

How Engineers and PMs should to collaborate on Evals

33 MIN · YOUTUBE · [HTTPS://WWW.YOUTUBE.COM/WATCH?V=7AXL9P0JHCK](https://www.youtube.com/watch?v=7AXL9P0JHCK)

<https://www.youtube.com/watch?v=7AXL9P0JHCK>

SUMMARY

The session focuses on the collaboration between engineers and product managers (PMs) in the context of AI product development, emphasizing the evolving roles and responsibilities in this space. Aman, an AI product manager, discusses the importance of labeling data, the need for collaboration in writing evaluation prompts, and the necessity for PMs to engage deeply with data to ensure product quality.

- AI product management involves three main roles: core product management, platform building, and AI-powered PMs who utilize AI in their workflows.*
- The collaboration between PMs and engineers is crucial, with PMs needing to understand data labeling and evaluation processes to enhance product outcomes.*
- There is a tendency for team members to pass the responsibility of labeling data, leading to a need for clear accountability in roles.*
- Effective AI product management requires PMs to be involved in writing evaluation prompts and understanding the data that informs product quality.*
- The quality and quantity of labels are critical; PMs should work closely with AI engineers to determine what is statistically significant for evaluations.*
- Companies that excel in AI product development often have technical PMs who understand data needs and maintain a well-scoped product focus.*
- Engineers are encouraged to adopt a product mindset, engaging with customer feedback and product insights to enhance their contributions.*
- Continuous iteration and collaboration between PMs and engineers are essential for aligning product performance with business metrics.*

session is about how engineers and PMs can collaborate. Um Aman is a AI product manager at Arise. He can describe what AI product manager is. We kind of had a lesson earlier that like went deep into that but we can briefly go over it. Um you know there's a lot of questions involved like who does the eval particularly like who's writing who's like labeling things. When you're part of a company, people like to point fingers like, "No, I'm you know, it's like like just pass around the labeling as a hot potato. No one wants to do it.

So, who's doing it? Is it the PM? Can you give the labeling to ask to your PM?" These are these are interesting questions and uh Aman and and I are going to try to come to an answer. Awesome. Yeah, this is super exciting and good to see you again, HL, and and a lot of folks here definitely see some familiar faces. Um, cool. I can I can do like a really quick uh sort of, you know, maybe maybe it's helpful just to do a bit of like level setting on the PM role in this AI LLM geni sort of world a little bit. Um, and then we can hop in from there. Hmel feel free to

like ask any questions and you know we have some some uh some good debate I think around so uh so for the folks that you know that that um that I I might be new to um my name is Aman I I lead up uh our product team at Arise uh we're an agent eval agent and AI application eval and observability company so you know we actually work really closely with HML he gives us really helpful and useful feedback on our product all the time.

So, we really really appreciate like, you know, people that are sort of getting their hands dirty and actually trying to build improvement into their their workflows. Uh, and just by way of background, I started my career working on self-driving car eval. So, I've been in the space for a little bit of time building and shipping AI products. Um, so, so pretty pretty familiar on both the the infrastructure side as well as building end products for users. Um, I thought it'd be kind of fun just to make this like less of a slide uh, show presentation and more of a whiteboarding sort of session. Uh, and so we can sort of like think of this as like marking up a whiteboard live. And if you guys have questions as well, just feel free to drop them in. I'm happy to try to to kind of take them live as well. Um, HMEL will sort of be making sure I'm asking I'm I'm going the right direction here. You can see him in the whiteboard, too. Um, so this is like a very quick just brain dump of like when you hear the term AI PM. This is kind of what what I think about. There's sort of three flavors of of AI product management. There's the core product itself where AI is a part of the product. Um, and that's really where a PM is thinking about like what should the form factor be? How do we package this technology up? What should the end product look like for a user? Uh there's sort of platform builders and this is, you know, within an organization or even it can be overlapping with the product person and product person, but they're thinking about infra managing costs, managing reliability, you know, do we have the right model? Are we picking the right model for the application? And then there's the the AI powered PM.

Again, this is like there's it's not really mutually exclusive. I I sort of view these as like almost you know sort of overlapping ven diagram of someone who's using AI in their job and this is where a lot of the like prompting guide you know prototyping with AI uh maybe even writing code sort of comes into the picture. I bring this up as sort of like a bit of a reflection on the PM role. Many people have worked with product managers in the past and um even me just being in the role I feel like the expectation has really changed on what product management means. Like back in the the old days a little bit, this was like the the diagram people would show.

