**Yilun Du:** 0:00

One of the fundamental limitations of RLHF is that you don't actually know if the model has Learned the thing you want to teach it right. You have no guarantees that the model has learned what you have taught it. Some of the most successful companies right now are all in charge of basically collecting data, and I guess they're not as pro-covered as much in the news, but I feel like it seems like a huge amount of people. If you have this AI feedback and it can give you answers for everything, right, you would hope that the coverage is actually a lot more than the coverage you could get from human feedback.

**Craig Smith:** 0:32

Hi, I'm Craig Smith and this is my AI. Large language models are notoriously inaccurate, stringing together coherent sentences that often have little grounding in reality. To fix this, companies like open AI deploy armies of humans to try and nudge LLMs toward more accurate responses using reinforcement learning, a process called reinforcement learning with human feedback, or RLHF. This is obviously an inefficient and labor intensive way to tune the world's most powerful AI models, and researchers are looking for ways to automate the task. In this episode, MIT PhD student In Du Talks about new techniques for improving large language models through reinforcement learning with AI feedback, or RALOF. Ilun explains how LIFE circumvents the need for expensive and inconsistent human ratings by having AI agents debate and critique each other's responses. He shares insights from his research, using multi-agent debate to enhance reasoning and accuracy, and we discuss the challenges of accessing proprietary models. I hope you find the conversation as interesting as I did. Hi, this episode is sponsored by Salonis, the global leader in process mining. Ai has landed and enterprises are adapting, giving customers slick experiences and the technology to deliver. The road feels long, but you're closer than you think. You see your business processes run through many systems, creating data at every step. Salonis reconstructs this data to generate process intelligence A common business language with process intelligence. Ai knows how your business flows across every department, every system in every process. With AI solutions powered by Salonis, enterprises get faster, more accurate insights, a new level of automation and a step change in productivity, performance and customer satisfaction. Process intelligence is the missing piece in the AI enabled tech stack. Search Salonis C-E-L-O-N-I-S to find out more. This episode is sponsored by Cruso Cloud. High performance cloud computing or low environmental impact? Cruso Cloud was built because the innovations of the future need both. Cruso Cloud is a scalable, clean, high performance cloud optimized for AI and HPC workloads and powered by wasted, stranded or clean energy. Cruso offers virtualized compute and storage solutions for a range of applications, including generative AI, computational biology and rendering. Visit crusoe cloud.com that's C-R-U-S-O-E C-L-O-U-D dot com To see what climate-aligned computing can do for your business.

**Yilun Du:** 4:06

I'm currently a PhD student at the Computer Science and AI Laboratory, so I've been working on generative models for the last five years, especially on their application to construct intelligent physical agents or robots. So I guess. Another name for generative models recently has been generative AI. So I've been working in this space for the last five years and, yeah, mostly I've been working. I've been working a lot on diffusion models for quite a few years and then recently also a reasonable amount on language models.

**Craig Smith:** 4:40

Yeah, so I'm interested in RLAI free enforcement, learning with AI feedback and I guess my first question. The reason I'm interested is because that's how open AI and the other large language model providers are trying to nudge their models into good behavior, to stop hallucinating and things like that, which to me seems like a very unsophisticated method to use armies of workers around the world, sort of you know, scoring responses. So in LIFE is a natural extension or solution. But am I wrong that reinforcement learning with human feedback really began as a strategy in robotics?

**Yilun Du:** 5:52

So not quite actually in 2018, I was working at Open AI, so that's when they just started actually this RLHF. So there was a paper between Open AI and DeepMind. I think this was in 2018. That was, I guess. Rl itself is a field that has been going on for like many, many decades. Deep reinforcement learning is also a thing that actually has been going on for even like for like two decades. Actually, even before DeepMind's DQN, there were quite a few works on deep reinforcement learning. But anyways, this LHF idea at the time in 2018 was that RL agents had these would often exploit your environment and they would like to exploit parts of the environment that you really would not want it to. So, essentially, like, maybe you wanted to complete this race course, but it finds that it gets a lot of points by looping around the ship. So instead of completing the race course, it will end up looping around the ship an infinite number of times. So initially, the goal was that by having humans give feedback to RL agents when playing these interactive games, you could prevent the agents from doing very odd behaviors, and then this involved I think around 2018 was when the first GPT model came out. So then there's the safety team out in AI, and I remember there are several people who wanted, who are interested in the idea that you could use this RLH, this idea of RL supervision, to train the language models, to like summarize or like to have humor, because it seemed like these are things are very hard to teach the language model if you just gave it a bunch of internet text. So I guess that might have been the start of this idea and then afterwards, afterwards, there were like a variety of papers and then eventually, I guess, like, like, I guess chat GPT was a version of this like RL from human feedback.

