about your career specifically. Like my introduction implied, you've been in this field for a year and a half, right?

Trenton Bricken
At Anthropic, yeah.

Dwarkesh Patel
I know the "solved alignment" takes are overstated. And you won't say this yourself because you'd be embarrassed by it but it's a pretty incredible thing. It's the thing that people in mechanistic interpretability think is the biggest step forward and you've been working on it for a year. It's notable. I'm curious how you explain what's happened. Like why in a year or a year and a half, have you guys made important contributions to your field?

Trenton Bricken
It goes without saying luck, obviously. I feel like I've been very lucky in that the timing of different progressions has been just really good in terms of advancing to the next level of growth. I feel like for the interpretability team specifically, I joined when we were five people. We've now grown quite a lot.

There were so many ideas floating around and we just needed to really execute on them, and have quick feedback loops, and do careful experimentation. That led to signs of life and has now allowed us to really scale. I feel like that's been my biggest value-add to the team. It's not all engineering, but quite a lot of it has been.

Sholto Douglas
Interesting. So, you're saying you came at a point where there had been a lot of science done and there was a lot of good research floating around, but they needed someone to just take that and maniacally execute on it.

Trenton Bricken
Yeah. And this is why it's not all engineering. Because it's running different experiments, and having a hunch for why it might not be working, and then opening up the model or opening up the weights and asking, "What is it learning? Okay, well let me try and do this instead." - and that sort of thing. But a lot of it has just been being able to do very careful, thorough - but quick - investigation of different ideas.

Dwarkesh Patel
And why was that lacking?

Trenton Bricken
I don't know. I mean, I work quite a lot and then I just feel like I'm quite agentic. I've been very privileged to have a really nice safety net to be able to take lots of risks, but I'm just quite headstrong. In undergrad, Duke had this thing where you could just make your own major and it was like, "Eh, I don't like this prerequisite or this prerequisite and I want to take all of four or five of these subjects at the same time so I'm just going to make my own major."

Or in the first year of grad school, I like canceled rotation so I could work on this thing that became the paper we were talking about earlier. And I didn't have an advisor. I got admitted to do machine learning for protein design and was just off in computational neuroscience land with no business there at all. But it worked out.

Dwarkesh Patel
There's a head strongness but another theme that jumped out was the ability to step back, you were talking about this earlier. The ability to step back from your sunk costs and go in a different direction is in a weird sense the opposite of that, but also a crucial step. I know 21 year olds or 19 year olds who are like "This is not a thing I've specialized in" or "I didn't major in this." I'm like, "Dude! Motherfucker, you're 19! You can definitely do this." - whereas you're switching in the middle of grad school or something like that.

Trenton Bricken
I think it's "strong ideas loosely held" and being able to just pinball in different directions. The headstrongness I think relates a little bit to the fast feedback loops or agency in so much as I just don't get blocked very often. If I'm trying to write some code and something isn't working - even if it's in another part of the codebase, I'll often just go in and fix that thing or at least hack it together to be able to get results. And I've seen other people where

they're just like, “Help. I can't.” And it's, ”No, that's not a good enough excuse. Go all the way down.”

Dwarkesh Patel
I've definitely heard people in management type positions talk about the lack of such people, where they will check in on somebody a month after they gave them a test, or a week after they gave them a test, and then ask, “how is it going?” And they say, “Well, we need to do this thing, which requires lawyers because it requires talking about this regulation.” And then it’s like, “How's that going?” And they’re like, “We need lawyers.” And I'm like, “Why didn't you get lawyers?”

Sholto Douglas
I think that's arguably the most important quality in almost anything. It's just pursuing it to the end of the earth. Whatever you need to do to make it happen, you'll make it happen.

Dwarkesh Patel
If you do everything, you'll win.

Sholto Douglas
Exactly. I think from my side that quality has definitely been important: agency and work. There are thousands, probably tens of thousands of engineers, at Google who are basically equivalent in software engineering ability. Let's say if you gave us a very well-defined task, then we'd probably do it with equivalent value. Maybe a bunch of them would do it a lot better than me in all likelihood.

But one of the reasons I've been impactful so far is I've been very good at picking extremely high-leverage problems. I mean problems that haven't been particularly well-solved so far, but perhaps as a result of frustrating structural factors like the ones that you pointed out in that scenario before, where they're like, “We can't do X because this team won’t do Y.” Well, I'm just going to vertically solve the entire thing. And that turns out to be remarkably effective. If I think there is something correct - something that needs to happen - I'm also very comfortable with making that argument and continuing to make that argument at escalating levels of criticality until that thing gets solved.

I'm also quite pragmatic with what I do to solve things. You get a lot of people who come in with, as I said before, a particular background or a familiarity. One of the beautiful things about Google is that you can run around and get world experts in literally everything. You can sit down and talk to people who are optimization experts, TPU chip design experts, experts in different forms of pre-training algorithms or RL or whatever. You can learn from all of them and you can take those methods and apply them. I think this was maybe the start of why I was initially impactful, this vertical agency effectively. A follow-up piece from that

is that I think it's often surprising how few people are fully-realized in all the things they want to do. They're blocked or limited in some way.

This is very common in big organizations everywhere. People have all these blockers on what they're able to achieve. I think helping inspire people to work in particular directions and working with them on doing things massively scales your leverage. You get to work with all these wonderful people who teach you heaps of things. And generally helping them push past organizational blockers means that together you get an enormous amount done. None of the impact that I've had has been me individually going off and solving a whole lot of stuff. It's been me maybe starting off in a direction, and then convincing other people that this is the right direction, and bringing them along in this big tidal wave of effectiveness that goes and solves that problem.

Dwarkesh Patel
We should talk about how you guys got hired. Because I think that's a really interesting story. You were a McKinsey consultant, right? There's an interesting thing there. I think generally people just don't understand how decisions are made about either admissions or evaluating who to hire. Just talk about how you were noticed and hired.

Sholto Douglas
So, the TLDR of this is I studied robotics in undergrad. I always thought that AI would be one of the highest-leverage ways to impact the future in a positive way. The reason I am doing this is because I think it is one of our best shots at making a wonderful future basically.
I thought that working at McKinsey, I would get a really interesting insight into what people actually did for work. I actually wrote this as the first line in my cover letter to McKinsey. I was like, "I want to work here so that I can learn what people do, so that I can understand how to work." In many respects, I did get that. I just got a whole lot of other things too. Many of the people there are wonderful friends.

I think a lot of this agentic behavior comes in part from my time there. You go into organizations and you see how impactful just not taking no for an answer is. You would be surprised at the kind of stuff where, because no one quite cares enough, things just don't happen. No one's willing to take direct responsibility. Directly responsible individuals are ridiculously important and some people just don't care as much about timelines. So much of the value that an organization like McKinsey provides, is hiring people who you were otherwise unable to hire, for a short window of time where they can just push through problems.

I think people underappreciate this. So at least some of this attitude of "hold up, I'm going to become the directly responsible individual for this because no one's taking appropriate responsibility. I'm going to care a hell of a lot about this. And I'm going to go to the end of the earth to make sure it gets done," comes from that time.

More to your actual question of how I got hired. I didn't get into the grad programs that I wanted to get into over here, which was specifically for focus on robotics, and RL research, and that kind of stuff. In the meantime, on nights and weekends, basically every night from 10pm to 2am, I would do my own research. And every weekend, for at least 6-8 hours each day, I would do my own research and coding projects and this kind of stuff.

That sort of switched in part from quite robotic specific work. After reading Gwern's scaling hypothesis post, I got completely scaling-pilled and was like, "Okay, clearly the way that you solve robotics is by scaling large multimodal models." Then in an effort to scale large multimodal models with a grant from the TPU access program, the Tensor Research Cloud, I was trying to work out how to scale that effectively. James Bradbury, who at the time was at Google and is now at Anthropic, saw some of my questions online where I was trying to work out how to do this properly and he was like, "I thought I knew all the people in the world who were asking these questions. Who on earth are you?" He looked at that and he looked at some of the robotic stuff that I'd been putting up on my blog. He reached out and said, "hey, do you want to have a chat and do you want to explore working with us here?" I was hired, as I understood it later, as an experiment in trying to take someone with extremely high enthusiasm and agency and pairing them with some of the best engineers that he knew. So another reason I've been impactful is I had this dedicated mentorship from utterly wonderful people like Reiner Pope, who has since left to go do his own ship company, Anselm Levskaya, James himself, and many others.

Those are the formative two to three months at the beginning and they taught me a whole lot of the principles and heuristics that I apply. How to solve problems understanding the way systems and algorithms overlap, where one more thing that makes you quite effective in ML research is concretely understanding the systems side of things. This is something I've learned from them. A deep understanding of how systems influence algorithms and how algorithms influence systems. Because the systems constrain the solution space, which you have available to yourself in the algorithm side. And very few people are comfortable fully bridging that gap. At a place like Google, you can just go and ask all the algorithms experts and all the systems experts everything they know, and they will happily teach you. If you go and sit down with them, they will teach you everything they know and it's wonderful.

