Dwarkesh Podcast #73 - John Schulman (OpenAI Cofounder) - Reasoning, RLHF, & Plan for 2027 AGI

Published - May 15, 2024

Transcribed by - thepodtranscripts.com

Dwarkesh Patel
Today I have the pleasure to speak with John Schulman, who is one of the co-founders of OpenAI and leads the post-training team here. He also led the creation of ChatGPT and is the author of many of the most important and widely cited papers in AI and RL, including PPO and many others. John, really excited to chat with you. Thanks for coming on the podcast.

John Schulman
Thanks for having me on the podcast. I'm a big fan.

Dwarkesh Patel
Thank you for saying that. Here's my first question. We have these distinctions between pre-training and post-training. Let's go beyond what is actually happening in terms of loss function and training regimes. Taking a step back conceptually, what kind of thing is pre-training creating? What does post-training do on top of that?

John Schulman
In pre-training you're basically training to imitate all of the content on the Internet or on the web, including websites and code and so forth. So you get a model that can generate content that looks like random web pages from the Internet. The model is also trained to maximize likelihood where it has to put a probability on everything.

The objective is basically predicting the next token given the previous tokens. Tokens are like words, or parts of words. Since the model has to put a probability on it—we're training to maximize log probability—it ends up being very calibrated. Not only can it generate all of the content of the web, it can also assign probabilities to everything.

The base model can effectively take on all of these different personas or generate all different kinds of content. When we do post-training, we're usually targeting a narrower range of behaviors where we want the model to behave like a kind of chat assistant. It's a more specific persona where it's trying to be helpful. It's not trying to imitate a person. It's answering your questions or doing your tasks. We're optimizing on a different objective, which is more about producing outputs that humans will like and find useful, as opposed to just imitating this raw content from the web.

Dwarkesh Patel
Maybe I should take a step back and ask this. Right now we have these models that are pretty good at acting as chatbots. Taking a step back from how these processes work currently, what kinds of things will the models released by the end of the year be capable of doing? What do you think the progress will look like if we carry everything forward for the next five years?

John Schulman
The models will get quite a bit better in five years.

Dwarkesh Patel
In what way?

John Schulman
Even in one or two years, we'll find that the models can do a lot more involved tasks than they can do now. For example, you could imagine having the models carry out a whole coding project instead of it giving you one suggestion on how to write a function. You could imagine the model taking high-level instructions on what to code and going out on its own, writing any files, and testing it, and looking at the output. It might even iterate on that a bit. So just much more complex tasks.

Dwarkesh Patel
Fundamentally the unlock is that it can act coherently for long enough to write multiple files of code? What has changed between now and then?

John Schulman
I would say this will come from some combination of training the models to do harder tasks like this. Most of the training data is more like doing single steps at a time. I would expect us to do more for training the models to carry out these longer projects.

That's for any kind of training, like doing RL, to learn how to do these tasks. Whether you're supervising the final output or supervising it at each step, any kind of training at carrying out these long projects is going to make the models a lot better.

Since the whole area is pretty new, I'd say there's a lot of low-hanging fruit in doing this kind of training. That's one thing. I would also expect that as models get better, they get better at recovering from errors or dealing with edge cases. When things go wrong, they'll know how to recover from it.

The models will be more sample efficient. You won't have to collect a ton of data to teach them how to get back on track. Just a little bit of data or their generalization from other abilities will allow them to get back on track. Current models might just get stuck and get lost.

Dwarkesh Patel
I want to understand specifically how the generalization helps you get back on track. Can you say more about that? I'm not sure why those two concepts are connected.

John Schulman
Right, they're not directly connected. You usually have a little bit of data that does everything. If you collect a diverse data set, you're going to get a little bit of everything in it. If you have models that generalize really well—even from just a couple of examples of getting back on track or if in the pre-training data there are a couple of examples of a model getting back on track—the model will be able to generalize from those other things it's seen to the current situation.

If you have models that are weaker, you might be able to get them to do almost anything with enough data. But you might have to put a lot of effort into a particular domain or skill. Whereas for a stronger model, it might just do the right thing without any training data or any effort.

Dwarkesh Patel
Right now these models can work coherently for five minutes. We want them to be able to do tasks that a human would take an hour to do, then a week, then a month, and so forth.

To get to each of these benchmarks, is it going to be the case that each one takes 10X more compute, analogous to the current scaling laws for pre-training? Or is it going to be a much more streamlined process of just getting to that point where you're already more sample efficient and you can just go straight to the years of carrying out tasks or something?

John Schulman
At a high level, I would agree that longer-horizon tasks are going to require more model intelligence to do well. They are going to be more expensive to train. I'm not sure I would expect a really clean scaling law unless you set it up in a very careful way, or design the experiment in a certain way. There might end up being some phase transitions where once you get to a certain level you can deal with much longer tasks.

For example, when people do planning for different timescales, I'm not sure they use completely different mechanisms. We probably use the same mental machinery thinking about one month from now, one year from now, or a hundred years from now. We're not actually doing some kind of reinforcement learning where we need to worry about a discount factor that covers that timescale and so forth.

Using language, you can describe all of these different timescales and then you can do things like plan. In the moment you can try to make progress towards your goal, whether it's a month away or 10 years away. I don't know if it's a phase transition but I might expect the same out of models where there might be some capabilities that work at multiple scales.

Dwarkesh Patel
Correct me if this is wrong. It seems like you're implying that right now we have models that are on a per token basis pretty smart. They might be as smart as the smartest humans on a per token basis. The thing that prevents them from being as useful as they could be is that five minutes from now, they're not going to be still writing your code in a way that's coherent and aligns with your broader goals you have for your project or something.

If it's the case that once you start this long-horizon RL training regime it immediately unlocks your ability to be coherent for longer periods of time, should we be predicting something that is human-level as soon as that regime is unlocked? If not, then what is remaining after we can plan for a year and execute projects that take that long?

John Schulman
It's not totally clear what we're going to see once we get into that regime or how fast progress will be. That's still uncertain. I wouldn't expect everything to be immediately solved by doing any training like this. There'll be other miscellaneous deficits that the models have that cause them to get stuck or make worse decisions than humans. I don't expect that this one little thing will unlock all capabilities. But some improvement in the ability to do long-horizon tasks might go quite far.