**Craig Smith:** 7:54

Right, and just for those listeners who don't understand how it works, can you describe the RHF how it works in this sort of process and when you were working on it in open AI? Was that that was before the GPT models or was that on early iterations of the GPT models, or were you using what we're using RLHF for?

**Yilun Du:** 8:27

So I personally was not using RLHF At the time when I was at open AI. It was a very small company, I think around 40 people, and then maybe, so we knew I knew about all the research that was going on at the time, so this was right when the first GPT one model came out, or before that. So this was very early on. The focus of the company at the time was RL and then there was focus on games like Dota. That was going on and the idea of ELF is that your model will generate different outputs and then you'll have people rate them Like. People will say this is very desirable, this is not desirable, and then you're basically and then this gives the model supervision on, like if you should generate this output more often or less often and, as I said, since these language models have blown up and hallucination is such an obvious problem, people have been using RLHF and it's actually to guide the models into better behavior.

**Craig Smith:** 9:40

Two things about that fascinate me. One is that you're only guiding, you're only kind of nudging the model, and is there some metric for how many times you need to push the model to stop it from doing something you're pushing it away from? But also, as this has, as LLMs have exploded, there is, like this new industry of human reviewers around the world, and do you have any sense of how many people are involved in RLHF for large language models? At this point, I mean the human reviewers.

**Yilun Du:** 10:24

Yeah, so I guess, going on the first question, so yeah, I think that's one of the fundamental limitations of RLHF is that you don't actually know if the model has learned the thing you want to teach it right. You have no guarantees that the model has learned what you have taught it. And then there's a variety of papers that have shown that, like you can take any of these large language models and like, just by adding some additional prompts, it's very easy to make them elicit behavior that you would have hoped that you had made the models forget. So yeah, exact issue. A big issue with RLHF is that the reward function nudges the model towards doing something you want it to do. It doesn't necessarily tell, doesn't necessarily guarantee that your model has learned this. I mean, on the other hand, though you could say that human learning also, like you as a teacher, you kind of nudges students with like this is the response you want, this is not the response you want, but you don't never necessarily know if the person has actually learned the thing you wanted to learn. Maybe they just copied the equations you've given them. But yeah, so along the first question, I think that is an issue with RLHF Along. The second question: how many people are rating these models? That is actually one thing I don't really know very well. My understanding is it's a lot. So I think that, based on what I've heard, some of the most successful companies right now are all in charge of basically collecting data, and I guess they're not covered as much in the news. But I feel like that seems like a huge amount of people and I think the job is very, I guess you could say, boring because and maybe scarring also because you like you see, you see all this stuff on the internet and then your goal is just the label things.

**Craig Smith:** 12:16

Yeah and the yeah. I guess that's right. I mean, I hadn't thought of it in that regard. It is just labeling, right, it's like supervised learning labeling. It's just for a different purpose. And when you said some of the most successful companies, you're talking about data prep companies, or yeah, like data companies that are focused on gathering data. Yeah, yeah. So bringing AI into the picture seems like a natural solution, and can you talk about how you do that? How do you get an AI to rate a response from a model?

**Yilun Du:** 13:16

Yeah. So when you do ROHF, what happens is your model will generate different outputs and then people will just say this is good, this is not good. So in a very similar vein, you can imagine that you have two language models. One language model generates the answer and then you give the other language model the maybe the ground truth answer and the question, and now it can basically assess how good the generating answer is. First, does it match the ground truth answer? And then it can check the reasoning. Does the reasoning make sense to the language model? So that's a very simplistic version you can imagine of ROHF from AI feedback, and then there's a variety of different ways to make it much more sophisticated. So one way that I've been working on a lot is this idea of using multi-agent debate. So what happens is now, let's say, I give you a question. What happens is you have multiple instances of your language model generating different possible answers to the question. So it's kind of like when I give you, when you're trying to solve a math problem or you're trying to answer a question, you have like different thoughts in your mind and like each of these voices in your mind kind of like talk to each other. So each of these language models talk to each other, like try to like. Is this answer consistent with my first thought? Is it consistent with my second thought? So you can imagine that you have these like language models debate with each other and then the answer you get at the end of this debate should be a more accurate answer than the original answer you have. So you can imagine that now this is a, this gets you better data, right, and then you can again use this data to retrain your model. Yeah, the core idea is basically you can use the model to like get better data Like you generate from your model, and then you can impose some structure on this generation process to encourage the model to like do a bit more like a reasoning by causal, causal, like chain of events or stuff like this, and then after, at the end, the answer you get is that improved version of what they, what the model would originally generate. So it gives you some improvement.

