SDS PODCAST EPISODE 789: ML FOR WIND-POWERED ENERGY GENERATION, WITH DR. JASON YOSINSKI

Show Notes: http://www.superdatascience.com/789

Jon Krohn: 00:00:00 This is episode number 789 with Dr. Jason Yosinski, co-founder and CEO of Windscape AI. Today's episode is brought to you by Crawlbase, the ultimate data crawling platform.

00:00:15 Welcome to the Super Data Science podcast, the most listened to podcast in the data science industry. Each week, we bring you inspiring people and ideas to help you build a successful career in data science. I'm your host, Jon Krohn. Thanks for joining me today. And now, let's make the complex simple.

00:00:46 Welcome back to the Super Data Science podcast. I'm delighted to have one of my all-time favorite AI researchers, Dr. Jason Yosinski as my guest on the show today. Jason is co-founder and CEO of Windscape AI, a startup using Machine Learning to increase the efficiency of energy generation via wind turbines. He's also co-founder and president of the ML Collective, a research group that's open to ML researchers anywhere. He was a co-founder of the AI Lab at the rideshare company Uber. He holds a PhD in computer science from Cornell during which he worked at crazy places like the NASA Jet Propulsion Lab, Google DeepMind, and with the eminent Yoshua Bengio in Montreal. His work has been featured in the Economist on the BBC and, coolest of all, in an XKCD comic.

00:01:30 Today's episode gets fairly technical in parts, so maybe of greatest interest to hands-on practitioners like data scientists and ML engineers. Although there are also parts that will appeal to anyone keen to hear how ML is being used to produce more clean energy. In today's episode, Jason details how ML can make wind direction more predictable, thereby making wind turbines and power grids in general more efficient. How to infer what individual neurons in a deep learning model are doing by using visualizations. Why freezing a particular layer of a

neural net prior to doing any training at all can lead to better results. How you can get involved in a cutting-edge research community no matter where you are in the world and what traits make for successful AI entrepreneurs. Are you ready for this mind-blowing episode? Let's go.

00:02:17 Jason Yosinski, welcome to the Super Data Science podcast. I'm blown away to have you here because I've been tracking you for almost a decade when your deep visualization toolbox came out. We're going to be talking about this later on in the episode, because Deep Vis toolbox allowed for amazing intuitive understanding of the way that deep learning networks, particularly convolutional neural networks, which at the time of me discovering you were near the state of the art of what we could do with AI at all. And so I have been using your Deep Vis YouTube video since about 2016 to teach students an intuitive appreciation of what's going on inside of neural networks. And we will for sure be linking that in the show notes and so people can check that out. Anyway, thank you for being on the show and glad [inaudible 00:03:11]. Yeah. Where in the world are you calling in from?

Jason Yosinski: 00:03:13 I'm calling in from San Francisco, Lower Haight.

Jon Krohn: 00:03:17 Nice classic choice for our AI entrepreneurs.

Jason Yosinski: 00:03:21 Yes, I'm sure. Yeah.

Jon Krohn: 00:03:22 Nice. Jason, before we get into the technical stuff in this episode, can you tell me why was six afraid of seven?

Jason Yosinski: 00:03:31 Because 7, 8, 9?

Jon Krohn: 00:03:32 Yeah, that's right. You're the guest in episode... You're the guest in episode number 789 and what a treat. We only

get to do that once. You'd have to have a whole other podcast and get to this episode number in order to be able to use that incredible joke. So beyond that, being able to guess the answer to my 7, 8, 9 joke, you're also co-founder and CEO at Windscape AI where you're using machine learning to help wind farms produce more energy. Tell us about that.

Jason Yosinski: 00:04:04 Yeah, so we are trying to make wind turbines more efficient. We're trying to make them generate more power and generate the power at lower cost. This will do two things. It will help wind energy be rolled out more quickly. It'll accelerate our transition to net-zero, so to a world in which we power our planet without using carbon. It'll make our customers, people that own the wind turbines more money and because the turbines will be more efficient and generating more energy for lower costs, it'll also make the energy cheaper for you and me, for people that use the energy. We do this with machine learning. We do this by looking at data from turbines as well as weather, and I can get into that as deeply or as shallowly as you'd like.

Jon Krohn: 00:04:51 Yeah, let's go. Let's dig into it. How does it work?

Jason Yosinski: 00:04:54 Okay, sure. So first we take data from turbines. So all turbines have on board a number of sensors. In particular, you can imagine a sensor on the back of the turbine that measures the wind speed and the wind direction. One thing not everyone knows is that the way turbines operate these days, all turbines track the wind. Some people think turbines are installed facing north, and if the wind is from the north, that's great, and if it shifts to the west, you're just out of luck. I'm happy to report that for many years now, turbines have all tracked the wind. So as turbines are tracking the wind, basically let's imagine the wind is coming from the north, the turbines facing north, all is going well, and then the wind

slowly starts to shift to the west or to the east or something. The sensor on the back of the turbine will pick that up and you'll see that the wind is shifting west and there's some control parameters that will involve some delays or some dead zones maybe, but eventually the turbine will start turning to the west and follow that changing wind.

00:05:57 Those sensors in the back are great in some ways. They provide obviously pretty good visibility into the wind at the correct height, so they're right up in the middle of the big circle inscribed by the blades, but they're also noisy in certain ways. The noise they receive is biased in certain ways. They're in the back of the nacelle, which is behind the blades. So every time the blades pass by, they generate vortices that come off the trailing edge and hit that sensor, and Jon is looking like this is one level two deep.

Jon Krohn: 00:06:29 No, it's not-

Jason Yosinski: 00:06:29 They generate noise, but the noise is shifted. It's not zero centered.

Jon Krohn: 00:06:34 So the wind turbine itself, as it adapts to where the wind is coming from, it makes the data worse that it's trying to use to detect wind direction?

Jason Yosinski: 00:06:46 Yeah, I would say it's actually not even a consequence of the turbine turning, although as it turns, the distribution of noise will probably shift. Even when it's not turning though there is still a lot of noise and that noise is not zero centered. So we're trying to help turbines deal with this problem, also deal with problems of these sensors not being installed perfectly correctly and, or as they wear over the years, as they in some cases become miscalibrated over the years, we're trying to learn that calibration, learn that distribution and kind of reset

conditions to be straight, so that the turbines are pointed the correct direction. There's some interesting studies showing that the median turbine right now, the median turbine around the world is mispointed by six degrees. There's other studies showing that the average turbine loses out on a couple percent of production because it's not facing the wind, it's not following the wind correctly.

00:07:41 We're hoping to fix those problems using our neural network models. So that's kind of one part of the company. We also are taking this data from the farms and feeding it into modern weather models. So a fun fact about wind energy is both wind energy and solar energy are variable, right? So you have an electrical grid, you have a bunch of producers, producing energy feeding into the grid, flowing all around transmission networks, distribution networks, and then being sucked out by consumers using that energy. So if your light is on right now, those electrons were generated somewhere probably within 50 or 100 kilometers of you, and it might've come from a wind turbine or a solar panel. Let's imagine you have solar panels and your light is on, and then a cloud just passes right over the solar panels, right? What happens? That production just drops in some cases nearly to zero. Or if you're a wind turbine, let's say it's windy, you're generating and then the wind dies.

00:08:44 Both of these factors, they're not factors that are controlled by humans, it's just something that happens to the grid and the grid needs to be robust to that and needs to react. So a second part of what our company is doing is trying to make wind itself predictable. The way we model weather as a... The way we model weather in general is sort of changing right now. So since the 1950s, we've been modeling weather by running physical simulations on supercomputers. You can imagine a bunch of little voxels.