Dwarkesh Patel
Would you say it's plausible? Does it seem quite likely that there will be other reasons why there might be bottlenecks? I'm also curious what the nature of these bottlenecks might be. It has all these representations of pre-training. Now it can work coherently for a long period of time because of long-horizon RL. What's remaining?

John Schulman
Maybe there's some other experience that human experts bring to different tasks such as having taste or dealing with ambiguity better. If we want to do something like research I could imagine those considerations coming into play. Obviously there are going to be mundane limitations around the affordances of the model and whether it can use UIs, interact with the physical world, or have access to things. So there might be a lot of mundane barriers that are probably not going to last that long but would initially slow down progress.

Dwarkesh Patel
Let's talk about the websites that are designed for these AIs. Once they're trained on more multimodal data, will they be in any way different from the ones we have for humans? What UIs will be needed? How will it compensate for their strengths and weaknesses? How would that look different from the current UIs we have for humans?

John Schulman
That's an interesting question. I expect that models will be able to use websites that are designed for humans just by using vision, after the vision capabilities get a bit better. So there wouldn't be an immediate need to change them.

On the other hand, there'll be some websites that are going to benefit a lot from AIs being able to use them. We'll probably want to design those to be better UXs for Ais. I'm not sure exactly what that would mean. Assuming that our models are still better at text mode than reading text out of images, you'd probably want to have a good text-based representation for the models.

You'd also want a good indication of what all the things that can be interacted with are. But I wouldn't expect the web to get totally redesigned to have APIs everywhere. We can get models to use the same kind of UIs that humans use.

Dwarkesh Patel
I guess that's been the big lesson of language models, right? That they can act within the similar affordances that humans do.

I want to go back to the point you made earlier about how this process could be more sample efficient because it could generalize from its pre-training experiences of how to get unstuck in different scenarios. What is the strongest evidence you've seen of this generalization and transfer?

The big question for the future abilities models seems to be about how much generalization is happening. Is there something that feels really compelling to you? Have you seen a model learn something that you wouldn't expect it to learn from generalization?

John Schulman
There have definitely been some interesting instances of generalization in post-training.

One well-known phenomenon is that if you do all your fine-tuning with English data, the model will automatically behave well in other languages. So if you train the assistant on English data, it'll also do something reasonable in Spanish. Sometimes you might get the wrong behavior in terms of whether it replies in English or replies in Spanish. Usually you get the right behavior there, meaning you get it to respond in Spanish to Spanish queries. That's one interesting instance of generalization where the model just latches onto the right, helpful persona and then automatically does the right thing in different languages.

We've seen some version of this with multimodal data where if you do text-only fine-tuning, you also get reasonable behavior with images. Early on in ChatGPT we were trying to fix some issues with the model understanding its own limitations. Early versions of the model

would think that it could send you an email or call you an Uber or something. The model would try to play the assistant and it would say "oh yeah, of course I sent that email." Obviously it didn't.

So we started collecting some data to fix those problems. We found that a tiny amount of data did the trick, even when you mixed it together with everything else. I don't remember exactly how many examples but something like 30 examples. We had a pretty small number of examples showing this general behavior, explaining that the model doesn't have this capability. That generalized pretty well to all sorts of capabilities we didn't train for.

Dwarkesh Patel
I still want to go back to this because I'm not sure I understood. Let's say you have this model that is trained to be coherent for longer periods of time. Setting aside these other bottlenecks which there may or may not be, by next could you have models that are potentially like human-level? I'm thinking of a model that you're interacting with like a colleague and it's as good as interacting with a human colleague. You can tell them to go do stuff and they go get it done. What seems wrong with that picture of the capabilities you think might be possible?

John Schulman
It's hard to say exactly what the deficit will be. When you talk to the models today, they have various weaknesses besides long-term coherence. They also struggle to really think hard about things or pay attention to what you ask them. I wouldn't expect improving the coherence a little bit to be all it takes to get to AGI. I guess I can't articulate exactly what are the main weaknesses that will stop them from being a fully functional colleague.

Dwarkesh Patel
It seems like then, you should be planning for the possibility you would have AGI very soon.

John Schulman
I think that would be reasonable.

Dwarkesh Patel
So what's the plan if there's no other bottlenecks. In the next year or something, you've got AGI. What's the plan?

John Schulman
If AGI came way sooner than expected we would definitely want to be careful about it. We might want to slow down a little bit on training and deployment until we're pretty sure we know we can deal with it safely. We would need a pretty good handle on what it's going to do and what it can do. We would have to be very careful if it happened way sooner than expected. Our understanding is still rudimentary in a lot of ways.

Dwarkesh Patel
What would being careful mean? Presumably you're already careful, right? You do these evaluations before deploying.

John Schulman
Maybe it means not training the even smarter version or being really careful when you do train it. You can make sure it's properly sandboxed and everything. Maybe it means not deploying it at scale or being careful about what scale you deploy it at.

Dwarkesh Patel
Let's just play with a scenario. AGI happens next year. You're not training a smarter system but you're deploying it in a somewhat measured way. Presumably the development wouldn't be particular to OpenAI. AGI just turns out to be much easier than we expected and that's why it happened. So you wait to deploy a little bit. Now other companies have a similar level of capabilities. What happens next? While you wait to deploy, what are you waiting for? What is every company doing in this scenario?

John Schulman
The game theory is a little tough to think through. First of all, I don't think this is going to happen next year but it's still useful to have the conversation. It could be two or three years instead.

Dwarkesh Patel
Two or three years is still pretty soon.

John Schulman
It's still pretty soon. You probably need some coordination. Everyone needs to agree on some reasonable limits to deployment or to further training for this to work. Otherwise you have the race dynamics where everyone's always trying to stay ahead and that might require compromising safety. You would probably need some coordination among the larger entities that are doing this kind of training.

Dwarkesh Patel
You'd be coordinating to pause deployment until what exactly? Until you figure out what's happening in the model?

John Schulman
We could pause further training. We could pause deployment. We could avoid certain types of training that might be riskier. We would set up some reasonable rules for what everyone should do to limit these things.

Dwarkesh Patel
Limit to what end? At some point the potential energy that's within this intelligence will be unleashed. Suppose in two years we get the AGI. Now everybody's freaking out. The AI companies have paused. What would be the thing we plan to wait until?

John Schulman
I don't have a good answer to that. If we can coordinate like that, that would be a pretty good scenario. Building these models is very capital intensive and there are a lot of complex pieces. It's not like everyone's going to go and recreate this stuff at home.