**Craig Smith:** 15:15

Yeah, and is there? So in the first, the more simplistic scenario where you're giving a model of the ground truth in the prompt is that you do that, you do that in the prompt, or you're connecting a vector database with, with you know, verifiable or ground truth data.

**Yilun Du:** 15:45

Yeah, I think probably the simplest way is to just give it as a prompt. So so you can ask the first language model here's a question, generated answer and then the second language model you give the original question again and then you give the generated answer and then you say here's the ground truth answer and then and then you can instruct the language model, reflect and like, describe what is correct versus not correct about the answers.

**Craig Smith:** 16:10

So walk me through that. Let's say you ask a model what's two plus two? And the model answers five. And then you have another model.

**Yilun Du:** 16:27

Yeah, so, so, yeah. So let's say the first model generates two plus two is equal to five. So what you do now is you give the second model, you say here's, so you give it the same question what is two plus two? And you and then you give it ground truth answer for and then you give it answer generated by agent two plus two equals five. And then you can say can you check if this answer is correct and provide like feedback on, like what portions were correct or portions were incorrect? Right? So one thing you could do is feed the intermediate generations of the language model, because the language model generates the answer step by step, right? And you can like, give it the first sentence and say is this consistent with the right answer. You can give it the second sentence, say, is that consistent? And you can get like much richer annotations than just this is correct. This is not correct, which is, in principle, something you could also do with people, but it's very tedious for people to do this. So in practice most people most of the time the RHF feedback that you get is just like holistic over the entire output.

**Craig Smith:** 17:32

Yeah, so in these LLMs talking to each other you don't have a human operator that's asking one LLM and then takes that answer and asks another LLM. This is all happening within the software between the two LLMs. Is that right, or is there some human oversight required?

**Yilun Du:** 17:59

Yeah, so in this debate procedure that I was talking about earlier, yeah, there is no human oversight. Well, I think one big thing about RO, ai, ro for an AI feedback is this idea that you do not want humans to be their bottleneck Right, because, like LLMs, right, or like all of these models, they can be accelerated very quickly on hardware accelerators, but people like operate at a specific speed and they're expensive, maybe they're inaccurate, right? So, yeah, the goal would be to completely automate this as much as possible.

**Craig Smith:** 18:35

Right, but where does the LLM, the reviewing LLM, get the ground truth data? From a vector database, or yeah?

**Yilun Du:** 18:48

Well, I think there are two ideas right. So one idea is that, yeah, you could give the ground truth LLM the correct answer Right, like if there are a set of problems that you want to teach the LLM to be more accurate on right, you could just generate a bunch of questions, get a bunch of answers right also, and then like use this LLM to like critique your answers. I think the thing that a lot of people want is to assume that you do not have the ground truth answer. Even if you do not have the ground truth answers right, like, let's say, let's say I want to write a proof like a mathematical theorem. I don't know the ground truth answer, but I can check to see if someone else's answer is incorrect. So I can carefully look at the first step and say, well, this step is logical or this step is not logical, I can check the second step. So one thought about this like a RL from AI feedback is in the same way as how people can prove mathematical proofs by sequentially just checking and making sure everything is correct to your knowledge. You can imagine each of these LLMs like checking each other and seeing if every single step is correct and if every single step is rational and you get the answer you want at the end, right. And again, you can, even if you don't know what, don't know what, the ground, truth, answer is right, we still know this is a satisfactory answer or this is not a satisfactory answer. So, like, just by like it's you can imagine it's a way to search through the space of possible solutions. So, having one LLM generates many different things and having the other one kind of prunes the ones that are probably wrong, and if you prune all the bad ones, then you get the right answer. And then if you have a model that can always generate the right answer, right, it's a more intelligent model than the one that's not. That's just generating all possible answers.

**Craig Smith:** 20:41

And this I mean. I understand how that could work for a logical proof or a mathematical equation, but how does that apply to more general and less precise questions and answers? I mean, if I ask an LLM what's the history of this town and it comes back with a hallucination, that's not correct. How does the other LLM know to check that first? Llm's answer, yeah.

