EPISODE 1354

[INTRODUCTION]

[00:00:00] KP: The first industrial deployments of machine learning and artificial intelligence solutions were bespoke by definition and often had brittle operating characteristics. Almost no one builds custom databases, custom web servers, or custom email clients. Yet technology groups today often consider developing home-grown machine learning and data solutions in order to solve their unique use cases.

Today's modern data stack is often a patchwork of interconnecting tools built to suit a variety of personas that need to interact with the data in notably different ways. In this episode, I speak with Leigh Marie Braswell, an investor with Founders Fund. We have a wide-ranging discussion about the technology landscape of data and machine learning solutions and the modern enterprise data stack.

[INTERVIEW]

[00:00:52] KP: Leigh Marie, welcome to Software Engineering Daily.

[00:00:55] LMB: Glad to be here, Kyle.

[00:00:56] KP: Can you talk a little bit about Founders Fund and your role there?

[00:01:02] LMB: Absolutely. So I started as an investor at Founders Fund about six months ago. Kind of prior to that I was early engineer at Scale AI and then a product manager there and started angel investing. And that's kind of what led me to want to join Founders Fund.

So Founders Fund, in general, we were founded about 15 years ago, Peter and his PayPal co-founders at the time saw a lot of problems with venture capital. So when Peter's at PayPal, they reinstated the CEO multiple times, overruled founder decisions, and he wanted to start a venture capital firm where we found really great founders. Really, that's our only thesis, find amazing founders and invest as much as we can in their companies.

So we invest anywhere from pre-seed to now with our growth fund pre-IPO across any type of company.

[00:01:50] KP: Do you have any particular sector or technology you're focused on?

[00:01:54] LMB: Yes, absolutely. So kind of given my engineering ML background, I'm particularly drawn to everything in dev tools, data infrastructure, ML applications and infrastructure, but I invest across stages.

[00:02:09] KP: How did you get started machine learning?

[00:02:12] LMB: Yeah, that's a good question. So I grew up, I loved math. So I did a lot of competitive math growing up, and that kind of led me at some point to discover programming, which I thought was super interesting. And then when I was in college, it just seemed like machine learning was kind of a marriage of math and programming and it turns into a little bit of magic.

So I took some classes in college. Thought it was extremely interesting. And you could see you know the applications starting to develop at the time. There were self-driving car companies, but they were extremely speculative, and some other sort of robotic applications.

But yeah, as I sort of graduated college and thought about what I wanted to do next, I joined Scale AI. So the sort of mission of the company is to advance AI applications. So we start with ground truth labeling. And primarily at first was serving autonomous vehicle customers, but now Scale has branched out to many different types of customers, NLP labeling and JsonML for products. So really being at Scale for four years, that's what kind of gave me a front row seat to everything that was going on in ML and, yeah, solidified my interest in it.

[00:03:31] KP: What sort of things do autonomous vehicles need labeled?

[00:03:35] LMB: Yeah, they need a lot. I think a great analogy with ML is you're teaching a baby or a child how to recognize certain things and act on them. So most autonomous vehicles, they have multiple sensors to figure out what's going on in the world around them. So all of them, or all that I know of, have cameras. And that's just cameras all over kind of placed in different spots around the car to give you that 360 degree view. And then many AVs, or autonomous vehicles, have LiDAR. So that's a 3d depth sensor perhaps supplemented with radar which can go even further.

So essentially when you are training an autonomous vehicle to see things, like to recognize pedestrians, to recognize where they are in the world, you have to give them lots of data that's been labeled by humans to teach them essentially. For example, something that we would do at Scale a lot was take images and then have humans and ML on our side and figure out, “Okay, here's where the pedestrians are on these images. Let's draw 2D bounding boxes around them. Or for LiDAR data, here's where a bicycle is traveling in the LiDAR data. Let's put a 3d box around it.” And once an ML algorithm sees enough of these examples, it can learn how to behave in any sort of circumstance. Obviously, that’s an over simplification.

But what's interesting about this is, with this sort of method, with deep learning, you need so many different examples labeled by humans to teach the algorithms. And that's where something like Scale comes in handy.

[00:05:15] KP: Well, that vision of AI, if I kind of in my mind extend that out, it seems like we could get into a world where development is just consuming all these AI microservices. I go to one vendor for pedestrian detection. I go to another for something else, sort of the antithesis of me building some AGI approach. Do you have any thoughts on what near-term industrial success is going to look like?

