CapTech

Machine Learning 201: Data Scientists Aren't Data Engineers

CapTech Trends Podcast | Episode 20

Vinnie

Hello, and welcome to CapTech Trends, a place where we meet with thought leaders and subject matter experts to discuss emerging technologies design and project methodology. I'm your host, Vinnie Schoenfelder, principal and CTO of CapTech Consulting. And today we're focusing on machine learning, specifically how to productionize machine learning models and looking out a few years to see what's on the horizon. I have with me, Calli Rogers. Calli is a director at our Richmond office, and is a leader in our data engineering and machine learning practices. Welcome, Calli. So, kick us off, when you and I first started talking about this, you mentioned that some of your expertise is in the productionizing of the models. I know what that means. But I don't know why it's a challenge for most of our clients that are facing that, and specifically what you do to help close that gap.

Calli

Yeah, so a lot of our clients and a lot of folks treat their data science side as a business unit. And it's really interesting, because data science is really the cross between business and technology. And a lot of the data scientists just really struggle with the productionization side of it. You hear the 80/20 rule, where they're doing data wrangling versus data science. It's really not just data wrangling, we just use that term, it's everything, it's making sure that the model can be used correctly, it's making sure that it's running monthly, everything like that they're doing all of that by hand. So, it's really a struggle for them to be able to connect the dots and do the engineering side when they really are data scientists.

Vinnie

Right, so we have seen this a bit too, where data scientists are not data engineers, I kind of think, and maybe you can confirm or reject this idea. I kind of think that developers, programmers are closer to data engineers and vice versa, than data scientists are the data engineers.

Calli

They are. I actually come from the developer side of things - started as an API Dev and moved over. And I use more of those tools than - even in machine learning engineering, where we're

really doing the productionisation and honing in on that specific data engineering pieces – more than I use what the data scientists use.

Vinnie

Right. So, a lot of times people are expecting data scientists to do data engineering work, because they have data in their name. But it's pretty foreign in terms of their skill set, right? And it's a funny thing, when you said that they're doing the data wrangling, and they're doing stuff outside of what they should be doing. I think about it, a lot of other technologies that have gone through that same thing; like I used to do human resource management software, and just doing employee and manager self service, you know, putting stuff on apps and websites, freed up the human resource people, so they weren't doing the manual, routine commodity-type stuff, and it freed them up to actually be HR professionals. So, we see this in technology a lot where the grunt work gets in the way of the expert work. And then the machines are really good at taking the grunt work away. So, with programming, DevOps takes care of a lot of that, for us. And application development is your equivalent on the machine learning side.

Calli

There is, they actually call it ML Ops. And it really is just the amalgamation of DevOps specifically for data science. So, it's being able to do their feature engineering for them automatically. Scheduling their models, making sure that if there are any sort of dependencies, all of that is taken care of. There's actually built into Sagemaker, Google's vertex as well, has different pipeline things built in specifically to be able to run those models be able to continuously train them and everything like that.

Vinnie

Right, so for the for the less technical audience members, when we talk about DevOps or ML Ops. What do we mean by that?

Calli

So, DevOps is the intersection between development and operations. It's the automation and the

production of things, so code management deployment, and automated testing.

Vinnie

Yep. Okay, so what are the equivalents on the machine learning side?

Calli

Automated training, experimentation, being able to automatically split your test train and validation, data sets, it really is deployment and code management as well.

Vinnie

So, I know in the DevOps space on the development side, there's been good maturity in terms of tools and processes. Are people still sort of hand building their ML Ops or are there good tools and processes to follow?

Calli

It's getting there. I'm seeing a lot of things that we apply to data engineering, which are concepts we took from DevOps and turn them into Data Ops. I'm starting to see those applied to ML Ops. I've mostly recently been working in GCP and they have a phenomenal Kubernetes tool called Kubeflow that actually automates a lot of that stuff for them.