Given the relatively small number of entities who could train the largest models, it does seem possible to coordinate. I'm not sure how you would maintain this equilibrium for a long period of time, but I think if we got to that point we would be in an okay position.

Dwarkesh Patel
Would we? I'm still curious because I'm not sure what happens next. Fundamentally, the benefit is that you push it to the server and now we have a bunch of intelligences, or they could push themselves to the server. Now we've got everybody coordinated but I'm not sure what we do next in this world. Why does that set us up for a good outcome?

John Schulman
If we had everyone reasonably coordinated and we felt like we could solve the technical problems around alignment well enough then we could deploy. We would be able to deploy really smart AIs that can act as extensions of people's wills but also prevent them from being catastrophically misused. That would be great. We could go ahead and safely deploy these systems and it would usher in a lot of prosperity and a much more rapid phase of scientific advancement. That would be what the good scenario would look like.

Dwarkesh Patel
That makes sense. I'm curious about something down the road in a couple of years. In the best case scenario, all these actors have agreed to pause until we've figured out that we're building aligned systems that are not themselves going to attempt a coup or not going to enable somebody else to do that. What would proof of that look like? What would evidence of that look like?

John Schulman
If we can deploy systems that are incrementally that are successively smarter than the ones before, that would be safer. I hope the way things play out is not a scenario where everyone has to coordinate, lock things down, and safely release things. That would lead to this big buildup in potential energy.

I would rather have a scenario where we're all continually releasing things that are a little better than what came before. We'd be doing this while making sure we're confident that each diff improves on safety and alignment in correspondence to the improvement in capability. If things started to look a little bit scary, then we would be able to slow things down. That's what I would hope for.

If there's more of a discontinuous jump, there's a question of "how do you know if the thing you've got is safe to release". I can't give a generic answer. However, the type of thing you might want to do to make that more acceptable would be a lot of testing simulated deployment, red teaming of sorts. You'd want to do that in a way that is much more likely to fail than the thing you're planning to do in the real world.

You'd want to have a really good monitoring system so that if something does start to go wrong with the deployed system, you can immediately detect it. Maybe you've got something watching over the deployed AIs, watching what they're doing, and looking for signs of trouble.

You'd want some defense in depth. You'd want some combination of "the model itself seems to be really well-behaved with impeccable, moral confidence in everything" and "I'm pretty confident that it's extremely resistant to any kind of severe misuse." You'd also want really good monitoring on top of it so you could detect any kind of unforeseen trouble.

Dwarkesh Patel
What are you keeping track of while you're doing long-horizon RL or when you eventually start doing it? How could you notice this sort of discontinuous jump before you deployed these systems broadly?

John Schulman
You would want to have a lot of evals that you're running during the training process.

Dwarkesh Patel
What specifically? Does it make sense to train on a long-horizon RL knowing that this is something that could happen? Or is it just a very low possibility? How do you think about this?

John Schulman
You'd want to be pretty careful when you do this kind of training if you see a lot of potentially scary capabilities. I would say it's not something we have to be scared of right now because right now it's hard to get the models to do anything coherent.

If they started to get really good, we would want to take some of these questions seriously. We would want to have a lot of evals that test them for misbehavior, mostly for the

alignment of the models. We'd want to check that they're not going to turn against us or something. You might also want to look for discontinuous jumps in capabilities. You'd want to have lots of evals for the capabilities of the models.

You'd also want to make sure that whatever you're training on doesn't have any reason to make the model turn against you. That doesn't seem like the hardest thing to do. The way we train them with RLHF, that does feel very safe even though the models are very smart. The model is just trying to produce a message that is pleasing to a human. It has no concern about anything else in the world other than whether the text it produces is approved.

Obviously if you were doing something where the model has to carry out a long sequence of actions which involve tools, then it might have some incentive to do a lot of wacky things that wouldn't make sense to a human in the process of producing its final result. However, it wouldn't necessarily have an incentive to do anything other than produce a very high quality output at the end.

There are old points about instrumental convergence where the model wants to take over the world so it can produce some awesome piece of code at the end. If you ask it to write you a Flask app, it'll be like "oh yeah, first I need to take over the world. At a certain point it's a little hard to imagine why for fairly well specified tasks like coding an app, you would want to first take over the world. Of course if you assigned a task such as "make money," then maybe that would lead to some nefarious behavior as an instrumental goal.

Dwarkesh Patel
Before we get back to that, let's step back and talk about today's RLHF systems and everything. I do want to follow up on that point because it is interesting.

With today's RLHF and the way in which it influences these models, how would you characterize it in terms of human psychology? Is it a drive? Is it a goal? Is it an impulse? Psychologically, what kind of thing is it? In what way is it being changed?

Not simply the persona of a chatbot but "don't talk that way, talk this other way" or "don't put out those kinds of outputs."

John Schulman
There are probably some analogies with a drive or a goal in humans. You're trying to steer towards a certain set of states rather than some other states. I would think that our concept of a drive or a goal has other elements, such as the feeling of satisfaction you get for achieving it. Those things have more to do with the learning algorithm than what the model does at runtime when you just have a fixed model.

There are probably some analogies though I don't know exactly how close it is. To some extent, the models do have drives and goals in some meaningful way. In the case of RLHF where you're trying to maximize human approval as measured by a reward model, the model is just trying to produce something that people are going to like and judge as correct.

Dwarkesh Patel
I've heard two ideas in terms of using that internal monologue to get better at reasoning. At least publicly, I've seen two ideas and I'm curious which one you think is more promising.

One is that the model learns from its outputs over a bunch of potential trains of thought, and it learns to follow the one that leads to the correct answer. It is then trained on that before deployment. The other one is you use a bunch of compute to do inference in deployment. This approach involves the model talking to itself while it's deployed.

Which one do you expect to be closer to the way a model has been trained when it gets really good at reasoning? Is it because it's doing just a bunch of inference clouds? Is it just because you've trained it to do well at that?

John Schulman
You could define reasoning as tasks that require some kind of computation at test time or maybe some kind of deduction. By definition, reasoning would be tasks that require some test time computation and step-by-step computation. On the other hand, I would also expect to gain a lot from doing practice at training time. So I think that you'd get the best results by combining these two things.