It's like product looks like this and then engineering does this and design does this and then sales and marketing are off doing their own thing. And now I feel like we're living in this world where these roles and responsibilities are more and more overlapping every single day. And instead of thinking about roles and responsibilities, I sort of sort of try to push people to think about like skills. Like what is your skill? What's your niche? the area you really care about, the area you're really good at. And if it happens to be product, you're not going to get away with, you know, just being able to do your own thing. You have to be able to collaborate with AI engineers and sort of vice versa if you're an engineer.

Thinking about how to get the most out of your AI product manager. That's really what I'm hoping to impart from this session a little bit. So that's sort of just to set the tone, the context of like where product management fits into this new AI uh development life cycle. I put a funny little I made a little funny edit to it. Do do you think uh you know this is there is it like we've seen this diagram before in like data science you know there's like

uh statisticians and software engineers and you know I forgot what the third one is and then in the middle was like this unicorn data scientist. Is there like are you trying to say like hey we're not you're not really saying there's a unicorn. It's like it depends like based on like your other graphics, you know, you have to like kind of choose where your strengths are sort of I another way to look at it is like I'll probably come back to you in a year hmel and draw this diagram like this again and be like actually what we really want is like a full stack AI product manager that can kind of do all of those things and then if you were spiky in an area great that's your strength but I don't think you're really going to get away without being a little bit of like thinking about all of these things to some degree as a PM. So I'll give you like another example. We were just at I was at AIG World Fair last week and it kept coming up like I gave this whole talk on like evalms and the Q&A 30 minute section at the end was like you know everyone really latched on to the point where I talked about like how our PMs contribute code back and a lot of people are like wait you guys write code you guys write labels you're writing eval. So, I think there's a lot of a lot of folks that are getting sort of, you know, more comfortable with the fact that product management is going to look a little bit different now that um the barrier for creativity is sort of like, you know, actually creating something has gone down. Um, makes sense. Yeah, but good point. I like that. I'm going to keep that in there. Um, great. Awesome.

Um, so I think maybe just going on a little bit in this journey. Um, I thought we could just talk about like what things kind of look like today. Like I really liked the I I took a look at the course reader and I love that like vibe coding is pulled up in there. And I thought I would kind of maybe just express the three phases of building AI products that we often see. So for context like from where we sit as a company and if you haven't if you haven't heard of us um I'll just like briefly show uh the like landing page just so that there's some context like we work with some of the leading companies out there that are building AI products like Reddit, Roblox, Booking.com, largest travel site in the world and what we often see is there's like a spectrum of where people come to us and say hey we should maybe start think should we start thinking about eval is usually like a very common question and uh one just sort of example workflow is like I would call this like level zero level one like you're a startup or maybe you're building something internally for your users and you just want to move fast right the vibe coding workflow maybe you're starting with like building an agent or AI product like we can build something over the weekend now with crew AI or langraph you take a look at the outputs you kind of do a like looks good to me and then you hit shift And what ends up happening is like, you know, usually that product actually does a pretty decent job for the most part for like some of your users, but then you're going to have some users that complain, right? And this is like the internal this, think of this as like an internal tool or code that's a little bit more um you know, it doesn't necessarily have to have staying power, right? You're okay with shipping something that's not great. This type of workflow works if you at the end of of shipping something you're not worried about like damaging your brand reputation or you're not worried about you know regulators coming after you because you accidentally leaked some customer data. It really isn't a realistic workflow for enterprise um you know sort of product builders. But I want to throw this up as like a lot of people who are vibe coding building stuff with like replet vzero shipping something quickly like the the thumbs up thumbs down vibe checking the outputs like I think that's great for like internal tools. So there's a certain category of tools where this makes sense. I think the fact that you're taking a course on eval means that you're probably further along on this workflow and you might be thinking about where eval sit stack and in this case when you when you're building the the agent or the AI product and you're writing an eval question that comes up is do we trust the eval? And uh we get asked a lot like do you guys have offthe-shelf evaluations for like toxicity or hallucination? And if you give us that eval,

should we be able to trust the output uh from this this eval system?

