SUPER DATASCIENCE
MAKING THE COMPLEX SIMPLE

SDS PODCAST EPISODE 971: 90% of The World’s Data is Private; Lin Qiao’s Fireworks AI is Unlocking It

Jon Krohn: 00:00 Over 90% of the world's intelligence is locked inside private enterprise data that no foundation model has ever seen. Today's guest is on a mission to unlock it. Welcome to episode number 971 of the SuperDataScience podcast. I'm your host, Jon Krohn. Today's guest, Lin Qiao, is the CEO of Fireworks AI, a bay area startup that has raised over $300 million to unlock the world's vast quantities of enterprise data for LLM training and inference revolutionizing capabilities and performance with a PhD in computer science from uc, Santa Barbara, and years of experience as a director of engineering at Meta Lin is now a highly successful technical founder with a rich perspective on AI today and what the future holds for all of us. Enjoy this one. This episode of SuperDataScience is made possible by Dell Intel, Cisco and Acceldata.

00:55 Lin, welcome to the SuperDataScience podcast. It's an honor to have you take time out of your busy schedule to be on the show. How are you doing today?

Lin Qiao: 01:03 I'm doing great. Thanks for having me Jon.

Jon Krohn: 01:05 Of course. And where are you calling in from?

Lin Qiao: 01:07 I calling from Portola Valley. That's where I live.

Jon Krohn: 01:10 Nice. And that's it's part

Lin Qiao: 01:13 Of Bay Area?

Jon Krohn: 01:14 Yeah, yeah, yeah. Palo Alto.

Lin Qiao: 01:15 Yeah, very close to Stanford.

Jon Krohn: 01:17 Nice, nice, nice. So we're here to talk about Fireworks AI your business, which has done incredibly well. I mean you've just grown so quickly. I believe you've now raised over 300 million in venture capital including a recent $250 million series C if I got that correctly. And so it's a platform built around open source model deployment at scale and the Fireworks AI platform is built around open source model deployment at scale and this idea of autonomous intelligence, tell us what that means Lin.

Lin Qiao: 01:56 Yeah, sure you're right. We raised our last round last year and we are growing really fast. So our mission is autonomous intelligence. This mission is very complimentary to a GI where the direction of a GI focus on investing in directing a lot of intelligence into this one model and have this model be able to solve various different kinds of tasks in a great way. So the idea is you just build your application on top of the A GI model as a utility, right? Great. I is a great direction. It's very scalable if it's successful, but the reality is only a very small fraction of data goes into the foundation models for a GI. If you look at the worst data, majority of the data, by majority I really mean more than 90 of the data is actually not in the public domain. It's not in public internet, it's not labeled by labeling companies which goes into the foundation models and majority of those data are private data locked inside applications and enterprises and data.

03:19 We all know data is intelligence, data is knowledge and those application specific enterprise specific data is not accessible by the A GI labs and we just leave a lot of intelligence on the table. My prediction and that's where we're betting on the future, is to be able to activate those private data and let the model absorb additional application specific intelligence and bring the model to the next level. And this kind of motion is more like customization, right? It is the model and the inference deployment we're customized towards applications in their specific pattern and this customization should not be just one time our application enterprise product that keeps evolving. So these customization should be continuous and ideally these continuous customization should be fully automated. That's autonomous intelligence. We are making great progress towards that direction and we believe the future is not one model all it's going to be millions of models, one per application per use case.

Jon Krohn: 04:39 Alright, so the term autonomous intelligence, it sounds kind of vaguely to me like the buzzword of 2025 in our field, probably the buzzword for 2026, which is age agentic ai, but it sounds like it's quite different from age agentic ai, this idea of autonomous intelligence,

Lin Qiao: 04:57 It's heavily connected with agentic ai. So think about agent is a way to automate many of our day-to-day task. So we have being living the world, that many expert intense task has been gradually automated so we can free up our time eventually some of the even professions will be redefined. So for example, I think there are interview agents or hiring agents where you give a job listing, it will source the candidates and into even do the first rounds of futuring and interview for you and there's marketing agent, you give your ICP list and will source the right company, right stakeholders and start to drive customized outbound emails and riches and their customer service agent just give the human agent some really good assistants who be smart and there are so many agents to doctors and so on. So this is happening, it's transforming our day-to-day life.