**Yilun Du:** 21:24

So I think this is where. So I guess I talked about a bit about how, like in some of my work on RLAF, our RLFRI feedback, I use this idea of having multiple models. So I think if you only have a single answer, right, like if I ask you like when was this person born, and then the LLM just makes up some number, there's no way to verify if it's correct or not. But the thing about LLM is you can, they can generate multiple answers for a question, right? So if I ask, if I ask the LLM, when was this person born, and it gives me like 1876, and then it tells me it's 1896, and then it tells me it's like 1895. Now you have multiple different answers, right? And now your critic can say well, clearly something's amiss here because none of these answers are consistent with each other, right? So I think when you introduce this idea of not just having one model generate, but like have multiple instances of the model, generate different solutions, then then you can, you can kind of, you can kind of like verify issues with the like causality or like the way that the chain of reasoning of language models, right, like you can identify things that are like logically inconsistent by just having multiple generations come out.

**Craig Smith:** 22:40

Yeah, so by having multiple generations of, or multiple agents debating, as you said at the beginning, is that? Is that what you mean?

**Yilun Du:** 22:51

Yeah, that's what I was thinking.

**Craig Smith:** 22:55

And I mean this is all happening in some vector data, vector space, right, it's not happening in actual language on a screen and that's then being fed to the other agent. How does it work mechanically? How does it work?

**Yilun Du:** 23:18

Yeah, so actually so far for the work that I've been working on debate, you actually have the language models, generate language outputs explicitly and then each of the language models takes in these language outputs from other models and then gives feedback on that. So it's purely in the language space and, like probably, it could be more efficient to be able to do it in some type of vector space. But actually there's some desirable properties in language. Right, If you can have all these agents' reasoning language, then we can kind of understand what the agents are doing. Also right, Like it's a hard constraint that forces the model to like be kind of reasonable in a sense. Right, Because if you do it in some effective space, maybe you, like you as a person, have no idea what the model is doing and it might also be finding all types of cheating ways to communicate. So like having it, having all of this self-improvement be directly in the language I think is a desirable thing also.

**Craig Smith:** 24:19

And so in your research, are you getting printouts of these conversations between AI agents? How do you monitor what the conversation or the debate is between agents?

**Yilun Du:** 24:33

Yeah, you can actually. Yeah, so you can actually see the entire printout of the conversation. So, it starts with like you give it a question and then model, and then agent A will say based on the order of operations I think it's x, y, z and then here's the final answer. And then agent two will say based off this other thing, I think the answer is A, w, a, b, c, and then each of these agents will respond to each other, so like when agent will say well, actually I think the second agent is wrong, I think it should be this, and the second agent will be like oh, no, I am correct. And then, like you can have this like multiple rounds of like conversation, and then, like, essentially, at the end of this conversation, each agent will be like based off my, based off the what I've seen of the other agents, I think the answer is like A, b, c, and then you just take this final answer.

**Craig Smith:** 25:19

And it will converge eventually to one answer, or do you end up with LLMs that will never agree?

**Yilun Du:** 25:28

Yeah, so most of the time you end up converging to the same answer, or like converging to one answer. I think this might be due to the fact that the models are trained with you. Like, these models have been trained to converse with people, right, and they've been trained to converse with people in such a way that they are like, not very stubborn, like if a person's constantly suggesting the answer, they will respond to it. So what we notice is, when you try to repurpose this mechanism to have multiple models talk to each other, right, then the models are very definite to other models, so, so it's almost all the time they will converge to one answer. And then you check that answer against ground truth or against sort of yeah, basically we just check against ground truth and we see how good it is.

**Craig Smith:** 26:19

And so what are they? Presumably you have metrics of how different strategies have improved response accuracy or something like that, but how do you measure that?

**Yilun Du:** 26:36

Yeah, so. So there are like a set of existing benchmarks and then what you can do is and roughly they're like they're in the style of here are a set of like math problems and I know the ground truth answer. So basically you take the final generation from all the agents and then just compare the accuracy in which that final answer matches the real answer. Is one thing you can do and in a similar sense, for like factually correct things, you can ask the model to generate all these facts right, and then you can get ground truth facts about the, about the subject of interest, and then you can basically check the accuracy in which the language model generated things matches the ground truth. So always there is so, so, so, so , all of these problems there are like for evaluating this approach, there's always like you can always come up with a database of answers and like check those. I mean the hope would be that you should. You would always have these benchmarks that you know the answer is for, so you can concretely monitor the progress, but then ideally you'd want to do it on all types of subjects where you don't have very clear ways to monitor how good it is. Also,

**Craig Smith:** 27:43

Yeah, and again, is this in the fine-tuning stage that you're doing this, where you're trying to improve the accuracy of one model by having it debate with these other models, or is this an architecture? For, as a consumer, you know, I interface with a language model on my screen but in fact, in the background, there's this debate going on between multiple models that then converge on an answer. Before supplying me with the answer, I mean, is this a training strategy or is this actually a new way of building large language models?