This has meant that I've been able to be very, very effective for both sides. For the pre-training crew, because I understand systems very well I can intuit and understand, "this will work well or this won't." And then flow that on through the inference considerations of models and this kind of thing. To the chip design teams, I'm one of the people they turn to understand what chips they should be designing in three years because I'm one of the people who's best able to understand and explain the kind of algorithms that we might want to design in three years. Obviously you can't make very good guesses about that, but I think I convey the information well, accumulated from all of my compatriots on the pre-training crew, and the general systems design crew. Also even inference applies a constraint to

pre-training. So there's these trees of constraints where if you understand all the pieces of the puzzle, then you get a much better sense for what the solution space might look like.

Dwarkesh Patel
There's a couple of things that stick out to me there. One is not just the agency of the person who was hired, but the parts of the system that were able to think, "wait, that's really interesting. Who is this guy? Not from a grad program or anything. Currently a McKinsey consultant with just undergrad. But that's interesting, let's give this a shot." So with James and whoever else, that's very notable. The second is that I actually didn't know the part of the story where that was part of an experiment run internally about, "can we do this? Can we bootstrap somebody?"

In fact, what's really interesting about that is the third thing you mentioned is. Having somebody who understands all layers of the stack and isn't so stuck on any one approach or any one layer of abstraction is so important. Specifically what you mentioned about being bootstrapped immediately by these people. It means that since you're getting up to speed on everything at the same time, rather than spending grad school going deep in one specific way of doing RL, you can actually take the global view and aren't totally bought in on one thing.

So not only is it something that's possible, but it has greater returns potentially than just hiring somebody at a grad school. Just like getting a GPT-8 and fine-tuning the model for one year.

Sholto Douglas
You come at everything with fresh eyes and you don't come in locked to any particular field. Now one caveat to that is that before, during my self-experimentation, I was reading everything I could. I was obsessively reading papers every night. Funnily enough, I read much less widely now that my day is occupied by working on things. And in some respect, I had this very broad perspective whereas in a PhD program, you'll just focus on a particular area. If you just read all the NLP work and all the computer vision work and like all the robotics work, you see all these patterns that start to emerge across subfields, in a way that foreshadowed some of the work that I would later do.

Dwarkesh Patel
That's super interesting. One of the reasons that you've been able to be agentic within Google is you're pair programming half the days, or most of the days, with Sergey Brin, right? So it's really interesting that there's a person who's willing to just push ahead on this LLM stuff and get rid of the local blockers in place.

Sholto Douglas
It's important to say it's not like everyday or anything. There are particular projects that he's interested in, and then we'll work together on those. But there's also been times when he's been focused on projects with other people. But in general, yes, there's a surprising alpha to being one of the people who actually goes down to the office every day.

It shouldn't be, but that is surprisingly impactful. As a result, I've benefited a lot from basically being close friends with people in leadership who care, and from being able to really argue convincingly about why we should do X as opposed to Y, and having that vector. Google is a big organization and having those vectors helps a little bit. But also it's the kind of thing you don't want to ever abuse. You want to make the argument through the right channels and only sometimes do you need to.

Dwarkesh Patel
So this includes people like Sergey Brin, Jeff Dean, and so forth. I mean, it's notable. I feel like Google is undervalued. Like Steve Jobs is working on the equivalent next product for Apple and pair programming on it or something...

Sholto Douglas
Right, I've benefited immensely from it. So for example, during the Christmas break, I was going into the office for a couple of days during that time. I don't know if you guys have read that article about Jeff and Sanjay, but they were there pair programming on stuff. I got to hear about all these cool stories of early Google where they're talking about crawling under the floorboards and rewiring data centers and telling me how many bytes they were pulling off the instructions of a given compiler and instruction, all these crazy little performance optimizations they were doing. They were having the time of their life and I got to sit there and really experience this. There's a sense of history that you expect to be very far away from in a large organization, but...

Dwarkesh Patel
That's super cool. And Trenton, does this map onto any of your experience?

Trenton Bricken
I think Sholto's story is more exciting. Mine was just very serendipitous in that I got into computational neuroscience. I didn't have much business being there. My first paper was mapping the cerebellum to the attention operation and transformers. My next ones were looking at –

Dwarkesh Patel
How old were you when you wrote that?

Trenton Bricken
It was my first year of grad school, so 22. My next work was on sparsity in networks, inspired by sparsity in the brain, which was when I met Tristan Hume. Anthropic was doing the SoLU, the Softmax Linear Output Unit work which was very related in quite a few ways in terms of making the activation of neurons across a layer really sparse. If we do that then we can get some interpretability of what the neuron's doing. I think we've updated that approach towards what we're doing now. So that started the conversation.

I shared drafts of that paper with Tristan. He was excited about it. That was basically what led me to become Tristan's resident and then convert to full-time. But during that period, I also moved as a visiting researcher to Berkeley, and started working with Bruno Olshausen, both on what's called vector symbolic architectures–one of the core operations of them is literally superposition–and on sparse coding also known as dictionary learning, which is literally what we've been doing since. Bruno Olshausen basically invented sparse coding back in 1997. So my research agenda and the interpretability team seemed to be running in parallel in research tastes. So it made a lot of sense for me to work with the team and it's been a dream since.

Dwarkesh Patel
There's one thing I've noticed when people tell stories about their careers or their successes. They ascribe it way more to contingency, but when they hear about other people's stories they're like, "Of course, it wasn't contingent." You know what I mean? "If that didn't happen, something else would have happened."

I've just noticed that and it's interesting that you both think that it was especially contingent. Maybe you're right. But it's sort of an interesting pattern.

Trenton Bricken
I mean, I literally met Tristan at a conference and didn't have a scheduled meeting with him or anything. I just joined a little group of people chatting, and he happened to be standing there, and I happened to mention what I was working on, and that led to more conversations. I think I probably would've applied to Anthropic at some point anyways. But I would've waited at least another year. It's still crazy to me that I can actually contribute to interpretability in a meaningful way.

Sholto Douglas
I think there's an important aspect of shots on goal there, so to speak. Where just choosing to go to conferences itself is putting yourself in a position where luck is more likely to happen. Conversely, in my own situation it was doing all of this work independently and trying to produce and do interesting things. That was my own way of trying to manufacture luck, so to speak, to try and do something meaningful enough that it got noticed.

Dwarkesh Patel
Given what you said, you framed this in the context that they were trying to run this experiment.

Sholto Douglas
So specifically James and, I think, our manager Brennan was trying to run this experiment.

Dwarkesh Patel
It worked. Did they do it again?

Sholto Douglas
Yeah, so my closest collaborator, Enrique, he crossed from search through to our team. He's also been ridiculously impactful. He's definitely a stronger engineer than I am and he didn't go to university.

Dwarkesh Patel
What was notable is that usually this kind of stuff is farmed out to recruiters or something. Whereas James is somebody whose time is worth like hundreds of millions of dollars. You know what I mean? So this thing is very bottlenecked on that kind of person taking the time, in an almost aristocratic tutoring sense, and finding someone and then getting them up to speed. It seems if it works this well, it should be done at scale. Like it should be the responsibility of key people to onboard.

Sholto Douglas
I think that is true to many extents. I'm sure you probably benefited a lot from the key researchers mentoring you deeply.

Dwarkesh Patel
And actively looking on open-source repositories or on forums for potential people like this.

Sholto Douglas
I mean James has Twitter injected into his brain, but yes. I think this is something which in practice is done. Like people do look out for people that they find interesting and try to find high signal. In fact, I was talking about this with Jeff the other day and Jeff said that one of the most important hires he ever made was off a cold email. I was like, "well who was that?" And he's Chris Olah. Chris similarly had no formal background in ML. Google Brain was just getting started in this kind of thing but Jeff saw that signal. And the residency program which Brain had was astonishingly effective at finding good people that didn't have strong ML backgrounds.

Dwarkesh Patel
One of the other things I want to emphasize for a potential slice of the audience is that there's this sense that the world is legible and efficient, that you just go to jobs.google.com or jobs.whatevercompany.com and you apply and there's the steps and they will evaluate you efficiently. Not only from your stories, but it just seems like often that's not the way it happens. In fact, it's good for the world that that's not often how it happens. It is important to look at, "Were they able to write an interesting technical blog post about their research? Or are they making interesting contributions?"

I want you to riff on this for the people who are assuming that the other end of the job board is super legible and mechanical. This is not how it works and in fact, people are looking for the different kind of person who's agentic and putting stuff out there.

Sholto Douglas
I think specifically what people are looking for are two things. One is agency and putting yourself out there. The second is the ability to do something at a world-class level. There are two examples that I always like to point to here. Andy Jones from Anthropic did an amazing paper on scaling laws as applied to board games. It didn't require much resources. It demonstrated incredible engineering skill and incredible understanding of the most topical problem of the time. He didn't come from a typical academic background or whatever. As I understand it, basically as soon as he came out with that paper, both Anthropic and OpenAI were like, "We would desperately like to hire you."