06:07 But similarly, another big transformation that's happening is in my domain is software development is being disrupted and today a coding agent can really start to behave like a junior engineer and I'm no kidding, this is kind of really happening and it's actually changed, it changed our interview process and the fundamental question we're asking ourself, is coding interview important anymore? So coding interview in the past is going to replace by how good you are at using coding agents. So it is actually happening across our day-to-day life. Now let's go back to this autonomous intelligence era. So without that, currently this work of continuously adapting the model and changing in customizing inference setup is done by a very, very small set of experts. So those experts are like they have been doing AI system for a long time. They have been researcher for a long time, accumulated their knowledge over years.

07:36 So only a few companies who has those strong density talent pool are able to do that. And the question is, can that part be automated and similar to others that has been disrupted and has been reshaped and can this part of doing product model and infusing more intelligence in the model and making the infra serving tier much faster and much more efficient, can that part be accessible by a wide range of application developers without them putting in a lot of work and carrying the burden of learning all the deep acknowledges. So that's what that means. So we heard a lot about AI is going to free up a lot of human labor and this wave is interesting because it will start from different angle. It will actually free up a human from the high intelligence level, not from the physical level. The robotics is disrupt the first level of engagement, but AI is going to free up a lot of high intelligence level of the tasks work. So that's kind of interesting change and we are also innovating and disrupting in that space from a baseline platform space.

Jon Krohn: 09:16 Really cool. So it sounds like yeah, autonomous intelligence builds on ag agentic things but also involves lots of things not associated with ag agentic like systems, like models automatically retraining and the whole kind of system humming along nicely. It seems like a key part of that working for you, especially in an earlier answer. You mentioned how you see the future as millions of different models. These could be like LLMs but millions of different AI M models that are tuned to specific tasks within specific enterprises. And it sounds like Fireworks offers a reinforcement fine tuning product that allows your customers to beat frontier closed models in under a month on specific tasks that relatively small tuned open model is fine tuned for.

Lin Qiao: 10:14 So we are very bullish in this direction. So think about AI has been following a lot of full step of human intelligence. So this artificial intelligence has been following that full step. Even the model architecture is called neural networks. It's kind of emulating the human's brain. So for reinforced learning, it is actually very similar to how we human learn knowledges. So we learn by various different angles. So one of the angle is we get positive feedback that we know, oh this is the correct thing to do by learning the principles or we get negative feedback and it's we get penalized for doing something bad from our behavior point of view, then we learn, no, don't do it anymore and we kind of change to a different direction. So this also happened to how we do agriculture for example, as I'm a food lover and today we all like to eat sweet fruit, the fruit also grows much bigger.

11:31 But this is not how originally it became right, it gone through multiple generations of selection process where we collect the seeds of the sweeter and bigger fruits and the planet and among those and collect another. So this is kind of another way of reinforcement learning, reinforcement seat collection, selection process. So a lot of things we do whether for ourself our own learning process or we have applied in other domains as following the same principle. And this is also similar to how model learn is you teach a model what is a positive reward, you teach a model what is a negative reward and the model is going to automatically scan through a search space of possibilities and through these feedbacks and find a path to specialize solve in solving certain kind of problems really, really well. With that said, it's not all only benefits, right? It's a trade off because if you let the model specialize in certain direction in the area, it's going to be less specialized in other areas.

12:45 So it's like us it exactly as Jon, you're specializing in driving this podcast and you're great cozy, you have very knowledgeable how to engage with guests and I specialize in building the best AI platform which can customize towards application specific patterns and so on. But then I'm not a good chef or cook and I don't know actually how to do gardening very well. So that's kind of a natural selection we have direct our attention to similar to the models. So that's kind of where we are betting on this direction to make reinforcement learning for models very, very accessible to all application developers so they can basically have a model in tune to their product all the time. And imagine that, imagine a model is just constantly learning the intelligence on your application and then you have your private model and then you have your mode where nobody else built on top of an existing API would have. So this is a special thing that we want to have every application developers to get hold of.