	00:09:17	Each voxel has what's the air doing on the left side, the right side, the top, bottom, back and front, what temperature is it, maybe what relative humidity and some other factors. And then you just click play on that physical simulation and you simulate all the voxels one time step at a time. This has worked fairly well for seven decades. About a year ago, a few groups around the world showed that in fact you can train neural networks to do the same thing. If you just have a bunch of recorded historical data, you just throw it all in, train in let's say an autoregressive way, and it works. Cool, right?
Jon Krohn:	00:09:53	It's nice.
Jason Yosinski:	00:09:54	Also, similar story to chatbots, train autoregressive models on text, make the big enough, make the dataset big enough, train for long enough, big enough computers and you get kind of magic out. We're just seeing that now with models for weather.
Jon Krohn:	00:10:09	Nice, nice. Yeah, and autoregressive there meaning predicting the next token in a series of tokens in the case of a large language model. So when you're speaking to a generative AI model that is putting natural language or code, it's outputting the next tokens. It doesn't have a future information, future tokens that it's outputting. And so what you're describing is similar in the sense that you have all of the data on say, wind up to a point in time and you're trying to extrapolate forward auto regression.
Jason Yosinski:	00:10:39	Yeah. Yeah. Some people would call this an unsupervised approach or a self supervised approach. Just requires a big dataset of what's happened in the past or in the case of chatbots, what was written on the internet in the past, and you can train those models then.

Jon Krohn: 00:10:52 Nice. Yeah, no explicit labels. Very cool. And that's great because then it means you have a lot of data to work with typically.

Jason Yosinski: 00:10:59 Exactly, exactly, and it's turned out, it's one of the lessons of the last few years is right, training models on these really huge datasets is what enables them to perform really well.

Jon Krohn: 00:11:10 Nice. Certainly exciting things that you're doing there on both fronts. So both with helping turbines attract wind direction more effectively, and so getting more bang for each buck out of each turbine, and then secondarily using ML to make wind direction more predictable, writ large across regions. And so I guess that allows... And you might've already said this and I'm just kind of being foggy, but how does that information get used by say, an energy provider. If they know wind direction... And you also gave the example of solar. Does this relate to the way that different renewable energy sources interplay within a grid?

Jason Yosinski: 00:11:57 Yeah, very much so. Again, we could jump in here for 35 minutes or we could talk for one minute. The problem with these renewables that I mentioned is that they're intermittent. So that they're coming online and going offline, not in a way that's under human control, just driven by nature. The grid needs to be able to react to this. We can do this in a couple ways. So one simple approach we could imagine is we take a bunch of batteries, we put them on the grid, what do the batteries do? Well, let's say they charge up when it's sunny and windy and their charge goes up to eventually 100%, and they hang out there at 100% for a while, and then as soon as the sun dies, as soon as the cloud comes over or the wind dies, they start discharging into the grid.

00:12:41 So this is one approach we could take. In order to use those batteries most optimally, you really want to know when is it going to be windy, when is it going to be sunny? You don't want to just be reactive. You want to anticipate periods of sun or wind so that you'll be ready to charge. The way the grid actually works is there's a real-time pricing signal that tracks how much energy is available and at what cost. If it's very sunny, if it's very windy, the energy is very, very cheap, sometimes literally free. So what you want to do if you own batteries is you want to charge. Then later when the wind dies, the price goes up and you want to discharge then and then you're paid that difference in price. This is already the way the grid works, but to make it more efficient, to make it more cheap for people and to make it more resilient, we need to make all these natural factors more predictable. And that's where the kind of weather modeling comes in.

Jon Krohn: 00:13:39 I imagine that in a lot of grids around the world today, we might not have the battery or renewable capacity always available. I suspect that that's the case actually on the majority of grids. Obviously we hope to fix that as soon as possible, but today there's probably also an advantage with this kind of planning that you're describing to allow us to know, Hey, we're going to have to turn a traditional kind of coal or oil-fired plant on in anticipation of a prolonged stretch of both darkness and no wind, say.

Jason Yosinski: 00:14:11 Yeah, absolutely, absolutely. So you mentioned a problem of if there are long stretches of no sun or wind, then we need to turn on dirtier fuel sources like gas and coal, and that's absolutely true. What's less immediately obvious is that by making things more predictable, just by knowing that period is coming, we can do certain things to prepare for it. But if you get into how energy actually moves around on the grid and is bought and sold and traded, having greater predictability does enable you to do some of those things.

00:14:41 And this is really important to work on right now. Certain grids, if you start out at 0% solar and wind and you slowly add a little bit of solar and wind, 1 or 2%, the whole grid kind of still works. But as you push that percentage higher and higher and a higher fraction ends up being intermittent, it can start to cause problems so much so that without all these batteries which are not yet deployed everywhere, like you said, you hit a limit and you need to keep using coal and gas and these dirty sources because the grid can't take any more wind and solar. So we're all around the world in the process of switching one step at a time, 1% at a time to wind solar. We are trying to kind of accelerate that process however we can.

Jon Krohn: 00:15:25 Today's podcast episode is brought to you by Crawlbase, the ultimate data crawling and scraping platform tailored for data scientists, AI developers, and Python developers. For ML and AI, high-quality data are of course essential. With Crawlbase, you get a powerful, user-friendly solution that guarantees seamless integration, lightning-fast performance, and unparalleled reliability.Crawlbase supports your needs with a 2-minute integration process, AI-powered efficiency, and 99.99% uptime. Crawlbase also excels in bypassing CAPTCHAs, avoiding IP blocks, and handling proxy failures, making them the go-to solution for all your data needs.Use the special code "SUPERDATASCIENCE", with no spaces, to unlock 10,000 free requests. Visit Crawlbase today and supercharge your data collection process with the best in the business!

00:16:15 It's interesting, this is digressing from possibly your expertise and it's certainly from the line of questioning that I had planned. But when you described the mix of energy sources that you were interested in using, you didn't mention fusion. Which I suspect was deliberately left out.

Jason Yosinski: 00:16:36 Sorry, I meant to mention nuclear fission and that is an important source in the US and in many countries. I didn't mention fusion. It doesn't work yet. If it does work someday, that would be amazing. That'd be great.

Jon Krohn: 00:16:50 That's true.

Jason Yosinski: 00:16:52 If we never get to fusion, we will need to power the world on those other four wind, solar, hydro, which we are using quite a lot in the US now and fission. If fusion does someday work, that will be amazing. We will build those plants as quickly as we can. It might be many years, many decades before they're all rolled out. So in the meantime, we still need quite a lot of wind and solar.

Jon Krohn: 00:17:13 Yeah, yeah, yeah, that makes a lot of sense. And as soon as I asked the question and you started answered, I started to feel silly.

Jason Yosinski: 00:17:18 No, no. Yeah. Even fission though. Even fission for whatever historical reasons, and this is the edge of my knowledge now, it's really hard to build these plants. They're very slow and very expensive. They're more expensive per megawatt hour than wind and solar, so they could be part of the right answer for the next decades, but they're not a silver bullet.

Jon Krohn: 00:17:43 Yeah. Conveniently they do fill that gap when it is dark and there's not enough wind, [inaudible 00:17:48] fusion. Yeah, yeah, yeah, the fission reactor.

Jason Yosinski: 00:17:51 Base load. Exactly.