[00:05:38] LMB: Yeah, I think it'll be super interesting. I think for these sort of super sophisticated use cases like self-driving that need 99.999% accuracy, there probably won't be sort of a micro service that offers that. That's kind of the core competency of these companies.

But then for more general use cases, we're already starting to see sort of ML in a box APIs that people are offering now. OpenAI obviously kind of famously with GPT3, but now we have multiple companies offering different vision and NLP APIs to consume. So I think it's super exciting. I mean, it'll be interesting to see, yeah, what's the sort of mix of people that are consuming these APIs versus building internally. And what's the persona of the user that ends up consuming these APIs, right? Is it an ML engineer who wants to get something spun up very quickly? Or is it a software engineer or even a less technical user? And I think the companies in the space are all developing opinions about who the right sort of end consumer is. At least I think it's more of a timing question, right? What's the right time to give people these tools?

[00:06:46] KP: Yeah, yeah. Well, autonomous vehicles are – We're definitely in the moment, right? There's a lot of money and interest in research and success flowing into that area. I mean, I mean you could disagree, but it seems like now would be the wrong time to start a from scratch brand new automated vehicle company, because you're up against such stiff competition. Maybe you disagree or don't. But yeah, I guess let's start there. Do you think this is the time a challenger could come into the market? Or is it more about finding other ways to contribute if you want to innovate in this space?

[00:07:16] LMB: Yeah. I mean, it's a great question. I think, as you've said, a ton of venture capital money has already flowed in to quite a few autonomous vehicle companies. And we've seen massive consolidation in the space. So multiple companies going public, multiple companies getting acquired. So the space is starting to mature. Now we have deployments where people are driving around certain cities. Or maybe we have cities where you can get a delivery by a robot. There are like clear real world examples, which is super exciting.

In terms of could you start a self-driving car company now? Actually I'm optimistic that there is always some sort of technological innovation that can unlock sort of new developments. So for example, one that was started very recently, Wabi, from Raquel, who used to lead Uber ATG's team, they have kind of a novel approach where they're overcoming their data disadvantage. Essentially all these companies have accumulated so much data over the years by doing really novel things in simulation. So that could potentially really work. And I think with AV, as with most companies, just the best strategy ends up winning. And wouldn't say that I'm completely convinced last mover can't be done effectively.

[00:08:34] KP: There's definitely a lot of other places if I was passionate about AV or just ML in general. AV is a long pipeline. You need the training you described and so many other pieces. Do you have any thoughts on the way the tooling is evolving? What are the big opportunities in data infrastructure today?

[00:08:51] LMB: Yeah, absolutely. I think we're seeing – If you look at where a lot of venture capital interest is going, we're seeing a lot of different categories that people now recognize are necessary both for AVs and for other ML applications. So you have the simulation, the synthetic data. Basically what this is, is can you bootstrap the training and testing of these models with data that isn't just from the real world labeled fully by a human, right? So can you like create fake data in some way to train or test things more quickly? A lot of interest there, applied intuition.

Now Scale is developing and offering. There's a lot of tooling around robotics deployment, like fleet management, debugging. How do you process all these logs very effectively? Because all these robots, especially AVs, generate a ton of data from all their different sensors. And so I see sort of startups starting to come into each one of these spaces and not only selling to AVs, but selling to other companies. So that's kind of broadly what I see more on the sort of, yeah, AV adjacent side. But then you have this whole category now of software, of ML infrastructure, that's being used by a variety of companies to either make their existing ML efforts more effective or kind of jump start their ML journey. And I can go into more detail there if you'd like.

[00:10:20] KP: Yeah, I would love to. Are you seeing any industry leaders emerging?

[00:10:24] LMB: Yeah. I mean, we have now kind of multiple examples of ML infrastructure startups that have gone public or are very mature. I mean, you have, of course, Databricks, and UiPath, and C3 AI, Scale, DataRobot, all of these companies that like have kind of found a smart wedge. Maybe it's around an open source. Maybe it's around some sort of hair-on-fire pain point and been able to expand across different types of companies and use cases.

