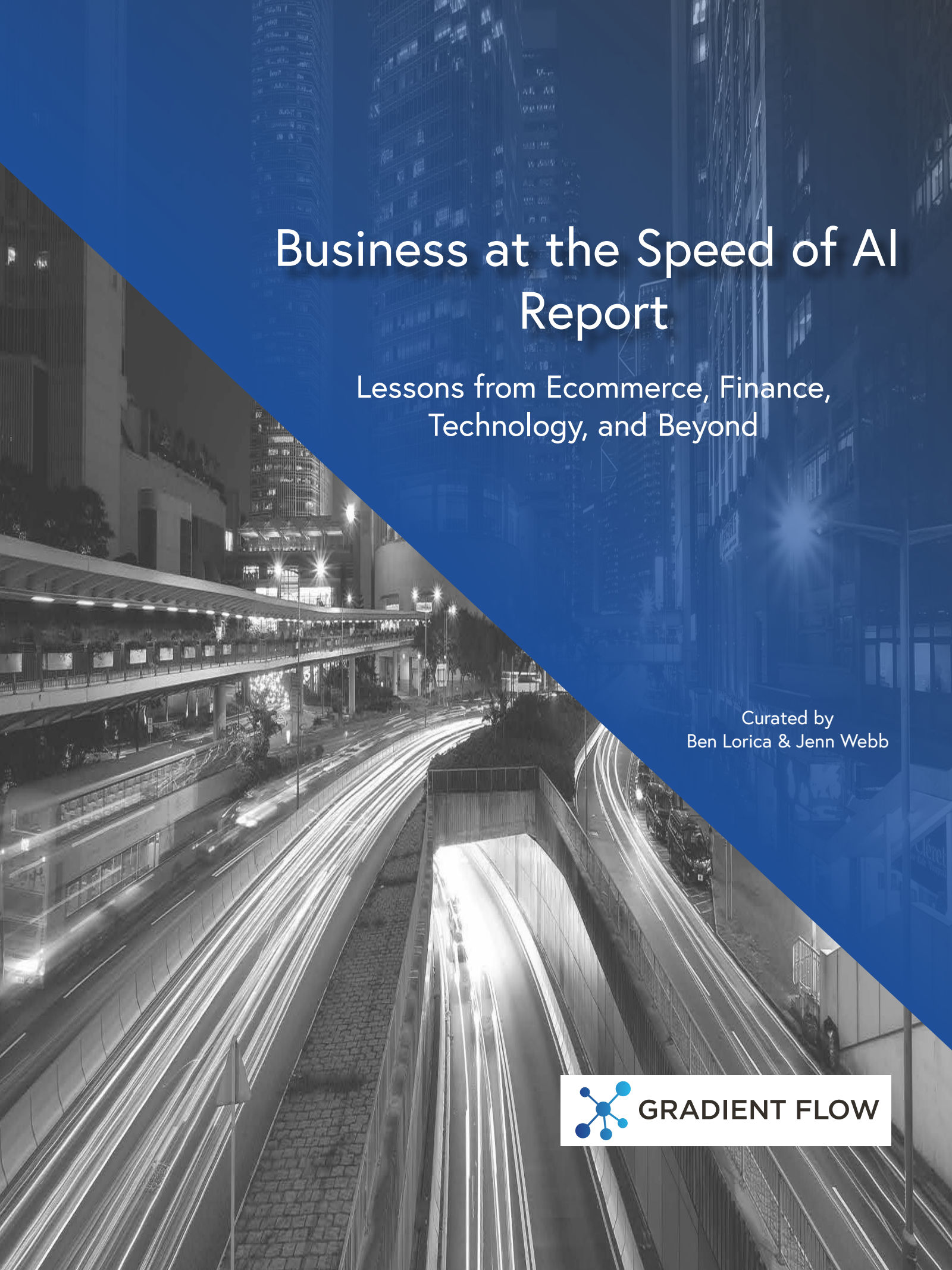


Business at the Speed of AI Report

Lessons from Ecommerce, Finance, Technology, and Beyond

Curated by
Ben Lorica & Jenn Webb

GRADIENT FLOW

Table of Contents

Introduction

Across industries, there is a growing interest in artificial intelligence (AI) and the supporting technologies—machine learning (ML) and deep learning (DL). Five years ago, easy-to-use tools for machine learning were readily available, but tools for deep learning were still largely confined to a few research groups. Today, companies have access to many more machine learning and deep learning tools: open source or commercial, and cloud or on-premise. We are now solidly in the implementation phase for AI technologies.