**Yilun Du:** 28:43

Yeah, I mean. So in the paper we wrote about this, we primarily focused on this application where you finish training this model and then this is a way to get better answers from the language model. But I mean, I think one thing that I've been very excited about is this idea that this is also an improvement operator right During training time. Also, this is a way to get, like our offer, AI feedback, and I've been recently talking to several companies that I've also been to. I guess I've been training the large language models and they're also thinking, like exploring one of these strategies. So, yeah, so I mean, I think it could be either right In general, I think, anything that you can do for any technique for AI, for an RL from AI feedback. Each of these strategies gives you some mechanism for how good the answer is. I always use that as a test time thing also, as well as a training thing. I think there are like two flips of the same coin, like I don't think one is very different from the other.

**Craig Smith:** 29:44

Yeah. But if you're using this strategy to train the model, that's one cost. But if you're building it into the model so that the model is going through this process to produce every inference, it seems to me that would be extremely costly. So actually, on one side, if it's built into the inference, every time you ask a question before you get the output there is this debate going on in the background that would increase the cost, right, yeah. But then, on the other hand, if you're using it for training and this is something about ELF also or trying to nudge these models, you do it, for example, on questions where there's a very clear ground truth answer, but that doesn't mean the model will necessarily generalize to all questions, right?

**Yilun Du:** 31:00

At training time. Well, yeah, so I think yeah, so definitely it's a case if you use that as a test time, that computational cost is much more. I always feel like it. It always seems like a thing, though, that if you have a thing that improves performance, the actual computational cost is always, like, not the thing to care about the most, because that cost always decreases year on year. So I feel like it's always been the case that every year on year, things get much, much cheaper and cheaper to do so. If you have something that can fundamentally improve the performance of your system, it seems like worrying about cost is not the most important thing, because people always find ways to make things faster. But yeah, on the latter point, though, yeah, so ideally, any type of like RL from AI feedback you would be able to apply on arbitrary prompts, hopefully, because, ideally, you construct this interface right that all the models can just autonomously talk to each other and generate answers, and as long as the models can autonomously generate answers, you can give it all types of prompts. One issue with having people rate the accuracy of something is it's really hard to rate the accuracy of not so well-known people, because you would have to search for ages on Google and maybe to verify when passes are generated. Maybe you need to spend two hours, so it's not very scalable and then for lots of prompts, maybe there's just no easy way to do it. But then if you have this AI feedback and it can give you answers for everything, you would hope that the coverage is actually a lot more than the coverage you could get from human feedback.

**Craig Smith:** 32:43

Right In the training phase, you mean.

**Yilun Du:** 32:46

Yes, in the training phase.

**Craig Smith:** 32:47

In the training phase, Because that's an issue right when I mean the hope in RLHF is that the model will eventually it'll learn improved behavior, It'll learn to reason more carefully or to learn to double check itself or something in the background, so that it then can generalize that behavior to any question that it's asked. But that's not necessarily the case. Is that right?

**Yilun Du:** 33:28

Yeah, I mean, yeah, I guess with all these things you can never know, like you can never guarantee, like this model will like to see every possible thing right. These models have like billions or trillions of parameters, so it's hard to guarantee, but the hope would be that you get more and more coverage. Like, the more scalable your automated critique is, the more possible prompts you could cover and hopefully, if you cover all possible prompts, maybe it will be very difficult to elicit odd behavior for the models.

**Craig Smith:** 34:00

Yeah, and on the improvement, on the measured improvement, this is there. How do you measure that? So, against the benchmarks, Like before RLHF, you know you get a 60% score by, and that's another question how do you score, I guess, on a benchmark? Yeah, so you have a 60% accuracy, and then, after RLHF, you've moved it up to 80%. I mean, what are the numbers that you're working with in terms of the improvements that you're seeing with these strategies?