And I think there are still though – Because ML is so new and the vast majority of companies that could use ML don't, there's still kind of categories where I wouldn't say there's a consensus tool yet. So I’m coming out with a blog post about this relatively soon, or probably by the time this podcast is posted it might already be out, hopefully. But how do you kind of broadly divide the opportunities that I see into five categories? One, how do you make just basic workflows easier to do cheaper, faster? Like right now, for example, imprints, which is just the act of an ML model doing predictions, can be extremely expensive depending on sort of how you set up the software and hardware around it. How do you democratize access to really fast, cheap predictions? That's one area I’m really excited about.

Sort of the other categories, I have a kind of a view that there needs to be more collaboration between ML engineers and the rest of a company. Otherwise you get these ml models built in a silo and they potentially aren't as useful as they could be because they don't have the context that an engineer and operations person uses. So there's this whole category of tools now to quickly allow ML engineers to deploy apps so that they can share their results very quickly with the rest of an organization.

I’m also extremely optimistic about – And this is a kind of a newer category, but now you have these companies who realize ML is important, who have seen success with early prototypes, but now they want to scale out to hundreds of models or they want to make sure that things don't break in production because they're for mission critical use cases. So how do you make that sort of production cycle more robust? Like how do you stress test ML? How do you monitor ML? How do you trigger retraining automatically or set up a more complex inference pipeline?

A lot of these big companies like Google and Amazon have internally, but the vast majority of startups that are using ML don't. So can a startup provide infrastructure to really scale ML in production across multiple use cases? And then finally, we've already talked about synthetic data simulation. Super important, since the data is just so costly and so necessary for deep learning. And then lastly, and this is kind of a little adjacent, but let's say I'm a company. I don't have an ML engineer. Maybe I just have a regular full stack engineer or maybe no technical people at all, but I have clear use cases where ML could be helpful. Like let's say I want to optimize conversion rate. Or I want to optimize like – Or I want to do better forecasting or something like that. Can a company come in and give me the power of ML but without me having that ML engineering context?

[00:13:46] KP: We've seen a growth in the number of startups and tools coming out, and maybe some of the artwork of putting together a company is stitching all of these together in the right way to have a full-blown ecosystem. Is that the way forward? Or do you think they will emerge some IKEA brand, put everything all in one standard way, that will be doing ML ops in the future?

[00:14:10] LMB: Yeah, I think that is an excellent question. I mean, right now, we're definitely seeing the former, I like how you describe it, of stitching together lots of best and breed point solutions. I think there's companies that are vying for that sort of IKEA all-in-one, whether it's the cloud providers. AWS SageMaker kind of comes to mind. Or you see Databricks expanding. You see Scale expanding. So you do see these companies that want to be your one-stop shop for all of your ML infrastructure.

However, I mean, if we maybe take a step back from the ML infrastructure and like we look at data or we look at performance observability like for regular engineering, you have companies that have wanted to own the full stack for both of these too. But in practice, we're still – And these are more mature sort of industries. We're still seeing people use a ton of, for example, different external tools set up ETL, different external tools to monitor their systems, like a multitude of them.

I think ML is kind of one of the least mature sort of places of developer tools. So it's going to be a really long time before we see a massive consolidation especially given all the excitement or funding for these point solutions.

[00:15:30] KP: When a company is going to adopt a new tool, something in their dev stack for either, I don't know, data processing, model training. There're so many different points here. I’m wondering if you could elaborate on the collaboration aspects. Generally speaking, who are the decision makers and how do they arrive at picking the right tool to suit their organizational needs?

[00:15:51] LMB: Yeah, I mean that's a great question. So it really kind of depends on – Yeah, I mean, I see a lot depending on the sort of particular point solution that's being considered. So it's interesting. I see a lot of bottoms-up adoption when it's something that is an open source tool or very easy to get started with. So like a tool for ML engineers to quickly prototype an application. Most of these tools are geared towards the ML engineer, finding it, being able to experiment with it very quickly and then eventually hopefully enough ML engineers in the company start using this tool and then it makes sense for them to get an enterprise license.

But then you also see companies doing top-down. So going through like an executive, an engineering, a CTO or something like that. The companies that, I think, have to be successful, it can't just be one user using them. It has to be a total either rip and replace or organizational tech stack redo. So a lot of, for example, the monitoring companies or the data infrastructure companies, they need sort of total organizational buy-in. So they might go in through that persona. So it's really tricky too, because that persona, the more executive could be removed from the problems on the ground and not realize which solution is actually more helpful? So it's really tough to kind of get the right balance of how do you prove to engineers this is going to be helpful for them but then get the right buy-in from the executive who ultimately has the power to sort of allocate the budget. It's tricky.

