Transcript

00:00:01 Speaker 2

This morning. OK. Yeah, I'm recording this session so. So we can start again for the 4th one. Can you please share your experience in the in your current position?

00:00:06 Speaker 1

No problem.

00:00:14 Speaker 1

Yeah. So because I am working with supporting data science teams.

00:00:22 Speaker 1

In the use of various technologies, the there's a couple of observations. I can make a lot of organisations are struggling to get machine learning into production and get.

00:00:36 Speaker 1

Value out of it.

00:00:37 Speaker 1

Partly because they find it hard to find the staff who are good at.

00:00:41 Speaker 1

Doing machine learning.

00:00:43 Speaker 1

Partly because they find it hard to find.

00:00:45 Speaker 1

The staff who are.

00:00:45 Speaker 1

Good at understanding how machine learning.

00:00:48 Speaker 1

Interface with the rest of the business.

00:00:51 Speaker 1

So that's a kind of a major hurdle for them. Those that are overcoming those hurdles are then facing a a new set of.

00:00:59 Speaker 1

Problems, which is?

00:01:01 Speaker 1

How do you set up machine learning systems in a way that you can manage them moving forward because inevitably staff move on, so you need some way.

00:01:11 Speaker 1

To hand over projects and to retrain models and understand models in a meaningful way. And that's a constant problem we need to solve the problem of who has responsibility for the models once they get to production, because one of the biggest issues is that data scientists are typically not.

00:01:30 Speaker 1

Are as good at writing software as dedicated software engineers, so the quality in general, the quality of the software isn't as good, which means that but no one it's it's it's unfair for IT teams for a data scientist to hand over poorly written code and have the IT team have to support that. So coming up with systems.

00:01:50 Speaker 1

Whereby data scientists can quickly build models that can then be sort of result in production ready code that can be maintained by software engineers and IT teams is a.

00:02:02 Speaker 1

Really hard problem to solve.

00:02:04 Speaker 2

Yeah, I agree. Yeah.

00:02:07 Speaker 2

So towards the 5th question, do you have any experience in in the previous company which is developing machine learning system?

00:02:14 Speaker 2

If so, then what was your old experience?

00:02:18 Speaker 1

Yeah. So I've worked in multiple companies building machine learning systems.

00:02:25 Speaker 1

I I think all of the problems that I just outlined, you know, were things I faced in various various kinds of levels.

00:02:34 Speaker 1

I think there's another there's another problem which may not be.

00:02:38 Speaker 1

As relevant to your to your to your.

00:02:40 Speaker 1

Research and that is that.

00:02:43 Speaker 1

You know, business owners.

00:02:45 Speaker 1

Stakeholders, because they don't have deep, they don't have the technical understanding of what the techniques do. It's hard for them to learn to trust them.

00:02:54 Speaker 1

They don't. They're not. They're much more used to managing teams of people with human metrics and human KPI's, and that's how they understand how to get value. They don't, they don't understand.

00:03:07 Speaker 1

When someone presents them, a set of statistics about the performance of model and understand how that's gonna result in outcomes so that.

00:03:14 Speaker 1

That conversation that you have to go through to get a modelling of production is hard and often often what happens is sub optimal models get deployed because you have to come up with something that the business finds acceptable.

00:03:32 Speaker 1

Uh, acceptable for their kind of what? What they view as risk tolerance, which is partly because they don't understand that they don't have.

00:03:40 Speaker 1

A deep understanding of what the.

00:03:41 Speaker 1

Roles do. Does that make sense?

00:03:43 Speaker 2

Yeah. Yeah. So is your company is service based or product based?

00:03:54 Speaker 1

So a WS is a.

00:03:55 Speaker 1

Little bit of both, mostly product based but has some savings. The company I'm moving to is again a little bit of both, but I'll.

00:04:03 Speaker 1

Be working on the product side.

00:04:08 Speaker 2

So towards the 7th question, what software development model do you practise in your company? In journal like Agile Waterfall?

00:04:21 Speaker 1

That's that's kind of a hard one to say. I don't I I mean.

00:04:24 Speaker 1

I'm not involved.

00:04:25 Speaker 1

In the product development in the AWSI certainly use something that's more akin to agile, but data science in general, like a lot of people try and force data science into an agile methodology.

00:04:26 Speaker 2

OK.

00:04:42 Speaker 1

And it's a little bit harder largely because.

00:04:45 Speaker 1

It's inherently experimental, whereas in a lot of agile situations you can at the start of a Sprint you set up and say I'm gonna build this exact thing and then we then we're gonna test it with users.