Vinnie

Gotcha. So, when we talk about helping companies productionize their models, we talked about wrangling the data and freeing up data scientists, but that, to me, freeing them up is more about allowing them to create more models. I don't understand necessarily like, what are the discrete steps that you need to help a company improve to get from having models to having models in production? Is it deployment at the edge? I'm trying to understand how you take a model and put it somewhere where apps are using it and using it real time and how those models get better over time.

Calli

Yeah, there are a couple different ways. There are really two buckets of models – batch models or offline models. And then there's online models, which is more along the lines of what you're talking about with the recommender engines. And you can deploy those on edge, as you just mentioned, which is really great. If you're in like low bandwidth areas, they have to be really lightweight. So, making sure a lot of the ops stuff will make sure that those artifacts are really lightweight that they only have the bare bones. You can also deploy them to endpoints, which is what I'm seeing a lot of recently. And they're just they become API calls, which is really neat to see. And the model just itself becomes an application.

Vinnie

Well, that's sort of where my head was where if it's a model in production, that there's an interface sitting in front of it that abstracts that complexity. So, I'm sending an API, something with parameters and I'm getting back some sort of response that I can parse and do something.

Calli

Yeah, especially for the online models. Offline models, less than the traditional sense of API and where you might actually just be calling the model artifact and doing like a dot run or something like that.

Vinnie

So, on the edge ones you're talking smartphones, tablets, IoT devices that are using these models. And I don't see there being a problem with scale, because it's happening at the edge. But the online models, especially if it's sitting behind an API, how do you handle scale and growing that elastically as demand for their services through the apps increases?

Calli

Honestly, if somebody is trying to do that, I'm going to tell them to use some managed services that they have available on the cloud. But the same way that you would do it with any application at that point, it's just a different artifact that you're serving up. If you're going to use a load balancer or anything like that.

Vinnie

Got it. So basically, it's running essentially as an application server at that point. Gotcha. What if you said it twice so far, and I can intuitively understand what you mean, but I'm curious to know more about continuous training. What are the parts of that working? How does it work?

Calli

Yeah. So continuous training is a really cool concept that I admittedly have only really recently gotten my hands into. Right now, when you're developing a model, a lot of times, you go through your exploratory data analysis, you find the data you're looking for. And then once you find the data that you're looking for, you start building your population, you start training it, and you'll have a subset that you train against. And then once you finish that, you have a model that's ready to deploy. That's if you're doing it all by hand; to continuously train, instead, what you'll do is you'll have a model that you can, you'll have a set of code that you can send different parameters into, and that can help train it. So, you'll have instead of deploying a trained model, you are deploying a trained model, but you are also deploying a training pipeline, and that training pipeline can be triggered to rerun, and then you're just updating your model.

Vinnie

I guess that's kind of thinking about maturity curve for models. And it seems like a good first step.

Calli

Yeah, I actually really like there's some documentation out there that Google has put together, they have level zero, level one, and level two ML Ops, and they consider level one to be you've gotten to a point that you can continuously train, so you can trigger your training pipeline to be able to keep working through it.

Vinnie

And what are the other levels?

Calli

Level two is where you've really got your CICD built in. You've got that test in there…

Vinnie

Define what CICD is again for our business.

Calli

Continuous integration and continuous delivery. That is, I want to make an update to my model. So, I make an update to the code and all I have to do is push it in the model will automatically test itself, train itself, make sure everything's good. And as long as everything passes and hits those green checkmarks. Then it's going to push it if you've got one of those online models out to that end point. If you've got an offline one, it'll update that artifact.

Vinnie

So CICD is a part of ML Ops.

Calli

It can be yes.

Vinnie

Okay. And then you spoke of monitoring and model drift. Is that part of level two also?

Calli

It is. So, model monitoring and monitor model drift is really watching to make sure that your model isn't skewing there; there's always bias in data science, there's always going to be bias in data science. That's just the nature of humans. We can't write something that doesn't have bias in it. But can we avoid the model from drifting too far in one direction and having too much bias and watching that. Likewise, watching features to make sure they're not drifting away from where we originally trained the model on.