Jon Krohn: 14:15 Right, I see. So all of that private data that you mentioned at the outset of the episode, all that accounts for most of the data in the world, enterprises can be using their particular private data to be creating a moat by not only having that private data but also having these fine tuned models, specifically specializing in particular aspects of their data applied to specific tasks.

Lin Qiao: 14:41 And this is not a new thing actually. For example, during the mobile first move, so before ai the biggest shift is mobile. So the application moves from desktop to mobile devices and that actually opens up a whole new domain of doing product development which is similar to autonomous intelligence we're heading towards in terms of thinking on mobile first, I think one thing that opens up is the access of end consumers to an application because the people who owns desktop and versus people who owns the phone, it is orders of magnitude different and that just means now your product could have reached to orders of magnitude higher group of people and it will go global much quickly and you will be able to access various different demographics of the cohort much wider. It changed how ER think about product evolution because now it's much broader and you cannot just deploy a group of PMs to understand what the customer want and the product no longer become monopoly just one design because for different cohort coverage you may want to highlight one feature versus the other. So then it becomes, oh, how do we even do product development? How do we incorporate that feedback? It has to be customized and that customization is being done through a new technology called AB testing. So the idea here is to use the insights from your product to compare A versus B of the product feature and make statistical decision based on statistical result. Make the decision where your product should look like within certain cohort of groups.

16:54 The idea is basic, similar like product has a lot of intelligence. We need to leverage that intelligence to make the product better. And here similarly product has a lot of intelligence. We need to leverage intelligence to make the model better for your product. Therefore your product built on top of a specialized model will be better. So right now it's a completely open space, it is a vacuum and there's no existing solution that has been solving this problem really well. I'm pretty sure industry will move towards in looking at this space very closely and I firmly believe there's a lot of value in there.

Jon Krohn: 17:38 Awesome, I love it. Yeah, really exciting. You're kind of defining a new category, which is an exciting thing to be doing. So when people are trying to figure out these relatively small, say LLMs for a particular task, does model selection really matter? I mean are your clients picking, oh okay, I'm going to use this llama model of this size or Quinn or do they make those decisions or is this something that's kind of handled automatically by the autonomous intelligence system?

Lin Qiao: 18:11 So model selection matters but it's also exhausting. So funny thing, there are two interesting phenomena that's so unique to ai. One is the model depreciation is very fast as you can observe. Every couple of weeks there's a new model launch, whether it's closed open, someone topped the leaderboard and then a couple of weeks later someone else topped the leaderboard. And we often, they all are strong in different ways. So we start to see the researcher focus start to diverge. It is clear some labs are really good at chat based models. Some labs are really good at coding to use age agentic models. Some labs are really good at multimodality models, some are really good at long context. They all start to diverging to focus on specializing in different areas, speaking about specialization. So different use case will suit different kind of model specialty very well, but it's really hard for people to figure out which one is the best for my use case and not my use case will also evolve over time.

19:41 And the benchmark result, public benchmark result is fully saturated and when to figure out how to pick and choose continuously, which is exhausting. So we do help our customer figure that out. But at the same time there's another level of complexity is hardware depreciation is also very fast. This is something new before this wave, usually every three years there's a new hardware skill and now last year myself, Nvidia launches three skills, three new skills and 2026 there will be a lot more new skills from all different vendors whether it's GPU or customer isic. So then how to manage hardware becomes a very hard problem and these two combined is causing so much headache to the application developer who want to stay on top of all different kind of wave. It's just too much deep knowledge and expertise to gain in order to pick the best for them. So we are the platform that we abstract out. We eventually want to abstract out hardware, we want to abstract hardware selection and we want to kind of provide the best model for various different use cases. So basically there are various different kind of mapping from specific use case specific workload patterns to the model, to the hardware, to the inference setup, to the fleet design. So all of that requires a big team in-house with deep experts, which is really hard to find and we want to make it super easy for our customers.