**Yilun Du:** 34:47

Yeah, so I think right now, yeah, so what you would see is like, maybe on this benchmark it used to be 60%, now it goes to 80%. So, yeah, you would want to see these consistent improvements across a wide variety of benchmarks. I think there is a question where, let's say, the models get really good, right, which I think they're very, very far away from at the moment. But if they were to get very, very good and maybe they do perfectly at all these human benchmarks, right, then how do you assess the performance of these models? And again, I think there's a nice idea from this multi-agent debate kind of thing, because you can use something like an MMR system. So, for all these chess and goal-playing bots, right, they are much better than people. But we can still quantify how much better they are by pitting these agents against each other. So if the agent is able to beat another agent 100% of the time, then it's probably a much more intelligent agent than the original one. So the hope would be that if you use this type of multi-agent competition as your improvement operator, you can also use the ease at which the language model is able to generate solutions to convince another agent as a measure of how intelligent it is. So if the model gets more and more intelligent, ideally it would be much better. It would maybe blow away the other model. Right, you can speak so eloquently or so correctly that you just cannot. You will always compete with the other one, and then maybe that could be a metric for improvement at that point. But I think that's very speculative. I think the models right now are very, very far away from being human level and probably my guess is, in the next year or so, or at least upcoming several years probably, we will have plenty of benchmarks to see numbers go up a little.

**Craig Smith:** 36:38

Yeah, but in your research, how much has the performance improved with these strategies? I mean, do you see a 2% improvement, or yeah?

**Yilun Du:** 36:51

I see like a 20% improvement, which is a decent amount, and I think so. We wrote this paper in May and I think there's been several follow-ups on this paper and I think some of the large companies I think, for example, anthropic might also be exploring debate. But, yeah, it seems like there's a boost. It doesn't seem like, well, we don't have access to the large language model, so we actually cannot do this full-on training at a large scale, so we don't see these new capabilities or anything like that. That would be the hope if it could. But yeah, right now we see this boost in these raw numbers.

**Craig Smith:** 37:29

Yeah, and if you don't have access to these proprietary models, what language models are you using in your research?

**Yilun Du:** 37:42

Yeah, so we have API access to these models as everyone else does. So this way we propose to do this RLAI feedback. It doesn't require the weights of the network, we can directly like. It's more of a test time thing, like we were talking about earlier, where you can still use this to improve your performance at test time. But yeah, the big issue is we can't fine-tune the model or use that to improve the performance. Yeah, I think that's a bit of an unfortunate thing at the time. I think there is a pretty large gap at the moment with all these open-source models and the ones that are being used commercially. So it feels like the pace of improvement could be improved a lot if the weights were made available to everyone. But the issue is that I think there's so much money and incentive on a commercial interest on this side that that's unlikely to happen.

**Craig Smith:** 38:42

But yeah, because of that. What about using Lama 2, for example, where you have the weights?

**Yilun Du:** 38:52

Yeah. So I think the issue with Lama 2 is still. It's pretty poor actually at the moment. So lots of people talk about how it's really good and stuff like this, but it still is. So some people claim that if you use Lama 2, you do lots of fine-tuning, it can match GPT 3.5 performance. What we find is that it can match GPT 3.5 performance on one task, which is the task they spent the last month optimizing for, and it's far worse on every other task. And then even GPT 3.5, it's much like this debate stuff and a lot of AI feedback stuff. So GPT 4 is much better at this than 3.5 is. So it feels like if you're stuck with things that are much worse than 3.5, it's very hard to get improvements. I mean, one thing that was kind of interesting is so after we released this paper, there was another group of people. They actually just tried it on these open source models and they actually also noticed the improvement, but it was like 2% or 3% compared to like our 20%. So I think there's a pretty big disparity between these commercial models and the ones that are available at the moment.

**Craig Smith:** 40:09

That's fascinating. That's another topic that I'm really interested in is whether open source for generative AI is the future or whether the open source community just can't marshal the resources, the financial resources, necessary to to compete with proprietary models and you know that is the only one that's that sort of step forward, to commit serious resources behind an open source LLM. But they're going to have to keep that funding at a very high level in order to compete. And do you think that open source generative AI will eventually? You know that Yanlacun says that that's the future, that all these things will be open source, but then I talked to other people who say no, because these resources are concentrated in these very large companies and open source will never be able to compete. Meta will not be able to keep up with Anthropica or OpenAI by, you know, improving the open source models. What's your view on that?