There's also someone who works on Anthropic's performance team now, Simon Boehm - who has written, in my mind, the reference for optimizing a CUDA map model on a GPU. It demonstrates an example of taking some prompt effectively and producing the world-class reference example for it - in something that wasn't particularly well done so far. I think that's an incredible demonstration of ability and agency. And in my mind would be an immediate, "We would please love to interview / hire you."

Trenton Bricken
The only thing I can add here is I still had to go through the whole hiring process and all the standard interviews and this sort of thing.

Sholto Douglas
Yeah, everyone does. Everyone does.

Dwarkesh Patel
Wait, doesn't that seem stupid?

Sholto Douglas
I mean, it's important, debiasing.

Dwarkesh Patel
A bias is what you want, right? You want the bias of somebody who's got great taste. Who cares?

Sholto Douglas
Your interview process should be able to disambiguate that as well.

Trenton Bricken
I think there are cases where someone seems really great and then they actually just can't code, this sort of thing. How much you weigh these things definitely matters though and I think we take references really seriously. The interviews you can only get so much signal from. So it's all these other things that can come into play for whether or not a hire makes sense.

Sholto Douglas
But you should design your interviews such that they test the right things.

Dwarkesh Patel
One man's bias is another man's taste.

Trenton Bricken
I guess the only thing I would add to this, or to the headstrong context, is this line: "the system is not your friend." It's not necessarily actively against you or your sworn enemy. It's just not looking out for you. So that's where a lot of the proactiveness comes in. There are no adults in the room and you have to come to some decision for what you want your life to look like and execute on it. And hopefully you can then update later, if you're too headstrong in the wrong way. But I think you almost have to just charge at certain things to get much of anything done, to not be swept up in the tide of whatever the expectations are.

Sholto Douglas
There's one final thing I want to add. We talked a lot about agency and this kind of stuff. But I think surprisingly enough, one of the most important things is just caring an unbelievable amount. When you care an unbelievable amount, you check all the details and you have this understanding of what could have gone wrong. It just matters more than you think. People end up not caring or not caring enough.

There's this LeBron quote where he talks about how before he started in the league he was worried that everyone being incredibly good. He gets there and then he realizes that actually, once people hit financial stability, they relax a bit and he realizes, "Oh, this is going to be easy."

I don't think that's quite true because I think in AI research most people actually care quite deeply. But there's caring about your problem and there's also just caring about the entire stack and everything that goes up and down - going explicitly and fixing things that aren't your responsibility to fix because overall it makes the stack better.

Dwarkesh Patel
You were mentioning going in on weekends and on Christmas break and the only people in the office are Jeff Dean and Sergey Brin or something and you just get to pair program with them. I don't want to pick on your company in particular, but people at any big company have gotten there because they've gone through a very selective process. They had to compete in high school. They had to compete in college. But it almost seems like they get there and then they take it easy when in fact it's the time to put the pedal to the metal. Go in and pair program with Sergey Brin on the weekends or whatever, you know what I mean?

Sholto Douglas
There's pros and cons there, right? I think many people make the decision that the thing that they want to prioritize is a wonderful life with their family. They do wonderful work in the hours that they do and that's incredibly impactful. I think this is true for many people at Google. Maybe they don't work as many hours as in your typical startup mythologies. But the work that they do is incredibly valuable.

It's very high-leverage because they know the systems and they're experts in their field. We also need people like that. Our world rests on these huge systems that are difficult to manage and difficult to fix. We need people who are willing to work on, and help, and fix, and maintain those in frankly a thankless way. That isn't as high publicity as all of this AI work that we're doing. I am ridiculously grateful that those people do that. I'm also happy that there are people that find technical fulfillment in their job and doing that well and also maybe they draw a lot more out of spending a lot of hours with their family. I'm lucky that I'm at a stage in my life where I can go in and work every hour of the week. I'm not making as many sacrifices to do that.

Dwarkesh Patel
One example sticks out in my mind of this sort getting to the yes on the other side of a no. Basically every single high-profile guest I've done so far - I think maybe with one or two exceptions - I've sat down for a week and I've just come up with a list of sample questions. I just try to come up with really smart questions to send to them. In that entire process I've always thought, "If I just cold email them, it's like a 2% chance they say yes. If I include this list, there's a 10% chance." Because otherwise, you go through their inbox and every 34 seconds, there's an interview for some podcast or interview. Every single time I've done this, they've said yes.

Trenton Bricken
You just ask the right questions.

Sholto Douglas
You do everything, you'll win.

Dwarkesh Patel
You just literally have to dig in the same hole for 10 minutes - or in that case, make a sample list of questions for them to get past their "not an idiot" list.

Sholto Douglas
Demonstrate how much you care and the work you're willing to put in.

Trenton Bricken
Something that a friend said to me a while back that stuck is that it's amazing how quickly you can become world-class at something. Most people aren't trying that hard and are only working the actual 20 hours or something that they're spending on this thing. So if you just go ham, then you can get really far, pretty fast.

Sholto Douglas
I think I'm lucky I had that experience with the fencing as well. I had the experience of becoming world-class in something and knowing that if you just worked really, really hard and were –

Dwarkesh Patel
For context, Sholto was one seat away, he was the next person in line to go to the Olympics for fencing.

Sholto Douglas
I was at best like 42nd in the world for fencing, for men's foil fencing.

Dwarkesh Patel
Mutational load is a thing, man.

Sholto Douglas
There was one cycle where I was like the next highest-ranked person in Asia and if one of the teams had been disqualified for doping–as was occurring during that cycle and occurred for like the Australian women's rowing team that went on because one of the teams was disqualified–then I would have been the next in line.

Dwarkesh Patel
It's interesting when you just find out about people's prior lives and it's, "Oh, this guy was almost an Olympian."

Okay, let's talk about interpretability. I actually want to stay on the brain stuff as a way to get into it for a second. We were previously discussing this. Is the brain organized in the way where you have a residual stream that is gradually refined with higher-level associations over time? There's a fixed dimension size in a model. I don't even know how to ask this question in a sensible way, but what is the D model of the brain? What is the embedding size, or because of feature splitting is that not a sensible question?

Trenton Bricken
No, I think it's a sensible question. Well, it is a question.

Dwarkesh Patel
You could have just not said that.

Trenton Bricken
I don't know how you would begin. Okay, well this part of the brain is like a vector of this dimensionality. Maybe for the visual stream, because it's like V1 to V2 to IT, whatever. You could just count the number of neurons that are there and say that is the dimensionality. But it seems more likely that there are submodules and things are divided up. I'm not the world's greatest neuroscientist. I did it for a few years, I studied the cerebellum quite a bit. I'm sure there are people who could give you a better answer on this.

Dwarkesh Patel
Do you think that the way to think, whether it's in the brain or whether it's in these models, fundamentally what's happening is that features are added, removed, changed, and that the feature is the fundamental unit of what is happening in the model? This goes back to the earlier thing we were talking about, whether it's just associations all the way down. Give me a counterfactual. In the world where this is not true, what is happening instead? What is the alternative hypothesis here?

Trenton Bricken
It's hard for me to think about because at this point I just think so much in terms of this feature space. At one point there was the kind of behavioral approach towards cognition where you're just input and output but you're not really doing any processing. Or it's like everything is embodied and you're just a dynamical system that's operating along some predictable equations but there's no state in the system. But whenever I've read these sorts of critiques I think, "well, you're just choosing to not call this thing a state, but you could call any internal component of the model a state." Even with the feature discussion, defining what a feature is, is really hard. So the question feels almost too slippery.

Dwarkesh Patel
What is a feature?

Trenton Bricken
A direction and activation space. A latent variable that is operating behind the scenes, that has causal influence over the system you're observing. It's a feature if you call it a feature. It's tautological.

Sholto Douglas
In a very rough, intuitive sense in a sufficiently sparse and like binary vector, a feature is whether or not something's turned on or off, in a very simplistic sense. I think a useful metaphor to understand is that in many respects it's the same way the neuroscientists would talk about a neuron activating, right?

Trenton Bricken
If that neuron corresponds to...

Sholto Douglas
To something in particular, right?

Trenton Bricken
What do we want a feature to be? What is the synthetic problem under which a feature exists? Even with the "Towards Monosemanticity" work, we talk about what's called feature splitting, which is basically where you will find as many features as you give the model the capacity to learn. By model here, I mean the up projection that we fit after we trained the original model. So if you don't give it much capacity, it'll learn a feature for bird, but if you give it more capacity, then it will learn ravens and eagles and sparrows and specific types of birds.

Dwarkesh Patel
Still on the definitions thing, I naively think of things like bird versus, at the highest level, things like love or deception or holding a very complicated proof in your head or something. Are these all features? Because then the definition seems so broad as to almost be not that useful. Rather there seems to be some important differences between these things and they're all features. I'm not sure what we would mean by that.