For companies that have the right tools, teams, and processes in place, AI technologies can be used to enhance many products and services. For example, AI tools can improve operational efficiency and decision-making, generate new or additional revenue, and improve machine learning models used in areas like forecasting, risk management, and fraud detection.

However, new technologies always come with challenges. AI and machine learning applications rely on access to data, the ability to train and tune models, and access to compute resources. Machine learning models (particularly deep learning models) require access to large amounts of data. Thus, companies that aspire to have a broad and sustainable AI practice need to understand the importance of having foundational data technologies.

Leading AI researcher and entrepreneur Andrew Ng has pointed out that data accumulation can often lead to a defensible business. He describes a Virtuous Cycle of AI where companies that have technologies in place to accumulate and use data end up with better products (and models) over sustained periods:

When implementing AI technologies, the need to align and educate multiple parts of an organization is often overlooked. At a minimum, many AI and machine learning solutions require the coordination of technical teams and business units. Most AI solutions are capable of augmenting humans, as opposed to functioning autonomously. This makes it imperative that project managers and technical teams developing AI solutions work closely with stakeholders and domain experts.

AI and machine learning approaches also introduce new risks and challenges that are best addressed by teams composed of people from diverse backgrounds and expertise. Concerns include issues around fairness, privacy, security, compliance, safety and reliability, and transparency. Additionally, while there are a growing number of professionals skilled at building AI products and services, the competition for talent remains fierce. Companies that are able to build an AI strategy, and assemble and retain an AI team have a tremendous competitive advantage.

In this report, we consult with industry leaders who have extensive experience implementing AI technologies. They share insights and clear, practical advice gathered from building and managing world-class AI teams delivering some of the most widely used AI products. We also take a deep dive into the Responsible AI space. Responsible AI includes concerns around safety and reliability, fairness, and transparency and accountability. The big takeaway is that, with responsible, efficient, and effective approaches, AI technologies are no longer reserved for large companies with expansive engineering teams. We're confident you'll find our report applicable across industries and use cases.

How Machine Learning Powers Ecommerce

Solmaz Shahalizadeh is VP and Head of Data Science and Data Platform Engineering at Shopify, an ecommerce powerhouse whose technology enables over a million businesses worldwide. Shahalizadeh and her team design and build data products and systems that are reliable, scalable, and that empower users. They are bringing software engineering rigor to data engineering and data science. As an example, data pipelines at Spotify undergo rigorous testing before they are deployed to production.

Highlights include:

- Data platform engineering and data science teams at Shopify fall under the same leadership. In addition, data teams are embedded in every product area. The result is that their data scientists and data engineers have domain and product context.

- Data pipelines at Shopify undergo complete unit testing, and they ensure that nothing gets deployed without collecting metrics and metadata on the models used in production. The majority of data pipelines are owned, maintained, and written by their data scientists.

- Shopify uses explainable models. They want the people who are going to make decisions

based on machine learning models to be comfortable in how models arrive at decisions.

- Shopify has a data-driven culture, and the company runs many experiments simultaneously.

- Models undergo a battery of tests before they get deployed. This includes tests for fairness and bias, and thorough backtesting designed to surface unintended consequences.

This conversation has been edited for clarity.

Ben Lorica: You wear two hats: data science and data engineering—which are very related, but somehow in some companies somewhat disconnected. First of all, is that a deliberate move on the part of you and the Shopify leadership, to put data science and data engineering under the same org?

"It is very intentional that we have our data platform, engineering, and data science teams under the same leadership."

Solmaz Shahalizadeh: It is very intentional that we have our data platform engineering and data science teams under the same leadership. The idea behind that is, for any data science team to be able to have an impact at the company, the right infrastructure needs to be in place. That includes systems to ingest data, to transform, process, load, and all of that.

Through experience, we have seen that having these teams close to each other with a similar mandate helps a lot in making progress and bringing machine learning and AI—and, in general, data-informed culture—to the company. We have had this set up for the past five years, and it has really served us well.

Ben: When we did surveys last year, we found that the companies that seem to do well in ML and AI were the ones that took some existing analytic or data use case and just started building on top of that. In other words, they may have a data warehouse already, they have some BI, and then they started using ML. In your experience, can you go from zero to ML?

Solmaz: It takes years. At Shopify, we started rebuilding our data warehouse in 2013, and we started using Spark. It was at that time that Shopify was going public. I was tasked to build the financial data warehouse, and build the team to do the work. We started from very simple business intelligence (BI) and analytics, doing all of the boring non-GAAP analysis for the company, and we built the financial data warehouse. As a result, we had data points that were very trustworthy, they were reproducible, and they were very well understood. Having that foundation allowed us to think about how we could use the data to create new experiences and products for our merchants.