Dwarkesh Patel
Right now, you have these two ways the model learns. One is in training, whether it's pre-training or post-training. Most of the compute in training is spent on pre-training, glossing over trillions of tokens, skimming trillions of tokens worth of information. If a human was subjected to that, they would just be totally confused. It's just not a very efficient way to learn.

The other way is in-context learning. Of course that is more sample-efficient, but it's destroyed with each instance.

I'm curious if you think that there's a path for something in between those, where it's not destroyed at each instance but it's also not as frivolous as just seeing trillions of tokens. Something more deliberate and active.

John Schulman
Do you mean models having some kind of medium-term memory? Too much to fit in context but much smaller scale than pre-training?

Dwarkesh Patel
It might be memory. I don't have context. Certainly when I'm trying to prepare for this conversation, I think of what I should understand, read it carefully, and maybe think about it as I'm reading it. I'm not sure what it naturally corresponds to in terms of models. What would that look like?

John Schulman
I see. So it's not just memory but it's also somewhat specializing to a certain task or putting a lot of effort into some particular project.

Dwarkesh Patel
I'm not even sure if it's specialization. It's more so "I don't understand this part, so let me look into it more deeply. I already understand this." I guess it's specializing to your existing knowledge base.

John Schulman
I see. So it's not just about training on a bunch of sources that are relevant and fine-tuning on some special domain. It's also about reasoning and developing some knowledge through your own reasoning and using some sort of introspection or self-knowledge to figure out what it needs to learn?

Dwarkesh Patel
Yeah.

John Schulman
That does feel like something that's missing from today's systems. People haven't really pushed too hard on this middle ground between large-scale training—where you produce a single snapshot model that's supposed to do everything like a deployed model—and on the other hand in-context learning.

Part of that is that we've just been increasing context length so much that there hasn't been an incentive for it. If you can go to a hundred thousand or a million context, then that's actually quite a lot. It's not actually the bottleneck in a lot of cases.

I agree that you'd probably also want to supplement that with some kind of fine-tuning. The capabilities you get from fine-tuning and in-context learning are probably somewhat complementary. I'd expect us to want to build systems that do some online learning and also have some cognitive skills, like introspecting on their own knowledge and seeking out new knowledge that fills in the holes.

Dwarkesh Patel
Is this all happening at the same time? Is it just a new training regime where all these things can happen at once, whether it's long-horizon or this kind of training?

Are they separate or not? Is the model smart enough to both introspect and act on longer horizons so that you get adequate reward on the long-horizon tasks?

John Schulman
If you're doing some kind of long-horizon task, you're learning while you do the task, right?

The only way to do something that involves a lot of steps is to have learning and memory that gets updated during the task. There's a continuum between short-term and long-term memory.

I expect the need for this capability would start to become clear when we start to look more at long-horizon tasks. To some extent, putting a lot of stuff into context will take you pretty far because we have really long context now. You probably also want things like fine-tuning.

As for introspection and the ability to do active learning, that might automatically fall out of the models' abilities to know what they know. Models do have some calibration regarding what they know. That's why models don't hallucinate that badly. They have some understanding of their own limitations. That same kind of ability could be used for something like active learning.

Dwarkesh Patel
Interesting. I want to step back and ask about your own history, at least at OpenAI. You led the creation of ChatGPT. At what point did you realize that these LLMs are the path to go? When did you realize a chatbot or some way to instruct them would be useful? Just walk me through the whole lineage from when this became your main focus and what the process was like.

John Schulman
Before ChatGPT, OpenAI had these instruction following models. The idea there was that we had base models that people could prompt them in elaborate ways. But they were also hard to prompt. They basically do autocomplete so you had to set up a very good prompt with some examples.

People at OpenAI were working on just taking the base models and making them easier to prompt. So if you just wrote a question it would answer the question, instead of giving you more questions or something. So we had these instruction following models, which were like base models but a little easier to use. Those are the original ones deployed in the API. Or after GPT-3, those were the next generation of models.

At the same time there were definitely a lot of people thinking about chat. Google had some papers like LaMDA and earlier, Meena. They had these chatbots. It was more like a base model that was really specialized to the task of chat. It was really good at chat. Looking at the examples from the paper, it was more used for fun applications where the model would take on some persona and pretend to be that persona. It was not so functional where it could help me refactor my code.

So there were definitely people thinking about chat. I had worked before on a project looking at chat called WebGPT, which was more about doing question answering with the help of web browsing and retrieval. When you do question answering, it really wants to be in a chat. You always want to ask follow-up questions or sometimes the model should ask a clarifying question because the question is ambiguous.

It was clear after we did the first version, that the next version should be conversational. So we started working on the conversational chat assistant. This was built on top of GPT-3.5, which was done training at the beginning of 2022. That model was quite good at language and code. We quickly realized that it was actually quite good at coding help. That was one of the things we were excited about.

We worked on that for most of the year. We had browsing as another feature in it although we ended up deemphasizing that later on because the model's internal knowledge was so good. The browsing wasn't the most interesting thing about it. We had it out to friends and family for a while and we were thinking about doing a public release.

Actually, GPT-4 finished training in August that year. The flagship RL effort at OpenAI was the instruction following effort because those were the models that were being deployed into production. The first fine-tunes of GPT-4 used that whole stack. Those models were really good and everyone got really excited about that after seeing the instruct fine tune GPT-4s.

They were really good. They would occasionally give you amazing outputs, but the model was clearly also pretty unreliable. It would sometimes hallucinate it a lot. It would sometimes give you pretty unhinged outputs. So it was clearly not quite ready for prime time, but it was obviously very good.

People forgot about chat for a little while after that, this alternative branch. We pushed it further and we ended up mixing together all the datasets, the instruct and the chat data, to try to get something that was the best of both worlds. The chat models were clearly easier to use.

It automatically had much more sensible behavior in terms of the model knowing its own limitations. That was actually one of the things that I got excited about as we were

developing it. I realized a lot of the things that people thought were flaws in language models, like blatant hallucination, could be not completely fixed but things that you could make a lot of progress on with pretty straightforward methods.

The other thing about chat was when we had these instruct models. The task of "complete this text, but in a nice or helpful way" is a pretty poorly defined task. That task is both confusing for the model and for the human who's supposed to do the data labeling. Whereas for chat, people had an intuitive sense of what a helpful robot should be like. So it was just much easier for people to get an idea of what the model was supposed to do. As a result, the model had a much more coherent personality and it was much easier to get pretty sensible behavior robustly.