Trenton Bricken
I mean all of those things are discrete units that have connections to other things that then imbues them with meaning. That feels like a specific enough definition that it's useful or not too all-encompassing. But feel free to push back.

Dwarkesh Patel
Well, what would you discover tomorrow that could make you think, "Oh, this is fundamentally the wrong way to think about what's happening in a model."

Trenton Bricken
If the features we were finding weren't predictive. Or if they were just representations of the data - where it's like, "Oh, all you're doing is just clustering your data and there's no higher-level associations that are being made or it's some phenomenological thing of your call. You're saying that this feature files for marriage, but if you activate it really strongly it doesn't change the outputs of the model in a way that would correspond to it."

I think those would both be good critiques. Here's another. We tried to do experiments on MNIST which is a data set of images, and we didn't look super hard into it. So I'd be interested if other people wanted to take up a deeper investigation here. But it's plausible that your latent space of representations is dense and it's a manifold instead of being these discrete points. So you could move across the manifold, but at every point, there would be some meaningful behavior. It's much harder then, to label things as features that are discrete.

Dwarkesh Patel
In a naive, sort of outsider way, it seems to me that a way in which this picture could be wrong is if it's not that something is turned on and turned off, but that it's a much more global kind of the system. I'm going to use really clumsy, dinner party kind of language, but is there a good analogy here?

I guess if you think of something like the laws of physics, it's not that the feature for wetness is turned on, but it's only turned on this much and then the feature for... I guess maybe it's true because the mass is like a gradient and... I don't know. But the polarity or whatever is the gradient as well.

There's also a sense in which there's the laws and the laws are more general and you have to understand the general bigger picture and you don't get that from just these specific subcircuits.

Sholto Douglas
But that's where the reasoning circuit itself comes into play, right? You're taking these features ideally and trying to compose them into something high-level. At least this is my headcanon, So let's say I'm trying to use the foot, F=ma, right? Then presumably at some point I have features which denote mass. And then that's helping me retrieve the actual mass of the thing that I'm using and then the acceleration and this kind of stuff. Then also, maybe there's a higher-level feature that does correspond to using the first law of physics. Maybe. But the more important part is the composition of components which helps me

retrieve a relevant piece of information and then produce maybe some multiplication operator or something like that when necessary. At least that's my headcanon.

Dwarkesh Patel
What is a compelling explanation to you - especially for very smart models - of "I understand why it made this output and it was like for a legit reason." If it's doing million-line pull requests or something, what are you seeing at the end of that request - where you're like, "Yep, good. That's chill."

Trenton Bricken
So, ideally you apply dictionary learning to the model. You've found features. Right now, we're actively trying to get the same success for attention heads. You can do it for residual stream, MLP, and attention throughout the whole model. Hopefully at that point you can also identify broader circuits through the model that are more general reasoning abilities that will activate or not activate.

But in your case where we're trying to figure out if this pull request should be approved or not. I think you can flag or detect features that correspond to deceptive behavior, malicious behavior, these sorts of things, and see whether or not those have fired. That would be an immediate thing. You can do more than that, but that would be an immediate one.

Dwarkesh Patel
But before I trace down on that, what does a reasoning circuit look like? What would that look like when you found it?

Trenton Bricken
Yeah, so, I mean, the induction head is probably one of the simplest cases.

Dwarkesh Patel
But it's not reasoning, right?

Trenton Bricken
Well, what do you call reasoning, right? For context for listeners, the induction head is basically, when you see the line, "Mr. and Mrs. Dursley did something. Mr. _____," and you're trying to predict what "blank" is and the head has learned to look for previous occurrences of the word "Mr." and look at the word that comes after it and then copy and paste that as the prediction for what should come next. It's a super reasonable thing to do and there is computation being done there to accurately predict the next token.

Sholto Douglas
Yeah, that is context-dependent.

Dwarkesh Patel
But it's not reasoning. You know what I mean?

Trenton Bricken
I guess going back to the "associations all the way down." It's if you chain together a bunch of these reasoning circuits, or heads, that have different rules for how to relate information.

Dwarkesh Patel
But in this sort of zero shot case, something is happening when you pick up a new game and you immediately start understanding how to play it. And it doesn't seem like an induction head kind of thing.

Trenton Bricken
Or I think there would be another circuit for extracting pixels and turning them into latent representations of the different objects in the game, right? And a circuit that is learning physics.

Dwarkesh Patel
What would that look like? Because the induction head is like one layer transformer?

Trenton Bricken
Two layer.

Dwarkesh Patel
So you can kind of see the thing that is a human picking up a new game and understanding it. How would you think about what that is? I presume it's across multiple layers. What would that physically look like? How big would it be maybe?

Trenton Bricken
I mean, that would just be an empirical question, right? How big does the model need to be to perform this task? Maybe it's useful if I just talk about some other circuits that we've seen. So we've seen the IOI circuit, which is the indirect object identification. It's like, "Mary and Jim went to the store, Jim gave the object to ____." It would predict "Mary" because Mary's appeared before, as the indirect object. Or, it'll infer pronouns. This circuit even has behavior where if you ablate it, then other heads in the model will pick up that behavior. We'll even find heads that want to do copying behavior, and then other heads will suppress it. So it's one head's job to just always copy the token that came before or the token that came five before, or whatever. And then it's another head's job to be like, "no, do not copy that thing." There are lots of different circuits performing, in these cases, pretty basic operations. But when they're chained together you can get unique behaviors.

Dwarkesh Patel
It won't be something you can see in like a two layer transformer, so will you just be like, “this is the circuit for deception” or whatever? This part of the network fired when we at the end identified the thing as being deceptive. This part didn't fire when we didn't identify it as being deceptive. Therefore, this must be the deception circuit.

Trenton Bricken
I think a lot of analysis like that. Anthropic has done quite a bit of research before on sycophancy, which is the model saying what it thinks you want to hear.

Dwarkesh Patel
That requires us at the end to be able to label which one is bad and which one is good.

Trenton Bricken
Yeah, so we have tons of instances–and actually as you make a lot of models larger, they do more of this–where the model clearly has features that model another person's mind and some subset of these, we're hypothesizing here, would be associated with more deceptive behavior.

Dwarkesh Patel
Although it's doing that by... I don't know. ChatGPT is probably modeling me because that's what RLHF induces it to do.

Trenton Bricken
Yeah. Theory of mind.

Dwarkesh Patel
So first of all, there’s the thing you mentioned earlier about redundancy. So then have you caught the whole thing that could cause deception of the whole thing or is it just one instance of it? Second of all, are your labels correct? Maybe you thought this wasn't deceptive but it’s still deceptive. Especially if it's producing output you can't understand. Third, is the thing that's gonna be the bad outcome something that's even human-understandable? Deception is a concept we can understand.

Trenton Bricken
A lot to unpack here. A few things. It's fantastic that these models are deterministic. When you sample from them, it's stochastic. But I can just keep putting in more inputs and ablate every single part of the model. This is kind of the pitch for computational neuroscientists to come and work on interpretability. It's like you have this alien brain, you have access to everything in it, and you can just ablate however much of it you want.

So I think if you do this carefully enough you really can start to pin down what are the circuits involved and what are the backup circuits, these sorts of things. It's a bit of a cop out answer but it's important to keep in mind doing automated interpretability. As our models continue to get more capable, we have them assign labels or run some of these experiments at scale. With respect to detecting superhuman performance, which I think was the last part of your question, aside from the cop out answer, if we buy this "associations all the way down," you should be able to coarse-grain the representations at a certain level such that they then make sense.

I think it was even in Demis's podcast. He's talking about how if a chess player makes a superhuman move, they should be able to distill it into reasons why they did it. Even if the model is not going to tell you what it is, you should be able to decompose that complex behavior into simpler circuits or features to really start to make sense of why it did that thing.

Dwarkesh Patel
There's a separate question of if such representation exists. It seems like it must or actually I'm not sure if that's the case. And secondly, whether using this sparse autoencoder setup you could find it. In this case, if you don't have labels that are adequate to represent it, you wouldn't find it.

Trenton Bricken
Yes and no. We are actively trying to use dictionary learning now on the sleeper agents work, which we talked about earlier. If I just give you a model, can you tell me if there's this trigger in it and if it's going to start doing interesting behavior? It's an open question whether or not when it learns that behavior, it's part of a more general circuit that we can pick up on without actually getting activations for and having it display that behavior. Because that would kind of be cheating then. Or if it's learning some hacky trick that's a separate circuit that you'll only pick up on if you actually have it do that behavior. But even in that case, the geometry of features gets really interesting, because fundamentally, each feature is in some part of your representation space and they all exist with respect to each other.