Jon Krohn: 00:17:52 Yeah. Nice. How did you get into this space in general? You had a storied history, including things like being a co-founder of Uber AI Labs. What led you to tackling this particular problem? I can imagine that there might be things like you want to be making a big social impact, but

then how did wind in particular end up being the problem that you're tackling?

Jason Yosinski: 00:18:19 Yeah, yeah, great question. I spent about 10 years of my life, I would say 2010 to 2020 working as a scientist. I did my PhD, worked on a startup, worked at Uber AI just as a scientist. So publishing papers, patents, giving talks, going to conferences. Honestly, super fun. Around 2020, COVID happened. I looked and saw the state of AI research, ML research, and I would say things were really... This may sound silly to say, but things were really slowing down. So the number of papers per year being published that I thought were really deeply interesting was kind of shrinking. They were being disproportionately published by a few large companies with great resources. And for example, grad students with one or 10 GPUs under their desk or set their friends' desks, couldn't really compete as much. So I really saw the process of research and ML research is changing and I had a life moment.

00:19:22 What am I going to work on for the next 10 years, right? What am I going to sink my teeth into that's got a longer runway? For various reasons, I decided to try to find something a little bit more applied. I got really interested in climate change. I started reading a lot of books, podcasts, talking to people, a friend and I... I interviewed probably 150 people by the end of it asking about their different industries, what do they work on, in what ways could they see data maybe mattering? I talked to farmers. I ran a pilot with this farmer to assess his soil health from space using satellite data and tell him which grasses his cows were eating. Worked on carbon credits, kind of spent a lot of time learning about a broad space of climate related topics. Became convinced that working on energy, is kind of one of the most meaningful things we could do. It's like a lever that we can pull today, that'll matter today

and tomorrow. It's not a technology that's 20 years off into the future.

00:20:30 And then I went looking for the right entry point. So if you want to work on climate change, one maybe problem of working on climate change as a data scientist or a machine learning person is that climate change is fundamentally about atoms, about physical infrastructure and electrons power flowing back and forth, atoms and electrons. What is our world about? What is data science about machine learning? It's about software and bits, data, right? GitHub repos. These are very different worlds. They certainly overlap sometimes, but if you imagine this Venn diagram, right? It's like a pretty small overlap in the middle. You have to find the right problem. Oftentimes climate change, what we really need is you really need people to vote. We need the governments to shift money and policy and infrastructure. We need to build big concrete and steel things. We need to build transmission lines across our whole country, AC lines, DC lines, right?

00:21:26 We need to get the energy from where it's sunny here to where it's cloudy somewhere else. But these are huge projects involving billions or trillions of dollars of investment and you aren't, and I'm not going to be able to affect all that change just by some clever code and models. Okay, but all is not lost. There are still entry points, right? So there is an intersection in this Venn diagram. If you want to work in energy using data, you should find something that's maybe not trivial to predict, but hard to predict but possible, right?

00:22:02 And so I started looking more at wind and solar and realized wind was a good place to be because it's fundamentally a pretty big, beautiful, chaotic, messy, turbulent system, but there are patterns. If you use ML cleverly, you can learn these patterns and you can use

that knowledge, those model predictions to really make a difference. To make your customers, people that own wind turbines more money and to make the grid more predictable and so on as we've been talking about. So it was a long path to find a foothold where data matters for climate. And there are many other footholds that... Don't let me discourage you, but this is the one that I found that I found was meaningful and a beautiful problem.

Jon Krohn: 00:22:46 That was a great explanation. For our listeners out there who may themselves be, say, searching for that startup idea and maybe it's their first startup idea, what do you recommend? It sounds like you had a bit of a process there, different consulting projects, soil health, different social impact projects, and so it seems like you use that as a period to land on what the right problem to start a company with was. And you mentioned podcasts. I don't know if you have more insight into the kind of structure that you had over that period, or did you formally say, "I'm going to give myself this much time before I make a decision. I have this much runway, or I'll just keep working on consulting projects until something starts to click and I'm like, wow, there's a whole startup idea here." How did that period go for you?

Jason Yosinski: 00:23:39 Yeah, I would say at the outset because for me it was a big shift into a completely new domain. I am mentally prepared for it to take quite a while, so I didn't give myself an immediate deadline. I said, "I'm just going to start reading and exploring and doing whatever I want." And I imagined that eventually I would start to feel nervous and start to realize, oh, I should hurry up otherwise I'm going to be jobless forever or whatever. But I was actually... I was pleasantly surprised that after a while I was still enjoying learning and I did not feel too much pressure too soon. I forget the other part of the question.

Jon Krohn: 00:24:14 I think that was basically it, it was just kind of if you had structure around-

Jason Yosinski: 00:24:17 Oh, yeah, structure. Yeah, not much. I just read whatever was interesting. A book that I really liked was called Rewiring America by Saul Griffith. There's now a new book which I imagine has a lot of the same content called Electrify, which is very informative. It was one of the first bits of content that I read that was written really by an engineer, a scientist, an engineer, not by either a politician or a hippie, both of which have their own kind of ways of presenting the world. And I found that it resonated much more with an engineering mindset. If you just want to solve the problem of climate change, imagine you have coordination and everything, what would it take? How would we solve it? How much would it cost? And he just goes through it all very directly and it made it feel much more simple for me.

Jon Krohn: 00:25:08 Very cool. Yeah, I'm sure that kind of engineering mindset is applicable to a lot of our listeners and it seems like your approach is working. So EDP, a large Portuguese utility company recently selected Windscape as one of nine startups for its renewable innovation program in Singapore to accelerate the global energy transition. What opportunities do you see emerging from Windscape AI's participation in this program?

Jason Yosinski: 00:25:34 Yeah, well thanks for mentioning that. We did apply for this program. We were selected. EDP is a huge utility. I believe they're the fourth-largest wind owner in the world, so they own tons and tons of turbines. They generate a lot of wind energy. When I met with folks from EDP, I found them to be a very, very forward-looking organization. Sometimes you get a big company and they're impossibly slow or something. But these folks are really pushing the boundaries, all the boundaries they can, which I thought was super cool. What we hope to get out of it and where

that collaboration might go is to pilot our technology, start working with them, see how it works on their wind farms around the world, and then if it does work really well, hopefully we roll out more broadly and we can also maybe use that as a demo for new potential customers.

Jon Krohn: 00:26:26 Very cool. So it sounds like EDP is forward-looking, but in general, do you counter resistance or hurdles as you try to come to energy utilities and say, "Hey, you could be using AI like Windscape's to be improving the efficiency of your systems." Do you encounter resistance or hurdles or is it relatively straightforward to convince people that you're doing something valuable?

Jason Yosinski: 00:26:49 I wouldn't say it's straightforward, no. Convincing people that what you're doing is valuable is maybe always hard. I would say saying the words AI or machine learning doesn't immediately open all the doors. It can open some doors. Some of these companies realize that AI might be revolutionizing things that happen internally and they're not quite sure how yet, but maybe we should talk to these randos from Windscape and see what they think. It does open some doors, but not all, just as probably within any industry. There are some organizations that are very forward-looking and others early adopters of any technology and others that are slower, that are later adopters. They literally... Some have told us, "We don't care what you're [inaudible 00:27:35], just show us when four other companies are using it and then we'll consider it because how we work," right? Which is potentially an efficient choice from their perspective.