Dwarkesh Patel
Interesting. Is it the case that anybody could have made ChatGPT using your publicly available fine-tuning API?

John Schulman
Not exactly. I don't remember which models were available for fine-tuning. Assuming we had 3.5 available for fine-tuning at the time, you could have made something decently close. I don't think you would have been able to do just one iteration of fine-tuning with purely human-written data. You'd want to do several iterations.

If not you're not going to do RL, which we did, you'd want some kind of iterative supervised fine-tuning where humans edit the model-generated outputs. If you train on human-generated data, even if it's really high quality, it's just hard for a model to fit that data perfectly because it might be something a model is capable of outputting. You need to do something iterative that looks a bit more like RL. If you'd done that, you could have gotten pretty close but it would have been non-trivial.

We also had another instruction-following model trained with RL, released a little before ChatGPT. If you put a chat wrapper on that you would've gotten decently close but that model had some differences in strengths. That model was good at writing and poetry but it wasn't as good at knowing its limitations, factuality, and so forth.

Dwarkesh Patel
Stepping back from 3.5, I think I heard you say somewhere that you were super impressed with GPT-2. Compared to your expectations in 2019, has AI progressed faster or slower than you would have expected?

John Schulman
Faster than I expected since GPT-2. I was pretty bought into scaling and pre-training being a good idea. But when GPT-2 was done, I wasn't completely sold on it being revolutionizing

everything. It was really after GPT-3 that I pivoted what I was working on and what my team was working on. After that, we got together and said, "oh yeah, let's see what we can do here with this language model stuff." But after GPT-2, I wasn't quite sure yet.

Dwarkesh Patel
Let's say the stuff we were talking about earlier with RL starts working better with these smarter models. Does the fraction of compute that is spent on pre-training versus post-training change significantly in favor of post-training in the future?

John Schulman
There are some arguments for that. Right now it's a pretty lopsided ratio. You could argue that the output generated by the model is higher quality than most of what's on the web. So it makes more sense for the model to think by itself rather than just training to imitate what's on the web. So I think there's a first principles argument for that.

We found a lot of gains through post-training. So I would expect us to keep pushing this methodology and probably increasing the amount of compute we put into it.

Dwarkesh Patel
The current GPT-4 has an Elo score that is like a hundred points higher than the original one that was released. Is that all because of what you're talking about, with these improvements that are brought on by post-training?

John Schulman
Yeah, most of that is post-training. There are a lot of different, separate axes for improvement.

We think about data quality, data quantity. There's just doing more iterations of the whole process of deploying and collecting new data. There's also changing what kind of annotations you're collecting. There's a lot of things that stack up but together they give you a pretty good effective compute increase.

Dwarkesh Patel
That's a huge increase. It's really interesting that there's this much room for improvement from post-training.

What makes for somebody who's really good at doing this sort of RL research? I hear it's super finicky. What is the sort of intuition that you have that enables you to find these ways to mess with the data and set up these environments?

John Schulman
I have a decent amount of experience at this point from the different parts of the stack, from RL algorithms, which I've worked on since grad school, to data collection, annotation processes, and playing with language models.

I'd say I've dabbled with these things and the people who do well at this kind of research have some view of the whole stack and have a lot of curiosity about the different parts of it. You want to be both empirical and let experiments update your views, but you also want to think from first principles. Assuming that learning works, what would be the ideal type of data to collect? That type of thing.

Dwarkesh Patel
Because there doesn't seem to be a model since GPT-4 that seems to be significantly better, there's a hypothesis that we might be hitting some sort of plateau. These models aren't actually generalizing that well, and you're going to hit a data wall beyond which the abilities unlocked by memorizing a vast corpus of pre-training data won't help you get something much smarter than GPT-4.

Do you think that hypothesis is wrong? We've talked about some examples of generalization, like Spanish to English. One example I think of is the transfer from code to reasoning in language. If you train on a bunch of code, it gets better at reasoning in language? Is that actually the case?

Do you see positive transfer between different modalities? If you train on a bunch of videos and images, it'll get smarter from synthetic data? Or does it seem like the abilities unlocked are extremely local to the exact kind of labels and data you put into the training corpus?

John Schulman
I'll try to respond to all that. First, are we about to hit the data wall? I wouldn't draw too much from the time since GPT-4 was released because it does take a while to train these models and do all the prep to train a new generation of models.

I wouldn't draw too much from that fact. There are definitely some challenges from the limited amount of data, but I wouldn't expect us to immediately hit the data wall. However, I would expect the nature of pre-training to somewhat change over time as we get closer to it.

In terms of generalization from different types of pre-training data, I would say it's pretty hard to do science on this type of question because you can't create that many pre-trained models. Maybe you can't train a GPT-4 sized model and do ablation studies at that scale. Maybe you can train a ton of GPT-2 size models or even a GPT-3 size model with different data blends and see what you get. I'm not aware of any public results on ablations involving

code data and reasoning performance and so forth. I'd be very interested to know about those results.

Dwarkesh Patel
I'm curious about something. One of the things is that the model gets smarter as it gets bigger. Would an ablation on a GPT-2 level model, which suggests that there isn't much transfer, provide evidence for the level of transfer on a similar set of domains in a GPT-4 level model?

John Schulman
Right, you might not be able to conclude that if transfer fails at GPT-2 size, then it's also going to fail at a higher scale. It might be that for the larger models, you learn better shared representations, whereas the smaller models have to lean too much on memorization. The larger models can learn how to do the right computation. I would expect this to be true to some extent.

Dwarkesh Patel
This might have a very simple answer. You train bigger models on the same amount of data and they become smarter. Or to get the same level of intelligence, you only have to train them on less data. Why is that the case? It's got more parameters, seen fewer things, and now it's equally as smart. Why is that?

John Schulman
I don't think anyone has a good explanation for the scaling law with parameter count. I don't even know what the best mental model is for this. Clearly, you have more capacity if you have a bigger model. So you should eventually be able to get lower loss.

Why are bigger models more sample efficient? I can give you a sketchy explanation. You could say that the model is an ensemble of different circuits that do the computation. You could imagine that it's doing computations in parallel and the output is a weighted combination of them. If you have more width... actually width is somewhat similar to depth because with residual networks, depth can do something similar to width in terms of updating what's in the residual stream.

