SDS PODCAST

EPISODE 818:

IN CASE YOU MISSED IT IN AUGUST 2024

Jon:	00:05	This is episode number 818, our "In Case You Missed It in August" episode.
	00:19	Welcome back to the Super Data Science Podcast. I'm your host, Jon Krohn. This is an "In case you missed it" episode that highlights the best parts of conversations we had on the show over the past month.
	00:30	In episode 809, I got to speak with Shingai Manjengwa. As head of AI education at ChainML, Shingai has had tons of experience in explaining data science and AI concepts. I wanted to know about AI agents. Specifically, how multiple individual AI agents can come together to tackle complex tasks.
	00:48	Tell us about AI agents, and then maybe get into what it means to have decentralized execution and utilization layer for these agents.
Shingai:	00:58	Just for everybody who's listening, an AI agent, let's think of it as software that leverages AI. When we say AI, for the moment, the agents that are the hottest right now are the ones that are using large language models, so think of a chatbot. If you build software around a chatbot so that it behaves in an autonomous way, it's able to do some planning and then execute on a task, let's call that an agent. There are multiple variations. We have computer vision, and if you wrap your software around computer vision, you could call that an agent. If you put that agent in hardware, we could start talking about robotics. The simplest way to think about an agent is really software that we wrap around artificial intelligence, and specifically these language models. The simplest implementation of an agent is where you enter a prompt, and that prompt executes on something.
	01:53	I would like a quote every day that inspires me to start my day, so I write that prompt into my large language model,

and then I scaffold that with some other instructions, software instructions for how I get that message delivered to me, whether that's in an app or maybe I just open up one of the products, the GPTs, that's what we've been calling them. That's how you might start interacting with an agent. Now, the reason we're talking about agents now, and why they're hot, is because agents specialize. If you're working with an agent that does something very specific, that agent can get very, very good at doing something specific. My favorite example is something like airline travel. You would have an agent that helps you plan an itinerary. Now, that's a task that does require specialization, because if you're going for a bachelorette party, or if you're traveling with your elderly parents, or with your kids, all of those are quite different itinerary. You want to have some sort of specialization on, what are the things that will make that a successful trip? That's one agent.

03:00 Now, you also now want to make some bookings. You need to book some accommodation, and you need to book some travel, like flights. Well, those platforms right now use dynamic algorithms. Wouldn't it be amazing if you had an agent that could support you in that process, that would be good at interacting with those different websites, getting all the right information, and then learning the timing for when to do the right booking, so it maximizes whatever your objectives are, let's say, to get you the lowest price? That's a specialized agent. And then finally, you might have a specialized agent that's really good at managing your wallet. You might have different credit cards, different types of accounts, even different types of currencies. I'll throw out digital currencies because that's an option.

03:45 You might have an agent that specializes in dealing in your wallet, dealing with your wallet. Now, the art of booking a trip would involve you typically, in a SaaS

model, going from one task with one bit of software to another task with another bit of software. The promise of AI is that we can knit together these different activities so that this happens in a seamless way. These three specialist agents, in our framework we call them a collective, this collective would be your travel collective, and it would be able to knit together those different tasks, and those different specialized agents in a way that makes a really seamless experience for the user that still, as the user, you would interact with in a conversational way.

Jon: 04:30 Very cool. Agents act autonomously, and that's what makes them a little bit different from the continuous conversation that we have with an LLM. Often, LLMs are used, as you said, to power some of the capabilities of an AI agent, and that's what makes it so easy to now suddenly interface with them. I think that's why they're so hot right now, because we can use an LLM as a natural language interface, and then, the agent can use information from the LLM to go out on, say the web, and come up with an itinerary for you. They could use its LLM weights maybe to come up with the itinerary, and then subsequently, like you said, the word scaffolding there, I like that. I'm guessing that's the kind of thing that ChainML helps with, is providing things like that scaffolding to allow you to have this agent be working autonomously, not just while you're in that conversation with it.

