SDS PODCAST

EPISODE 1048: THE HUMANS WHO TRAIN FRONTIER AI, WITH GABE GREENBERG

Jon Krohn:	00:00	Recursive self-improvement could arrive within two years. So why isn't my guest, whose entire business is human training data, losing any sleep? Welcome to another episode of the Super Data Science Podcast. I'm your host, Jon Krohn. Today's guest is Gabe Greenberg, founder and CEO of G2i, which grew from a one-person shop to $15 million in revenue without any VC backing. Today, G2I's bench of 8,000 vetted engineers creates the reinforcement learning data that Frontier AI labs train on, and G2I paid those engineers $50 million over the past year. That's five zero. In this episode, which G2I kindly sponsored, Gabe explains why frontier training data now demands staff level experts, why so many popular AI benchmarks are broken, and how his agentic platform orchestrator writes production code at scale. Enjoy this great episode. Gabe Greenberg, welcome to the Super Data Science podcast. It's great to have you on the show.
	00:59	You have a really fun name to say.
Gabe Greenberg:	01:01	Really? No one's ever told me that before.
Jon Krohn:	01:05	Really? Gabe Greenberg, it really rolls off the tongue. If you created a novel with that kind of character name, you'd be in your mind instantly.
Gabe Greenberg:	01:13	I'll have to write my parents and thank them.
Jon Krohn:	01:16	Yeah, exactly. Exactly. Perfect. So you are known, I think, quite well in the Miami area as co-founder of something called React Miami. And more recently you co-organized AI Engineer Miami. What has caused you to be building these communities and how's it been going?
Gabe Greenberg:	01:38	COVID. We were in the middle of COVID and I was up too late on Twitter and I decided to talk about somebody should build React Miami. I bought the domain name and a hundred people responded that night saying, "You should." And I was ready to transfer the domain name

out. I didn't mean me. So it was just, we were all wanting connection. And honestly, this is not the React community of the Southeast Florida. There's nothing down here when it comes to deep, deep, deep, deep tech. It's growing, but this is the people in San Francisco and New York and London and Australia that are involved in the JavaScript space that in April they're like, "Hey, Miami makes sense." So it's been great to get Jordan who created React, Chang, the rest of the team, the core team 10 years ago kind of all together. And it's really been about community building, deep connections for the last six years with people who really get to know each other.

02:50 It's five, 600 people. AI Engineer Miami is about the same. Swix from latent space. He's a friend of mine. He's on our board at G2I and he started AI engineer in SF. So kind of bringing it to Miami, the vibe, it feels good. And yeah, deep connection. That's what we do.

Jon Krohn: 03:12 Yeah. The in-person connection is so important and maybe everyone felt like going to Miami because if I remember correctly, that is one place that people were still partying when a lot of the US was shut down. So it might've been the one place you could get those kinds of events together. But you've been running G2I, which I can't help but think might be related to Gabe Greenberg G2I.

Gabe Greenberg: 03:36 I don't know where you came up with that. Yeah. Many moons ago it had to do with my middle name, Gabriel Ian Greenberg. When this was a one man show in 2010, that's where G2I originally came from. You nailed it.

Jon Krohn: 03:51 And you've grown quite a bit since then. It's not just a one man show anymore.

Gabe Greenberg: 03:56 Yeah, we have 900 people right now.

Jon Krohn: 03:59 Wow.