Vinnie

How can you detect drift? Like, because these models are what, 80% accurate? And they're making recommendations, personal recommendations, for what, buying habits or fraud detection? How do you codify something that can detect drift if there's a built-in error, and you don't really know what the person had a satisfactory result or not?

Calli

A lot of times, it's honestly more statistics. At that point, you're doing a lot of checking.

Vinnie

That's why I don't understand it.

Calli

It's a little bit out of my wheelhouse. That's not really where I focus. But I was actually just talking with a couple of data scientists who are in the process of building something out for this. And you know, it's a lot of standard deviation, they're checking to make sure. And I'm working with data scientists who are working on models that have to be closer to like 95 98% accurate. And so, they really need to make sure that they don't have future drift that they don't have these skews and those sorts of things.

Vinnie

Gotcha. That makes sense. So, math and statistics, great. So, talk to me a bit about where you see things going over the next five years.

Calli

Yeah, so there's this really new concept, I don't even think that there's really a marketed product, yet a couple of different clouds have them in places like Netflix. And those sorts of places have built their own, but it's called feature stores. And the idea is to abstract the feature engineering, the feature drift, detection, model, monitoring, everything like that, out of the data scientist's job and automate it. And what's really cool about it is that it would have it has the ability to have

both online and offline feature serving. So, when you retrain, you don't want to do that on your online model, you don't want to put that load on that model. So, you would do that in your offline data store, which is more of a columnar format, your traditional database, the online feature store is probably more likely going to be a no SQL database, it's going to be row based. And that's going to be able to get back those recommendations in milliseconds. I could talk about future stories all day; I will avoid doing that. But they are really neat. And they automate a lot of that 80% that the data scientists have been doing.

Vinnie

Right. It feels like commoditizing, some of those big parts of the machine learning development. And it's interesting to me, because it means that maybe five years ago, people were competing on how they could innovate by hand coding or creating your own features. And now, if that's going to be commoditized, you're going to be innovating on how well you know your data, how the right way to build a model the right questions to ask how to ask them, as opposed to the actual coding and feature development. So, we're kind of shifting from competing on the build, and now we're competing on the models. Does that make sense?

Calli

That's exactly where we're going. The other thing that I really see is explainable AI. I'm sure that there have been times that you've gotten an ad recommendation and it's like, why did I get this? This doesn't even make sense. Actually, I think it's on Facebook, they can say, "why did I see this ad" and you'll start to see why you saw those. And sometimes it's as simple as I'm a female somewhere between the age of 22 and 50. And it's like, well, great. Yes, I do qualify for that range but is there any other reason I got this ad?

Vinnie

That doesn't seem very smart. That seems like a decision system. Like when I think of when I get the wide audit, I see this, most times it feels like an if/then statement. If you are in this age group, and if you are this gender and you are in this income range, you're getting it. That to me is not machine learning. That's a decision support system that's just nested. I am more intrigued by

the machine learning models predicting something I may like based on my actions and other people's similar actions. That the computer saw something that maybe the person running a case statement wouldn't see.

Calli

A lot of those times those are classification models. Especially with classification models, we're trying to get to explainable AI, where, without using statistics, or with using very little statistical language, were able to explain in layman's terms why you got the recommendations you did. And it's hard. It's really hard. And even if the data scientists who actually coded this up many times, can't do it without talking about standard deviations and means and variations. Getting them to explain it in English is just difficult. We have to, though, because it's legally required, or we can't use it in a lot of fields. Like you can't use data science right now for a lot of loan processing, because if you do it sometimes you can't explain why. Same thing with health care and accepting rejecting claims, like you have to be able to explain why so until we can get there, a lot of those fields can't use that.