As you know, Shopify is a leading commerce platform that powers over one million entrepreneurs of different sizes and in different stages of growth. As a result, we have a very good view into commerce, and we had a really well-maintained financial data warehouse. From there, we felt that the company was also ready to build products on top of it, so we started investing time figuring out what we could build. One of the first areas we found, as you said, is an area where there was an existing system, and it was rule based: we started by looking at order fraud detection. So, for every order that is processed through Shopify as a platform, our merchants have to make a decision very quickly about whether or not they are going to fulfill that order. That's when the fraud detection occurs. If our algorithms or our system can, in a fraction of a second, inform the merchant of the likelihood of fraudulent orders, then they would probably not fulfill the order, or they would ask for extra information from the buyer.

We already had an existing system that was rule based. So, in 2015, we said, "OK, this is probably a good candidate to start data products." Why? Because, first of all, we have a lot of data. We have a lot of historical data around order processing, so there's a rich set of features we can look into. The other part is that there is an existing system that has the rules, so the basic ideas and basic understanding of how our features result in an outcome might be there. The last reason is that it's a problem that matters for the business. By solving it, actually, we're going to get usage, we're going to get feedback, and then we can see if we are having an impact or not.

We started with very basic, very simple models, like logistic regression. It's fascinating that, of everything that was involved in bringing this model from working in our data systems to powering millions of transactions per second, the least important part was the actual model itself. What was really important and crucial was what was done for feature engineering and what was done to bring it to scale, to not slow down the checkout process or any part of the existing systems, and how we continued to maintain and govern models as they grew.

Starting where you have some sort of basic understanding and some basic analytics is the right thing to do. But the other side is also making sure you're tackling a problem that people will actually care whether you solve or not, because bringing machine learning brings complexity to the systems. You're going from deterministic systems, for the most part, to probabilistic models, and that itself brings a lot of interesting challenges. So, being sure that what you're investing time and energy in solving actually provides value is also another key point.

Ben: Let me return to what you brought up around feature engineering and scaling in a second, but as you were describing that transition from rule-based to a simple logistic regression, I started thinking, in the case of the rule-based system, some of the business or domain experts may have had a hand in building it, so they had a sense of ownership, and they understood how it worked—in other words, there's some amount of explainability there. So, now you have to convince them somehow that, "Hey, this is not quite a black box because it's logistic regression. This algorithm that you have less to do with is the better way to go." I guess at the end of the day, if this new logistic regression model approach outperformed the rule-based model, was that enough to

encourage adoption and usage?

Solmaz: That's a very good question. I'm going to take a step back and tell you how we have actually designed our data science and engineering organization, because I think one of the key points lies in there. At Shopify, the way I have built the team is that data science engineering is embedded in every single product area we have. So, for financial services, the area where we were working on order fraud detection, there's a mining data science team that lives and breathes that space. They're not just data scientists; they are data scientists with the domain context. As part of that, they have already established relationships with domain experts, the existing risk analysts we have. In the beginning, there was a lot of back and forth in understanding the rule-based system and the features.

The other part of our system is that we don't really think of AI *versus* humans. What we think about is AI *with* humans. Even for our fraud detection system, we have a streaming solution that samples and surfaces predictions to risk analysts. We get feedback from them on how they are doing. The other part is that when we started this approach of using machine learning, it was really important for us to understand what the features meant. I'm sure you've heard many stories about people picking a feature with a definition, not knowing where it's coming from, then later on realizing it doesn't have the right frequency or the meaning they thought. Working very closely with the domain experts allowed us to understand the meaning of every single feature. On the other side, our predictions would sample and go through a human verification as well.

"One of the main principles we have put into teams to ensure we don't just build models for the sake of models: we build models for better user experience in the domain."

The other side is that the risk analysts are always ahead of us in understanding new features that might be useful. So, we had a collaborative process where we said, "We're going to continue to comb through the data we have, but can you please also tell us the things you see day-to-day that you think would be very good features?" That was really nice because by working together, we also made a simple framework for the analysts to be able to introduce a feature to us and then be able to see, okay, what's the false positive rate on this feature? Because as humans, we see a signal and we think it's very important, but it's also good to look at the other side - the times the signal is not correct. That collaboration allowed us to gain the trust of the risk analysts and the company to put it in front of the merchants.

Ben: At the scale of Shopify, just rolling out these models, I imagine, requires rigor and discipline. As you pointed out, feature engineering and data pipelines is probably where a lot of the IP lies. What is your secret sauce to make sure data pipelines are under control, reliable, and meet the

SLAs that you need? Is there anything you've learned over the years that you're willing to share?