Gabe Greenberg:	04:00	It's grown. There's a large contingent of that is contractors coming in, doing the data sets and building out RL environments. And then we have a pretty large core team as part of that.
Jon Krohn:	04:14	Wow. Well, we'll get into the reinforcement learning in a second. I think I'd actually like to focus a lot of this episode on that. But in terms of this genesis, the story of how G2I grew from one to 900, when I talk about React, for people who, it's a data science show, probably most people do know this, but we're talking about the React front end framework for web development specifically, just to clarify, it's not some kind of reactionary conference or something, React Miami. And it seems like G2I has a specialization in curating a bench of React engineers historically, React and other open source worlds. Yeah?
Gabe Greenberg:	04:55	Yeah. I mean, we started in the JavaScript pre-TypeScript community back in 2012 to 2015, 2016 when React Native was growing massively. You just had the big boom of SaaS, the golden era where everything was growing and Figma was new at the time and you could do more in the browser. So yes, the front end world became more complex, more important. And that's how it started, just talent marketplace. We got to about 15 million in revenue, no VC backing, and we did the screening and then broke out into everything from data science to machine learning to full stack engineers to insert here.
Jon Krohn:	05:44	Wow.
Gabe Greenberg:	05:44	Yeah, it was fun.
Jon Krohn:	05:46	Congratulations. $15 million in revenue with no VC backing is extraordinary. Definitely deserves lots of plaudits. These days,
Gabe Greenberg:	05:55	You

Jon Krohn:	05:55	Guys are making a lot of progress in reinforcement learning. So you mentioned RL earlier and reinforcement learning has become the key thing in training frontier models. So GPT-2, GPT-3, I think even a lot of GPT-4 kind of era, it was mostly about the pre-training and the reinforcement learning was maybe a few percent up to 10% of the compute. I recently read that what they call the post-training, although now it seems like it's a big chunk of the training, it doesn't really necessarily make sense to call it post-training, but that reinforcement learning post-training step is now it takes up about 50% of the compute and the cost of training frontier models.
Gabe Greenberg:	06:36	Yeah. I mean, again, our transition from talent marketplace where you have the handshakes of the world, Turing, there's lots of players in the market that started as a talent marketplace because we had screened domain experts. But back in 23, 24, when we were asked to do more and more, you were talking about 6,000, 7,000 people on a single project. The amounts of data that you needed were unbelievable. However, it was lower end expertise, a junior or mid-level domain expert software engineer could do the work. And so it's changed a lot.
Jon Krohn:	07:15	And so what you're saying specifically is to create the data for training frontier models with reinforcement learning. It used to be, of course, with relatively simpler models like a GPT-3 era model, it didn't require very advanced software engineering expertise, but these days you need very advanced people to be pushing frontier capabilities.
Gabe Greenberg:	07:35	Very, very advanced on pretty much everything. I mean, legal probably has the benchmarks are less saturated. That being said, just everything requires a staff level type person and it's still intense. It's intense like it was three years ago. It's just not quite as many people per project, not quite as much data per month, the units of data.

Things have moved over to RL environments where you have real code bases or environments built, there's more enterprise data. I mean, it's matured and the labs are maturing too with how they deal with human data or RL environments. It's a completely different mindset.

Jon Krohn: 08:20 I don't want this to be a controversial question given your business model, but it must be something that you think about as well. If we go from this over the past few years from being able to have 6,000 relatively junior engineers creating the problems for reinforcement learning models to be training frontier models, and then now it's a much smaller group of these, as you say, staff level engineers, maybe there's only six or 7,000 of those on the planet. And how do we go beyond them? Something like recursive self-improvement and then what's the human role in that? Well, I mean, I guess fundamentally, if it really is RSI, there is no human in the loop. So I don't know, this must be something that you think about. I don't mean to put you in a...

Gabe Greenberg: 09:02 No, I mean, it's a fair question. I think about it. It doesn't keep me up at night because there's so much that we can't... It's a very Jevin's paradoxy problem in terms of, it's not like electricity where there's a plateau and then 40 years later we need more because of AI and compute. It's like the desire for human intelligence doesn't just... It's just more is better and we keep wanting more and more and more. And the problems that we can solve in software are going from these crud apps that are relatively small to what Anthropic just did. They rebuilt their own internal GitHub and now that's the new app that software engineers would work on amongst many others around multi-agent orchestration and the complexity. We're going up a level of complexity. So I do think talking to people at these frontier labs for self-improvement at V1, version one is going to be here in the next year or two.