Jon Krohn: 21:41 So instead of needing to find the D experts, they can just come to Fireworks and work with you guys, work with your solutions.

Lin Qiao: 21:48 Absolutely. Interestingly, last year solve, I haven't counted every month there are few new models get deployed and launched and it has been very exciting, but also I would say early on we better on open models. That was a big debate in the company that we debate on many what ifs, right? But because our background in PyTorch, which is open source project PyTorch is now we built PyTorch from ground up. It is now the dominant AI framework. We firmly believe in the power of open science and it will in the long run. We believe that is going to really shape the industry in an unprecedented way. So that's why we better open models from very early on and it's great to see that open model performance is converging with close model. I think 2026 is the year that that convergence will become more prominent and super excited about that.

Jon Krohn: 23:04 Yeah, something that I haven't mentioned on air yet is that for seven years before founding and being CEO of Fireworks AI, you were at meta as a senior director of engineering where you led over 300 engineers and a big part of what you were doing there based on what I could find online is developing PyTorch. So thank you for that.

Lin Qiao: 23:25 This is definitely a very big team effort and I have a very talent team at that time. Yeah, so I'm very thrilled about the industry-wide impact of PyTorch. There are many great researchers and leaders, engineers I'm lucky to work very closely with and some of them start a company with me Fireworks. So really great crew and we continue to going down the same path of democratized AI to the whole entire industry.

Jon Krohn: 24:03 Really cool. All right, so back to Fireworks and what you were saying about open source, not just PyTorch but open source models and how you at Fireworks have made this bet to embrace open source AI models. Do you think that a lot of organizations, a lot of enterprises underestimate what's possible with open models?

Lin Qiao: 24:23 I don't think it's about underestimation that much, it's just about not being familiar with open models. I think especially I think I will put simplify this answer, there's two groups, there's AI native startups, there is enterprises, obviously enterprise has digital natives and traditional enterprise and various different categories. But let's just take a look at this group. Two groups, they're reactions different and native startups are, they just want to try the best option and they do not have any baggage. They're very, very brave in testing open models. They're always very curious. And at the same time there's a very interesting phenomenon that's also unique to ai. So as you know, there are many companies, whether startups or incumbents, they're trying and experimenting with new user experiences that our day-to-day is interacting with. And those kinds of experiments have seen a lot of success in terms of product market fit.

25:48 But product market fit doesn't at AI time doesn't mean a viable business. There could be a big 10 that people are willing to pay the product, pay for the product, but the cogs of running the business could be much higher than the revenue you get. And then people just cannot scale their business after they hit the product market. And it's funny, literally they're going to scale into bankruptcy. So obviously startups have limited funding and even including incumbents, they cannot have the distribution and they have to be so careful that CFOs have to so careful in controlling the cost. So the kind of budget and this RI totally makes sense and because of that they have to hold a lot of people on the waiting list cannot open up the gate. So while the startup going back to AI native startups, they're very brave in embracing open models, but part of motivation is open models are they have a lot more control.

26:58 They can customize hardware they want, but also at the same time open models are much more economical because we as a provider, we do not have pre-training costs. Billions of dollar pre-training costs amortized over and we only focus on bringing the best quality speed and cost like from post-training, which is much cheaper and inference customization of the deployment. So the unit of economics of Fireworks operating open models very different from Frontier labs. So that's where we see the A native startups are very brave in adopting and going down the path of customization with open models. The second group of enterprises, obviously they're much more careful because many of those are public companies and they are under certain kind of obligation, especially their legal team to understand what are implication of using open models license, what are license limitation, just kind of trying to understand what is this beast? But from my observation, the enterprise start to open up and they start to understand much better, oh this model actually is not going to send data to other countries. It's just a model and if the model provider, it's hosting a model in US and then they can provide privacy and security around the ins and outs.