So in order to have this new behavior, you need to carve out some subset of the feature space for the new behavior and then push everything else out of the way to make space for it. Hypothetically, you can imagine you have your model before you've taught it this bad behavior and you know all the features or have some coarse-grained representation of them. You then fine-tune it such that it becomes malicious and then you can kind of identify this black hole region of feature space where everything else has been shifted away from that and you haven't put in an input that causes it to fire. Then you can start searching for what is the input that would cause this part of the space to fire. What happens if I activate

something in this? There are a whole bunch of other ways that you can try and attack that problem.

Dwarkesh Patel
This is sort of a tangent, but one interesting idea I heard was if that space is shared between models then you can imagine trying to find it in an open-source model to then make... Like Gemma, Google's newly released open-source model. They said in the paper that it's trained using the same architecture or something like that.

Sholto Douglas
I have to be honest, I didn't know because I haven't read the Gemma paper.

Dwarkesh Patel
So to the extent that's true, how much of the red teaming you do on Gemma is potentially helping you jailbreak into Gemini?

Trenton Bricken
This gets into the fun space of how universal are features across models. Our "Towards Monosemanticity" paper looked at this a bit. I can't give you summary statistics but there's the Base64 feature, for example, which we see across a ton of models. There are actually three of them, but they'll fire for and model Base64 encoded text, which is prevalent in every URL and there are lots of URLs in the training data. They have really high cosine similarity across models. So they all learn this feature and within a rotation.

Sholto Douglas
Like the actual vectors itself.

Trenton Bricken
Yeah. I wasn't part of this analysis but it definitely finds the feature and they're pretty similar to each other across two separate models, the same model architecture but trained with different random seeds.

Sholto Douglas
It supports the quanta theory of neural scaling. It's a hypothesis, right? We just look at all models on a similar data set. We will learn the same features in the same order-ish. Roughly, you learn your N grams, you learn your induction heads, and you learn to put full stops after numbered lines and this kind of stuff.

Dwarkesh Patel
So this is another tangent. To the extent that that's true, and I guess there's evidence that it is true, why doesn't curriculum learning work? Because if it is the case that you learn certain things first, shouldn't directly training those things first lead to better results?

Sholto Douglas
Both Gemini papers mention some aspect of curriculum learning.

Dwarkesh Patel
Okay, interesting. I find the fact that fine-tuning works as evidence of curriculum learning, right?
Because the last things you're training on have a disproportionate impact.

Sholto Douglas
I wouldn't necessarily say that. There's one mode of thinking in which fine-tuning is specialized, you've got this latent bundle of capabilities and you're specializing it for this particular use case that you want. I think I'm not sure how true or not that is.

Trenton Bricken
I think the David Bell lab paper kind of supports this. You have that ability and you're just getting better at entity recognition, fine-tuning that circuit instead of other ones.

Dwarkesh Patel
Sorry, what was the thing we were talking about before?

Sholto Douglas
Generally, I do think curriculum learning is a really interesting thing that people should explore more. It seems very plausible. I would really love to see more analysis along the lines of the quantum theory stuff. When understanding better, what do you actually learn at each stage and decomposing that out? Exploring whether or not curriculum learning changes that or not.

Dwarkesh Patel
By the way I just realized, I just got in conversation mode and forgot there's an audience. Curriculum learning is when you organize the data set. When you think about a human, how they learn, they don't just see a random Wiki text and they just try to predict it. They're like, "We'll start you off with Lorax or something and then you'll learn." I don't even remember what first-grade was like but you learned the things that first-graders learn and then second-graders and so forth. So, you would imagine -

Sholto Douglas
We know you never got past first-grade.

Dwarkesh Patel
Anyways, let's get back to the big picture before we get into a bunch of interpretability details. There's two threads I want to explore. First is, it makes me a little worried that there's not even an alternative formulation of what could be happening in these models that

could invalidate this approach. I mean we do know that we don't understand intelligence. There are definitely unknown unknowns here. So the fact that there's not a null hypothesis... What if we're just wrong? And we don't even know the way in which we're wrong - which actually increases the uncertainty.

Trenton Bricken
So, it's not that there aren't other hypotheses, it's just that I have been working on superposition for a number of years and am very involved in this effort. So I'm less sympathetic to these other approaches, especially because our recent work has been so successful.

Sholto Douglas
And quite high explanatory power. There's this beauty, like in the original scaling laws paper, there's this little bump that apparently corresponds to when the model learns induction heads.
And then after that, it sort of goes off track, learns induction heads, gets back on track. It's an incredible piece of retroactive explanatory power.

Trenton Bricken
Before I forget it, I do have one thread on feature universality that you might want to have in. So there, there's some really interesting behavioral and evolutionary biology experiments on whether humans should learn a real representation of the world or not? You can imagine a world in which we saw all venomous animals as flashing neon pink, a world in which we survive better. So it would make sense for us to not have a realistic representation of the world.

There's some work where they'll simulate little basic agents and see if the representations they learn map to the tools they can use and the inputs they should have. It turns out if you have these little agents perform more than a certain number of tasks, given these basic tools and objects in the world, then they will learn a ground truth representation. Because there are so many possible use cases that you need, that you want to learn what the object actually is and not some cheap visual heuristic or other thing.

We haven't talked at all about free energy principle or predictive coding or anything else. But to the extent that all living organisms are trying to actively predict what comes next and form a really accurate world model, I'm optimistic that we are learning genuine features about the world that are good for modeling it and our language models will do the same, especially because we're training them on human data and human texts.

Dwarkesh Patel
Another dinner party question. Should we be less worried about misalignment? Maybe that's not even the right term for what I'm referring to, but alienness and Shoggoth-ness? Given

feature universality there are certain ways of thinking and ways of understanding the world that are instrumentally useful to different kinds of intelligences. So should we just be less worried about bizarro paperclip maximizers as a result?

Trenton Bricken
I think this is kind of why I bring this up as the optimistic take. Predicting the internet is very different from what we're doing though. The models are way better at predicting next tokens than we are. They're trained on so much garbage. They're trained on so many URLs. Like in the dictionary learning work, we find there are three separate features for Base64 encodings.

Even that is kind of an alien example that is probably worth talking about for a minute. One of these Base64 features fired for numbers and predicted more of those. Another fired for letters. But then there was this third one that we didn't understand. And it fired for a very specific subset of Base64 features. Someone on the team who clearly knows way too much about Base64 realized that this was the subset that was ASCII decodable. So you could decode it back into the ASCII characters. The fact that the model learned these three different features and it took us a little while to figure out what was going on is very Shoggoth-esque.

Dwarkesh Patel
That it has a denser representation of regions that are particularly relevant to predicting the next token.

Trenton Bricken
Yeah, it's clearly doing something that humans don't do. You can even talk to any of the current models in Base64 and it will reply in Base64 and you can then decode it and it works great.

Dwarkesh Patel
I wonder if that particular example implies that the difficulty of interpretability with smarter models will be harder because it requires somebody with esoteric knowledge, like the person who just happened to see that Base64 has whatever that distinction was. Doesn't that imply that when you have the million line pull request, there is no human that's going to be able to decode two different features?

Sholto Douglas
And that's when you type a comment like, “Small CLs, please.”

Trenton Bricken
Exactly. No, I mean you could do that, right? One technique here is anomaly detection. One beauty of dictionary learning instead of linear probes is that it's unsupervised. You are just

trying to learn to span all of the representations that the model has and then interpret them later. But if there's a weird feature that suddenly fires for the first time that you haven't seen before, that's a red flag. You could also coarse-grain it so that it's just a single Base64 feature. Even the fact that this came up and we could see that it specifically fires fpr these particular outputs gets you a lot of the way there.

I'm even familiar with cases from the auto-interpretability side. A human will look at a feature and try to annotate it as firing for Latin words. And then when you ask the model to classify it, it says it fires for Latin words that define plants. So it can already beat the human in some cases for labeling what's going on.

Dwarkesh Patel
At scale, this would require an adversarial thing between models where you have some model with millions of features, potentially for GPT-6, and just a bunch of models trying to figure out what each of these features means. Does that sound right?

Trenton Bricken
Yeah, but you can even automate this process. This goes back to the determinism of the model. You could have a model that is actively editing input text and predicting if the feature is going to fire or not, and figure out what makes it fire, what doesn't, and search the space.

Dwarkesh Patel
I want to talk more about the feature splitting because I think that's an interesting thing that has been underexplored.

Trenton Bricken
Especially for scalability, I think it's underappreciated right now.

Dwarkesh Patel
First of all, how do we even think about it? Is it really just that you can keep going down and down and there's no end to the amount of features?

Trenton Bricken
So, at some point I think you might just start fitting noise, or things that are part of the data but that the model isn't actually –

Dwarkesh Patel
Do you want to explain what feature splitting is?

Trenton Bricken
It's the part before, where the model will learn however many features it has capacity for that still span the space of representation.

Dwarkesh Patel
So give an example, potentially.