	10:00	However, talking to them, they don't think that necessarily no human data will be needed. They're still like, yeah, just because you have an engine that is very efficient on the research side doesn't mean that we're not going to have data coming in that's not synthetic. So I think it's still going to be needed. I think there's still a lot of room for enterprise data. There's a lot of room for complexity around reinforcement learning environments, and there's a lot of things we just can't even comprehend now that we'll need to do. But are we going to be in a gold rush five years from now? I mean, not necessarily. I don't know. But again, I didn't think we would need more software engineers and more data scientists right now. We actually do. So how long is this going to continue going up because the enterprise space has not been saturated?
	10:47	You're talking about barely any real Fortune 500 companies having AI truly installed.
Jon Krohn:	10:54	For sure. I had an interesting... So I'm running my own kind of consulting firm here that's at a stage that you were at seemingly like 15 years ago where I co-founded the business a year ago and we're now up to 10 consultants working with us and I've been doing all the sales up until now as the CEO of this business, why Karat? And so now we're starting to interview salespeople for the first time. And these salespeople have a lot more experience doing structured sales than I do and thinking about addressable markets and these kinds of things. And so our very first sales interview, the guy's like, "All right, so what's the TAM on this opportunity?" I was like, "It doesn't matter because it's orders of magnitude bigger than there are people that can do the work. I don't have a number for you. I wish I knew that number and that's a useful thing for me to think about, but it hasn't mattered up until now in the business and it doesn't concern me in terms of the growth of the business because the demand

in enterprises for the kind of talent and for the ways that we can be completing projects for them is so much faster.

12:07 The demand is so much faster orders of magnitude than there are people available to fill it.

Gabe Greenberg: 12:12 It is mind blowing. There's such a hunger, but the problem is there's so many snake oil salesmen. There's so many people that are sprinkling something here or there or have a harness that sort of works in this direction. But in the enterprise space, the complexity usually doesn't mesh with an out of the box SAS solution. They need something that's their own and really can work with their security needs and all of the legacy stuff. I mean, you're talking about we're dealing with some legacy stuff that's 20, 30, 40 years old. How do you rewrite that and then deal with all the data in different places and go to governments? We're talking with a government overseas about having orchestrate our multi-agent operating system do all their procurement, their critical infrastructure. Yeah, it's a greenfield opportunity and there's 10 companies I can think of that have really made a difference in building their own harness and it's because they have such great engineering power.

Jon Krohn: 13:23 Yeah, there was a report that made a lot of headlines the morning that you and I are recording where the chief economist at Apollo said that basically all of the productivity gains in the S&P 500, so the biggest 500 firms in the US have been 100% of the gains from AI have been in the magnificent seven, the big seven tech companies and that 493 businesses have not had any productivity gain from AI.

Gabe Greenberg: 13:53 That's the reality. You're not going to have productivity gains when you're at level one where it's like, hey, either you're not using it or you're starting to go into it's augmenting, it's like a chat interface. Until you can actually have the agents orchestrate long horizon work

and until you can have you orchestrating multiple agents orchestrating multiple harnesses really across long horizon work, you're not really replacing a lot of money coming in the door and there's not a lot of ROI when it's just a chat interface. It's babysitting. As long as we're babysitting, it's harder.

Jon Krohn: 14:37 Yeah, I agree with you. Yeah, there's a lot of data backbone to get set up in these organizations, permissions to figure out before we can have agents doing very much. You were talking about a long horizon work there versus short horizon work. And that reminds me prior to us starting recording, you were talking about Meter. So I always, for years now, I have had charts from METER, which is model evaluation and threat research. You can check them out at meter.org. And so they're an independent research outfit that does great work assessing frontier models specifically on how capable are AI models on human tasks where those human tasks are. We know how long those human tasks take based on just like with your business, you have humans evaluating tasks and coming up with new kinds of scenarios and you can measure how long it takes people to do tasks.