00:04:54 Speaker 1

Whereas in data science, you're saying here's a set of experiments I'm gonna run, but I don't know. There's no guarantee that any of.

00:05:01 Speaker 1

Them are actually going to work.

00:05:04 Speaker 1

If that makes sense.

00:05:04 Speaker 2

So yeah. Yeah, exactly. So towards the 8th, 8th question, can you could you please share your experience with the interesting project in machine learning that you have worked on recently?

00:05:23 Speaker 1

Yeah. So I guess recent just recently working with publishers to try and there's a lot of publishers are very interested in systems that help them categorise articles, tag them automatically and display related content. But.

00:05:39 Speaker 1

It's it's there's many, many ways of doing that and it all depends.

00:05:44 Speaker 1

To some extent it depends.

00:05:45 Speaker 1

On how that is displayed to the user.

00:05:48 Speaker 1

They because the user need to see named categories, in which case you need something like topic modelling that have topic modelling produced category names. Or are you just need to know that?

00:06:00 Speaker 1

Things you know. What is it that what is it that determines whether someone will be interested in reading another news article and it's not it. It's not very clear because it's not necessarily just about W.

00:06:12 Speaker 1

The the the like sport for example, take sport. You know people can be interested in cricket but have absolutely no interest in badminton.

00:06:20 Speaker 1

So just because they've read something in sport doesn't mean they wanna read other sport. You know they may. It may be the fact that it and it may be they care about only when their.

00:06:32 Speaker 1

Their own country is playing so that you know, some people aren't actually that interested in the sport, but they just they care because they're kind of.

00:06:40 Speaker 1

That's a bit.

00:06:40 Speaker 1

Of national pride, they'll watch things where their own country is is sort of in there. So it's it's really difficult.

00:06:47 Speaker 1

Thing to determine what makes a A an article similar to another, so you have to sort of propose multiple ways that you can do those groupings and then experimentally test them which are the ones that actually results in people you know clicking through and reading more content.

00:07:08 Speaker 2

So towards the 9th question in your working experience, how many software architecture design techniques?

00:07:15 Speaker 2

Of machine learning you work with.

00:07:19 Speaker 1

Yeah. OK. So I guess when I first started doing machine learning, I did lots of Java work, so it was all object oriented.

00:07:29 Speaker 1

Java, Java systems, we were tending to build them from the ground up most of the time, but we used a couple of libraries, Weka.

00:07:38 Speaker 1

And Lib SVM, both of which were Weka, was at least written in Java. Comma have lived that life. SVM had to have a wrapper around it in time, so all object oriented and typically you write some kind of main.

00:07:53 Speaker 1

You know, sort.

00:07:53 Speaker 1

Of main script that use the libraries.

00:07:55 Speaker 1

That you've built to sort of run simulations.

00:07:58 Speaker 1

I've since worked on.

00:08:02 Speaker 1

I I I worked on I guess more kind of like.

00:08:05 Speaker 1

What would you call it?

00:08:08 Speaker 1

Like when I first worked on machine learning learning systems in something which was kind of like a microservice.

00:08:19 Speaker 1

Like more of a kind of a rudimentary. No, actually, no. Sorry, that's that's pretty not great. It's more like it was more like scheduling based batch jobs.

00:08:28 Speaker 1

So basically you'd have a standalone script which would be executed by a Cron job on a regular basis to, say, compute a bunch of predictions and write them back.

00:08:36 Speaker 1

Somewhere and then make use of them so they decoupled in the sense that this batch job just did. One thing you know, processed input, wrote it to an output and then downstream systems could deal with it but not in micro sense. Service in the sense that it was an online API.

00:08:53 Speaker 1

That was, you know, ready to sort of just to take inputs, produce outputs. I have worked on.

00:09:03 Speaker 1

Yeah. And I did a similar I've done. I've done that particular kind of approach of like individual batch processing jobs decoupled from each other by kind of input output mapping mapping. I've done that quite a few times.

00:09:19 Speaker 1

I think largely because that's a very in in machine learning. That's a quite a robust way to do things because you you don't have to build the infrastructure of of API's and maintain them and have online thing and you can it's a simple way to decouple just by input output matching. I have I've built systems.

00:09:39 Speaker 1

I've built sort of like simple kind of Docker container type things for API's and that kind of thing, but I haven't done that. I would say in a kind of large scale production environment.

00:09:51 Speaker 2

OK, so towards the 10th question, which common software architecture design techniques of machine learning you found being used in most companies through your experience?