You're learning all these different computations in parallel and you have more of them with a bigger model. So you have a higher chance that one of them is lucky, ends up guessing correctly a lot, and gets upweighted.

There are some algorithms that work this way, like mixture models or multiplicative weight update algorithms, where you have—I don't want to say mixture of experts because it means something different—basically a weighted combination of experts with some learned gating.

I actually said something slightly wrong, but you could imagine something like that. Just having a bigger model gives you more chances to get the right function.

Of course, it's not just totally disjoint functions you're taking a linear combination of. It's more like a library where you might chain the functions together in some way. There's some composability. So I would say a bigger model has a bigger library of different computations, including lots of stuff that's dormant and only being used some of the time, but it has more space to look for circuits to do something useful.

Dwarkesh Patel
Stepping back from the current research questions, I want to understand your modal scenario of what happens for the next few years. Towards the beginning of the conversation, we were talking about the case in which it progresses really fast, but let's just take the modal scenario.

You're unlocking long-horizon RL at some point, but as you said, there are potentially other bottlenecks. What's happening? How good are these models? How are they being deployed? What other modalities are part of them and at what stage are these being unlocked? I want to understand your broader picture of what the next few years look like.

John Schulman
I would expect new modalities to be added over time or pretty soon. I would expect the capabilities to generally keep getting better through a combination of pre-training and post-training, and that'll open up new use cases.

Right now, AI is still not a huge part of the economy. There's a pretty small fraction of jobs that it can help with at all. I'd expect that to be higher over time, not just from the models improving but also from people figuring out how to integrate them into different processes. So even if we just froze the models at their current state, you would still see a lot of growth in how they're being used.

I would expect AI to be used much more widely and for more technically sophisticated tasks. I gave the programming example earlier, doing longer projects, but also helping with various kinds of research. I hope that we can use AI to accelerate science in various ways, because you can potentially have the models understand all the literature in a given field and be able to sift through tons of data. It's more than a person would have patience to do.

I hope the form factor would be such that people are still driving all of this and you have your helpful assistants that you can direct and point to lots of different problems that are useful to you. Everyone would have all these AIs helping them do more and get more done.

Dwarkesh Patel
Obviously at some point they're going to be better than everyone at whatever they want to do. What would that process look like? Right now, they're clearly only helping you. At some point, they'll be able to just do things for you and maybe run entire firms for you. Is it going to be a smooth process? At that point, is the hope that we have systems that are aligned with the user enough that they can count on the firm being run in the way they expect.

John Schulman
We might not want to jump to having AIs run whole firms immediately. We might want to have people overseeing these important decisions and calling the shots, even if the models are good enough to actually run a successful business themselves. To some extent, there might be choices there.

I think people will still have different interests and ideas for what kind of interesting pursuits they want to direct their AIs at. AI doesn't necessarily have any kind of intrinsic desire, unless we put it in the system. So even if AIs become extremely capable, I would hope that people are still the drivers of what the AIs end up doing.

Dwarkesh Patel
I wonder if the economic equilibrium is so far from that, where you have the equivalent of Amdahl's law in a firm. The slowest part of the process is the one that's going to bottleneck you.

Even if AI makes all the non-human parts of the firm 10X more efficient, the firm is still bottlenecked by that step. If one company decides to proceed by keeping humans in the loop on all the things that you really want human oversight on, then they'll just be outcompeted by other companies. If one country decides to go this route, other countries will beat it. I wonder if this is a sustainable plan for keeping humans in the loop.

John Schulman
If we wanted to keep humans in the loop, which seems reasonable, and it turned out that firms with any humans in the loop were outcompeted by firms that didn't have any humans, then you would obviously need some kind of regulation that disallowed having no humans in the loop for running a whole company.

Dwarkesh Patel
But there are so many companies in any country, let alone the world. I wonder if it's better to do the regulation on companies and say you've got to keep humans in the loop in important processes, but then you have to define what important processes are.

You've got to monitor every single company and you also have to get collaboration from every single country which has firms. If this is a problem, should it be solved before the

model is even deployed, such that hopefully if you did decide to build a firm and depend on these models, it basically does what you want it to do and you don't need a human in the loop?

Does that question make sense? I'm just wondering, in this situation, how do we actually monitor every single firm to ensure a human is in the loop? And what happens if China doesn't decide to do that?

John Schulman
You would either have to have every country agree to this regulatory regime, or you would need all of the model infrastructure or the model providers to agree to this kind of requirement.

It's definitely going to be non-trivial. This is looking a ways ahead, so it's a little hard to imagine this world before seeing anything like it.

For example, are we actually confident that AI-run companies are better in every way. Do we think they're better most of the time, but occasionally they malfunction because AIs are still less sample efficient in certain ways? Consider when they have to deal with very wacky situations.

AI-run firms might actually have higher tail risk because they're more likely to malfunction in a big way. There might be some practical questions like that that would determine how things play out. Maybe if you just require people to be accountable for various liabilities, this would also change the incentives a bit.

Let's say it turned out that AIs are better at running everything and they're also completely benevolent. Let's say we've totally solved alignment, and they're better at being accountable to people than people are. Then maybe it's okay having the AIs run the firms. But that's pretty far out.

We're more likely to be in a situation where they look better in the short term, but they still have some serious problems. It's actually practical considerations that push you more towards having humans in the loop, at least for the near future.

Dwarkesh Patel
So this is a problem we have to deal with today with RLHF. You have to aggregate preferences across a lot of different humans. It'll be maybe more marked with future, more powerful systems. But when you say we want these eventual AI systems that are going to fully replace humans as part of these firms to be aligned, what does that mean?

Will it mean that they basically do what the user wants them to do? Does it mean that they have to result in some sort of global outcome that we're happy with as the stakeholders in OpenAI? Concretely, what would that mean?

John Schulman
If the models are being used for these higher stakes use cases, then we would have to think about RLHF in a much different way than we are right now.We're not quite ready for that or the current methods might not be completely sufficient. We would need to make compromises between the needs of the different stakeholders involved. We have this document that we're releasing called the Model Spec. It's about how we want our models to behave in the API and in ChatGPT.