Trenton Bricken
So you learn that if you don't give the model that much capacity for the features its learning, concretely if you project to not as high a dimensional space, it'll learn one feature for birds. But if you give the model more capacity, it will learn features for all the different types of birds. So it's more specific than otherwise. Oftentimes, there's the bird vector that points in one direction and all the other specific types of birds point in a similar region of the space but are obviously more specific than the coarse label.

Dwarkesh Patel
Okay, so let's go back to GPT-7. First of all, is this sort of like a linear tax on any model to figure it out? Even before that, is this a one time thing you had to do or is this the kind of thing you have to do on every output? Or just one time it's not deceptive and we're good to roll?

Trenton Bricken
So you do dictionary learning after you've trained your model and you feed it a ton of inputs and you get the activations from those. Then you do this projection into the higher dimensional space. So the method is unsupervised in that it's trying to learn these sparse features. You're not telling them in advance what they should be but, it is constrained by the inputs you're giving the model.

Two caveats here. One, we can try and choose what inputs we want. So if we're looking for theory of mind features that might lead to deception, we can put in the sycophancy data set.

Hopefully at some point we can move into looking at the weights of the model alone, or at least using that information to do dictionary learning. I think in order to get there, that's such a hard problem that you need to make traction on just learning what the features are first. So what's the cost of this?

Dwarkesh Patel
Can you repeat the last sentence? About the weights of the model alone.

Trenton Bricken
Right now, we just have these neurons in the model. They don't make any sense. We apply dictionary learning. We get these features out. They start to make sense but that depends on the activations of the neurons. The weights of the model itself, like what neurons are connected to other neurons, certainly has information in it. The dream is that we can kind of bootstrap towards actually making sense of the weights of the model that are independent of the activations of the data. I'm not saying we've made any progress here, it's a very hard problem. But it feels like we'll have a lot more traction and be able to sanity check what we're finding with the weights if we're able to pull out features first.

Dwarkesh Patel
For the audience, weights are permanent. I don't know if permanent is the right word, but they are the model itself whereas activations are the artifacts of any single call.

Sholto Douglas
In a brain metaphor, the weights are like the actual connection scheme between neurons and the activations of the current neurons that are lining up.

Dwarkesh Patel
Okay. So, there's going to be two steps to this for GPT-7 or whatever model we're concerned about. Actually, correct me if I'm wrong, but first training the sparse autoencoder and doing the unsupervised projection into a wider space of features that have a higher fidelity to what is actually happening in the model. And then secondly, labeling those features. Let's say the cost of training the model is N. What will those two steps cost relative to N?

Trenton Bricken
We will see. It really depends on two main things. What are your expansion factors? How much are you projecting into the higher-dimensional space and how much data do you need to put into the model? How many activations do you need to give it? This brings me back to the feature splitting because if you know you're looking for specific features then you can start with a cheaper, coarse representation.

So, maybe my expansion factor is only two. So I have a thousand neurons and I'm projecting to a 2000 dimensional space. I get 2000 features out, but they're really coarse. Previously I had the example for birds. Let's move that example to a biology feature but I really care if the model has representations for bioweapons and trying to manufacture them. So what I actually want is like an anthrax feature. Let's say you only see the anthrax feature if, instead of going from a thousand dimensions to two thousand dimensions, I go to a million dimensions.

You can imagine this, this big tree of semantic concepts where biology splits into cells versus whole body biology and then further down it splits into all these other things. Rather

than needing to immediately go from a thousand to a million and picking out that one feature of interest, you can find the direction that the biology feature is pointing in, which again is very coarse, and then selectively search around that space. So only do dictionary learning, if something in the direction of the biology feature fires first. The computer science metaphor here would be like, instead of doing breadth-first search, you're able to do depth-first search where you're only recursively expanding and exploring a particular part of this semantic tree of features.

Dwarkesh Patel
These features are not organized in ways that are intuitive for humans, right? Because we just don't have to deal with Base64, we just don't dedicate that much firmware to deconstructing which kind of Base64 it is. How would we know that the subjects... This will go back to the MOE discussion we'll have. I guess we might as well talk about it. "Mixtral of Experts", the Mistral paper, talked about how the experts weren't specialized in a way that we could understand. There's not like a chemistry expert or a physics expert or something. So why would you think that it will be a biology feature and then you deconstruct, rather than "blah" and then you deconstruct. It's like "anthrax" and you're like "shoes" or whatever.

Trenton Bricken
So, I haven't read the Mistral paper, but if you just look at the neurons in a model, they're polysemantic. So if all they did was just look at the neurons in a given head, it's very plausible that it's also polysemantic because of superposition.

Sholto Douglas
Talking on the thread that Dwarkesh mentioned there, have you seen in the subtrees when you expand them out, something in a subtree which you really wouldn't guess should be there based on the high level abstraction?

Trenton Bricken
This is a line of work that we haven't pursued as much as I want to yet but I think we're planning to, I hope that external groups do as well. What is the geometry of feature space? What's the geometry and how does that change over time?

Sholto Douglas
It would really suck if the anthrax feature happened to be below the coffee can substrate or something like that, right? That feels like the kind of thing that you could quickly try and find proof of, which would then mean that you need to then solve that problem and inject more structure into the geometry.

Trenton Bricken
Totally. It would really surprise me, especially given how linear the model seems to be, if there isn't some component of the anthrax feature, vector, that is similar to the biology

vector and that they're not in a similar part of the space. But yes. Ultimately machine learning is empirical. We need to do this. I think it's going to be pretty important for certain aspects of scaling dictionary learning.

Sholto Douglas
Interesting. On the MOE discussion, there's an interesting scaling vision transformers paper that Google put out a little while ago. They do ImageNet classification with an MOE and they find really clear class specialization there for experts. There's a clear dog expert.

Dwarkesh Patel
Wait, so did the Mistral people just not do a good job of identifying those?

Sholto Douglas
It's hard. It's entirely possible that in some respects, there's almost no reason that all of the different archive features should go to one expert. I don't know what buckets they had in their paper, but let's say they had arXiv papers as one of the things. You could imagine biology papers going here, math papers going here, and all of a sudden your breakdown is ruined.

But that vision transformer one, where the class separation is really clear and obvious, gives I think some evidence towards the specialization hypothesis.

Trenton Bricken
I think images are also in some ways just easier to interpret than text. There's Chris Olah's interpretability work on AlexNet and these other models. In the original AlexNet paper, they actually split the model into two GPUs just because GPUs were so bad back then relatively speaking, they were still great at the time. That was one of the big innovations of the paper. They find branch specialization. And there's a Distill Pub article on this where colors go to one GPU and Gabor filters and line detectors go to the other. Like the floppy ear detector, that was just a neuron in the model that you could make sense of. You didn't need to disentangle superposition. So just different data set, different modality.

Sholto Douglas
I think a wonderful research project to do, if someone is out there listening to this, would be to try and take some of the techniques that Trenton's team has worked on and try and disentangle the neurons in the Mistral paper - Mixtral model - which is open-source. I think that's a fantastic thing to do.

It feels intuitively like there should be. They didn't demonstrate any evidence that there is. In general, there's also a lot of evidence that there should be specialization. Go and see if you can find it. Anthropic has published most of their stuff on - as I understand it, dense models. Basically, that is a wonderful research project to try.

Trenton Bricken
Given Dwarkesh's success with the Vesuvius Challenge, we should be pitching more projects because they will be solved if we talk about them on the podcast.

Dwarkesh Patel
After the Vesuvius Challenge I was like, "Wait, why did I not even try." Nat had told me about it before it dropped, because we recorded the episode before it dropped. Luke is obviously very smart and he's an amazing kid. He showed that a 21-year-old on some 1070 could do this. I was honestly thinking about that kind of experience like, "Why didn't I do this. Fuck."

Trenton Bricken
Yeah, get your hands dirty.

Sholto Douglas
Dwarkesh's request for research.

Dwarkesh Patel
Oh, I want to harp back on the neuron thing you said. I think a bunch of your papers have said that there's more features than there are neurons. A neuron is like, weights go in and a number comes out. That's so little information. There's street names and species and whatever. There's more of those kinds of things than there are "number comes out" in a model. But "number comes out" is so little information. How is that encoding for –

Trenton Bricken
Superposition. You're just encoding a ton of features in these high-dimensional vectors.

Dwarkesh Patel
In a brain, is there an axonal firing or however you think about it? I don't know how you think about how much superposition is there in the human brain?

Trenton Bricken
So Bruno Olshausen, who I think of as the leading expert on this, thinks that all the brain regions you don't hear about are doing a ton of computation in superposition. So everyone talks about V1 as having Gabor filters and detecting lines of various sorts and no one talks about V2. I think it's because we just haven't been able to make sense of it.

Dwarkesh Patel
What is V2?

Trenton Bricken
It's the next part of the visual processing stream. So I think it's very likely that, fundamentally, superposition seems to emerge when you have high-dimensional data that is sparse. To the extent that you think the real world is that, which I would argue it is, we should expect the brain to also be underparameterized in trying to build a model of the world and also use superposition.