Solmaz: Oh, of course that's where my interest and passion is. Before Shopify, I also worked in cancer research, and after that, I worked in investment banking. I'm really used to people making crucial decisions based on what we predict or what we put in front of them, be it in healthcare or finance—or now, working with one million merchants. That's the part we've put a lot of emphasis on to make sure we do a good job.

As you mentioned, feature engineering, then training and testing models, they all have "mini got-yous" with them. The best thing we did from Day One is that we said, "OK, we're going to invest in our pipeline the same way we invest in any other piece of software we build." So, we're going to have complete unit testing. We're going to ensure that nothing goes to production without collecting very good metrics and metadata on the models we put out there.

"The majority of our data pipelines are owned and maintained and written by our data scientists."

The other side comes from my finance background—I was very well aware of practices like backtesting. Oftentimes when you're launching a model, you don't really know the ground truth for six months or longer. So, before any model sees the light of day, we put a lot of practice around making sure we know how it compares to existing models. We know if the experience of a group of users is going to drastically change as a result of introducing a new model or not. Then, the most important piece is that, at the end of the day, if you're building a data product, what your users care about is their experience. The users are not going to care about your F1 score or accuracy or what kind of models you use. If the experience delights them, they're going to come back. Otherwise, they could care less what the machine learning or AI algorithm is powering.

We put a lot of focus from Day One on making sure every data product we build has product metrics we care about. To give you an example, in the case of fraud, we showed the recommendation to the merchant. Do you accept this order? Do you cancel? Or do you investigate? Regardless of our accuracy or F1 score, what is important is what we call our "trust score"—how often are merchants actually following what we recommend. That's one of the main principles we have put into teams to ensure we don't just build models for the sake of models: we build models for better user experience in the domain.

Ben: Let me segue into our next stop, which is building and scaling data teams. So, let's start by identifying some of the key roles. There are analysts, data scientists, and data engineers, but as you move more into ML, are you starting to have specialized roles, like ML engineer, deep learning engineer, or even ML ops?

Solmaz: The solution we have right now is what has worked well for us so far, and talking to other peers in the industry, I don't think there is a perfect team structure. You should always be aware of what works in the context of your company and product, and design for that. So, for Shopify, right now, the way we have designed the teams is that we have a data platform engineering team that supports all the data scientists and the data engineers who build applications that integrate with our staff. We have data scientists, so we have the data science role. We don't have data analysts or machine learning engineers.

There's a combination of skills needed in a team to make a product successful. We offer training and learning in each of these areas if an individual is interested in growing in one area, but is already strong in another one. For example, you might have people who are excellent product analysts and they may want to become better with their data engineering skills. We have education and training around that.

But having this broad group of work has allowed people to learn all the parts and not be blocked. I always tell people who join our team, and we are always hiring, is that by joining our team, 12 months from now, you will look back and you will know how to bring something that was just an idea all the way to production. By doing the product analytics, by building the ETL pipelines, by getting the data, you get to really understand the nature of data of the domain and the problems. By doing product analytics and understanding the story of the existing product or what you're going to build, you become an expert in the domain, not just in the data.

Then we have an ML platform team that helps us to scale the machine learning models, but having data scientists care about the end-to-end also means that we don't have instances of very complex models built in isolation that would take a very long time to hit production. Because our personal philosophy is that simpler models scale faster; they give the benefit to the users faster than waiting two years to have the perfect model out there.

Ben: Right. I guess you answered one of my questions as you were describing your expectations for people, which is: do your data scientists build their own pipelines? Number one. And number two, what is the boundary between prototype and production? In other words, does the data scientist build something in a notebook and then someone has to rewrite it in order to deploy to production?

Solmaz: That's a very good question. Let me answer the first one, the data pipelines question. The majority of our data pipelines are owned and maintained and written by our data scientists.

The reason we do it that way and it works for us is because the product context is very important. We didn't want to have a ticketing system where people are throwing work over the wall. For our organization, this model really works well. But it brings some challenges that, as a leader, we have to plan for. Part of it is understanding the data scientists who are on the market are not necessarily going to have the skillsets or see right out of the box the need for building this ETL model. Helping

them understand what they're actually unlocking by creating this is also an important part of onboarding in their journey at Shopify.

Ben: As far as production, say I built something in a notebook—how do I get it to production?