00:27:45 There's also small energy companies and large energy companies and there's a spectrum there of how you sell to these companies and how you get adoption and so on. Yeah, and convincing everyone, it can be hard. You have to convince people that your technology will work, that it won't be a huge headache to adopt. The people in the field

need to buy into it. It can't ruin their workflow or something. It has to be possible to actually integrate. So some of these systems run software that's hard to work with and simply integrating can be difficult at times. So I don't know, there's a lot of factors probably as in any industry.

Jon Krohn: 00:28:28 Yeah. It makes so much sense and hopefully I'm not going too deep here, and if I am asking a question that would give away some kind of IP or just feel free to not answer this. But it seems to me like in a situation like yours, where you are providing software to hardware companies, say the turbine manufacturers, you are not at least in the immediate term planning on building say your own turbines, your own wind farms, your software company. You need to be partnering with turbine and manufacturers, with wind farm operators. How does that work? I guess maybe your response is going to be similar where there's a range of responses where some turbine manufacturers are relatively early adopters. They see the potential. They say, "Wow, Jason's done a lot of amazing research in the past. He seems like the kind of person we should be working with to accelerate our roadmap." And then other folks are just like, "Yeah, we've got our own team," or I don't know. How does it look for you?

Jason Yosinski: 00:29:26 When we started this whole endeavor, what we imagined would happen is we would first build products that we would sell to people that own the turbines. Why do they want them? Because our product would help them make more money starting next month, right? We help them make more money. They like our product, we roll out, they tell their friends, we deploy to more and more farms, more and more companies. As we start to increase our market penetration in the industry, then much later turbine manufacturers would notice. And they would say, "Hey, everyone's using these windscape people. Maybe we should talk to them and consider integrating their thing

off the factory floor rather than as aftermarket add on." That's kind of still the process we're following, although we've been surprised that some OEMs are kind of interested in chatting early, I think they just want to have on their radar what's going on in the world and if there's any promising technology, they want to be there first. So I guess we're already having some of those conversations too.

Jon Krohn: 00:30:25 Mathematics forms the core of data science and machine learning. Now with my Mathematical Foundations of Machine Learning course, you can get a firm grasp of that math, particularly the essential linear algebra and calculus. You can get all the lectures for free on my YouTube channel, but if you don't mind paying a typically small amount for the Udemy version, you get everything from YouTube plus fully worked solutions exercises and an official course completion certificate. As countless guests on the show have emphasized to be the best data scientist you can be, you've got to know the underlying math. So check out the links to my Mathematical Foundations and Machine Learning course in the show notes or at jonkrohn.com/udemy. That's jonkrohn.com/udemy.

00:31:09 Nice. That's cool. All right, so I'm going to switch gears a bit now from Windscape to more broadly the research you've been doing. So you were describing from roughly 2010 to 2020, if I'm remembering correctly, that was kind of like your research phase and in that phase you were prolific. So one main focus of your research has been interpreting and understanding deep learning models. We already talked about the Deep Vis toolbox. So being able to visualize intermediate layers in the many layers of a deep neural network, which is what makes it deep for people who aren't already familiar with the way that deep learning works, that you have layers of these things called artificial neurons, which are very loosely based on the

way that biological neurons, biological brain cells work in your own brain in a human brain or an animal brain. And by layering these together, there's a lot of capabilities.

00:32:01 So when you are having a conversation with your ChatGPT or your quad or your Gemini, that incredible amount of nuance and understanding comes from just layers of these artificial neurons being able to do increasing complexity, increasing abstraction as you go deeper, but simultaneously that all of those layers, all of those neurons make it difficult to interpret what's going on inside of a model. So tools like your Deep Vis toolbox allow you to see... And people should really check out this YouTube video. It's amazing. It allows you to see not just layer by layer what's going on, but neuron by neuron. And so for example, my favorite part in the video is when you are on camera and you are highlighting a specific neuron in... It's in convolutional layer five of the network. There's 256 artificial neurons in that layer and one of those neurons based on the particular training data that this machine vision model was trained on.

00:33:06 So the machine vision model had to become capable in a broad range of different kinds of images. So some of the neurons became specialized in detecting text, some became specialized in detecting dog faces, and one of them in particular became specialized in seeing human faces. And so in real time in this video, you're on camera and as you move your head to the left to the right, you can see this white-hot activations happening for that specific neuron on the pixels that your face is in. And to make it even more compelling, there's a point where you bring a colleague in to the video and he joins you in the frame, and then we have these two white-hot areas of pixels representing where the faces are in the video. So really cool tools for being able to allow us to understand what's going on. And I think convolutional neural networks-

Jason Yosinski: 00:34:00 It's funny that we just had a one-minute explanation of that, but if we could actually just show it would be more obvious in 10 seconds or something, which is maybe-

Jon Krohn: 00:34:08 Yeah, yeah, for sure.

Jason Yosinski: 00:34:09 In the first place.

Jon Krohn: 00:34:11 Absolutely. And there's all kinds of things we would love to show people, but because almost 90% or more, 95% or more of our listeners in a typical are audio only. So yeah, although you and I-

Jason Yosinski: 00:34:23 Describing a white-hot activation is as good as we're going to get. Yeah.

Jon Krohn: 00:34:26 Yeah, exactly. So yeah, my point is getting all this with convolutional neural networks with machine vision problems, those are cool because visualizations are... There's something that kind of comes quite naturally from that. Whereas today, some of the most impressive generative models, certainly the most widely used generative models are outputting text. And so that becomes harder to visualize in a cool way like you did. Anyway, as models continue to get bigger and bigger and bigger... I mean, that's a whole other dimension here. So as we're talking about models that aren't visual and as they get bigger and bigger and bigger, how can we continue to understand what's going on in them or does that matter at all?

Jason Yosinski: 00:35:19 Yeah. No, great questions and questions that I don't really have the answers to. And a lot of people maybe don't necessarily have the answers to. A lot of topics, so as models get bigger, they certainly get more complicated. They certainly get harder to understand. Even back in 20... I think it was '15 when we published the Deep Vis toolbox, that model had 60 million parameters and like

you mentioned on one of the layers, conv5 had 256 neurons. Even that model, even I who wrote the paper played with the toolbox maybe as much as anyone scrolling around to all the different neurons, even I can't claim I understand what was going on inside, right? We found a face detector, that was great, but it just happened to fire a lot for faces. We didn't actually know whether it would fire .6 and not .5 for another part of an image that was not really face-like, but a little face-like.

Jon Krohn: 00:36:15 Yeah.

Jason Yosinski: 00:36:16 Or we know that that-

Jon Krohn: 00:36:18 .5 was very important, like maybe that was-

00:36:20 [inaudible 00:36:21].

00:36:21 Or a dog face or a clock face.

Jason Yosinski: 00:36:24 Yeah, for sure. For sure, it would fire for monkey faces and dog faces, but we just by seeing it fire doesn't mean you understand everything it's doing in all the downstream layers. And actually the layer right after conv5 was in this network FC6. So the sixth layer fully connected layer, it had 4,000 neurons, [inaudible 00:36:45], right? So you see these 4,000 little things spiking. You could try to scroll through every single one, but what would it take for you to claim you understand what's happening? We're probably not going to get there.

00:36:56 Nevertheless, showing the Deep Vis toolbox, I think teaches people who don't know how networks work a lot about how they work very quickly, which is I think something I was proud of in that paper. Also, if you work with convolutional networks, it teaches you subtleties of how these networks work, that might not have been obvious before as a practitioner, as someone trying to

debug a network that might be broken or not training well or not generalizing well, just seeing high bandwidth visualizations, I think can often help. So I think that fact, let's now fast-forward a couple of years to get to where we are today with much, much larger models, I think that fact still holds. So seeing visualizations for how networks are working is probably still useful to people trying to debug problems with those networks. But it has not led to and probably will not lead to full human understanding of what's going on inside. So I think it's useful as a tool in a practitioner's tool belt, but will not make models on their own explainable.