15:32 And so you can, for example, have finding a fact on the web was a simple GPT-3, GPT-4 era task that you could ask a human to do and benchmark an AI model against. More recently with Claude Opus 4.6 capabilities earlier in 2026, you're talking about tasks like how do you optimally reduce the size of a language model, which is something that takes humans like eight, 10, 12 hours on average. And with the release of Claude Methos this summer, the Northern Hemisphere summer, Claude Methos broke the kind of meter evaluations on these charts because the time horizons that Methos was suddenly being able to handle were plus 16 hours and Meter hadn't been collecting data on tasks that take humans more than 16 hours. And can you imagine it

would cost so much more, it would take so much longer obviously to do that.

Gabe Greenberg: 16:25 Yeah, I think there's a lot of saturation in the 16 hours or less, which is Meter considers short horizon. The long horizon tasks, which is where you really take a team of four, six software engineers, data scientists, researchers, whatever, insert here, and they're building an entire software application with lots of depth and complexity to it. If you can start to do that autonomously, you're really bringing a different level and that's what the enterprises are looking for. Salesforce reached out to me, so this is public, but we announced that we were replacing Salesforce. The Salesforce team reached out to me and emailed me said, "Okay, is there anything we can do?" I said, "I'm addicted to being able to, if I need a feature, being able to add it." It took us about two weeks. I did the front end work in Cloudcode, and then we moved everything into Orchestrator orc.ai, our autonomous software factory, but it's a substrate.

17:27 It just handles... It's an agentic operating system, so it can handle all kinds of use cases. It took Orchestrator seven, eight days, about $4,000 to put a D plus in code quality into an A minus where it's secure, it's scalable, it's performant, and we use it. It's going to replace Salesforce officially at the end of the year when our contract ends. So I can do that over and over again. Anytime I have an idea, anytime somebody on the team has buy-in, they can vibe code something, get the user interface working really well, put it through Orchestrator, it handles the rest, has someone polish it off for a day, and all of a sudden you have that used to take three to six months, somewhere in that range, depending on the size of your org. If you were a startup, it might take you three months.

18:22 If you're an enterprise, it might take you nine. So a lot of the enterprises are asking for, "Hey, how do we take the average person on our team, what they need? How do we

put it in production? How do we have governance, security, all those pieces?" So this is happening over and over again. How do we patch security vulnerabilities in 24 hours? We can't do any of this. What if we get attacked by one of these agentic attacks over the last week? What happens there? We need agents running twenty four seven. How do we do this efficiently? How do we pick the right model? And there lies why Open Router was acquired is because a lot of the game will be picking the right model that's cost effective and does just enough for what you're looking for, for said task. And after adoption, you have cost.

Jon Krohn: 19:10 So Orchestrator, which our listeners can go to at orc.aiorc.ai, that is a platform built by G2I, yeah?

Gabe Greenberg: 19:22 Yes.

Jon Krohn: 19:22 And if I understand what you were saying correctly, G2I have themselves used the orc.ai platform to create your own Salesforce. Is that right? When you say replace Salesforce, so instead of having a Salesforce license, you now have your own in-house built one.

Gabe Greenberg: 19:40 Yeah. I mean, it was costing us a quarter of a million dollars a year at our size and was going to go up. So what's the point when I really want an agentic driven operating system to run software? I want agents to figure out and learn the things that we need, and I want three or four agents on certain tasks that are mission critical, whereas sometimes single players find. And that's the issue that we're seeing. And we built this because we needed to build RL environments at scale. Our orchestrator's probably written 20 million lines of code for us, maybe 20 to 30. 10 million of those are in Frontier Labs today, right now, being used every day by researchers. So yeah, we had the scale to use and we were spending $200,000 on tokens a month at a time at

that scale. Now it's down again because we're more efficient.