And that's usually where I think things start to get pretty interesting because that's what involves, you know, kind of forcing you to like look at your data, look at the outputs uh a little bit more. So I kind of call this like the okay, we wrote an eval, we we tested something, but we're still at this point where we're like, do we trust the eval? And this is really where HL and I have like have had a ton of debate, which is who writes the eval who writes the labels that you're using to actually check those eval. Um, so H, I'll pause there. I don't know if you want to add anything else more here, but we can go a little bit deeper on this.

Oh, you're muted by the way. Um, so I think the undercurrent is everybody wants to pawn the labeling on someone else. Like if we were to announce today or declare that PM should write the labels, then everyone would be like, "Hey, like I'm not writing the labels. I'm going to give it to my PM." And that might kind of result in some sort of backlash from PMs. And they might say, "Hey, like I'm not the person necessarily going to write the labels."

Like I don't know how they would feel about that. Um it's it's interesting. And so I think this like historically labeling even like personally I love labeling and looking at data. I think maybe the students in this course will appreciate the value of that. But I think in companies and organizations like people are really busy for whatever reason that labeling tends to Yeah. It's like minimized in importance somehow. Yeah. Could I get a quick like show I know this is like a virtual one, but maybe we can do a quick show of hands.

like could you drop in the chat if you've actually like labeled data for an email like how many people here are just like rough you can say like yes me or whatever but I just want to kind of see like what percentage of your your audience is like okay cool I have and I'm the PM yes amazing okay so note made the scientist do it I think that that this like kind of what I was curious is like how many people have written labels versus how many people have pawned off and like what that ratio looks like. And I I guess what I'm getting at is like these two workflows are actually pretty interlin. Like I threw in PM here as like maybe someone who should be writing label, but um to be honest, like these two things are a little bit one and the same to me in that you can't really like the the label that you're writing should impact the eval, right? Because at the end of the day, if you're a PM, you are responsible for the end product and you're responsible for that end user experience. And so when you think about an eval system that contains an eval prompt, an LLM, and some data, you're ultimately trying to say, okay, maybe I can tweak some knobs on my LLM. Maybe I can pick a better LLM, you know, cost performance, latency, accuracy, tuning.

But the data that you're using to check that against that eval maybe that's a data set of like input output expected label expected output and label. This is this is really really important right because this ultimately determines what the goodness of your overall system is going to be compared to like you know like what that end product experience will look like for your user. And I think similarly the eval prompt which is defining what should be evaluated that I feel like is also a collaboration between your PM and AI engineering because your eval prompt is should be representative of your data. Um, I'll kind of show this this graphic here a little bit of like what a eval prompt sort of looks like just to kind of like you know I know I know everyone here is probably familiar with

this but when you think about setting the role providing context providing the goal and defining the terminology and the label these are all like natural language and you have to usually provide examples that fit into the eval prompt as well.

So deciding what examples what few shot examples go in defining what that label look like looks like you can't really do that unless you're labeling the data and you're in the data itself. So I would actually push back a little bit and say like you know as much as like the PM is is the right person to write the label they're also probably the right person to kind of help write the eval here to some degree. And so I think that there's like this this good kind of healthy tension of like maybe it's not as mutually exclusive and it's a little bit more collaborative in terms of a workflow where you know the PM takes on some of the labels and then the the AI kind of can help iterate on those. I would like I would like to see the PM I think the closer you can get the labeling to the domain expert the better and usually the PM is more of a domain expert than anyone else if things are going correctly and organized correctly.

Right. That's that's exactly right. And then and then at the same time, if the AI engineer is the person accountable for the end product delivery, then the eval prompt and the final prompt that goes into the system also like kind of rests on the AI engineer, but it also is a little bit of like some domain expertise should enter that as well. So like you should see, you know, a little bit of that like back and forth. I think in a typical workflow, we see this a lot. We see human, we see PMs writing labels. We also see PMs writing eval prompts and writing the end prompts um quite often. Okay. What I want to kind of where this kind of leads me is like I'm not sure if we're going to get a packaged up answer of who writes the labels and who writes the prompts necessarily. Though I do think that like leaning towards PM should write labels is probably a fair assessment. Like the PM needs to be in the details of the data. the PM needs to be in the details of the eval but kind of who does what and what that split looks like. Your mileage may vary at your company. What I would urge people to do is think about these three questions which is when you go back and you leave this course and you move on to like writing and building AI products. I kind of ask people how many labels do you think are good enough and for each example how many labels should you have? And so this is a really important question because like a lot of times people will come to us and say like do we do you know is 10 labels enough 100 labels a thousand labels and I think that this is a question that PM should be working with their AI engineers on which is what is statistically significant for you to know that this eval is good enough and what you know does that matter per row?