Jon Krohn: 00:38:01 Nice. Makes sense. And so do you think that that experience... And I realize that now as we get to models, large language models behind things like GPT-4 might have, and we don't know because they haven't officially published it, but there might be a couple billion artificial neurons in there. There's no chance of us in the way that you were able to go over the 4,000 neurons of the 256 neurons in an architecture. It was [inaudible 00:38:31], I think that-

Jason Yosinski: 00:38:32 It was AlexNet or... AlexNet, yeah, yeah, yeah.

Jon Krohn: 00:38:34 AlexNet. Yeah.

Jason Yosinski: 00:38:34 AlexNet. Yeah.

Jon Krohn: 00:38:40 And so of course... What was I thinking? It's way too many convolutional layers for [inaudible 00:38:45]. Anyway, yeah, so as you're saying, it becomes even there where you're looking at hundreds of thousands of neurons in a layer, millions of neurons total, it is difficult to interpret too much, but you can still get some sense of what's going on, maybe learn a bit, understand where things are working well, things aren't working well.

Jason Yosinski: 00:39:08 Yeah, just having the basic starter visibility into something about the network was at the time really important and is still important. So when I started working on this, I was at... It doesn't matter where I was. Somewhere working for the summer and I was working on kind of my real project, which had some possibility of publishing like a normal paper. And I realized nobody had made plots of what's happening inside of the network, just like a live video plot of what's happening as you feed some video stream. And so this became just like a weekend project. I just wanted to see it for myself. And as I worked on it, I was realized, I was really frustrated with just how little we know about what's actually happening in the network. As you train a network, what do you do? You watch the loss, it starts out really high and it shrinks over time.

00:40:00 You hope, and that's the training loss, and maybe you watch the validation loss too, and you see that shrinking a little bit too, but maybe not quite as much. But that's like you're watching a scalar and there's so much magic or maybe broken parts happening inside of a network training, and you can only see this one little number decreasing. So for example, let's say I went into your network, you initialize it, I went into your network and I just randomly set half of one of the layers to zeros and I left the other half, okay, the same. If you start training, it'll probably still train, the loss will go down. It might be really subtly broken in some tiny way, but you as a network trainer have no basic visibility into the fact that I just broke half of one of your layers. Because there's no visualization, you even watch as a matter of regular course to see this.

00:40:49 Or let's say I go in and multiply one layer by 10 and divide the next layer by 10. The net Jacobian or whatever beginning to end is the same, but the training process will proceed very differently. Would you be able to detect that

by looking at your whatever plots you normally look at? Probably not. So it was funny, this was true in 2015, and so we started working on a couple of these papers to try to produce greater visibility, but I think it's still true today. I think most people that train networks, they click go, either it works or it doesn't, and if it doesn't, they try something else. Why don't we have the oscilloscope for training? Why don't we have the oscilloscope for network operation representation and so on?

Jon Krohn: 00:41:30 Yeah, you're right. It seems like there is still a lot of potential in there. I also realized I misspoke earlier when I was talking about model size. So as these models get bigger, these kinds of things like something that could act as in oscilloscope over the whole network as opposed to having to individually probe yourself neuron by neuron or layer by layer. I said that some of the biggest networks, I think I said one to 1 to 2 trillion is where we're at now.

Jason Yosinski: 00:42:00 We can also clarify the number of parameters versus number of-

Jon Krohn: 00:42:03 Oh, neurons, that's right. Of course. Of course. Course.

Jason Yosinski: 00:42:08 Very, very big networks though is the point.

Jon Krohn: 00:42:09 Yeah, and it's funny, I do that embarrassingly often where I interchange between parameters and neurons. When you can have orders of magnitude more parameters trivially relative to the number of neurons, and often I am quite aware of that fact. I mean obviously when I'm working through the math or when I'm building a network, but it's funny how we can use it interchangeably and you often catch people in arguments say, talking about AGI, they'll say, we now have a network with 2 trillion artificial neurons, the human brain... Or sorry. So really there's one to 2 trillion parameters, connections.

Jason Yosinski: 00:42:50 I think neurons to a lay audience is easier to map to the concept of a brain, which I guess you could say is connections like dendrites or something.

Jon Krohn: 00:43:00 Exactly. But then you're starting to really get into some... You'd have to at least have five minutes of neuroanatomy before explaining, so you end up in the situation where you have one... There might be about 2 trillion parameters in GPT-4, and not all of those are active on a given call. So it seems like there's eight based on the rumors, you have these eight expert networks in this mixture of experts. But anyway, but you have that many say 2 trillion parameters, but the number of neurons could be many orders of magnitude, fewer than that. Anyway, I am kind of off on an unnecessary-

Jason Yosinski: 00:43:42 It's hard to talk about in a convolutional network. It's almost like the neurons are replicated in space, so are the unique neurons are not in a transformer, the neurons might be applied at every single token or those the same neuron or not. So even using these words is under-specified.

Jon Krohn: 00:43:59 Yeah, yeah, yeah, that's right. So when you were at Uber Labs, Uber AI Labs, your research, it seems like there's a little bit of a connection in terms of forecasting. So in particular, some of your papers from then were on extreme event forecasting, and so that highlighted the challenges of making accurate predictions during high variance periods. So say when you are hailing a Uber and a Taylor Swift concert just ended nearby, it's going to be harder or New Year's Eve. And so some of these New Year's Eve, that's probably relatively predictable. You could hand code something in to be expecting that kind of situation. But all kinds of things happen like there are manmade or natural disasters that happen that are completely unexpected or other kinds of events protests that could completely change the forecasting that a car

hailing app needs to be able to do in order to get a car to you and set a market appropriately. So is there any kind of relationship between that kind of extreme event forecasting or forecasting that you're doing at Uber AI Labs and the kind of now casting that you're doing today with Windscape?

Jason Yosinski: 00:45:26 That's interesting. I had never thought of making that connection, but yeah, actually the problem formulation is not so different. In both cases, you might formulate a network to model a problem. You might use a loss, which is mean squared error or something which optimizes for the general case, but doesn't directly optimize for extreme events for Uber. Uber might care about... Yeah, like that Taylor Swift concert. That happens very rarely, but there's a huge price surge and you might care about forecasting the probability of these four sigma, five sigma events. For example, I think at ride-sharing companies, they would directly message drivers to try to get them to get on the road because they have a chance of making a lot of money at these specific events.

00:46:17 So it can be worth a lot of extra effort just for that one time thing. In wind, we might also think about optimizing for the general, just like is it windy or not? Use a regression loss or something to track expected amount of wind. But it turns out that a lot of the instability of what happens with the grid and a lot of the money that changes hands changes hands in very rare cases where something is mispredicted by a lot. So you might've heard about the blackouts in Texas when there was the freeze, I want to say 2021 winter.

00:46:52 So I don't know two or four sigma events happened then that led to, I want to say... I might [beep] the figure. I want to say it was $5 billion worth of energy changed hands. A lot of people made money, a lot of people lost money. Also, the grid, parts of the grid went down, they

blacked out, chunks of the grid to save other chunks around near hospitals, for example. Okay, back to the modeling side. Yes, to model rare events explicitly might be useful at Uber, might be useful for the grid as well for these reasons.

Jon Krohn: 00:47:30 Nice. Yeah, and I did quickly look this up. February 2021 was when that happened.