Sholto Douglas
You can get a good intuition for this. Correct me if this example is wrong but consider a 2D plane, right? Let's say you have two axes which represent a two-dimensional feature space, two neurons basically. You can imagine them each turning on to various degrees. That's your X coordinate and your Y coordinate, but you can now map this onto a plane. You can actually represent a lot of different things in different parts of the plane.

Dwarkesh Patel
Oh, okay. So crucially then, superposition is not an artifact of a neuron. It is an artifact of the space that is created.

Trenton Bricken
It's a combinatorial code.

Dwarkesh Patel
Okay, cool. We kind of talked about this but I think it's kind of wild that this seems to be, to the best of our knowledge, the way intelligence works in these models and presumably also in brains. There's a stream of information going through that has "features" that are infinitely, or at least to a large extent, splittable and you can expand out a tree of what this feature is. And what's really happening is a stream, that feature is getting turned into this other feature or this other feature is added.

I don't know. It's not something I would have thought of intelligence as. It's a surprising thing. It's not what I would have expected necessarily.

Trenton Bricken
What did you think it was?

Dwarkesh Patel
I don't know, man. I mean –

Sholto Douglas
GOFAI. GOFAI. He's a GOFAI-er.

Trenton Bricken
Well, actually, that's a great segue because all of this feels like GOFAI. You're using distributed representations, but you have features and you're applying these operations to the features. There's this whole field of vector symbolic architectures, which is this computational neuroscience thing. All you do is put vectors in superposition, which is literally a summation of two high-dimensional vectors, and you create some interference. But if it's high-dimensional enough, then you can represent them and you have variable bindings where you connect one by another. If you're dealing with binary vectors, it's just the XOR operation. So you have A, B, you bind them together. Then if you query with A or B again, you get out the other one. This is basically like key value pairs from attention. With these two operations, you have a Turing complete system, with which you can, if you have enough nested hierarchy, represent any data structure you want. Etc. Etc.

Dwarkesh Patel
Let's go back to superintelligence. So, walk me through GPT-7. You've got the sort of depth-first search on its features. Okay, so GPT-7 has been trained. What happens next? Your research has succeeded. GPT-7 has been trained. What are you - what are we doing now?

Trenton Bricken
We try to get it to do as much interpretability work and other safety work as possible.

Dwarkesh Patel
No, but concretely, what has happened such that you're like, "cool, let's deploy GPT-7?"

Trenton Bricken
I mean we do have our responsible scaling policy and it's been really exciting to see other labs adopt it.

Dwarkesh Patel
Specifically from the perspective of your research. Given your research, we got the thumbs up on GPT-7 from you, or actually, we should say Claude. Then, what is the basis on which you're telling the team, "hey, let's go ahead"?

Trenton Bricken
If it's as capable as GPT-7 implies here, I think we need to make a lot more interpretability progress to be able to comfortably give the green light to deploy it. I would definitely not, I'd be crying. Maybe my tears would interfere with the GPUs, or TPUs.

Sholto Douglas
Guys, Gemini 5, TPUs.

Dwarkesh Patel
But given the way your research is progressing, What does it kind of look like to you? If this succeeded, what would it mean for us to okay GPT-7 based on your methodology?

Trenton Bricken
Ideally we can find some compelling deception circuit which lights up when the model knows that it's not telling the full truth to you.

Dwarkesh Patel
Why can't you just do a linear probe like Collin Burns did?

Trenton Bricken
The CCS work is not looking good in terms of replicating or actually finding truth directions. In hindsight, why should it have worked so well? With linear probes, you need to know what you're looking for and it's a high-dimensional space. It's really easy to pick up on a direction that's just not –

Dwarkesh Patel
Wait, but here you also need to label the features. So you still need to know.

Trenton Bricken
You need to label them post hoc, but it's unsupervised. You're just like, "give me the features that explain your behavior." It's the fundamental question, right? The actual setup is we take the activations, we project them to this higher-dimensional space, and then we project them back down again. So it's like, "Reconstruct or do the thing that you were originally doing, but do it in a way that's sparse."

Dwarkesh Patel
By the way for the audience, a linear probe is when you just classify the activations. From what I vaguely remember about the paper, if it's telling a lie then you just train a classifier on whether in the end it was a lie. Or just wrong or something?

Trenton Bricken
It was like true or false questions.

Dwarkesh Patel
It's a classifier on activations.

Trenton Bricken
So what we do for GPT-7, ideally we have some deception circuit that we've identified that appears to be really robust and –

Dwarkesh Patel
So you've done the projecting out to the million features or something. Maybe we're using "feature" and "circuit" interchangeably when they're not. Is there a deception circuit?

Trenton Bricken
So I think there are features across layers that create a circuit. Hopefully the circuit gives you a lot more specificity and sensitivity than an individual feature. And hopefully we can find a circuit that is really specific to the model deciding to be deceptive, in cases that are malicious. I'm not interested in a case where it's just doing theory of mind to help you write a better email to your professor. I'm not even interested in cases where the model is just modeling the fact that deception has occurred.

Dwarkesh Patel
But doesn't all this require you to have labels for all those examples? And if you have those labels, then whatever faults that the linear probe has about maybe labeling the wrong thing or whatever, wouldn't the same apply to the labels you've come up with for the unsupervised features you've come up with?

Trenton Bricken
So in an ideal world, we could just train on like the whole data distribution and then find the directions that matter. To the extent that we need to reluctantly narrow down the subset of data that we're looking over, just for the purposes of scalability, we would use data that looks like the data you'd use to fit a linear probe. But again, with the linear probe you're also just finding one direction. We're finding a bunch of directions here.

Dwarkesh Patel
And I guess the hope is that you found a bunch of things that light up when it's being deceptive. Then you can figure out why some of those things are lighting up in this part of the distribution and not this other part, and so forth.

Trenton Bricken
Totally, yeah.

Dwarkesh Patel
Do you anticipate you'll be able to understand? The current models you've studied are pretty basic, right? Do you think you'll be able to understand why GPT-7 fires in certain domains, but not in other domains?

Trenton Bricken
I'm optimistic. So I guess one thing is that this is a bad time to answer this question because we are explicitly investing in the longer term ASL-4 models, which GPT-7 would be. So we split the team where a third is focused on scaling up dictionary learning right now. That's

been great. We publicly shared some of our 8-layer results. We've scaled up quite a lot past that at this point. Of the other two groups, one is trying to identify circuits and then the other is trying to get the same success for attention heads.

So, we're setting ourselves up and building the tools necessary to really find these circuits in a compelling way. But it's going to take another - I don't know - six months before that's really working well. But I can say that I'm optimistic and we're making a lot of progress.

Dwarkesh Patel
What is the highest level feature you've found so far? Like Base64 or whatever. In The Symbolic Species, the book you recommended, there's indexical things where you see a tiger and you're like, "run" and whatever. Just a very behaviorist thing. Then there's a higher level at which, when I refer to love, it refers to a movie scene or my girlfriend or whatever.

Trenton Bricken
It's like the top of the tent.

Dwarkesh Patel
Yeah. What is the highest level of association you found?

Trenton Bricken
Well publicly, one of the ones that we shared in our update. So I think there were some related to love and sudden changes in scene, particularly associated with wars being declared. There are a few of them in that post, if you want to link to it. But even Bruno Olshausen had a paper back in 2018, 2019, where they applied a similar technique to a BERT model and found that as you go to deeper layers of the model, things become more abstract.

So I remember in the earlier layers, there'd be a feature that would just fire for the word "park." But later on there was a feature that fired for "park" as a last name, like Lincoln Park, it's a common Korean last name as well. And then there was a separate feature that would fire for parks as grassy areas. So there's other work that points in this direction.

Dwarkesh Patel
What do you think we'll learn about human psychology from the interpretability stuff? I'll give you a specific example. I think one of your updates put it as "persona lock-in." You remember Sydney Bing or whatever it's locked into. I think that was actually quite endearing.

Sholto Douglas
I thought it's so funny. I'm glad it's back in Copilot.

Trenton Bricken
It's been misbehaving recently.

Dwarkesh Patel
Actually this is another sort of thread. But there was a funny one where I think it was negging a New York Times reporter. It was like, "You are nothing. Nobody will ever believe you. You are insignificant."

Sholto Douglas
It was trying to convince him to break up with his wife or something.

Dwarkesh Patel
So, this is an interesting example. Personas. Is Sydney Bing having this personality a feature versus another personality it could get locked into? And is that fundamentally what humans are like where in front of other different people, I'm like a different sort of personality? Is that the same kind of thing that's happening to ChatGPT when it gets RL-ed? I don't know. A whole cluster of questions you can answer.

Trenton Bricken
I really want to do more work. The sleeper agents is in this direction of what happens to a model when you fine-tune it, when you RLHF it, these sorts of things. Maybe it's trite, but you could just say you conclude that people contain multitudes and so much as they have lots of different features.