And then let's say that there's a large data component to whatever ML infrastructure tool there is, then you're dealing with data engineers, data analysts, head of data. And I think people are realizing now that deep learning is going to be sort of the future of ML. I think at one point there's perhaps more skepticism around that. In deep learning, an essential component is really great data, clean data, labeled data. So it's really intertwined.

[00:17:57] KP: Well, one of the, I guess, promises if you want to be optimistic about deep learning is to think that it'll replace a lot of what maybe a data scientist or an ML engineer does today. If it can do all the feature engineering and data cleaning on its own, then that person can achieve greater things or be replaced depending on how you want to look at it. What's your vision for where things are going in industry? Are we always going to have someone fine-tuning in a lab somewhere working on these models? Or can that really be more industrialized?

[00:18:29] LMB: I definitely think there's a ton of room for it to be industrialized and a lot of stuff to be automated. Right now, when you're an engineer, an ML engineer, you also – Judging from my time at Scale, you have to know developer operations, you have to know data engineering. You have to know all these adjacent fields that are not the core of what you were trained on, which was how do I really effectively train a machine learning model? So yeah, there’s just so many things that could be automated and that I think are great opportunities for startups to go after.

[00:19:00] KP: When you're looking at potential investments, I mean, if they're data companies, they're entering arguably a pretty crowded space. There's no shortage of data companies out there. But it's also a growing demand. Are you looking for companies that are finding new markets or do they have to compete for market share when they're entering something?

[00:19:20] LMB: I’m definitely open to both. I think this sort of core thing that I’m always looking for regardless of whether it's a new market or they're taking a new approach to an existing market even if it's quite crowded. What is this company's moat going to be is kind of the question that I always come back to. Is it a new category? So potentially they can brand themselves as sort of category creators and like develop a lot of sort of thought leadership in the space, or perhaps it's an open source software and they can create a community and find ubiquity, and that could be a moat.

I’m open to sort of that approach. And then also if they're going to go and enter a crowded market, there has to be some sort of story about why the rest of the companies aren't doing what you're doing. Why is what you're doing challenging and different? And ultimately kind of more strategic, if you do want to, like, for example, become a data platform and you don't just want to be a data point solution. So yeah, I’m quite open to sort of investing in any sort of market dynamic, but some are definitely trickier than others. And having a strong strategic roadmap is essential.

[00:20:33] KP: It seems like any data company these days has to maybe stay up late at night worrying that one of the cloud providers is about to release their solution as some check box feature. How do you consider investments and different products as you're considering, “Hey, could the cloud provider just offer this and even make it harder for people to get to this new upstart because they've already have all the buy-in from organizations?” Yeah, what are your thoughts on cloud as a threat to innovation?

[00:21:02] LMB: Yeah, a great question. And when I joined Scale I get asked this a lot. Mechanical Turk is there. This problem is already solved. Why is a startup even trying? And I think what it boils down to is these cloud providers have an excellent understanding of the things that people want. And their distribution is second to none. Their marketing is great. But when it comes to actually using the cloud providers’ products, I think some are excellent. And then some have clear problems when it comes to developer experience when it comes to cost, incentive alignment, especially if it's something that would make sense on sort of a multi-cloud architecture or something like that. So there are clear sort of pain points even if a cloud provider offers a seemingly exact replica of what a startup is offering.

So I’m actually very – I mean, one of the sort of investment theses that I have personally is we've got these cloud providers with all these products, some of which I had never heard of. People have probably not heard of their names, but they're massive businesses. And the people that are using these products are using them more out of necessity than anything else. And if something came along that provided a better developer experience, a better community, potentially could be very compelling for them to switch. So I’m not particularly bearish if the cloud providers already have something or offer something that no startups can be successful in that area.

[00:22:27] KP: Some applications of machine learning, automated vehicles are a great example, they require real-time solutions. You have to make split-second decisions. There are other cases, I know some fraud analysis, and like loan approvals can work this way where a company might be happy running a batch of machine learning inferencing overnight because they don't make decisions at the same scale. There're a lot of use cases for machine learning. Should we expect them all to be on the same platform?