Jason Yosinski: 00:47:34 Okay, yeah.

Jon Krohn: 00:47:38 Yeah, so interesting that there ended up being a little bit of a parallel there. Very cool. So another presentation or another paper that you had at Uber, it was probably both, that we thought was particularly interesting and wanted to highlight here was you talked about the learning process in neural networks. So you talked about top down versus bottom up or even synchronized. And so that isn't something that I've thought about before. So when I think about training a neural network, I think about having what's called the forward pass where you go from some input to some output. So for example, if it's a machine vision algorithm, the input could be the pixels of an image and then the outcome is the prediction of the class of that image. It says, Hey, this is a cat or this is a truck or this is Jason Yosinski or whatever.

00:48:31 So you have this forward pass from the input to the output and then the gradient descent that allows us to update all of the parameters throughout all the layers of the neural network that goes backwards. We call it back propagation from the output layer back towards the input layer. Yeah, so Jason, fill us in and let me know if my high level summary or any of my ideas there made any sense related to your paper from 2019, which was in the most prestigious AI conference called NeurIPS and the paper was called LCA, Loss Change Allocation for Neural Network Training.

Jason Yosinski: 00:49:06 Yeah, prestigious conference also basically the last conference for a couple of years. The last conference we all met in person in Vancouver before COVID killed in-person conferences for a couple of years. Yeah, so Loss Change Allocation was a paper. The first author is Janice Lam who was at Uber at the time. Our goal here was to really start to build something like that oscilloscope or maybe a microscope that helps us examine training. So let's imagine you have a network and it has 10 layers and you start training that network and you watch your loss go down. Now let's imagine as you watching that loss go down, I'm sneaky and I grab one of the layers and it just freeze it, and the rest of the nine layers keep training, but layer four in the middle or something is frozen and stops learning. Do you think you would notice that in the actual loss signal?

00:49:58 I guess if you had two runs and they were identical in every way except for that, you might notice a little deviation, maybe the learning slows down a little bit. But more or less the fact that I just grabbed 10 million out of your 100 million or something parameters and completely froze them is mostly not visible to you, which we thought was just silly. And so we tried to build a method that would let you see learning but on an individual neuron level. This is really tricky to do because in some sense, if you define learning as just the function represented by the network changing over time to better fit the data set, then learning is really a property of the entire network. But we came up with a way of breaking down the change in loss, allocating it to individual neurons in such a way that the little score of all the neurons, if you add them all up, you get a score for the entire network, which exactly matches the change of the loss.

00:50:47 Okay, so cool idea. So we took this and implementing it efficiently is a little tricky, but we found some approach that worked well enough. We took this around and

started training networks and as they're training, we watch every single neuron and we kind of see is it learning? Is it going the right way or is it kind of anti-learning? So going the wrong way. We can assess that by looking at the training loss. Is the training loss going down for that neuron or up for that neuron? Separately, we can look, is the validation loss going down or up for that neuron or that parameter? Thanks to Jon and I's earlier discussion about parameters versus neurons. You could do this on a neuron level or a parameter level. So you can watch validation versus train and figure out if a neuron or a parameter is individually fitting as in fitting both train and val or over-fitting as in fitting train but not val.

00:51:41 So we did all this. We ran the method, we generated lots of plots. A method like this generates a ton of data. You basically have one number per parameter, per time step of the training, which is a huge volume of numbers. It's like you have to snapshot the network every single step. And what we found was more or less a huge mess. So we found a lot of data, a lot of neurons doing a lot of things, and it was really hard to sift through this data and make sense of it all in a way that led to a clear story. So I would say that was one of our first conclusions from this paper. We did find a few things. So if you just look at all the parameters together, it turns out that throughout most of training, most parameters are swinging back and forth, and if you think of it as being in a little valley, in a little hue, they're more or less going up and down the walls of this valley.

00:52:35 So about half the time, they're going the right way to the bottom of the valley, decreasing loss. The other half of the time they're just going up the valley increasing the loss. So I think our number from one of the networks was 50.3% of the time parameters were going in the right direction and 49.7% of the time they were literally going

the wrong direction. But because there's so many parameters and so many time steps on average, the network is actually learning and the loss actually does go down. We also found some fun things, for example, some layers for the entire course of training, if you add up all the progress they made, literally take the network backwards, so they literally hurt the network.

00:53:15 We found this I think in many cases for the last layer of networks. And so our hypothesis was, well, if you just freeze the last layer and don't let it learn anything, you might literally help the network. And so we did that and it did help the network. It worked. Why does this happen? Interesting question. Could be a follow-up paper, I can go into that maybe. But just the fact that it was happening and that we were able to measure it and that maybe people should consider measuring these sorts of things, I think was a partial fulfillment of our goal. There's a lot of other plots in the paper, a lot of other work, and honestly a lot of other data that's hard to analyze because there's so much happening in a high-dimensional space.

Jon Krohn: 00:53:54 Is kind of the intuition there behind because the final layer is closest to the cost function that you're optimizing, that it can be easiest in a way to figure out with gradient descent that, that is what we should be changing. So I think you were talking about just there freezing the final layer and so maybe freezing that final layer has the positive impact because by freezing that final layer, you're allowing the penultimate layer and the third last layer to be able to do more learning than they otherwise could.

Jason Yosinski: 00:54:25 All right, so you want to dive... Dig into this? Okay, let's do it. Okay. First, let's assume the network is configured in a way that's relatively sane. In other words, known layers are configured super horribly. Let's imagine we're in a case of classification. Let's imagine for the sake of example, it's something like ImageNet. So there's a

thousand classes on the output layer, so there's a thousand neurons in that output layer. Let's say we initialize the network randomly, so the loss is really high, and then we run a bunch of separate experiments. Let's say we froze all the layers and then just thawed one, and we just trained that one layer at a time. No matter what layer we choose, do you think it would work? Do you think it would learn something?

Jon Krohn: 00:55:08 I think so. I can't immediately think of a reason why not?

Jason Yosinski: 00:55:12 Yeah, so pretty much as long as you didn't [beep] up the initialization in some pretty catastrophic way, any individual layer will indeed learn. It will move the network in the right direction and it'll push that loss down. Of course, in practice, we don't train one layer at a time, even though we could because it's much slower. You do all this forward and backward computation, you may as well use all those gradients that you just spent forever getting and train all the parameters instead. Okay, so how did we find that this last layer doesn't learn net? We found this kind of complicated kind of beautiful story where the layers are individually learning, swinging back and forth, and there's this sort of periodic motion.

00:55:56 If that periodic motion is synchronized between all the layers, they're all kind of learning together. In some cases, if networks, if the layer gets too far behind, then it's kind of doing the right periods. But it's always so far behind the other layer's representations, that it's learning based on the old representations. And on average it's more wrong than right. And this basic fact is why freezing the last layer helped. So if you just say, "Stop, stop trying, you're always too slow. You're always learning too far behind everyone else that actually approved the situation." It's a completely different way of seeing why this might be a reasonable idea. And I can give you that explanation as well. So let's say we have a classification

network. It's got a thousand classes, like we said, one of them is dog, one of them is cat, one of them is lion.

00:56:41 What is dog? Okay, we have to decide as a human engineer, we have to decide a dog in this network is going to be represented by a one hot vector, 100000. What is cat? 0100. We chose that arbitrary vector starting with one, one in the second spot, one in the third spot to represent the concept of dog. But why do we choose that? Well, we wanted to spread out the ones I guess. We wanted to have them all be unique, I guess. Okay. But let's imagine another representation. Let's take those vectors and sort of back-project them through the last layer.