20:38 But yeah, we have our own little data center through TailScale with GDX Sparks and Mac Ultras, and then Daytona takes care of the rest. Daytona's phenomenal for just spinning up sandboxes. Wow. We can have 100 to 200 instances of Orchestrator running at a time writing real production code.

Jon Krohn: 21:02 Really cool, Gabe. You really are at the frontier. It is not surprising that you're supplying Frontier Labs with the data that they need. And so to give listeners a sense of the scale, you have 8,000 engineers on your bench that are vetted, ready to provide data, particularly for reinforcement learning at the Frontier Labs. You paid out $50 million over the past year, five-zero to those 8,000 engineers for the work that they're doing. As you scale to that kind of number of people working for you, how do you keep quality from degrading as volume grows?

Gabe Greenberg: 21:48 Well, we've hired from a few, specifically, we brought in somebody from XAI actually, and who's very passionate about this, to join our team and increasing our research team, as well as just the managers involved. Each manager is very focused on signal versus noise, what really matters. We found there's a few things that matter to quality data, and it is somebody that knows how to do that type of work. It's tedious, it's not easy, it's not a dream job, but it pays very well. I've actually tweeted about that transparently. If you want to join our team, this is not traditional software development work. So we asked them to go through a pretty short interview where they asynchronously do some human data work and get a feel for it, and we're looking for their level of English. We're looking for if they cheat, if they're using a large language model, which is going to contaminate the data, what their level of ability is within the specific domain.

	22:50	Specifically for us, it's a lot of different levels of different languages in software engineering today. So once you get that piece done, it's a lot of training. You want to teach engineers or anyone for this matter, lawyers, doctors, there's probably 15 main domains right now going on on a regular basis. AAA Games is being trained. You want to teach every single person what the outcome looks like for a good post-training run, and you can do that through benchmarks, you can do that through letting them read papers, and they're like, "Oh, if I use a large language model to develop my annotation or my data set, it's synthetic and it's going to reduce the capability of the model. It's going to contaminate the data, that's why we're being hired, so let's not do these types of things." So I think those are the two areas. The last area is just paying people well, you're going to attract better talent.
	23:53	I think where two, three years ago, the pay rates were like 20 to $40 an hour, and now on some projects we're able to pay between 200 to $300 an hour. So those three elements I think go into quality at scale.
Jon Krohn:	24:10	Yeah, that makes a lot of sense. You'd have to be paying those kinds of rates to be getting staff level people for sure. So let's talk about benchmarks, which you mentioned there. So evaluations and benchmarks. You have done reviews of a lot of the software engineering benchmarks. So take us through the journey from... A lot of my listeners will have heard of SWEBENCH, S-W-E Bench, Software Engineering Bench as a benchmark, but as AI models have become wildly better, especially within their agentic harnesses in 2026, Sweebench has become too easy in a lot of ways. And so tell us about the follow-up benchmarks to Sweebench.
Gabe Greenberg:	24:47	Yeah, Sweebench was a standard for a long time, and Sweebench and Sweebench Pro lasted for longer than they should have, but you'll see this with any benchmark as it goes along. There's reasons that it becomes

contaminated, and there's also reasons that it becomes saturated. And the saturation, just the benchmark wasn't good. It wasn't designed well, and a lot of it came down to the instructions. The prompts were not realistic. So if you're coding, everyone codes now at some level, right? We're all writing code through prompting. You're not going to tell a Claude Code to find a bug and then go through the steps in how that bug should be solved. But Sweebench Pro did. Their instructions were way too detailed. So in a sense, it was cheating. You have issues with... Yeah, so prompts are just one of the pieces. Verifiers can be too easy, too hard, et cetera, et cetera.