Vinnie

I think about college acceptance as well. You know, if you reject somebody, you can't say, well, the model predicts that your child won't do well; like that's not a good answer. Do you have to explain it, even if it's used optimistically? What I mean by that is, let's say, we have a college admissions model, but instead of rejecting people, it auto accepts, and then if it rejects you, you're not rejected, you go to a human process that a human reviews it, so that you're being optimistic, and there's a positive outcome. And any negative outcome will go through a normal human workflow, you still have to be explainable in those terms. It doesn't matter if it's a positive or negative result, I guess, is the question.

Calli

Yeah, you still have to explain that. Because if I'm the dean, and we do have a student that was auto accepted, and does not do well, I want to know what happened that we ended up accepting that person, you know, there's, there's even with the most positive intentions, at some point, you

do have to explain what happened.

Vinnie

It’s interesting, right? Because that provides a strong limiting factor on the models we can build. It does, right? But at the same time, it helps a ton with bias too, because if we didn't, and we weren't able to explain it, and let the models do as much as they could do, then I think it would be a multiplier in the bias.

Calli

Oh, yeah, absolutely. But I mean, you also have to be careful, Facebook wrote a model, and with the best of intentions, ran resumes through for their hiring process. And they ran resumes through of people they had already hired, what they didn't realize was that they had unintentionally given a bias to white males in their early 20s. And so, the process was rejecting resumes that didn't apply to those people who didn't speak that way, or who didn't write that way. So sometimes, if you're not careful, you can actually build bias into your process without intending to.

Vinnie

Let’s dive into that. Because I think people have a hard time differentiating between the more descriptive sequel based. If you're a white male and, in this age group, you're going to be accepted. That's not what you're saying. What you're saying is that the people who were accepted were white males. And when the computer learns their language, and how they wrote their resume, there was a similarity there, that doesn't apply to other people. So, the bias was inferred. Right?

Calli

Exactly. It wasn't that it was only looking at resumes, but it had picked up on particular language. And this is something that we at CapTech have actually recently gone through, we started looking at language and trying to make sure nothing was too masculine or too feminine, too excluding or to including in our job postings. We recently went through that just for our diversity, inclusion

and belonging, because it's so easy to, for a male to write something that comes off very masculine, whereas I may write it completely different. Just based on gender, there's always this little bit of difference. So, the computer learned the language of the people who worked there, which had unintentionally had a bias and then continue to propagate that bias.

Vinnie

That seems incredibly daunting. It's terrifying. I mean, even when you're not trying to have bias, then you're gonna have like, inverse bias. I guess that comes back to the monitoring, and the continuous training so that you put tools in place to help keep it in check. Because by its nature, it's going to drift.

Calli

Yeah, absolutely. And that's something I said, their bias is inherent in data science. You can't code it out, you will never be able to code it out. Because at the end of the day, it's humans coding.

Vinnie

I wouldn't blame the coders that have the model that saw the bias. It was the data that it was fed. Gosh, there's so many different nuances to think about, that's why I'm glad I'm not a data scientist. So, would you consider yourself a data engineer then?

Calli

Yeah.

Vinnie.

Any other future predictions? I have a quote here that I don't necessarily believe. So, it's who was it, Ray Kurzweil? Yeah, the American inventor and futurist. "Artificial intelligence will reach human levels by around 2029. Follow that up further, say 2045 will have multiplied the intelligence, the human machine a billion old. I don't think by 2029, we're going to reach human levels. That seems crazy to me.

Calli

So, he made that prediction in a book in 1990. So that's a little bit out of date. And his dates haven't always been accurate. But there's been a handful of different things that Ray Kurzweil has predicted that have come true, like self-driving cars, those sorts of things. He's a little out there. But I think you have to be to be a futurist. And he's got some really interesting books on AI and those sorts of things, all theoretical and everything. But he also hits on a lot of the ethical considerations and stuff like that. So yeah, I'm not sure that he's going to hit the mark on the 2029. But I think it's coming faster than we realize it is.

Vinnie

And we mentioned this on a previous podcast, it was about machine learning and AI. And AI, in particular, is good for automating human tasks that take humans less than a second to do. Like, I know a stop sign means stop. If I see a red light it means stop. But if I've got to figure something out, if I've got to determine something and like look at your face and hear the in the tone of your voice? As things get to be like five to 10 seconds of human brain analysis, the models break down already, like it's not much more than a second. So, it's weird to me to think that an AI will be as complex as a human brain.