[00:22:57] LMB: Yeah, I mean, not in the near term, for sure. I mean, the sort of infrastructure of the use cases that you enumerated right now is just so different. Like if you look at companies that are doing them, real-time machine learning I think is something that the vast majority of people aren't even doing. And so there's sort of newer startups where that's a specialty that are trying to solve the pain points around, yeah, acting on data streams or doing lightning fast inference. Yeah, it's a different set of challenges than if you were doing a batch workflow.

And quite frankly, with companies, I think you can only have a – The whole point of focus is that you only can only have a few focuses at a time even if you're a huge company. And yeah, just right now, for example, real-time ML doesn't seem to be a focus of a lot of existing companies. And so that's one area that I'm particularly excited about, whether it's streaming or whether it's other sorts of infrastructure. Yeah, I don't think there will be massive consolidation for all those use cases at least in the short to medium term.

[00:24:10] KP: Yeah, streaming has been growing in popularity, at least in my anecdotal observations. Are you seeing that or any other trends that are kind of contemporary and interesting to follow?

[00:24:21] LMB: Yeah, that's definitely one of them. I mean, I think other trends, I guess there's more like basic stuff. I mean, with developer tools in general, I think people are now quite accepting of a few things that I think has been relatively recent. So like, one, open source can work as like a company model. I think that's no longer like a contrarian opinion that cloud is inevitable and that also we're not like fully there yet. So how do you create tools, whether it's very basic monitoring tools, or more complex ML tools that play nice with cloud?

And then, yeah, I mean, deep learning, as I kind of said earlier, realizing that that's sort of the dominant paradigm and how do you make sure that the data is there for that? So those are kind of like general trends that I'm seeing in this sort of broader dev tools space. But yeah, if you're talking about ML, real time. I guess we already talked through synthetic data, testing, collaboration. And yeah, I think we're still very early in – If I wanted to create infrastructure to like spin up something in ML very quickly, like that is not a solved problem. And if you ask an ML engineer at let's say the 20 hybrid growth startups, how they set up their ML stack? I can guarantee you that you'd get different answers and you'd probably get people who set up the stack multiple times because there was a problem with it originally.

Now, at least with data, it's like if you want to set up ETL, you kind of know. I'm going to be Fivetran and Snowflake. And that's a pretty good place to start. With ML, I don't think that there's a consensus on, "Okay, I'm going to use these sorts of tools."

[00:26:11] KP: Yeah, in my mind, I feel like software engineering in general is the furthest ahead in maturity. They've developed CI/CD procedures. And everyone's kind of trying to copy that model, although it's not a perfect analog for machine learning. Data engineering might be further along. As you'd mentioned, there are some common tools, and processes, and best practices, and standards. Are we going to get there on the machine learning side? Or is the fact that every problem is a little different going to mean that everyone's kind of a special case?

[00:26:41] LMB: I think we'll get to at least something that looks closer to the data engineering ecosystem in the next five to ten years. I think there's a lot of sort of promising open source projects that sort of – At least I'm optimistic. At least one, if not a few of them, will become very mainstream on how do you just quickly, yeah, train and deploy ML models as an ml engineer with minimal sort of overhead from having to deal with other DevOps or infrastructure concerns.

And then I do think you'll see like more popular ways of non-technical or non-ML engineers being able to use a ML. I think right now it's very early. But yeah, five to ten years, I do think the ecosystem will start to mature. You'll see more proof points of people being really successful with certain solutions. And, yeah. But then I almost imagine, ML at least is a little different in that. There's just so much research that can happen. So if for some reason real-time ML really takes off or a different type of ML, such as not deep learning, but single oneshot learning or something like that. There's some chance that that happens and then we're back to earlier maturity when it comes to the whole stack. So I'm excited. I'm very excited to see things play out, but I'm sort of hopeful given all of the concrete proof points of AV, of self-delivery of robotics in general, of these big companies using ML very successfully to recommend things or to forecast things that we're on to something and that the majority of companies can also get a lot of benefit out of machine learning.

[00:28:30] LMB: Yeah, the tooling for ML has advanced pretty quickly. And I agree, I think we're headed towards a place where less technical contributors can help. Maybe the ultimate horizon for that is some really good no-code tooling. Are you bullish or bearish on no-code?

[00:28:47] LMB: I'm bullish. I think that it's just much harder to build, right? So if you're dealing with someone who does not have an understanding of what's going on underneath the hood, you have to just design it in a way such that things aren't going to break and become unreliable for them.