**Yilun Du:** 41:38

Yeah, I feel okay. So I do think at the moment there's a pretty massive gap between the open source language models and the internal ones. I think a huge reason for this gap is actually because of the data. So I think OpenAI has spent a lot of time gathering very customized data to make the agent look very appealing to people, and I think this lack of data makes it feels like it's possible, I don't know. So I think one thing is that OpenAI and Anthropica, which I guess was from OpenAI actually they were doing this large language model scaling for four or five years and they were also trying to like looking for how to commercialize it for the last 2.5 years also, and I think everyone else kind of just suddenly realized they had to do this, maybe a couple, I guess now at this point, almost like nine months ago, but like about nine months ago, people were suddenly like we need to spend time getting these large language models. So I think there's a big time difference at the moment and I think, yes, open source models are much worse. I think even the company models. I think Facebook's model internally my guess is, I don't know, I don't know any information about this is probably much worse than the OpenAI ones, and I think a lot of it is just a time thing. So I would hope that maybe in a year or two, I think, or maybe even in one year, maybe things will catch up. But yeah, at the current moment things are very different. I think one thing that people seem to constantly say is they feel like there's a lot of technical knowledge to train these models and that's one reason why OpenAI is so much better than the other ones. I actually don't think that's the case. I do believe that I mean, I think there are a couple of things that people don't appreciate about training the models. I think one big one is like finding hyperparameters on smaller models and having the exact same hyperbrander skill for larger ones. But besides a few technical tricks, I think the main thing that is missing right now is just the data and like yeah, so I think because it's just the data. I feel like maybe in a year or two people should catch up.

**Craig Smith:** 44:05

I don't think there's a technical know-how that's missing. Yeah, and the GPU constraints and the cost of the computer, don't you think that's a barrier to open source?

**Yilun Du:** 44:17

Yeah, I think that could be a barrier. So, yeah, it is true that the GPU cost is quite large, like probably several, like maybe even 100 million, I guess, like maybe hundreds of millions, maybe even a billion was a huge model. Yeah, I mean, I feel like maybe we need some type of like a research center or something like this where, like government funded things, so like I guess, like with particle accelerators and particle physics, you have the large hadron collider and then that's like hundreds of millions of dollars or more. So if we had an equal size effort, I think you could totally do this. Yeah, but actually I do feel like at the moment, we don't even have a. I feel like there's a disconnect between what people are. Yeah, it does feel like there's a big disconnect between what people constantly say online right now in the open source community, like everyone's like oh, you just need a small model, you just need to fine-tune it a little and then it will reach their actual performance. Like, yeah, I think people maybe should focus a bit more on using more GFUs. Yeah, that could be a thing that could be problematic.

**Craig Smith:** 45:26

Yeah, and then you're also working in robotics, is that right? So how does this work relate to robotics?

**Yilun Du:** 45:43

Yeah, so I guess throughout my PhD, I've been very interested in the idea of artificial intelligence and, in particular, I think that, like to have artificial intelligence, we really need to have it physically in the world. So we need an agent that can visually understand the world, that can physically act in the world, has intensity in the world and likes to interact with other agents. So I really believe in this physical AI process. So, yeah, and I see that robotics is basically the instantiation of having a physically intelligent AI agent. So to me, I guess this multi-agent debate stuff it's one way to try to get AI, to get a more intelligent AI. I think this idea in general could be applied a lot in different ways. So right now, I guess it's just language models, but you can imagine that you have different models, like in the robot processing system, right? So maybe you have a video generation model that generates possible actions you want to do, right. You have some type of action model that takes images and predicts actions that you want, right, and you can imagine each of these models talking with each other, like the action model says you gave me this image. This image is not possible. Please refine it a little right, so you can imagine that you have a network of these like individual modules that, like all are talking with each other and then, like, when you have like this set of these decentralized models that like are all talking with each other, maybe that whole theme can be like some like distributed operating system to like construct an intelligent robot. But like, I feel like this idea of like a debate or like the self-improvement thing, like can be broadly applied across like multi-modal models, like on a robot also.

**Craig Smith:** 47:23

Yeah, yeah, it seems you know. I've had Peter Abiel, I'm sure you know, on the podcast and Sergey Levine and different people working in robotics and it seems that the barrier in robotics is not so much the brain AI but it's just the mechanics, the miniaturization of motors, and you know all of that stuff. Do you feel that? Or is that? Is the hardware going to catch up with the software, with the AI?