Should you have agreement between all of your subject matter experts or are you okay with disagreement in your human labels? So think of this as like what's the quality and quantity of the human labels that we have. The second is what happens when the eval says that the output is good but the human label disagrees and what happens during this tie break scenario. And you'll kind of see this when you actually get your your your product into production. It's going to happen a lot where your eval system is going to say this went well, but your human in the loop or your feedback from your end user is going to say this was a bad response. So, who takes a look at those examples and what happens and who's the tiebreaker there? That's a that's a really important question that comes down to like your team. And then I'd say last but not least, what happens when you linger on that for a second?

One of the pieces of advice we have in our course reader and otherwise is wherever you can try to anoint a benevolent dictator. And um you know that benevolent dictator is someone again who has the most context and the most domain expertise ideally when it comes to doing the tiebreak and maybe you would be an advocate of the product manager being the benevolent dictator. I I think it's I think it's important to look I think uh there should be a process in defining who that person is. I I think that's what I'm advocating for here which is like whether it's democratic or not I don't know what the process looks like that depends team by team but there should be like think of for each of these questions who's responsible and who's accountable and I think in the same way like human labels maybe the PM is responsible right but like who's accountable for the end quality of the eval it's going to be the AI engineer so maybe the AI engineer needs to say I need a thousand labels you know but the PM has to go figure out how to get those labels Like that's the split in responsibility and accountability. The same goes for what happens when the eval is good but the human disagrees like if the AI engineer is signing off on the eval but knowing that there's going to be you're going to have to go back and iterate on that like who's responsible for making that call and then who's accountable. And then the same same goes for what happens when the eval is good but the business metric goes down. I think in this last one, I'm pretty confident the PM is going to be responsible if the product suffers, but I think that kind of just reiterates how important it is to get that eval right in the first place. And so like all of these kind of I would say like kind of show like you're going to have a product manager in the loop here at some point for each of these because it's ultimately going to come down to the business metric. It's you can't really get away from it. Yeah. No, this is really good. And we're so a lot of these things, you know, like figuring out why the eval is good but the human label disagrees or figuring out why the eval is good and the business metric goes down. That often requires a fair amount of data analytics. Uh what you know some people might call data quote data science. Yeah. Um whatever you want to call it. You know how does that mean that you need a data scientist? Does that mean that PMS are also doing the data science work or the tools there yet where data scientists can do a fair amount of data science work? I've seen a lot of variance in the skills.

Uh where are we at today in terms of expectations and skills of product managers and in this dimension or how do you feel it should be? It's a good question. I mean, I think we like I'm going to be a little biased because I'm kind of working on a tool that helps with this, but I'll try and like pull out some principles for that too for like you don't have to use this tool. I I I don't at the end of the day whether or not you use the tool I build matters less than like you walk away with the right principles. I think you ultimately you want a tool or a platform that shows you this single pane of glass between your eval metrics, your human labels and your business metrics in one place because you want to be able to plot those correlations. And whether that's a data scientist or data analyst that's helping you get the data in the right place and build those correlations, I think that depends if you have the resources for that. And if not, if you're a PM and you're ultimately responsible for the business metric, you might just need to roll up your sleeves, right? Like that that's part of this as well. It kind of comes back to like the earlier point like you you should be the bull in the china shop like breaking things down to make sure that you have the right data in place to answer these questions. And if that means you have to hire someone or you do it yourself, like I think that's Let's linger on that for a second. Do you mind the scrolling up to that? Yeah. Ven diagram like the ones where they kind of overlap heavily where it's like more about skills. Yeah, this one. Yeah. And so, um, actually I've been talking to Shrea a lot about this and one thing that she said that stuck with me that I thought seemed right to me is like, hey, it's not about job title anymore. It's just about, you know, it's about roles. Yeah. Or like you you might have to fill many roles, right? And this so this is not necessarily

about like this is not like a war it's not like some adversarial way people might think oh it's like PMS versus AI engineers like who's like some turf but I feel like everyone maybe like you know it's like engineers will have to do product management too right it's like goes in the other direction it goes in the other direction as well yeah it goes it definitely goes both ways like I I kind of view this more like you remember okay I'm dating myself here a little bit but like you know back in the day if you have like baseball cards cards, right? You're going to have like skills on those baseball cards, right?