I mean, I'm more optimistic about low-code than no-code. So I think it's more reasonable to say, "Okay, now, a software engineer, or a data scientist, or a data analyst can use this ML model with minimal – Maybe they just need to know SQL. Or they need to know how to operate in a data warehouse or something." I think that's more likely to happen in the super short term. But, I mean, at some point, I do think there will be sort of these bespoke use cases where everyone's using ML to do X, Y and Z, and you can package that in a way with a sort of interface for non-technical people, perhaps one that can be – The best tools, they accommodate many types of users. So maybe it's no-code until you need it to be low-code depending on the person that's interacting with it.

[00:29:55] KP: Data infrastructure is a crowded space, lots of options. I think some of the entrepreneurs who aren't intimidated by that and want to create their own product must see that the opportunity for like a lightning in a bottle moment where you just have the right product market fit and your tool could be so useful to a couple of companies that overnight uh you've scaled up and you've really got something there. Have you seen any success stories like that in either your portfolio or companies you're profiling?

[00:30:26] LMB: Yeah. Well, I mean, I don't want to continuously talk about Scale. But it's definitely the success story that I know the most intimately. But yeah, I mean, I think a few things made Scale a really sort of successful and sort of the strategy was really there from day one, which was you want to find a problem that is hair-on-fire for somebody. And in this case, it was data labeling. The majority of ML companies, the data labeling is managed by an ML engineer and is literally someone who likes to spend their day training models now managing a workforce of tens, hundreds of people and teaching them how to label things accurately. And it's an operations problem.

And so like that person is just so eager to get that part of their workload out of their day that they're willing to work with a third-party startup even one that at the beginning was very small and pretty unknown. So I think it's all about finding like these sorts of hair-on-fire problems. And with the sort of caveat, maybe it's a hair on fire problem for only a few people, but they're willing to pay a lot of money to solve it. Or maybe it's a hair-on-fire problem for many people, but they have lots of options and/or some other barrier and aren't willing to pay much for it. So I think, yeah, kind of figuring out where that key pain point is that you're solving. Why people want you to solve it? That's what leads to these sort of breakout successes like Scale and like some of the other companies that I mentioned earlier.

[00:32:06] KP: What does it take beyond the right product? Maybe team-wise, who do you need on your team to really build up that sort of success?

[00:32:14] LMB: Yeah. No, that's a great question. I think it depends a bit on your strategy. I mean, if part of your strategy is I'm going to build an open source community, definitely someone – I think this is becoming very hot recently in developer relations, somebody who will educate your community, who will write content. We're seeing so many really successful examples of people having really strong content or community presences. Let's see. And then if you're doing something a little bit more on the other side, the sort of top-down enterprise approach, then at that point you've got to build a sales team. You've got to figure out your marketing strategy just sort of depending on your contract value and how many contracts you expect to sell.

But, yeah, and then I think another thing that I always tell people that's maybe a little counterintuitive, but regardless of the type of company, I think hiring a designer sooner rather than later is almost always a great choice. I think people maybe wait a little too long to hire a designer. But one thing that makes a lot of these companies very successful is if they can develop a really strong sort of engineering fashion brand, if you will, around what they're offering and having a designer with a really sort of cohesive vision of how your website and your product, like your product sandbox looks and feels, is just as important for a dev tools company as it is for other types of companies.

[00:33:44] KP: We've talked about a few areas you're excited about. Is there anything you're willing to call out as overhyped?

[00:33:50] LMB: Overhyped. I mean, there's definitely some crowded spaces, but I don't necessarily want to say anything is overhyped per se. I don't know. I'm pretty optimistic that the majority of categories in developer tools, in data, in ML are kind of going after real problems just from my experiences as an engineer. But yeah, I mean, obviously there's other fields that I don't know as much about that seem overhyped at least to me. But I'm not the expert to be talking about anything, for example, in crypto or something like that.

[00:34:31] KP: In, I guess it would be the mid-90s, there was this great transformation taking place and businesses were moving to the Internet. And you could have a lot of success just by digitizing. I don't know if you would agree, but I feel like we're in that sort of moment for ML. It doesn't solve every problem, but every industry should be having a serious look at how they could adopt machine learning. Do you agree? And if you do or don't, what do you think are the major opportunities for ML in the next five to ten years?