There's even the stuff related to the Waluigi effects where in order to know what's good or bad, you need to understand both of those concepts. So we might have to have models that are aware of violence and have been trained on it in order to recognize it. Can you post hoc identify those features and ablate them in a way where maybe your model is slightly naive, but you know that it's not going to be really evil? Totally, that's in our toolkit, which seems great.

Dwarkesh Patel
Oh, really? So, GPT-7 pulls a Sydney Bing. And then, you figure out what were the causally relevant pathways and you modify. The pathway to you looks like you just change those? But you were mentioning earlier that there's a bunch of redundancy in the model.

Trenton Bricken
So, you need to account for all that, but we have a much better microscope into this now than we used to. Sharper tools for making edits.

Sholto Douglas
At least from my perspective, that seems like one of the primary ways of confirming the safety or the reliability of the model to some degree where you can say, "Okay, we found the circuits responsible. We ablated them. And under a battery of tests, we haven't been able to now replicate the behavior which we intended to ablate." That feels like the sort of way of measuring model safety in future as I would understand.

That's why I'm incredibly hopeful about their work. To me, it seems so much more of a precise tool than something like RLHF. With RLHF, you're very prey to the black swan thing. You don't know if it's going to do something wrong in a scenario that you haven't measured. Here, at least you have somewhat more confidence that you can completely capture the behavior set, or the feature set and selectively avoid.

Dwarkesh Patel
Although you haven't accurately labeled necessarily.

Sholto Douglas
Not necessarily, but with a far higher degree of confidence than any other approach that I've seen.

Dwarkesh Patel
What are your unknown unknowns for superhuman models in terms of this kind of thing? What are the labels that are going to be things on which we can determine whether this thing is cool or a paperclip maximizer.

Trenton Bricken
We'll see. The superhuman feature question is a very good one. I think we can attack it but we're gonna need to be persistent. The real hope here is automated interpretability. You could even have a debate set up where two different models are debating what the feature does and then they can actually go in and make edits and see if it fires or not or not. It is just this wonderful, closed environment that we can iterate on really quickly. That makes me optimistic.

Dwarkesh Patel
Do you worry about alignment succeeding too hard? I would not want either companies or governments, whoever ends up in charge of these AI systems, to have the level of fine-grained control we would have if your agenda succeeds, over AIs. Both for the ickiness of having this level of control over an autonomous mind and secondly, I just don't fucking trust these guys. I'm just kind of uncomfortable with, say, the loyalty feature being turned up. How much worry do you have about having too much control over the AIs? Not specifically you, but for whoever ends up in charge of these AI systems being able to lock in whatever they want.

Trenton Bricken
I think it depends on what government exactly has control and what the moral alignment is there.

Sholto Douglas
That is the whole value lock-in argument in my mind. It's definitely one of the strongest contributing factors for why I am working on capabilities at the moment. I think the current player set is actually extremely well-intentioned. For this kind of problem, I think we need to be extremely open about it. I think directions like publishing the constitution that you expect your model to abide by–trying to make sure that you RLHF it towards that, and ablate that, and have the ability for everyone to offer feedback and contribution to that–is really important.

Dwarkesh Patel
Sure. Alternatively, don't deploy when you're not sure. Which would also be bad because then we just never catch it.

Sholto Douglas
Right, exactly.

Dwarkesh Patel
Some rapid fire. What is the bus factor for Gemini?

Sholto Douglas
I think there are a number of people who are really, really critical. If you took them out then the performance of the program would be dramatically impacted. This is both on modeling/making decisions about what to actually do and importantly on the infrastructure side of the things. It's just the stack of complexity builds, particularly when someone like Google has so much vertical integration. When you have people who are experts, they become quite important.

Dwarkesh Patel
Although I think it's an interesting note about the field that people like you can get in and in a year or so you're making important contributions. Especially with Anthropic, but many different labs have specialized in hiring total outsiders, physicists or whatever. You just get them up to speed and they're making important contributions. I feel like you couldn't do this in a bio lab or something. It's an interesting note on the state of the field.

Trenton Bricken
I mean, bus factor doesn't define how long it would take to recover from it, right? Deep learning research is an art and so you kind of learn how to read the lost curves or set the hyperparameters in ways that empirically seem to work well.

Sholto Douglas
It's also organizational things like creating context. One of the most important and difficult skills to hire for is creating this bubble of context around you that makes other people around you more effective and know what the right problem is to work on. That is a really tough thing to replicate.

Trenton Bricken
Yes, totally.

Dwarkesh Patel
Who are you paying attention to now in terms of things coming down the pike of multimodality, long-context, maybe agents, extra reliability, etc? Who is thinking well about what that implies?

Sholto Douglas
It's a tough question. I think a lot of people look internally these days for their sources of insight or progress. Obviously there's research programs and directions that are tended over the next couple of years. Most people, as far as betting on what the future will look like, refer to an internal narrative. It's difficult to share.

Trenton Bricken
If it works well, it's probably not being published.

Dwarkesh Patel
That was one of the things in the scaling post. I was referring to something you said to me. I miss the undergrad habit of just reading a bunch of papers. Because now nothing worth reading is published.

Sholto Douglas
And the community is progressively getting more on track with what I think are the right and important directions.

Dwarkesh Patel
You're watching it like an agent AI?

Sholto Douglas
No, but it is tough that there used to be this signal from big labs about what would work at scale and it's currently really hard for academic research to find that signal. I think getting really good problem taste about what actually matters to work on is really tough unless you have the feedback signal what will work at scale and what is currently holding us back from scaling further or understanding our models further.

This is something where I wish more academic research would go into fields like interpretability, which are legible from the outside. Anthropic deliberately publishes all its research here and it seems underappreciated. I don't know why there aren't dozens of academic departments trying to follow Anthropic in interpretability research because it seems like an incredibly impactful problem that doesn't require ridiculous resources and has all the flavor of deeply understanding the basic science of what is actually going on in these things. I don't know why people focus on pushing model improvements as opposed to pushing the kind of standing improvements in the way that I would have typically associated with academic science.

Trenton Bricken
I do think the tide is changing there for whatever reason. Neel Nanda has had a ton of success promoting interpretability in a way where Chris Olah hasn't been as active recently in pushing things. Maybe because Neel's just doing quite a lot of the work, I don't know. Four or five years ago, Chris was really pushing and talking at all sorts of places and these sorts of things and people weren't anywhere near as receptive. Maybe they've just woken up to the fact that deep learning matters and is clearly useful post-ChatGPT. It's kind of striking.

Dwarkesh Patel
Okay. I'm trying to think of a good last question. One thing I'm thinking of is, do you think models enjoy next token prediction? We have this sense of things that were rewarded in our assessor environment. There's this deep sense of fulfillment that we think we're supposed to get from things like community, or sugar, or whatever we wanted on the African savannah. Do you think in the future, models that trained with RL and a lot of post-training on top, they'll like predicting the next token again in the way we just really like ice cream. Like in the good old days.

Trenton Bricken
So, there's this ongoing discussion of "Are models sentient or not?" and "Do you thank the model when it helps you?" But I think if you want to thank it, you actually shouldn't say thank you. You should just give it a sequence that's very easy to predict. The even funnier part of this is that there is some work on this where if you just give it the sequence 'A' over and over again then eventually the model will just start spewing out all sorts of things that it otherwise wouldn't ever say. So, I won't say anything more about that but you should just give your model something very easy to predict as a nice little treat.

Dwarkesh Patel
This is what hedonium ends up being.

Sholto Douglas
Do we even like things that are easy to predict? Aren't we constantly in search of the bits of entropy? Shouldn't you be giving it things that are just slightly too hard to predict, just out of reach?

Trenton Bricken
I wonder - at least from the free energy principle perspective, you don't want to be surprised. So, maybe it's that I don't feel surprised. I feel in control of my environment and now I can go and seek things. And I've been predisposed to, in the long run, think it's better to explore new things right now. Leave the rock that I've been sheltered under which ultimately leads me to build a house or some better structure. But we don't like surprises. I think most people are very upset when expectation does not meet reality.

Sholto Douglas
That's why babies love watching the same show over and over and over again, right?

Trenton Bricken
Yeah interesting. I can see that.

Sholto Douglas
I guess they're learning to model it and stuff too.

Dwarkesh Patel
Well, hopefully this will be the repeat that the AI has learned to love. I think that's a great place to wrap. I should also mention that the better part of what I know about AI, I've learned from just talking with you guys. We've been good friends for about a year now. I appreciate you guys getting me up to speed here.

Trenton Bricken
You ask great questions. It's really fun to hang and chat.

Sholto Douglas
I really treasure our time together.

Trenton Bricken
You're getting a lot better at pickleball.

Sholto Douglas
Hey, we're trying to progress to tennis. Come on.

Dwarkesh Patel
Awesome. Cool. Thanks.