Like it's not like a pitcher never bats, you know, like you have to be able to do multiple things on the team and there might be areas where you're stronger and someone else is stronger and you complement each other, but it doesn't mean that you never do the other person's job or vice versa. And I think that that's the level of like PMs, engineers need to be comfortable with this kind of overlapping territory, you could say, of like who does what. and and it seems like okay so with AI coding stuff and AI you know whatever um cloud code cursor it seems like a lot more tractable for a product manager to you know roll up their sleeves and do some of the stuff that an engineer was doing previously. What about in the other direction though like how can is there some tools or does AI help the engineer become a better product manager as well?

Absolutely. I mean I think um the the interesting thing here is like product management is in a way it's like talking to users synthesizing data and trying to figure out what to go and build. And the good news is that for a lot of companies that a lot of that data is democratized. So you have tools from the sales team like Gong which you can go and you know if you're building B2B applications you can go and listen to like what are your users saying you can build tools that help you get customer data insights like you know looking at session replay data.

So like these tools are are you know they're meant to be used by everyone on the team. In the same way that you know we at Arise we're trying to build a tool to help you take your eval data and correlate it to business metrics there's tools that help upstream of that which are take signals from the market and turn it into product insights and so I think so am I understanding correctly like okay so like in this course you know we focus a lot on looking at your traces looking at your eval quote data and what I'm hearing from you is like as a product manager you're already like you know you should be looking at data of other kinds like the sales data that's where you're getting that's like a lot of the the work is like synthesizing all of this like sales information customer conversations.

Yeah. Can I if I could double click on that one? It's like at the end of the day does it matter if your chatbot is friendly or accomplishes your goals if it's not actually doing the conversion that you want it to have? you know, and so like do you have the right data in place to say our chatbot is completing cases or it's converting users to a purchase and is that represented because that's not going to be an eval right like that that signal often comes weeks weeks later maybe days later and you you still need a way to correlate it back to a single session or some feedback and so PMs need to be thinking about how to get those signals back in and so do engineers because engineer an AI engineer would be responsible for making sure that we have the data and in the right place to answer that question. So that that's like if an if a PM comes to you and says like hey can we track this back to conversions to sessions like you might need to think about how that how that relates to your eval plot the two together in a way it's kind of seems unifying because you're again saying look at your data look at your data yeah exactly it's the same it's just what's the data in this case right and so um so actually that that kind of brings me to

maybe like one of the last points here um to kind of tie this all together which is let's say you have the right eval in place, right? You have eval you've kind of solved the messy in the middle of like who labeled is do we trust the eval metric and you have eval and then you ship you ship that product.

Well, guess what? It's going to work for more users than it did before, but you're still going to have users that complain. That's just how building products goes. You're going to have users that are like, "No, I'm I'm not happy with this product experience." That's what forces you to look at your data. And in this case, you have the same input, output, eval label, human label. But you probably have some business metric, too, here. And you're going to be looking at all of these and you're going to have a much stronger signal of like where things went wrong.

Let me just go ahead and pull this out a little bit. Right? So, you're going to have like the human label was bad. Maybe the business metric is going down. And then that means that you probably need to go back and iterate on your eval. It means that there's probably something missing here where the eval didn't catch something and you want to go back and iterate and improve on it further. And you're never really going to get I mean this is kind of a bit of a loop. But the good news is that as you kind of squash more examples where the the product didn't perform well, you're going to have that eval metric going up, your business metric going up ideally because these things are now correlated.

and when they aren't going up where they disagree, you have a system and process in place to go in and figure out how to iterate. So that's really what I think that for me a little bit of what the takeaway is is that, you know, you're you're really trying to pull people together to look at the same data, make sure it matches what your business cares about and the right people are there kind of asking these questions and answering, you know, kind of who's accountable for what.

Is there any companies that you think are doing this really well? I know it's kind of a hard difficult question because we're so early and you know and maybe we don't want to pick favorites. Um yeah, anyone anyone come to mind off hand? I think um I I can maybe like you know rather than like specific companies out there. I could probably list like what I've seen like common patterns of what those companies look like. And they're they're actually pretty they're they're pretty um like they they do really come to mind because they're outliers a little bit. So there's maybe two or three traits that come to mind.