05:28 If you ask the agent to, every day, provide you with some inspiring quote, then that's what makes it different. If you go into ChatGPT, or Claude from Anthropic, and you go in manually every day, and you ask for an inspirational quote, that isn't an agentic AI situation, because it's just reacting to you. It's just having that conversation with you at that instant. If you can say it in a natural language to an agent, "I want to have an inspiring quote every day,"

and then it is pushing to you, say via email, or SMS, or whatever you prefer, pushing that to you automatically, then it is an agent.

Shingai: 06:15 Maybe one layer before that level of automation, if you think about the GPT store, where it's a function. You pull down an agent to perform a function, so it's not yet automated. It's not being automatically delivered to you via an app. At that point, those are agents, too, but you have to do the work of going to get it from the store and then incorporating it into your workflow. Anthropic also now has a store, and I believe they call them projects at this point, but we're all leaning towards this idea of having a marketplace where you have multiple itinerary agents. Anybody can do that now, especially with the no-code solutions. Anyone can go and build an itinerary agent, and if you're particularly good at building one for a family with small kids, you would put up your agent in the store.

07:08 Now, where Theoriq, the product that ChainML Labs has built, where Theoriq really shines is, how would you make a decision about which agent to use? You've got access to this GPT store, or the projects, or Hugging Face, or a platform that has multiple different agents. How do you make a decision about which is the right agent for me to use in this collective? What we've built is a protocol that has some mechanisms for that to happen in an assisted way. It's an automated way, but let me for now call it an assisted way.

Jon: 07:43 Shingai's example of using multiple AI agents to organize travel made understanding agentic AI concrete and crystal clear. We continue on our journey into understanding decentralization with a clip from episode 807. In it, I ask the renowned futurist and entrepreneur, Dr. Daniel Hulme, how AI can help us become better and faster at our jobs by circumventing the traditional

corporate hierarchies that today seem only to slow us down.

08:09 In doing that, we also got into talking about alignment. There's a lot of discussion around alignment in AI, aligning AI to common human values and goals. Like you said, there's different ways of thinking about this. Is it just maximizing curiosity, like Elon Musk suggested some time ago? What are the kinds of things that we need to do to get an AI aligned with our goals? In the way that our society is set up today, we tend to reward short-term profits over things like long-term sustainability. You have mentioned elsewhere the potential for an ultra capitalistic system that avoids the pitfalls of the traditional capitalism that you're in now. Can you explain to our audience how decentralization and tech can create a more efficient and equitable allocation of resources?

Daniel: 09:02 There's two things here I want to talk about. Maybe we can come back to the alignment problem in a bit, because I want to share an idea about a way that we might be able to capture human morality. I'll come back to it in a bit. I was, for a long time at Satalia, very interested in decentralization. I've worked in large organizations, I've worked in small companies, I've had my own startup, and I think a lot of companies start in the same way. They want to be decentralized, they want to get processes and bureaucracy and hierarchies out of people's way so they can go and do the things that they need to do. Of course, as you get bigger as an organization, you end up creating hierarchies, putting in these processes and structures that slow organizations down. They prevent them from being innovative.

09:53 For the past 15 years, I've been trying to figure out, how could we use AI to create decentralized organizations? What I mean by that is, organizations that, they're able to identify the best, diverse group of experts to be able to

make the decision. Whether it's feedback or hiring or firing or pay, rather than those decisions being made in the hierarchy, can you identify those people? It's often called a liquid organization, or liquid democracies. I think that now, AI can, by the way. I think that we are able to now ingest all of the digital footprint that exists across an organization, and an AI could make sense to it, and they could say, "Look, you've worked very closely with that person. You're very knowledgeable about their domain. You understand about the company's strategy, so therefore, you should have more rights in their salary than somebody else."