Calli

I think you're spot on though, because at the end of the day, our brain is whether we realize it or not, it is breaking it down into those sub second tasks. And whatever it's doing, that does take those longer periods of time, you know, you're at a four-way stop, who gets to go next. That's going to take more than half a second. But if you can break it down, and if the model, if we can get to a point where models can start breaking it down and start putting complexity into those, and I think we're getting there with NLP and starting to do better at listening to a whole conversation instead of early NLP was just listening to the next word that came up. And it was like, if you heard not, okay, the sentence is negative. So, it would take that and it would stop. But sentiment analysis has gotten a lot further and listens to the whole sentence now. And I think that's where we have to get with more of the AI in the ML.

Vinnie

Right? And yet Siri still frustrates so many people. I think that's the counter argument. There's someone who works with us, whose last name is mispronounced by Siri and has been mispronounced for the last five years. And so, where's the continuous learning and the training? Apple, come on.

So, anything you can tell the audience in terms of what some of their next steps can be? If they're in a spot where they've taken a step into machine learning? They don't feel like they're pros at it, yet. They're getting some models built. What advice would you give someone to get that next level of maturity? Is it hiring a particular person? Is it putting particular process in place? What are some of the early things that our audience members can do to advance where they are?

Calli

Yeah, the machine learning engineer role is a very nice role that you're really not going to find one person who can do it all, which is what everybody is looking for. A small team of strong data scientists with a solid engineer is going to go a lot further. And it can be a data engineer but can also just be a software engineer.

Vinnie

So don't look for a single role that doesn't exist, that's probably a unicorn.

Calli

That unicorn is going to cost you too much money, Google and Amazon and all of those folks are paying them hundreds of thousands of dollars and they're not leaving, right? So, building a small team of specialists is probably your best bet.

Vinnie

Right? And then thinking about the maturity of the data they have to work with, you know, is this a big bang thing where you have to wait till all your data is clean and in the right systems? Are there small wins you can get along the way?

Calli

Oh, there's absolutely small wins. I think back to a project, one of my first data projects when I moved over. We predicted whether or not a house had gas or electric heat. So, we didn't have all of the data available. But we started with what we did have, and we could get with like 80% certainty, this has gas heat. And then as we got more and more data, the model got better. So, you can absolutely start there and start with things that like, this will help us. That was a large utility that helped them make better predictions on their finances, they were able to predict what type of energy costs different houses were going to have.

Vinnie

That's what I see a lot when I talk to clients. It's having a data scientist who's not only good at the math and statistics, but know that industry vertical really well, that's the tough intersection, because you can't just hire someone. It's funny. When I meet someone who's just graduated college, and they say, I'm a data scientist, and I feel like saying, well, you studied data science. And maybe you'll be doing some data science work, but it feels to me like there's a requirement of industry expertise, have to know what questions to ask, and how to apply what you've learned. Because a lot of people don't know what questions to ask or how to ask them, or what's valuable to the business. So maybe, am I being a little pedantic?

Calli

No, I think you're absolutely right. The team I'm working with right now are all in the healthcare industry. And most of them have degrees in health economics, or something to that effect, like their PhDs are not in data science. Those were, you know, they have minors in. I think one of them has a dual major in computer science. But most of them all have minors in data science, or data analytics or something like that. But their primary focus is their industry.

Vinnie

Yeah, that would be a good recommendation for people who are coming out of college with data science degrees, and that is to understand what verticals, what industry verticals, interest them

and start getting deep into that because it's as much about knowing the business as it is knowing the models. Well, Calli, thanks for joining me today. Really appreciate it as always. I always learn something when I when I sit down and talk to you. So, thank you. And for our listeners, if you're enjoying these podcasts, please share them on social media, and subscribe. It really does help us out.