25:53 So when you finally got to... And you could see it in the output around Sweebench and Sue Bench Pro, all of the models were very close to each other on these charts. They were always 10, 15% apart. And even Gemini 3.8 Flash, who everyone knew it was a lighter, smaller model, and it'd be closer to the fables of the world. DeepSuite was the first one to finally get the prompts better, the verifiers better. Everything was in a better spot. It still could be improved upon, but when you get that right, you start to see these... There's finally just a better spread in how the benchmark looks and it aligns to people's experience inherently if you're using these every day. I know this is twice as good as this. So I think that was some of the change in the coding models. Now you're seeing these models look at a lot of more long horizon tasks.

26:54 And that was the issue is the saturation is coming through short horizon work. If you have a task that's taking five hours, it's much more likely to be done by the model. But the newest benchmarks are looking at 10, 15, 20, 30 hours. Some are looking at no instructions or no spec, just build the software, extremely difficult things that are under 10% saturation. So I love what everyone's figuring out, that it's not always that the models are where we think they should be, it's that the benchmarks

are improperly made or too easy. Computer use is an area that we're personally releasing a benchmark in the next month, month and a half. OS world is a fantastic start, but if you look at the Microsoft research they talked about publicly, because it makes sense. Think about Microsoft, what do they want to build for? They want to build for their operating system.

27:51 So you think computer use would be pretty important to them. Publicly, they were not allowed to have their own model until October of last year because of their relationship with another lab. This public information, and so they just started over the last 12 months. So Satya has talked about reinforcement learning environments and they know, they've published the research that if you don't have high fidelity research environments or RL environments that really match the enterprise applications you're trying to handle and work better with or the depth and complexity, the surface area, it doesn't help the model very much. You can't be like 80%, you have to be like 98% of fidelity. And this is all across the stack from the front end, the UI, all the way to the MCP and the API. It gets very complex, so it becomes a true software engineering problem, and this is a space that's heavily research and machine learning focused.

28:58 So you have machine learning and research focused people trying to write production software, and that's where there's been a flip in the RL environment space because it's difficult.

Jon Krohn: 29:08 As one last final technical question for you, many of our listeners who might have experimented with reinforcement learning in the past, for example, in 2019, I published a bestselling book called Deep Learning Illustrated where I had a chapter on deep reinforcement learning. And in that chapter, I had my readers work with the cartpool game, which is an extremely simple reinforcement learning game. Or people might be aware

how a decade ago, Atari was a really important reinforcement learning environment for those kind of earlier deep reinforcement learning models to be able to get better at. And a key thing in reinforcement learning is the reward. So how do you model when... So basically the reinforcement learning algorithm gets points for performing well. It has points taken away for doing things incorrectly. In Atari video games, most or all Atari video games have a score built in. So in Tetris, if you place a block, you get a slightly higher score.

30:14 And so it becomes relatively easy in those earlier problems to be able to define what your reward is and to reward the algorithm for doing well at taking rewarding actions. Tell us about how things have evolved. You gave us a picture there in a hand-wavy way in your most recent response, but how have reinforcement learning environments evolved and how do we track reward effectively when the actions are so complex?

Gabe Greenberg: 30:45 I wish I had a PhD and I could answer this in a less hand-wavy way.

Jon Krohn: 30:49 No, I didn't mean that as a dig, by the way.

Gabe Greenberg: 30:52 No, no, no, no. I think it was funny because I was literally waving my hands. So I could tell you what goes into it, and that's my expertise and what our company does. And the researchers, we're onboarding somebody who just left OpenAI to lead research, and I'm really excited to see where some of the understanding with rewards and what's missing. But once you get the complexity and depth of an RL environment, whether it's coding or whether it's an enterprise application or consumer application, a year ago, consumer and enterprise applications became forefront of what RL environments truly mean. There's an element to you need to put really, really messy data in to start to figure out what should be reward and what shouldn't, where the gaps are. Well, you

need to go out and buy all of this data. So a lot of these human data companies are going and buying a couple million dollars worth of data.