00:10:04 Speaker 1

Yeah. So I would say that kind of that sort of.

00:10:08 Speaker 1

Single purpose batch scoring and writing results either to a file or to to a database is one of the the most common patterns.

00:10:19 Speaker 1

Of of deploy.

00:10:21 Speaker 1

I've seen a lot of organisations write those predictions to a database and then have downstream systems pull them from a database.

00:10:28 Speaker 1

Whether that's sales force, if it's kind of for humans to look at it, but they've even seen plenty of people build other systems on, you know, pulling, pulling all of the the.

00:10:40 Speaker 1

A bunch of different predictions from a A Teradata or or MySQL or sorry SQL Server database, that kind of thing, yeah.

00:10:51 Speaker 2

Towards the 11th question according to your experience, what are your best software architecture design techniques for machine learning and what are the benefits of using them?

00:11:06 Speaker 1

So I mean that tech, the tech thing is really common that I outlined before like large scale batch, that's kind of an independent process that only requires.

00:11:15 Speaker 1

It can either be scheduled or it can be data presence sensitive, so sometimes it's scheduled to run a certain time every day.

00:11:24 Speaker 1

Sometimes it's executed whenever a system detects like a new input file is there. So basically, using an event on an S3 bucket on on a WS services is a really common pattern.

00:11:35 Speaker 1

When the data arrives that triggers the process to run to make do the scoring and rights to another S3 bucket. So that kind of decoupling is super common and the advantage the advantages of that are that it's.

00:11:48 Speaker 1

It's easy to replace because there's just no dependencies whatsoever other than the inputs and outputs, and it's easy to deploy quickly and and in a world where you're still experimenting with different machine learning approaches to a problem, it's it's some sense. It's probably. It's probably better to have something that's decoupled and easy to replace.

00:12:09 Speaker 1

I guess the downside of that is that it's.

00:12:15 Speaker 1

You it's. It's harder to do things like more sophisticated deployments like a like a Canary deployment or a a blue-green deployment or a or any kind of AB testing, that kind of stuff because you have to, you might you either have to.

00:12:33 Speaker 1

You need then.

00:12:34 Speaker 1

Have software around it that determines which batch job does, which records, or you score all records with both batch jobs and then have some kind of downstream system that then sort of mixes and pulls from them. So it could be a little bit computationally efficient, inefficient in that Reg.

00:12:50 Speaker 1

So not suited for I guess, large scale experimentation with large numbers of models, the kind that you know, we hear that, that the Googles and the Facebooks of the world do a lot of.

00:13:02 Speaker 1

I would say that you know, most organisations just don't. They don't have the talent and the time and the money.

00:13:09 Speaker 1

To set up systems like that.

00:13:13 Speaker 2

Yeah. Interesting. So towards the 12th question, do you have any recommendation for software architecture, design techniques of machine learning system?

00:13:25 Speaker 1

Yeah. So other than like the deployment pattern we spoke about quite a bit. The only other the I guess the other thing that I would say is.

00:13:33 Speaker 1

I think there's organisations can do a lot more around standardisation of the training testing process and more rigorous testing, so you see high variability in the way models are trained and tested, which makes them difficult to compare. I think that that you can.

00:13:52 Speaker 1

Yet you can the more rigour an organisation, particularly the management of data science, puts into that process, the easier it is to evaluate new models and the the easier it is for for for data scientists to get up to speed. What's done before, because there's a standardised pattern and I think the other thing that.

00:14:12 Speaker 1

Organisations can do if once they've got those kind of standard patterns for training test.

00:14:17 Speaker 1

And evaluate. They can also have. They should have a bunch of simple baseline models that are easily to to to build and and evaluate so you know, so you know quickly whether some fancy model that and a data scientist has just has just built, whether it's actually adding any value because what you see.

00:14:37 Speaker 1

Again, because the field is so new, it's moving so rapidly that so much of management doesn't really understand the technicalities. You there are needlessly complicated models being deployed all over the place.

00:14:50

OK.

00:14:51 Speaker 2

So towards the 13 question, which would be the best practise that could be useful in applying software architecture designing?

00:15:00 Speaker 2

Of machine learning systems.

00:15:11

UM.

00:15:15 Speaker 1

I think so, I think.

00:15:19 Speaker 1

I think in general data science teams can learn a lot from software engineers. There's a lot, you know, software engineering is discipline has developed a lot of great ideas over the last 30 years. It's just simple things like the movement into automated testing on on commits.