[00:34:59] LMB: I agree. I think very frequently, depending on the industry, it could be at first though – The 80/20 might just be getting your data clean and using it in some more simple model than some really complex ML deployment. I think that might be sort of the digitizing moment. But yeah, then there are other industries where perhaps most people have at least their data being recorded and stored and some very basic predictions that ML could be that sort of 10X improvement that they really need. And yeah, I think a lot of successful companies will just be targeting one major industry, one sort of vertical application. Yeah, I mean, sort of at Founders Fund, we have multiple portfolio companies that use ML as part of their core product and are very successful. And then, sorry. What was the second part of your question?

[00:35:50] KP: Opportunities. Next five to ten years, where can AI really make an impact?

[00:35:55] LMB: Yeah. I mean, let's see. Already kind of talked about the infrastructure, but when it comes to, yeah, more sort of ml applications. I mean, we're obviously still early in the rollout, a lot of these sort of more flashy solutions like self-driving cars and the robots and things like that. But where I'm excited to see ML being used a lot more, definitely in medicine and healthcare. I think we're starting to now have the tools where companies can use sensitive data. So whether it's through synthetic data, or federated learning, or different ways of anonymizing the data, or sharing it in an encrypted way, I think that unlocks a lot of data. And so now we're going to be able to have health tech companies who use ML to like help doctors make decisions or help patients have a better view of their total health. So I think that that's super exciting, especially if you consider maybe places where there aren't enough doctors or there isn't enough expertise. Can we use ML to help provide better healthcare?

And then I'm, yeah, excited to kind of see the sort of automation of rote work in a lot of industries. I think RPA, or robotic process automation, has been kind of proven as a category in recent years. But still there's a long way to go. Like there're still a lot of people doing data entry in their day-to-day. Or doing sort of repetitive tasks, whether it's lawyers, or accountants, or anybody that's interacting with documents or a computer screen, can RPA sort of evolve to taking on more and more of that work and leaving time for the real core decisions that they have to make every day? I'm excited to kind of see that play out as well.

[00:37:52] KP: Well, the Software Engineering Daily audience is pretty diverse, but everyone is – A software engineer or something adjacent, maybe a data engineer something like that. And in my anecdotal survey, people like that are also kind of wannabe entrepreneurs at heart. Everybody sees an opportunity to improve something in the process. We've got to have at least some listeners with really good ideas. And let's also trust they know how to build the idea. Those are two important criteria, but not necessarily enough. What else does someone have to do to be in a good position to collaborate with someone like you if they want to seek funding for a venture?

[00:38:26] LMB: Yeah. I mean, I am extremely open to collaborating with anybody in the audience with ideas in this area. I am very responsive on Twitter or email. But yeah, this is the stuff I really love thinking about. And yeah, just kind of reaching out and, yeah, letting me know how I can help and what you're thinking about.

[00:38:50] KP: But what do you look for? I mean, you kind of mentioned the stages of investment previously. Are you looking – What sort of traction makes somebody appealing to you?

[00:38:59] LMB: Yeah, it really depends on the stage. and I have a few kind of like Twitter threads on this that give my general heuristics that I use. But I mean, at Founders Fund, across the investment team, what gets us really excited is just really great founders. And obviously there's many ways that you can define a great founder. But it's someone who is sort of uniquely positioned to solve a problem who really has thought strategically about the way that they're going to solve it. And yeah, I think it can really depend the traction that we sort of want to see depending on, yeah, the history of your company. Like why you're trying to raise the amount that you're trying to raise now? So I don't want to say any sort of heuristics that are disqualifying, but I do have some that are like good references in case you're unsure on my Twitter.

[00:39:48] KP: Very cool. Well, Leigh Marie, give us your Twitter handle and let us know anywhere else people can follow you online.

[00:39:55] LMB: Awesome. My Twitter handle is lm_braswell.

[00:40:04] KP: And we touched on a lot. Is there anything you think we missed or should have gone back to cover in more detail?

[00:40:10] LMB: Not really. Yeah, we did touch on a lot. Super fun conversation, great questions, and I really appreciate you having me on the show. Thanks so much.

[00:40:18] KP: Leigh Marie, thanks for coming on Software Engineering Daily.

[00:40:22] LMB: Awesome. Thank you so much.

[END]