31:59 Look at Spirit Airlines, theirs is bought for 12 million, probably overpriced because to pull it out is another thing. But when you get complex data, and let's just say you're working across many different RL environments, you talked about hiring salespeople. They're working across HubSpot, email, and Salesforce, and they have a contact that's Gabriel Tavera and another one that's Gabriel Tavera. The model is going to get confused today. Today's model gets confused. And there's lots of gaps like that when you don't have complexity in your data. So I think that's one piece of it. We could probably talk for another hour on where this is all heading, but yeah, that's the start is having the complexity of the data. There's also not a lot of people front running the gaps and making the tasks break the best models in the world. And that's because if you look at researchers in these labs, they're like, "Hey, I'm for finance, I'm for health, I'm for coding." But RL environments is infrastructure for everything.

33:17 So they actually lack some research bandwidth to focus on, "Hey, how do we take eight different environments and how do we make tasks that are above the complexity that's saturated?" That's where these human data companies are starting to come in. As a consultant and front running the labs, which is not normal, a lot of the labs are like, "Hey, you follow our lead." And when you talk about rewards, we're starting to find some of the gaps there, but it's expensive.

Jon Krohn: 33:48 Well, congrats, Gabe, on all your success building this data provider that is ahead of the frontier. Being ahead of the frontier labs, it's wild. Congratulations. It's been amazing to hear a bit of your story. We could have easily talked for hours, but the Friday episodes are shorter like

this. So thank you so much for taking the time out of your super busy week to give us this highly concentrated episode with a very high reward, indeed. Thanks. Before I let you go, something that I ask all of my guests for that I didn't give you a warning about is that I always ask for a book recommendation.

Gabe Greenberg: 34:34 Ooh, that's easy.

Jon Krohn: 34:35 Okay, great.

Gabe Greenberg: 34:37 The Science of Scaling. It is Elon Musk's philosophy of company building, but it never mentions Elon Musk. And the 10 second version is stop using goals as achievable and start using them as though experiments and then work backwards and find the limiting factor. And we've seen a 6X growth since we've... There's one business book I'd ever recommend and it's that. If you have a technical business or a non-technical business that it's the real deal.

Jon Krohn: 35:09 Very cool. Thank you for that recommendation. I actually had not heard of that book. And based on everything you've said in this episode, it does seem like I should be taking some pointers from you. So thank you for that, Gabe. Final thing, how should people be following you or your business after this episode? Obviously we've talked about org.ai, but yeah, what about you personally?

Gabe Greenberg: 35:27 Yeah, so g2i.ai is our human data business. Org.ai is our enterprise focused business and Twitter's the main way. Twitter or X as we call it now, x.com/gabe, G-A-B-E, Greenberg, Green, B-E-R-G, is the best place to get me.

Jon Krohn: 35:45 Nice, thanks. We will of course have those in the show notes. Thanks Gabe for taking the time and hope to catch you again soon.

Gabe Greenberg: 35:50 Yeah, looking forward to it. Thanks, Jon.

Jon Krohn:	35:54	Inspiring episode today. In it, Gabe Greenberg detailed why he isn't losing sleep over recursive self-improvement, even though people at Frontier Labs expect a first version within a year or two because our appetite for intelligence keeps growing like a Javon's paradox. He talked about why Sweebench and Sweebench Pro saturated with overly detailed prompts and weak verifiers leaving every model bunched together on the leaderboard. And he talked about how reinforcement learning environments need around 98% fidelity to real enterprise software and messy realistic data to improve a model, which is why data companies pay millions for data sets like Spirit Airlines. I hope you enjoyed this conversation today. To be sure not to miss any of our exciting upcoming episodes, subscribe to this podcast if you haven't already, but most importantly, I hope you'll just keep on listening. Until next time, keep on rocking it out there and I'm looking forward to enjoying another round of the Super Data Science podcast with you very soon.