**Yilun Du:** 48:14

Yeah, I think so. I agree that hardware for robots is a lot worse than what people have. Yeah, we don't have tactile sensing. The arms are very awkward to use, they are very constrained, but I think when you give a person a remote control of the robot, they can do a lot of tasks in the kitchen, right, like most almost every task you can imagine you can actually do by remotely controlling the robot. So, yes, those are big issues, but it feels like we're also missing the fundamental intelligent person also, like I, as a person, can do a lot more than I can autonomously code my robot to do at the moment, and I think that if we could do more autonomously with the robots, then people would be much more interested in robotics. I think then we would have the interest in hardware to get better hardware to train robots. I think the thing is there's no proof of concept of AI working in robotics. Still. That makes it so that people aren't very interested in trying to get better hardware for robots.

**Craig Smith:** 49:16

Yeah, that's interesting. Yeah, okay. Well, we're almost up to an hour. I don't want to keep you over, but are there areas of this that you're working on that I haven't asked about that? Listeners might be interested in?

**Yilun Du:** 49:38

Oh, I mean, I think there is one kind of interesting idea that I feel like. So I feel like everyone and it's kind of related to this thing I was talking about , like multi-agent debate is a way to improve these language models. I think right now everyone is working on these like one very giant monolithic model that can do everything it can sense, it can perceive and stuff like this. But I feel like our brain is actually very decentralized. We have areas for motor control, we have areas for vision, we have areas for memory. So I think it's cool to think about this idea of having not a single AI, but having a society of AIs. In your mind, you have a bunch of different small agents that are talking to each other, and then that's how you actually function. You have one thought process that's trying to remember things that you're doing, another one that kind of perceives what you're doing. So, yeah, I think that's an interesting idea that I feel like maybe people would also find interesting to think about.

**Craig Smith:** 50:33

Yeah, and where did you do your undergraduate?

**Yilun Du:** 50:37

I actually did my undergraduate at MIT also.

**Craig Smith:** 50:42

Oh, okay, and where are you going to go after this? Are you going to stay in academia?

**Yilun Du:** 50:48

Yeah, I'm planning to stay in academia at the moment.

**Craig Smith:** 50:52

Yeah, and I'm curious why I mean these days you don't have to leave academia to go into industry. I mean all these guys Yanlacun, nyuf, metta or Jeff Hinton I'm used to retiring now but Google and the University of Toronto, I mean there's a lot of opportunity to do both.

**Yilun Du:** 51:16

Yeah. So I think if you had asked me this maybe one or two years ago, I would have been a lot less certain, because it feels like at the time, yeah, I felt like you could do very open-ended research, both in industry as well as in academia. I feel like nowadays, because there's so much commercial success with these models, the research you can do in industry is actually very limited. So if you want to work on RL agents without any language model, you can't really do that. Like there's a huge push for you to work on RL for language models. Or like if you say I don't believe that a single large language model will solve everything, which I don't know. I've been playing around. So during the summer I was working at Google, earlier I was working at Facebook. I generally haven't played a lot with these vision language models. They still seem very, very far away from actual intelligence with the vision side. But even if you don't believe in this hypothesis, you still have to work on this Because that's the bold thing and people think that it's so promising commercially that that's the idea to work on. So I feel like nowadays you have very little freedom in industry and I think it's actually just been in the last year that this has happened.

**Craig Smith:** 52:29

Hi. This episode is sponsored by Solonus, the global leader in process mining. Ai has landed and enterprises are adapting, giving customers slick experiences and the technology to deliver. The road feels long, but you're closer than you think. You see your business processes run through many systems, creating data at every step. Solonus reconstructs this data to generate process intelligence a common business language With process intelligence. Ai knows how your business flows across every department, every system in every process. With AI solutions powered by Solonus, enterprises get faster, more accurate insights, a new level of automation and a step change in productivity, performance and customer satisfaction. Process intelligence is the missing piece in the AI enabled tech stack. Search Solonus C-E-L-O-N-I-S to find out more. This episode is sponsored by Crucio Cloud, a high-performance cloud computing or low environmental impact. Crucio Cloud was built because the innovations of the future need both. Crucio Cloud is a scalable, clean, high-performance cloud optimized for AI and HPC workloads and powered by wasted, stranded or clean energy. Crucio offers virtualized compute and storage solutions for a range of applications, including generative AI, computational biology and rendering. Visit crucio cloudcom that's c-r-u-s-o-e c-l-o-u-d dot com to see what climate-aligned computing can do for your business. That's it for this episode. I want to thank Yilun for his time. If you'd like to read a transcript of today's conversation, you can find one, as always, on our website, ionai, that's e-y-e-o-n dot. Ai. And remember, the singularity may not be near, but AI is already changing our worlds, so pay attention.