28:42 But anyway, there are other concerns as in the kind of content generated does that follow certain kind of guardrail and so on. But overall I've seen the whole entire industry start to understand better and start to understand the bounds of operating open models and I only see an upper trend of embracing open models.

Jon Krohn: 29:06 Nice. That was a very cool explanation. Of those kind of two customer types, the AI native startup, the enterprise, it sounds like regardless of which category a customer of yours falls into, a huge advantage of Fireworks must be that they can not only are running open source models cheaper like you said, but by running them through you it means that they don't need to themselves be buying the GPUs, be managing all the ops around that, physical ops, software ops, they just can rely on Fireworks to get things up and running and then having the right GPUs running behind the scenes and controlling those kinds of costs is handled automatically.

Lin Qiao: 29:53 So at the beginning of this conversation I mentioned AI is changing a lot of things in our day-to-day. I think one fundamental thing is changing is the velocity of product development on AI is extremely fast including the enterprises. So because of that it's really hard for enterprise internally to forecast how many much traffic this product is going to generate. It may generate no traffic because the experiment doesn't go to the face, they can go into production or it can generate massive traffic. It's kind of the variation is very broad in this fast evolution of product development. So then it cause a big problem because without a steady prediction of how to do capacity from finance point of view and then what does that even mean to procure hardware and you either overprovision or you either or you underprovision kind of the range is very broad.

31:11 So that's where we can help because we are aggregator, we aggregate demand across all different customers and we take that risk off the table from being you want to worry about that and we're very flexible being accommodating varying requests because we can kind of add them together and drive the adoption. So I think that's kind of the unique nature of ai, but it's not that unique. But think about that, right? So in the early days of cloud first, before cloud first everyone, every big enterprise they have mainframe. So data center locking with keys in a cage of machines, this is the most rigid way of doing capacity planning and the cloud. When cloud comes into the picture, I usually people think you are crazy, why would you move from the most secure enclo deployment of your infrastructure into renting someone else infrastructure and you run your most important application there, but guess what? That cloud infrastructure provides so much flexibility that over the long term it's much more efficient to run. So similar things is happening in AI space and that's why it's fascinating. The velocity of AI development is shifting and changing and even disrupting how we do business in the traditional way.

Jon Krohn: 32:48 Really cool answer again from you there. Thank you so much. I've learned so much from every response and so it seems like you might have an interesting and informative perspective on reasoning models or slow thinking models. So that's been a big thing last year. OpenAI, anthropic, Google, there's a big fuss around the slow thinking models that they released and particularly their performance on complex tasks like math olympiads and writing academic papers and these kinds of things that require more processing before just outputting some kind of train of thought. I noticed on the Fireworks website that you mentioned very fast latency like sub two second or even sub 500 millisecond latency in a lot of real world deployments. Do you think that the slower reasoning models that take time before they output something, do you think that there's a lot of enterprise need for that?

Lin Qiao: 33:52 So our cost of classify, again extremely simplified way of looking at application. So one type of application is real time response. So it's either human facing interactive response latency or it's kind of fast transaction facing, for example fraud detection. So I aggregate way is kind of very fast going, the type application is offline. So for example, I have a case study legal case study, I need my legal assistant to go off and analyze similar cases and come back in three days, give me a report. Now an agent, legal agent can go off and do this study for a couple hours and come back. So those are two very big categories of agents or agent applications that has different latency requirements. So typically for the real time, extremely low latency requirement, you cannot afford to think, you have to basically react and usually the customization go goes back to customization, right?

35:08 Usually the customization process is very interesting also is you go small, big and then small for real time use cases. What does that mean is when you customize a lot of time it's about your data. Is your data high quality? Keep in mind this kind of garbage and garbage same apply here. If your data is not high quality, then your end model is not high quality. So for you to test data quality and the fixed quality issue, you start with small model to see if the model is going the right trajectory and then if it is going not going, you go back to fix our data and then small model help you irate fast, but that small model is not important. The trajectory of the model quality change is important, which you like the trajectory and then you move to the biggest model and usually those are biggest MOE model really hard to get it running.