10:44 I'm very interested in how to do that, and the reason why I'm interested in how to do that is because, let's assume that over the next decade, AI will free people up from tasks, and maybe even whole jobs. And by the way, that's a good thing, because it means that we can reduce the cost of goods. But if it happens very quickly, as I've mentioned before, it could create some sort of social mess. How can we use AI to identify what granular pieces of work people could be doing, and recommend and enable those people to do that work? Rather than being a job role, can you understand people's skills, their plethora of different ways that they can contribute, and then decentralize that across different tasks that need to be done across an organization? One of my dreams was to try and figure out, how could you scale that to a planet?

11:34 How could you create a planet where we are free to be able to contribute in different ways, facilitated by AI? Something I still think deeply about, and we're still working on in Satalia. What's interesting, what's happened over the past several years is the birth of open source. I think this is a much more interesting and semi-related concept, which is, the reason why Meta open sourced their large language models, it's not necessarily for the good of their hearts, it's because it doesn't hurt

their business model. If Google or Microsoft open sourced their models, then it potentially has a revenue impact. Meta can do it because their business model is not dependent upon it. What happens with Meta is that they're able to now capture data, more data, which is, as we know, makes AI smart, data is valuable, but also, they can access talent contributing to those open source technologies.

12:37 What I'm seeing is this impulse, more and more, to actually open source innovations rather than just sitting on them until somebody else comes along and out innovates you. You open source it, you make it available to your competitors, et cetera, because what's valuable is the data and the talent. Ironically, it's that open sourcing, which is a very sage business decision, it's that open sourcing also creates a world of abundance. Large language models, which are perhaps the most intelligent technology we've ever built, are pretty much free now to everybody, or they will be free to everybody soon. That power is in the hands of everybody. There's a possibility where, I don't know how many phones you go through every few years, but I go through a couple of phones every few years. Now, if we gave those phones to people that don't have computers, if those phones were connected to the internet, then not only is education then free, because pretty much all education is free on the internet now anyway, but those people then have access to these technologies to get them to go and create new innovations.

13:49 This idea of open sourcing and decentralization, I think is a very powerful idea, and what we're seeing is, actually, the capital model, the capitalistic model forcing the open source movement.

Jon: 14:00 Getting granular and doing away with job titles altogether might sound radical, but it seems to be where the future

of work could indeed be headed. In the meantime, how can we keep ourselves challenged in an increasingly automated work environment? For an answer, let's head to my next clip from episode 813, in which I speak with mathematical optimization guru Jerry Yurchisin about the future of continuing education.

14:23 If we have listeners out there who write Python code, or write R code, and they want to be getting started on using Gurobi or mathematical optimization on some real world business problem that they have today, how hard is it for them to set up the problem? We talked about how the hardest part of this is having the optimizer work efficiently. Gurobi handles that for us automatically under the covers. The thing that is bespoke and different for every circumstance, for every business problem, is figuring it out how to set that up in our code. How tricky is that? How often can somebody do that on their own versus needing to engage with a consultant that is expert at this kind of stuff?

Jerry: 15:10 To model the hardest of hard problems out there, it does take some experience, with anything. It takes some understanding of how these particular sets of constraints work, and again, taking the logic that someone says or writes down and translating that into the algebraic logic, and then that into code can be tricky. There are some things that specifically Gurobi, and other solvers and other platforms and things, be they competitors or something, there are shortcuts out there, where you don't have to know all of this stuff, and you don't have to understand all of it down to its most minute detail. It is very easy to get started, because our simple problems are extremely simple. But then building on top of that, it's going to take some understanding, it's going to take a little bit of work, going to take some research, and a lot of Stack Overflow and things like that, to really get to a production level type of model, I'd say, at a large scale.

16:31 The journey there, and this is one of the hangups of mathematical optimization in the past, is that you needed to have a PhD in order to make this journey from basic problem to actually doing something at scale for a business, and making an impact. Now, that's not the case. You can just be really good at coding and understanding logic, and you can have an impact, and you can solve problems, and you can provide solutions that are really doing something. That's part of why I joined Gurobi, is to help get those resources out there. I'm just going to talk about what we put out there a little bit, but there's a ton. I think at the end of the last episode, I referenced an optimization book that would be really, really good, again, to dive into. From our perspective, from things that we released, I did two online training sessions that we called Optimization for Data Scientists, Opti 101, Opti 201.