00:57:15 So let's say that very last layer had a thousand neurons. The one before it had let's say 4,000. So back-project, those one hot vectors through that 4,000 by 1000 matrix, and you'll get a thousand vectors of length, 4,000. And now they're not one hot, they're just a random vector in that 4,000 dimensional space. Happens to be almost guaranteed, mostly orthogonal just from the way random vectors work. So if we freeze that last layer, all we're doing is we're saying to a slightly shorter network, please learn to represent dog. And instead of this one hot thing we chose, we're choosing a random vector in this 4,000 dimensional space. Does that work better or worse? I don't know, turns out it works better. As measured-

Jon Krohn: 00:57:57 Really?

Jason Yosinski: 00:57:58 Not every time, but actually quite a lot. And there's a paper, and I can't remember the author off the top of my head. There's a paper that showed this worked in a number of cases and I wouldn't be surprised if they recommended you just do this all the time. Because it's just a good idea.

Jon Krohn: 00:58:11 Right. And so to kind of recap that idea back to you, it's instead of having an arbitrary one-

Jason Yosinski: 00:58:17 One hot vector.

Jon Krohn: 00:58:19 You take your randomly initialized initialization where there are random vectors in the penultimate layer, but those map to our one hot and it gives a much more nuanced-

Jason Yosinski: 00:58:37 Dense, much more dense.

Jon Krohn: 00:58:39 A much more dense, right, yeah. A much more dense representation to be mapping into. Yeah. That's cool. I had never thought of that. How did you think to do... Oh, I guess that came out from freezing different layers and seeing what happens.

Jason Yosinski: 00:58:53 Yeah, exactly. Exactly. Just came out of the measurement. Now there is still a nice feature of going through that last layer and finishing in the network. It's that you get to use actual cross entropy loss and have an actual probabilistic interpretation. You don't want to just randomly initialize vectors and then use a mean squared loss because that's a different loss. So probably you should still be using cross entropy unless you have a really specific reason not to. But yeah, choosing that random representation seems like a fine idea though.

Jon Krohn: 00:59:27 Wow. Very cool. I learned something that sounds like a very fundamental understanding of neural networks today.

Jason Yosinski: 00:59:34 Cool.

Jon Krohn: 00:59:35 So that is cool. That doesn't always happen on the show. I'd say that that doesn't happen very often at all. Awesome. So one final topic area before I let you go. You

co-founded and are president of a organization called the ML Collective. What is the ML Collective and why did you found it?

Jason Yosinski: 00:59:54 What is the ML Collective? Yeah, so at Uber, when I was at Uber, my friend Rosanne and I had a research group, which we called Deep Collective, and we were kind of the crew of people that worked on deep learning, on neural networks, just pushing the fundamentals and neural networks forward. We had a fun group, we liked it. A lot of people liked it. We had people both from AI Labs and from outside of AI Labs. We would just have a meeting once a week, kind of an all hands and jam on whatever we were working on. It was more or less a cool crew, a productive crew. We had a good time. Middle of COVID, Uber Post IPO decided to let go of scientists, including me and I think everyone, or maybe everyone, but one of the people on the deep learning Deep Collective team.

01:00:40 And so we thought, well, maybe we should just take this thing and rename it slightly and make it an external open entity. We did that. At first, it was more of a closed but slightly open research group, and then over the years it kind of has morphed into just a completely open group that does a couple things. We have a Discord, people can sign up for the Discord, anyone can. You can look for collaborators there. You can look for resources there. You can literally post drafts of papers and have editing help. We also run a weekly speaker series. Rosanne runs a weekly speaker series that was actually going on inside of Uber, and that's been going for I think over five years now. So many people have come by and presented their papers. We also have a few events. We do events at conferences sometimes.

01:01:32 So actually at [inaudible 01:01:33] last week... It was last week, Rosanne ran a social kind of open collaboration, social, I think it was called, where people that want to

meet collaborators can meet collaborators. I would say the mission of ML Collective is to be a research home base for people that don't have one. So if you imagine first principles, what does it take to be an effective scientist? Let's imagine you start as a grad student and you want to be a great scientist. What does it take? You should be good at some individual skills like programming. You should have access to resources like a computer. That's great. Some GPUs you can use to run experiments on. You should have access to mentorship in the form of maybe an advisor. Also, mentorship in the form of postdocs or PhD students that are further along that can help you.

01:02:25 And it sounds simple, but you should have a place to show your stuff. So when you make a cool plot or you make a confusing plot, you should have people to show it to. So some people as they start grad school maybe have access to all these things and maybe they do well or they tend to do well because of it. Many people would like to get started, but they don't have access to some of those components. Maybe they have some, they don't have some. We help people find whatever kind of components they're missing. Some of them are easier to provide than others. So Discord, a forum where people can meet and self-organize, works well for some people. Actually giving people GPUs is relatively easier. We can't give people OpenAI level, ChatGPT level GPUs, but we can give them a couple GPUs and if you use [inaudible 01:03:12] enable someone that can count for a lot.

01:03:15 Providing incentive-aligned mentorship, I would say is the hardest thing to do. So we try to do that. In some ways we try to help people coordinate these types of relationships, but I would say that's still a hard problem to solve. That's a problem solved by a lab relationship where you have an advisor and they work with you for many years and your incentives are really aligned because you both want to

produce lots of great papers over years. That's harder to provide in a distributed way, but I would say we do what we can there. We also have office hours, so anyone can stop by, book a time and stop by any of the office hours with some of our mentors just to brainstorm, ask questions, ask about the process of research or literally look at a plot together and try to interpret it together.

Jon Krohn: 01:04:00 Nice. Very cool. That sounds amazing. It sounds like something I would love to have in my life. And it reminds me, pre-pandemic, I had a group called the Deep Learning Study Group, which-

Jason Yosinski: 01:04:10 Yeah. Cool.

Jon Krohn: 01:04:12 Yeah, I set up... The genesis of that was I was at ICML, which happened to be in New York that year.

Jason Yosinski: 01:04:19 Yeah, 2016?

Jon Krohn: 01:04:22 Yeah, 2016. And so I went because I live here in New York, so it was super easy to get there. And there was a talk... I can't remember, you might even know who this researcher is, I can't remember off the top of my head. At the time, he wasn't someone that I had heard of, but it's someone who had been working on machine vision with neural networks since the '80s, [inaudible 01:04:48] was sitting in the front row at this talk and nodding his head vigorously to everything that the guy on stage was saying. And it included things like... So the guy was providing a history of neural network research, so he was talking about the formation of NeurIPS.

01:05:03 He was... Oh, NIPS the neural information processing systems. And he talked about how the... Because there is so much kind of like you're describing, there's so much around understanding neural networks and making breakthroughs in neural networks that is collaborative.

People come from different areas, from physics, from neuroscience, from computer science, and together are able to say, oh, borrow ideas from different places. And some of those end up having great effects. He showed a video if this helps you figure out maybe who it was. He had a video of a self-driving car in the '80s driving around a closed track and it was entirely neural network driven.

Jason Yosinski: 01:05:49 I don't think I remember this exact talk or I can't remember the person giving it, but I can picture a little bit. And I think your point is the collaborative nature.