00:15:38 Speaker 1

Into a repo like. That's a. That's a really simple thing that most data science teams don't do to the idea that you can. Not only you can have automated commits, you can automate you. You know you can have.

00:15:52 Speaker 1

On automated initiation of testing of model artefacts, if they're committed into something like a code repository for deployment.

00:16:00 Speaker 1

It's it's all.

00:16:01 Speaker 1

It's a little bit it's a, it's a little bit glib to make that comment because data science is different from software engineering in the sense that it's heavily data dependent.

00:16:12 Speaker 1

Like the things that you test.

00:16:14 Speaker 1

Depend on the datasets you're using, and it's not binary, whereas you can say definitively you know you can put binary conditions that say if this happens, the software is growing, you've introduced a bug, whereas in in data science the it's stochastic. It's like what's the performance metrics you know for this particular model so.

00:16:34 Speaker 1

So it it requires a little bit more thought, not the least of which because you need you need data sets and not the kind, not the kind of data sets that are typically used in software development.

00:16:46 Speaker 1

Thing like dependency injection where there's synthetic data sets and it doesn't really matter. You actually need real data sets to test on.

00:16:54 Speaker 1

With data science, you know projects and that that makes the whole process hard. So I would not be surprised if we see the emergence of something like a Jenkins, but specifically for data.

00:17:06 Speaker 1

Science so you can host a bunch of test data sets on there and and set up process that run automated sort of tests on those data sets. When model artefacts are injected or or a that kind of thing. Does that make sense?

00:17:25 Speaker 2

OK, towards the 14th question, what are the most common software architecture design challenges?

00:17:31 Speaker 2

In machine learning system.

00:17:35 Speaker 1

Well, I think I think one of the the, the IT, it's one of the simplest one is that a lot of data scientists will develop models inside like a single jump.

00:17:45 Speaker 1

Notebook and then it's not. They don't. There's not. There's not many. There's no functions, there's no parameterization. It's just not even vaguely close to being production ready.

00:17:57 Speaker 1

Or even if.

00:17:59 Speaker 1

You can sort of.

00:18:00 Speaker 1

You know, there are systems for deploying Jupyter notebooks if you convey just if you sort of hold certain conventions.

00:18:06 Speaker 1

But it's just not very reusable I think. While the white basically the way that data scientists reuse code is just copy and paste from from notebook to notebook which is which is terrible.

00:18:17 Speaker 1

So I think just getting getting data science teams to adopt any convention for writing standardised libraries that they reuse.

00:18:29 Speaker 1

Would be a win.

00:18:30 Speaker 1

Doesn't matter what the convention is.

00:18:38 Speaker 2

So towards the last question, what are the main architectural decision on software architecture design of different machine learning systems?

00:18:48 Speaker 2

Like what are the major architecture design decisions?

00:18:53 Speaker 1

Yeah, I I'd sort.

00:18:54 Speaker 1

That I think the main one, the main one typically really is is is this a batch system or is it real time that's that's that that drives a lot of things because that's it's going to.

00:19:08 Speaker 1

It'll it'll affect model choice because latency matters in real time, but it doesn't in batch and it will affect the whole kind of the the whole everything you right around the soft the model for how it's kind of executed and and and and runs.

00:19:26 Speaker 1

As a parallel to that, the other thing that really matters is to some extent the.

00:19:35 Speaker 1

The the like the horizon of the predictions, how the predictions relate to the business. So you know for for example, sometimes you're you might just be predicting what the customer's going to do next or you know the next purchase or something. So something really.

00:19:54 Speaker 1

Sometimes it might be. What's the probability they're going to churn in the next three months, so the time scale for the prediction can be anything from the next event to some kind of aggregate measure over the next period of time. And those kind of decisions affect the kinds of data that you're able to use.

00:20:13 Speaker 1

And and how the model will then kind of.

00:20:16 Speaker 1

Interface for the business.

00:20:19 Speaker 2

Yeah, really interesting for me. Yeah, I learned a.

00:20:22 Speaker 2

Lot from you?

00:20:26 Speaker 2

Yes. Yeah. So these are the, these are all questions that I have.

00:20:33 Speaker 2

Yeah, I don't have any other questions.

00:20:36 Speaker 1

All good, Roger.

00:20:37 Speaker 1

I hope that's useful. Good luck with your masters.

00:20:40 Speaker 2

Yeah. Thanks a lot. I really appreciate your time.

00:20:44 Speaker 1

Cheers. Thanks mate.