17:41 It was the bare basics of optimization, and then, a more intermediate level. The Opti 101 series can find on our YouTube page, and the Opti 201 is going to be on our YouTube page probably in the next month or so. There's going to be an Opti, I think 202 is how we're phrasing it, or thinking about it internally, which is more intermediate level stuff. All of it has hands-on exercises, hands-on notebooks, me looking at a camera just like this, and talking to people for hours and hours, making mistakes like everyone does. It's just a lot of fun. It's a great way to take a day or so to really improve some skills.

18:33 And then, we also have recently launched, I think in April, something that's a lot more massive. On Udemy, we have optimization through the lens of data science. It's a four part course. We teamed up with one of the best optimization minds out there, Dr. Joel Sokol from Georgia Tech, and he walks you through everything that you need to know about mathematical optimization, from the

absolute bare beginnings to creating real models that, again, will have real impact, and just makes that journey step by step by step, very incremental, nothing too crazy, all in Python, and then weaves in his experiences, and everything with his consultancy stuff that he's worked on the side, and stuff that he's done. It is a wonderful way to set yourself on the journey, and again, it's through the lens of data science, so it's, again, saying how these two things really, really work well together, how they are super complimentary. Between those two things, I think you're set, but again, you can-

Jon: 19:57 Between the Opti 101 course that's available on YouTube now, as well as the Udemy course that's through the data science lens? We'll be sure to include links to both of those.

Jerry: 20:05 Awesome. I think we were saying, how can one person make this journey? Again, I was talking about incremental, building upon, "I know this, now I know a little bit more, and a little bit more, and a little bit more. Now, I can actually get to something that makes a lot of sense." That type of progress isn't just for learning optimization as a whole, but it's how you build a model. You start with a very basic premise, a very basic problem that someone talks to you about, and then you build a model, and then you'll get a solution that makes no sense when you talk about it. That's 100% expected and fine. It'll say, "Put all of your," maybe we can talk about, "Put one burrito truck here, and it'll serve everybody, and you'll make infinite profit." You'll be like, "Whoa, whoa, whoa. That makes no sense. Oh, I forgot this type of constraint, or I modeled something slightly wrong."

21:09 Just building a model within itself is iterative, not just learning how to do optimization is iterative. You're going to make mistakes, you're going to get weird answers. Mathematical optimization, it's the best cheater of all

time. If you give it the smallest little opening to have infinite profit, it will find it.

Jon: 21:30 Reward hacking, to take the reinforcement learning terminology.

Jerry: 21:33 Yeah, precisely. It will do that each and every time if it's possible. If you find yourself stumbling a little bit, like, "This doesn't make sense, or this doesn't make sense. Why am I getting weird solutions, or no solutions?" That's perfectly normal. The best of the best of us still do that, and it is just part of the learning process.

Jon: 22:00 Jerry is right to emphasize our need to stay aware of weaknesses in models. In my final clip, taken from episode 811, you'll hear my guest, Nick Elprin, address the need for companies to define their AI infrastructure. The stronger the infrastructure, the more on board your team is, and the better you'll identify weaknesses all across the board, from the granular to the structural.

22:21 Let's dig right into what you're doing at Domino Data Labs, where you are co-founder and CEO. The Domino Data Lab promotes a unified and open platform for AI. It emphasizes end to end models and data science life cycle management, as well as MLOps automation. That's a lot of different things going on. Maybe you could summarize for us, I'm sure as the CEO, this is your bread and butter, to tell us what the pain points are for your users, and how the Domino Data Lab platform addresses them.

Nick: 22:53 I started building Domino about 10 years ago, and over the last decade, we've built what I believe is the most comprehensive and the best platform that gives large organizations, particularly enterprises, everything they need to do mission-critical AI at scale. I think of that as really having three elements, or three facets of what a large organizations need to do data science, ML, AI at

scale. The first is, we make it really easy for data scientists to orchestrate, get access to, weave together all the different infrastructure that you need when you're doing AI work. You need a lot more compute, you need access to data, you need agility to use all the new, latest and greatest software tools, packages. The state of the ecosystem is changing so fast that every week, there's a new thing you might want to use.