Jon Krohn: 01:06:00 Exactly. So seeing that the collaborative nature, I was... Like a light bulb went off in my head. I was like, I need something like your ML Collective to accelerate my own learning of these concepts, my deployment of these concepts. And so that evening there was a meetup in New York, it still happens. It's called the New York Open Statistical Programming Meetup, a very cool thing. Wes McKinney is often there and he was involved in the starting of it. He's probably the biggest name. Hilary Mason is often around there, Hadley Wickham. So really cool big people go to this meetup and there was a meetup, it was a monthly meetup and they had a meetup the same evening that I had seen this presentation at ICML. So I stood up at the beginning, the guy who hosts this meetup, his name is Jared Lander, he always gives the opportunity at the beginning for people who have hiring opportunities to stand up.

01:07:04 And I stood up and I was like, "I don't have a hiring opportunity, but I was just at this ICML talk today and Neural Network research seems to be highly collaborative. We can do a lot together." And so I said, "I would love to... There's people here that would like to join me and study on a weekly basis, meet up, cover particular Stanford lectures or textbook chapters together, papers, and then review them, talk about our own problems." And I guess

that's one of those... You get a lot of nerds in a room together. At the time that I was standing up and looking around the room, I didn't seem to see any head nodding. I just felt really awkward. And so I sat back down and I was like, well, whatever.

Jason Yosinski: 01:07:45 Yeah. Did you give your contact? Did people contact you afterwards though?

Jon Krohn: 01:07:49 Yeah, so at the end of the talk... So that was the beginning of the talk, someone spoke for an hour and at the end of the talk, about a dozen people crowded around me and that formed the first email list as well as the ideas for our first... So we ended up working through Michael Nielsen's Neural Networks and Deep learning ebook initially. And so that initial 12 people, I shared out to a couple... There was women in machine learning. It was a particular network that I had already spoken at a few times in the New York area.

01:08:23 So I reached out to their organizers and they sent out an email blast. We had 200 people on the email list before the first meeting. And the first meeting had 60 people show up, which was a really cool experience. And for years, that is what allowed me to develop the deep learning material that I taught, that I turned into a book. And that in a way led to me hosting the show and that kind of stuff. So I think community is huge. That was a very [beep] whoop. They'll bleep that out. Very long-winded story to say that I completely understand where you're coming from with the ML Collective and I really miss... Since the pandemic hit, I haven't rekindled that and it's a big gap for me. So you meet regularly in person in San Francisco?

Jason Yosinski: 01:09:17 No, it's almost entirely virtual. That way people can participate around the world. I would say one of the-

Jon Krohn: 01:09:22 Yeah, yeah.

Jason Yosinski: 01:09:23 Which is finding times that work for people and every time zone and we always leave someone out or require that someone wake up at 3:00 A.M. or something. Which we have had people do. We've had people wake up at 2:00 A.M. and give a talk and go back to sleep.

Jon Krohn: 01:09:39 Oh, my God.

Jason Yosinski: 01:09:39 But no, I think to your point, right? If you ever see value that you can add to 20 people's lives just by getting them together, that's such a clear win. The cost might be minimal. It's just like a win-win, win-win, win for everyone. Go for it. And I'm really glad you did and it sounds like it was really valuable. I would say we're trying to do the same thing to the extent we can.

Jon Krohn: 01:10:00 Nice. Yeah. Yeah, I missed it a lot. So yeah, so very cool. So ML collective, check that out. We'll be sure to have a link. It's at mlcollective.org and anyone who's listening can go and check it out.

Jason Yosinski: 01:10:09 Anyone can join. I would say we have our weekly talks. Our most common event is the weekly talks on Friday. It's Friday, 10:00 AM Pacific, 1:00 P.M. Eastern. And almost every Friday, there's a speaker presenting a paper.

Jon Krohn: 01:10:23 Nice. So final question for you, final technical question is you're an angel investor and advisor to several ML startups. What key qualities or strategies do you think set successful AI companies apart from the others?

Jason Yosinski: 01:10:37 I mean, I have very limited experience compared to, for example, VCs that have been investing for years and things. To me, having really, really great people seems to matter a lot. And really understanding the problem you're trying to solve seems to matter a lot. Like why that

problem matters to someone, why that problem matters to a customer as opposed to really great people just in a room trying to build something that they think is cool but they can't sell. Or people that understand the market but don't have the technical chops to build something great. They know AI might be a solution, but they don't have the machine learning chops, maybe. If you get both of those, I don't know, it seems you'd have a fighting chance.

Jon Krohn: 01:11:18 Nice. Very cool. Well, thank you for that. Put you on the spot with that one. All right, Jason, this has been an amazing episode. You've been very generous with your time as well. Before I let you go... Oh, yeah, sorry.

Jason Yosinski: 01:11:30 It was really fun talking. Thanks.

Jon Krohn: 01:11:32 Nice. Yeah. Well, my pleasure. We'd love to have you back, that's for sure. And you did mention a couple of books already in this episode, particularly Rewiring America seems like one that was very interesting to you. But do you have any other book recommendations for our listeners?

Jason Yosinski: 01:11:47 Yeah, I really liked Rewiring America, like I said, really goes through from an engineering perspective, what do we need to do to fix climate change? Because you asked for another one. I'll pitch a completely separate type of book. I really like 100 Years of Solitude by Gabriel Garcia Marquez, just one of my favorites, classic. Yeah, if you read it, hope you like it.

Jon Krohn: 01:12:10 Awesome, thank you so much. And other than Mlcollective.org, how should people be following your work?

Jason Yosinski: 01:12:15 Well, I used to post a lot more and these days I don't much at all. I guess, I'm technically on Twitter, but I don't really use Twitter anymore or X. Yeah, I don't know. You

		can follow me on Twitter and someday maybe I'll resume posting.
Jon Krohn:	01:12:30	Nice. All right, great. Well, we are all cheering you on with Windscape. We love social applications of AI and we know that we all benefit. So fantastic. So glad that you have found that and we're excited to see what happens next.
Jason Yosinski:	01:12:46	Cool. Yeah, thanks for having me, Jon.
Jon Krohn:	01:12:54	Eight years of waiting for me to meet the brilliant Dr. Yosinski and he did not disappoint. In today's episode, Jason filled us in on how Windscape's ML-infused tech allows turbines to track wind direction to adjust, making them more efficient. That wind direction can be modeled with an autoregressive neural network allowing grids to handle capacity planning better. How his Deep Vis toolbox is useful for understanding individual neurons in a deep learning network, but that doesn't make the network fully understandable. How his Loss Change Allocation research revealed that freezing the final dense layer before doing any training at all improves model fit and how the ML Collective allows researchers anywhere in the world to benefit from a lab group with functions such as study groups, mentors, and GPUs. As always, you can get all those show notes including the transcript for this episode, the video recording, any materials mentioned on the show, the URLs for Jason's social media profiles, as well as my own at superdatascience .com/789. Yes, that's fun.
	01:13:51	Thanks to my colleagues at Nebula for supporting me while I create content like this Super Data Science episode for you. And thanks of course to Ivana, Mario, Natalie, Serg, Sylvia, Zara, and Kirill on the Super Data Science team for producing another mind-blowing episode for us today. For enabling that super team to create this free podcast for you, we are deeply grateful to our

sponsors. You can support this show by checking out our sponsor's links, which are in the show notes. And yeah, if you yourself would ever like to sponsor an episode, you can get the details on how by going to jonkrohn.com/podcast.

01:14:23 Otherwise, please share, please review, please subscribe and all those kinds of good, helpful things for us. But most importantly, I hope you'll just keep on tuning in. I'm grateful to have you listening and I hope I can continue to make episodes you love for years and years to come. 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.