36:08 So obviously that's where we will definitely help you and use the clean data to tune the model, get the best special ed model to solve your real-time problem. But usually those largest model is not as fast because they're very big, they're trillion trillions of parameters, very big but very good in terms of quality and then you go small again, but you still found the biggest model. You have tuned into small model because you need to fast response and then you launch the smallest model. You can get to high quality from that process and can go from there. So small, big, small right now we move to offline asynchronous agents and usually those agents are doing some deep research in certain kind of domain, whether it's about legal, whether it's about finance, whether it's about software engineering or whether it's about something else. But usually it takes time to think like us human being, we're going to hey, do some homework, go hiding a cave, figure it out. So similarly and those model requires the highest level of intelligence, highest level. So then the kind of slow thinking mode becomes extremely important. But not just that.

37:35 Those big models for offline research also requires a lot more context, a lot more context will make the right thinking process, get the right design of the flow, but at the same time a lot oftentimes those model also need to get specialized in solving certain kind of problem really well. The legal process, the finance process, the process of working, building PowerPoint, the process of building a executive overview, a pitch to an investors and the process of doing customer service, those are all very, very different. So we have also seen a lot of application agent developers, they start to customize offline agent where that just means they need to figure out how to tune with the thinking tokens. So that's kind of a very different classes of occasion. I'm oversimplify here.

Jon Krohn: 38:52 I mean oversimplifying, but making it very easy to understand as you have with all of your responses in this episode. I love particularly the small, big, small approach to finding the right model for your use case and having that be performance in real time. As you were discussing that, something that came to mind for me is how one of the most difficult things with deploying such complex models that have stochastic outputs is evals. And so even when you're doing that small, big, small, how do you ensure that when you go say from the big to the final distilled small model, you're still getting the same kind of performance evals can be so difficult. Do you have thoughts on that or does Fireworks have any tooling that helps out?

Lin Qiao: 39:34 Yeah, so for evals, we do not offer eval product. We instead partner with the companies whose specialty is doing evals. So here again, we really respect specialty and the customization and the focus. So you're right, eval is where people get started. But it's interesting if you talk to the startups or people building AI native agents, there are so many different way to do evals. First and foremost, people do vibe evolving, you build an application. The first thing, it's very different from software and development practices. Usually when we write piece of software, we start from unit test first and then we build different kind of guard rail to make sure the quality, we have guarantees of quality, find various different unit tests to integration tests, to then testing production. You collect some metrics and so on. But a gene development people usually start with some hypothesis and just look at result and do vibe testing.

40:56 It's very interesting because they do not want to bound their creativity in the imagination by forming small test to big tests. They test big ideas, but those kinds of vibe testing is really hard to harden and the drive further model customization because it's very subjective and then the question is how you turn vibe testing into something more concrete that a model can understand or a tuning new platform can understand. So I think it is a very important area and where we have been contributing is to solve another complexity. So today there are various different two new platforms and the various different evaluation platform. As you know, evals are directly feeding into tuning to drive the tuning process. Without eval, you basically don't have a guide, hey, this direction is correct or not. But there are so many different eval systems, there are different tuning systems and we're part of the uni systems and there's no standard across them. The integration of cross product complexity is very high. So we opened sourced a project called Eval Protocol. The idea here is to standardize the format so any eval system can talk with any tuning system. So for our developers they can pick any combination as they want, but because of eval protocol, it kind of bridge the gap across and they can have optionalities on both sides. So that's our intention to help bring more commonality across these two sides because these two sides are deeply integrated.

Jon Krohn: 43:00 Cool. So as you were talking there, eval protocol from Fireworks hadn't showed up actually in our research, but I quickly found the GitHub repo for it, so I'll be sure to include that GitHub repo in the show notes. A different offering of yours that did show up in our research was something called 3D Fire Optimizer. And so this seems like it's in a way it goes over a hundred thousand possible ways of optimizing an LLM stack. Do you want to tell us about that?