We try to talk about this issue where there are different stakeholders involved and sometimes there are conflicts between what they might want. In our case, we were thinking of the stakeholders as the end user (someone sitting in front of ChatGPT or some other app), the developer (someone using the API who might be serving other end users with their app), the platform (OpenAI, we don't want the models to expose us to legal risk), and the rest of humanity (including people not part of the users or customers).

Obviously, the user might ask the model to do something that we think is actively harmful to other people. We might have to refuse that. By the way, this isn't the order of priority necessarily. These are just the four or so classes of stakeholder. Actually, you could maybe also say in the future, the model itself. We're not there yet.

Anyway, we have these different stakeholders. Sometimes they have conflicting demands. We have to make some call on how to resolve those conflicts.It's not always obvious how to do that. We had to think through the trade-offs and basically the rough heuristic is that we mostly want the models to follow your instructions and be helpful to the user and the developer.

But when this impinges on other people's happiness or way of life, this becomes a problem and we have to block certain kinds of usage. We mostly want the models to just be an extension of people's will and do what they say. We don't want to be too paternalistic. We want to be neutral and not impose our opinions on people. We mostly want to let people do what they want with the models.

Dwarkesh Patel
I got a chance to read the Spec beforehand. This is a question of how well that transfers over to how the model itself behaves. I was impressed with how sensible the trade-offs were. I believe the actual edge cases were explicitly stated rather than the kinds of things where are obvious. In this case, you really are going after the edge cases.

John Schulman
We wanted it to be very actionable so that it wasn't just a bunch of nice sounding principles. Each example tells you something about some non-obvious situation and reasons through that situation.

Dwarkesh Patel
I have a couple of questions about the state of the research itself. Famously in the social sciences, things are really hard to replicate. There's a question about how much of the science there is real versus these manufactured, bespoke sorts of experiments. When you look at the average ML paper, does it feel like a really solid piece of literature or does it often feel like the equivalent of what p-hacking is in the social sciences?

John Schulman
Everyone has their complaints about the ML literature. Overall, I think it's a relatively healthy field especially compared to some others like in the social sciences. It's largely grounded in practicality and getting things to work. If you publish something that can't be replicated easily, people will just forget about it.

It's accepted that often you don't just report someone's number from their paper. You also try to reimplement their method and compare it to your method on the same training dataset. If you publish methods that are really hard to implement or are really finicky, they'll tend to get forgotten.

As a result, people actually try to open source their work a lot. There are also various unfavorable incentives. People are incentivized to make the baseline methods they're comparing to worse. There are other mild pathologies, like trying to make your methods seem sophisticated mathematically.

But overall, I feel like the field makes progress. I would like to see a little bit more science and trying to understand things rather than just hill climbing on benchmarks and trying to propose new methods. There's been a decent amount of that recently. We could use more of that. I think that's a good thing for academics to work on.

On a slightly different note, I'd be really excited to see more research on using base models to do simulated social science. These models have a probabilistic model of the whole world and you can set up a simulated questionnaire or conversation and look at how anything is correlated. Any traits that you might imagine, you can see how they might be correlated with other traits.

It'd be pretty cool to see if people could replicate some of the more notable results in social science, like moral foundations and that sort of thing, by just prompting base models in different ways and seeing what's correlated.

Dwarkesh Patel
What is that Stanford experiment? The Asch conformity test? It'd be fun if that replicated with the language models as well. It's very interesting.

I want to ask about the rest of the research that happens at big labs. How much of it is increasing or decreasing the amount of compute you need to get a certain result as an actual compute multiplier versus how much of it is just making the learning more stable and building out the infrastructure?

The broader question I'm trying to ask is, since GPT-4, does it feel like with the same amount of compute, you can train a much better model? Or does it feel like you've made sure that learning can happen better and in a more scalable way with GPT-5, but it's not like we can train GPT-4 with GPT-3.5's budget now?

John Schulman
There's definitely always progress in improving efficiency. Whenever you have a 1D performance metric, you're going to find that different improvements can substitute for each other. You might find that post-training and pre-training both improve the metrics. They'll have a slightly different profile of which metrics they improve.

But at the end of the day, if you have a single number, they're both going to substitute for each other somewhat. For something like a human evaluation, what do humans prefer, we've definitely made a lot of progress on both sides, pre-training and post-training, in improving that.

Dwarkesh Patel
A couple of rapid-fire questions about RLHF. Obviously, RLHF is important to make these models useful. So maybe the "lobotomized" description is inaccurate.

However, there is a sense in which all of these models, once they're put in a chatbot form, have a very similar way of speaking. They really want to "delve" into things. They want to turn things into bullet points. They often seem to have this formal and dull way of speaking.

There are complaints that they're not as creative. Like we were talking about before, they could only do rhyming poetry and not non-rhyming poetry until recently. Is that a result of the particular way in which RLHF happens now? If so, is it because of who the raters are? Is it because of what the loss function is? Why is this the way all chatbots look?

John Schulman
I would say there's a decent amount of room for variation in exactly how you do the training process. We're actively trying to improve this and make the writing more lively and fun.

We've made some progress like improving the personality of ChatGPT. It is more fun and it's better when you're trying to chit chat with it and so forth. It's less robotic.

It's an interesting question how some of the ticks came about, like the word "delve." I've actually caught myself using that word recently. I don't know if it rubbed off on me from the model.

Actually, there might also be some funny effects going on where there's unintentional distillation happening between the language model and providers. If you hire someone to go do a labeling task, they might just be feeding it into a model. They might be pulling up their favorite chatbot, feeding it in, having the model do the task, and then copying and pasting it back. So that might account for some of the convergence.

Some of the things we're seeing are just what people like. People do like bullet points. They like structured responses. People do often like the big info dumps that they get from the models.

So it's not completely clear how much is just a quirk of the particular choices and design of the post-training processes, and how much is actually intrinsic to what people actually want.

Dwarkesh Patel
It does seem persistently more verbose than some people want. Maybe it's just because during the labeling stage, the raters will prefer the more verbose answer. I wonder if it's inherent because of how it's pre-trained and the stop sequence doesn't come up that often and it really wants to just keep going.

John Schulman
There might be some biases in the labeling that lead to verbosity. There's the fact that we tend to train for one message at a time rather than the full interaction. If you only see one message, then something that just has a clarifying question, or maybe a short response with an invitation to follow up, is going to look less complete than something that covers all possibilities.