One is you have PMs that sort of look more like this but more overlapping meaning you you usually have like technical PMs in the product because they they know what data they need to have in place and how to get it. And they're usually engineers that transition into product management. Though we're seeing a lot of PMs also starting to trend this way. But it's like you you have a strong technical AI product manager that knows okay here's how to pull this data together and have it correlate. So that's I think like a secret like a superpower. It's like sometimes I talk to these PMs and I'm like shocked that they're product managers because they're not just thinking about you know PRDS they're really thinking about code and that's a huge light bulb moment. I think another one is um a well scoped AI product. So often times like where you see like products in the market kind of you know sort of not do so well is they're trying to do too much and you you see people trying to throw more

agents at the problem like let me just try to like build more complexity into the system.

The simpler your AI product is the higher likelihood that you're going to actually have the right data in place to help answer these questions to at least get started. And that I think that's really where we're at. like you don't need some overblown abstraction or a crazy framework. Like start with simple principles and be able to track when things go wrong. I think those are usually like two principles I see really stand out for me um with strong with strong AI products right now, AI product teams.

Um last question I have is I think we have heard a lot about how PMs can be more technical. I mean not just from you just in general in the ether of like hey this is how everyone can start coding this is how everybody can like you know whatever what are the are there any like good resources or some starting point of how like engineers might be uh can train themselves to be like have better product thinking act more like PMs in this like way so they can kind of you know get towards this like overlapped ven diagram.

Yeah, I I think that that's a really good question. I I was talking to um we we were talking to Brian about this, right? Like in the in our last like lightning session, he kind of expressed the story of like when he was a tech like a lead, like a team lead for a product and he was trying to do it all. And I think uh you know, he was responsible for like going 0 to 1 shipping the end product. And where I'm kind of going with this is like Brian got really really far doing that until he just had too many customers and he couldn't really scale himself much more.

He was like I'm I'm talking to too many customers and I can't write code. I would urge engineers and tech technical leaders to think of themselves in that way which is like even if you have a PM on the team like offer or help split the PM's responsibilities but think of yourself as a product person and keep doing that until it actually starts to eat into the time where you can't do both jobs anymore and you have to like figure out how to scale yourself further because the more you do that or think that way like it's just hands-on sort of uh experience to be honest a little bit of you know you have to talk to customers you have to figure out what's real and not and you kind of develop a little bit more skill there. Um, I'm biased. There's a ton of like great resources out there like Lenny's podcast. Um, you know, Akasha is another great person to listen to. There's a bunch of these like product thinkers or leaders that kind of help put out content um that helps you like use the same vocabulary, but that's really more I would say that's like more for style points because it's you still have to do the job at the end of the day um to really learn what what product thinking looks like. And um are you still teaching a course uh in the future coming up? Can you tell us a little bit more about that? Yeah, absolutely. So, kicking off um also kicking off a Maven session or Maven cohort here. This one is um I I would kind of call it like a lot of this information, you know, is you could go to YouTube and like process it or go to like, you know, I I think you you've done an awesome job with the course reader of eval dive. This cohort is going to be more of a playbook for how to help you write your H1 or H2 AI strategy roadmap. So, think of this as like personal training versus going to the gym and figuring it out on your own. Um, so I'm going to in part at least show what we've seen those really effective teams do like what are the processes that they follow? What do they get right? Um, it's going to be a fairly fairly small cohort to start with. Uh but uh yeah, you know, feel free to email me for a a you know, discount code on this one, too. We'd be happy to to share that. I my goal is to build something that your employer is going to reimburse because it's just so obvious like you know this is valuable for you to learn. Um

and ideally build a community around as well. So yeah, that's the prototype to production. Uh the AIPM playbook is the name of the course. Um how should people email you?

Is that I don't necessarily how do you want to distribute that email or how can people get in touch with you for that? Yeah, so you can just email me at ammani.product@gmail.com and I'll throw it in the chat as well. Feel free to email me with any follow-ups or reach out to me on LinkedIn as well. I tend to respond to like anyone that reaches out to me. So happy to be helpful for anyone that's watching this asynchronous as well. like just reach reach out to me on LinkedIn and I'll drop we can maybe send out the Maven link as well um for folks that want to follow Async definitely u yeah it was really great having you this was really useful um definitely answered some questions I had so yeah thank you for coming on awesome thank you for having me and yeah let's get more people labeling the right data so that we build better eval products out there that's really what the world I want to live in as well yeah I love it. All right. Thank you. Cool. All right. Thanks. Thanks.