Lin Qiao: 43:32 Yeah, definitely. So when it comes to customizing, it goes back to our mission of autonomous intelligence. We are customizing across three dimensions, across model quality, model speed, and the model efficiency, which is cost. So when we started our journey, people were sure, hey, they're really concerned about cost or hey, they're really concerned about speed because they're very interactive or hey, they're very concerned about quality. Wherever they start their pain points at the end they are concerned about all three dimensions. There's no time we're like, Hey, the cost is really good. Here we are like 10 times, 10 times cheaper compar to alternatives and they're like, go for it. They're always like, oh, we also need a better quality to be launch this. Oh, we need to be faster to launch this. It's always all three dimensions has to be much better. So we're like, hey, this is actually not a linear problem.

44:45 It's a complex problem. It's with the exponential search space because the three dimensions are evolved, but it's actually more than three dimensions. If we decompose this problem, then it becomes so many different building blocks to stack onto each other and each building block has five to 10 different options to pick and choose from. So that's where in combination there are more than a hundred thousand options in this search space and it becomes a search problem. Again, that's not a new problem across our system research, for example, databases. Databases has query optimizer where a data engineer or analyst write a SQL query, but the execution, so SQL Query is a semantic description of what they want to achieve, but in reality, based on whether you have an index, whether how the data's being laid out, how's data being sorted or how data is being partitioned, there are different way to retrieve that data and process data in the most efficient manner. And the Query optimizer basic convert this query at preserving the symmatic level of meaning and turn that into specific query plan customized towards the current state of database. And inquiry optimizer basically is the brain of a database engine and there has been massive innovation in the first decade of this manal, I was part of the research group building quite optimizers, but this in the space we are operating, we are optimizing and customizing across quality, speed and cost and that is much more complicated than Query Optimizer, which only optimize for efficiency.

47:05 So that's what we build is to free up our customer, those app developers from carry the burden of learning how to do this three dimensional optimization and to find the sweet spot, the best spot among all these candidates, the best suited for their requirements. So yeah, that's kind of one of our innovations as we build our platform, but it is directly fitting to our mission of autonomous intelligence.

Jon Krohn: 47:38 Very cool, Lin, love it. That's the end of my technical questions for today, but I suspect that we have a lot of listeners out there who having heard what an amazing speaker and thought leader you are with all the capital that Fireworks has raised with the impact that they're making, the challenging problems that they're solving, I bet we have a lot of listeners that would love to work for you. Do you have any open roles and what do you look for in people that you hire?

Lin Qiao: 48:03 Absolutely. The reason we raised CRC last year is to massively accelerate our growth. So we are actually hiring across the board from all the way from the GTM side. We have a lot of openings from sales marketing as well as product engineering. And in the finance business operation across the board, I really mean it because we just have so much demand we cannot manage and we love, love to work with very creative and high aptitude people who are willing to join us take a big chunk of ownership and drive a lot of impact. So yeah, if you're interested, love to please send, please contact us and send your application.

Jon Krohn: 49:03 Wonderful. Alright, so now we're just to my last two questions that I ask every guest, but I think I'm going to get an interesting answer based on our conversation before we started recording, I always ask my guests for a book recommendation and I think you have something else.

Lin Qiao: 49:19 Yeah, I think this is an interesting time of AI and everything has changed and how we learn has changed. In the past I've been reading books from mostly around, I love to read books around business and leadership and I really like to read books about certain individuals I'm very curious about, but nowadays I listen to a lot of podcasts. I think Jon, your podcast will be very impactful as well. Particularly fundamental reason is the following. So this AI innovation is changing how we do business, how we create technology, all these agents that is emerging is going to change our day to day, but at the same time, fundamentally it's also changing how we learn, how learn what's happening in a fast moving world. Particularly I think we have a lot of people, including my funding team, including my top tier engineers, they learn a lot from a lot of posts.