There's also a question of whether people's preferences would change depending on how fast the model is streaming its output. Clearly, if you're sitting there waiting for the tokens to come out, you're going to prefer that it gets to the point. But if it just gives you a dump of text instantly, maybe you don't actually care if there's a bunch of boilerplate or if there's a bunch of stuff you're going to skim. You'd rather just have it all there.

Dwarkesh Patel
The reward model is such an interesting artifact because it's the closest thing we have to an aggregation of what people want and what preferences they have. I'm thinking about models that are much smarter. One hope is that you could just give it a list of things we want that are not trivial and obvious, something like the UN Declaration of Human Rights.

On the other hand, I think I heard you make the point that a lot of our preferences and values are very subtle, so they might be best represented through pairwise preferences. When you think of a GPT-6 or GPT-7 level model, are we giving it more written instructions or are we still doing these sorts of subliminal preferences?

John Schulman
That's a good question. These preference models do learn a lot of subtleties about what people prefer that would be hard to articulate in an instruction manual. Obviously, you can write an instruction manual that has lots of examples of comparisons. That's what the Model Spec has. It has a lot of examples with some explanations. It's not clear what the optimal format is for describing preferences.

I would guess that whatever you can get out of a big dataset that captures fuzzy preferences, you can distill it down to a shorter document that mostly captures the ideas. The bigger models do learn a lot of these concepts automatically of what people might find useful and helpful. They'll have some complex moral theories that they can latch onto. Of course, there's still a lot of room to latch onto a different style or a different morality.

So if we were to write a doc, if we're going to align these models, what we're doing is latching onto a specific style, a specific morality. You still need a decently long document to capture exactly what you want.

Dwarkesh Patel
How much of a moat is better post-training? Companies distinguish themselves currently by how big their model is and so forth. Will it be a big moat for who has figured out all the finickiness that you were talking about earlier with regards to all this data?

John Schulman
There's something of a moat because it's just a very complex operation and it takes a lot of skilled people to do it. There's a lot of tacit knowledge and organizational knowledge that's required.

With post-training, to create a model that actually has all the functionality people care about, it's pretty complicated. It requires a pretty complicated effort and accumulation of a lot of R&D. That makes it somewhat of a moat. It's not trivial to spin this up immediately. It

does seem like the same companies that are putting together the most serious pre-training efforts are also putting together the most serious post-training efforts.

It is somewhat possible to copy or to spin up more of these efforts. There's also one force that sort of makes it less of a moat. You can distill the models, or you can take someone else's model and clone the outputs. You can use someone else's model as a judge to do comparisons.

The more big league people probably aren't doing that because it goes against terms of service policies. It would also be a hit to their pride. But I would expect some of the smaller players are doing that to get off the ground. That catches you up to a large extent.

Dwarkesh Patel
I guess it helps clear the moat. What is the median rater like? Where are they based? What are their politics? What is their knowledge level?

John Schulman
It varies a lot. We've definitely hired raters with different skills for different kinds of tasks or projects. A decent mental model is to just look at people who are on Upwork and other platforms like that. Look at who's doing odd jobs with remote work.

It's a pretty international group. There's a decent number of people in the U.S. We hire different groups of people for different types of labeling, like whether we're more focused on writing or STEM tasks. People doing STEM tasks are more likely to be in India or other middle or lower-middle income countries. People doing more English writing and composition tend more to be U.S.-based.

There've been times when we needed to hire different experts for some of our campaigns. Some of the people are very talented, and we even find that they're at least as good as us, the researchers, at doing these tasks and they're much more careful than us. I would say the people we have now are quite skilled and conscientious.

Dwarkesh Patel
With regards to the plateau narrative, one of the things I've heard is that a lot of the abilities these models have to help you with specific things are related to having very closely matched labels within the supervised fine-tuning dataset. Is that true?

Can it teach me how to use FFmpeg correctly? Is it like there's somebody who's seeing the inputs, seeing what flags you need to add, and some human is figuring that out and matching to that. Do you need to hire all these label raters who have domain expertise in all these different domains? If that's the case, it seems like it'd be a much bigger slog to get these models to be smarter and smarter over time.

John Schulman
You don't exactly need that. You can get quite a bit out of generalization. The base model has already been trained on tons of documentation, code, with shell scripts and so forth. It's already seen all the FFmpeg man pages, lots of Bash scripts and everything.

Even just giving the base model a good few-shot prompt, you can get it to answer queries like this. Just training a preference model for helpfulness will, even if you don't train it on any STEM, somewhat generalize to STEM. So not only do you not need examples of how to use FFmpeg, you might not even need anything with programming to get some reasonable behavior in the programming domain.

Dwarkesh Patel
Maybe a final question. We've touched on this in different ways but let's put it together. You said you're training on much more multimodal data. Presumably, these things understand what screens look like and will be able to interact with them in a much more coherent way. Also you're going to do this long-horizon RL, so they'll be able to act as agents in the systems and be part of your workflow in a much more integrated way.

What do you expect that to look like? What will be the next steps from there? Suppose by the end of the year or next year, you have something that's an assistant who can work with you on your screen. Does that seem like a sensible thing to expect? Where does it go from there?

John Schulman
I definitely expect things to move in that direction. It's unclear what's going to be the best form factor. It could be something that's like a Clippy on your computer helping you or if it's more like a helpful colleague in the cloud. We'll see which kinds of form factors work the best. I expect people to try all of them out.

I expect the mental model of a helpful assistant or helpful colleague to become more real. It'll be something where you can share more of your everyday work. Instead of just giving it one-off queries, you would have a whole project that you're doing and it knows about everything you've done on that project so far.

It can even proactively make suggestions. Maybe you can tell it to remember to ask me about this and if I've made any progress on it. Proactivity is one thing that's been missing. I'd love to see us moving away from one-off queries, using the model like a search engine, and more towards having a whole project that I'm doing in collaboration with the model. Something where it knows everything I've done. It's proactively suggesting things for me to try or it's going and doing work in the background.

Dwarkesh Patel
That's really interesting. This is the final question. What is your median timeline for when it replaces your job?

John Schulman
Oh, it replaces my job? Maybe five years.

Dwarkesh Patel
Pretty soon. Interesting. John, this was super interesting. Thanks so much for making the time. This seems like one of the parts of the AI process that is super important and people don't understand that much about. It was super interesting to delve into it and get your thoughts on it.

John Schulman
Thanks for having me on the podcast. It was fun to talk about all this stuff.