23:57 In enterprises, making it easy to get access to all those infrastructure components, all those resources can be very frictionful, because IT is putting up bottlenecks and gates and things like that. At the same time, if you're a data scientist, you're not a DevOps expert, you're not an infrastructure engineering expert, so you may not want to be setting up your Spark cluster, and debugging parallel jobs and things like that. The first thing we do, we provide self-serve access to all the infrastructure you need to be really productive, and experiment rapidly, and stay on the cutting edge. By the way, we do that in a way that also meets the needs of security conscious IT organizations. Imagine providing infrastructure access with security controls in place and templates, because again, in an enterprise context, you don't want to just give all the data scientists free-reign to build a wild west, or whatever they want.

24:58 That's the infrastructure layer of our stack, and we're, as far as I know, the only advanced data science platform that has native capabilities for multi and hybrid cloud orchestration. As data scientists, whether you're running an interactive development experience like Jupyter, or whether you're running a batch training job, or a real-time inference, you're deploying a model for real-time inference, we can push that to any cloud or on-premise. On top of that, next layer of the stack once you've got all the infrastructure stuff you need, we've built, like you said, an integrated experience that facilitates the model

development lifecycle. I think about this as like what the Microsoft Office Suite is for workspace productivity, or what the Adobe Creative Suite is for designers. It's all the tools you need to go through your workflow, your productivity apps, so interactive development, experiment management, model deployment, model monitoring, and then critically, making that a tight, closed loop.

26:09 If you go to deploy a model, we'll automatically set up the monitoring rules for you. If we detect that drift has occurred, we will automatically create a development environment for you, with all your raw materials, your code, your data, your software package definitions, precisely as they were when you first deployed your model. We really believe that data science, it's not a straight line, it's a loop, and the most effective organizations are the ones that speed up this iterative development lifecycle. It's not just deploy a model, it's deploy it, monitor it, and then continuously improve. We think a lot about, how do we streamline the workflow for data scientists as they go through that whole iterative, continuously improving model development lifecycle?

27:00 That's layer two, and then layer three, which, it's one that I think is most exciting, and probably most valuable, and it's one that unfortunately, doesn't get a lot of attention, because it's not really sexy, it is creating the system of record for data science work and artifacts. The models that you deploy, the history of experiments that you've done, the projects you've tried, keeping that all in one place so there's a single source of truth, and so that organizations can find and build on past work instead of reinventing the wheel, and that organizations can have standards and consistent ways of working.

27:50 Let me just unpack that for a little bit. I was talking to an SVP of data science at a big media company a couple of weeks ago, and he said, his is the only team in the whole

company that can onboard a new team member in 24 hours, and that has been critical to them accelerating their productivity, increasing their productivity. He said, over the last couple of years, they've seen a 6X improvement in the throughput of the productivity of his data science organization, and he attributed that largely to the ability to reuse past work and stop reinventing the wheel all the time. A new person joins the team, they're ready to go with, "Hey, here's how we do a project. Here's the life cycle. Here's how you access your data. We give you access to the platform. Everything you need is at your fingertips, ready to go. You don't have to go hunt around, and ask people, and find wikis, and have people send you connection strings or whatever."

28:57 I know it was a really long answer, but I thought that'd be good to set the stage a little bit. The way we see the world is, what enterprises need to scale mission-critical AI, self-serve access to governed flexible infrastructure, great productivity suite to streamline the life cycle, working through the model development life cycle for data scientists, and then, the system of record that helps organizations build on past work instead of continuously reinventing the wheel.

Jon: 29:25 All right, that's it for today's, "In Case You Missed It" episode. To be sure not to miss any of our exciting upcoming episodes, be sure to 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. I'm looking forward to enjoying another round of the Super Data Science Podcast with you very soon.