50:47 We were always on the cutting edge. We read a lot of paper and the paper reading velocity is very high and paper generating velocity is very high. But we read a lot of best practices from social media and there are a lot of creative ideas, a different way of thinking, approaching problems and you have to think differently to be very innovative in the AI time because a lot of fundamental assumptions has been disrupted. For example, we're rethinking our interview process because now coding agent are very good at general code almost as a fresh graduate, junior engineer. So then the question is, hey, is writing code quickly and correctly a important area to test or not? So we're questioning that because with a system of coding agent that's no longer a problem and it's more a problem to have the eyes and the mental framework to judge how good is the code and have a strong way to steer a coding agent to design very well coming from the source of design and system architect thinking to drive the implementation. So it start to shift the focus and bottleneck of the software development to just give you an example of recruiting in the hiring process that has changed. So I will say the whole community is a book, the whole community is writing a book of ai. I think that's fascinating to me and I just feel like I'm very lucky to live in this time of this world to fully embrace fast velocity of changes and there's so much to learn from each other.

Jon Krohn: 52:52 Cool. What an answer. Yeah, certainly we've never had an answer like that before on the show, but I love it. You're a really progressive thinker and I have personally learned tons from this episode. I'm sure a lot of our listeners have as well. Lin, how can people follow you? Where on social media or how can people be getting your thoughts after this episode?

Lin Qiao: 53:18 Yeah, I'm learning also to transition my interaction more towards social media. So you can follow me on L-I-N-Q-I-A-O, my handler on Twitter x ai x.com. But also you can follow my LinkedIn. I share my thoughts, I will do more in the future, but also I talk about our product and the product launches and direction we behind towards. I love engagement. So if you have for all the audience here, you have thoughts, feedbacks, ideas, I would love to talk with you to exchange notes and go find there.

Jon Krohn: 54:05 Awesome. Thank you so much for opening up your inbox to our listeners, Lin, really appreciate it. And yeah, that's the end of the episode. Thank you so much for joining us. I can only imagine how crazy your schedule is and so to take this time out and speak with me and share your thoughts with our audience, we really appreciate it. Thank you Lin.

Lin Qiao: 54:26 Glad to be here. Thanks Jon.

Jon Krohn: 54:31 Super episode today with the exceptional engineer and entrepreneur Lin Qiao. In it, she covered how over 90% of the world's data live in private enterprise systems and never make it into foundation models, representing a massive untapped source of intelligence, how autonomous intelligence is about continuously and automatically customizing models with private enterprise data resulting in millions of specialized models rather than one a GI. To rule them all, she talked about her small, big, small approach, which means starting with a small model to iterate on data quality, moving to the largest model for best quality tuning, then distilling back down to a small model for fast real-time inference. And she talked about how coding agents now perform at the level of junior engineers fundamentally changing what matters in technical interviews from writing code quickly to having the judgment to steer and evaluate AI generated code. As always, you can get all the show notes including the transcript for this episode, the video recording, any materials mentioned on the show, the URLs for lens social media profiles, as well as my own at superdatascience.com/971.

55:36 Thanks to everyone on the SuperDataScience podcast team, our podcast manager, Sonja Brajovic, media editor, Mario Pombo, our partnerships team Natalie Ziajski, our researcher, Serg Masís writer, Dr. Zara Karschay, and our founder Kirill Eremenko. Thanks to all of them for producing another fantastic episode for us today for enabling that super team to create this free super data science podcast for you. We are deeply grateful to our sponsors. You can support the show by checking out our sponsors links, which are in the show notes, and if you'd ever like to sponsor an episode, you can get the details on how to do that by making your way to Jon crone.com/podcast. Otherwise, help us out by sharing this episode with anyone that would like to listen to it, review the show on your favorite podcasting app or on

YouTube, but most importantly, just keep on tuning in. I'm so grateful to have you listening and I hope I can continue to make episodes you love for years and years to come. Till next time, keep on rocking it out there and I'm looking forward to enjoying another round of the SuperDataScience Podcast with you very soon.