Solmaz: Excellent question. We still ask the data scientists to think about, and depending on the model, try to bring it to production themselves because our production stack for data is the same—for notebooks ,we use Spark; our Python notebooks can do the same thing. But there are challenges sometimes that are out of the skillset we expect the data scientists to have. So, for example, if it's a model that is scoring things in batch, then you can easily go through our data platform, and we have built services that allow you to serve the results back to our application, so that's fairly simple. Then we have other things like the fraud case where we need to score something in real time and it has very strict operational constraints around it in terms of SLA, freshness, availability. In those cases, we have partnered closely with the engineering team that owns the product.

Again, we try to box the machine learning models very well with well-defined APIs. And then when extra engineering is needed, we work with the application developers in the area to do that. The reason for that is twofold. If our monitoring is able to easily diagnose what issues are model issues and what issues are application issues, then maintaining these things is easy because your alerts would be very informative as to when there's a model issue versus when there are other application concerns. The fact that we do a lot of testing and shadowing before we put things in production allows us to have a good sense of that before something goes live.

Ben: By the way, since you are in the area of Ecommerce, one of the things that strikes me as far as Ecommerce and media and advertising is that there's a whole discipline and appreciation for experimentation. In other words, in media and advertising, there are experimentation platforms like Optimizely. As companies do more and more ML, we're starting to see tools that encourage experimentation and building model catalogs—tools like MLflow from Databricks and other open source projects. What's your sense? Do you think over time that companies will develop the ability to scale their ML experiments more? Or is ML just too challenging to expect that companies will be running hundreds or thousands of experiments?

Solmaz: Experimentation is really important in building products, period. We have been doing experimentation at Shopify—literally since the time I joined, we've been doing experiments regardless of ML. Across industry, what makes ML experiments interesting, as you said, in advertising or in general commerce, is that with ML, you can create very complex, different experiences for different groups within the experiment. That's really interesting. There are companies even now doing thousands of experiments with different machine learning models. I'm sure, like different ad companies, the large ones are doing broad experiments. To your question, too—do you think companies should do that?

Ben: In the case of the media and advertising and Ecommerce companies, for example, they have many years of building these experimentation platforms to allow them to scale and share experiments. In ML, we'll need tools that will allow me, for example, if I'm on your team, to log on and see what experiments are going on—"Did Solmaz try something similar to what I was thinking of before? What did she learn?" Because even failed experiments have value, right?

Solmaz: Oh, yes. They're amazing actually, because when you have failed experiments, you often have more reason to look why things didn't work than when they're successful; sometimes you attribute them to the change you've made. I think it would be really great to have that capability to run experimentation in ML, but I want to go back to something—doing experiments on their own is OK. As you said, what's important is being able to curate the results of different experiments and have governance around them. It goes back to the field of model governance that, listening to your podcast, I know you're passionate about as well. If you have good model governance in place, and you have a way to catalog the features you've used, catalog the findings, then doing experimentation for ML would be much easier. What I don't want to see happen is just doing experiments for the sake of experiments without really learning what's going on, and as a result, giving a less-than-ideal experience to the users of a product.

Ben: The great Thomas Edison had the saying that the most valuable area in his lab was a junkyard where he stored all his failures.

Solmaz: Exactly. Actually, we talk a lot about being data-informed at Shopify, and experimentation and celebrating failure is part of that, and data plays a crucial role in that.

Ben: You raised the notion of model governance, which as you mentioned, I'm passionate about. One idea people have raised is this notion of cross-functional teams to manage risks in machine learning. In other words, as machine learning becomes more important, should we bring together teams that come from different backgrounds. In other words, should we have privacy, compliance, and security experts earlier on in the model-building process rather than bringing them in later? What's your sense moving forward? Are we going to see more of these cross functional-teams for ML and AI?

Solmaz: I think so. As with any product, when you add a diverse group of experiences, you're going to build a better product. Privacy is very important, and it will be until we figure out how to build systems that are by default privacy-compliant. Even in our case, when we started, we would meet very regularly with our privacy leader. The best thing we can do is build systems that do the right thing by default. We have made sure our data platform only allows you to access data that doesn't have personally identifiable information (PII), for example—they're compliant with various things. As a result, you don't necessarily need to have the cross-functional people in every single meeting or every single chat; you can talk to them in the beginning of the project to get their sense overall, but having systems that ensure what you say is actually happening is very important.

I'm a big believer in trying something once, twice, or three times and then seeing if we can bring it into a platform or a system so that people can use their energy for more creative pursuits in their craft. That's why, for example, we say, "Okay, now we want to make sure we've tried a few times. Now we want to make sure we can do it." So, out of the box, our platform is privacy-compliant. But in terms of cross-functionality, the other part is making sure that when you're building a machine learning product, remembering it's still a product—you have a product manager, you think about the engineering aspects and the design, and you don't say, "OK, this is a data product, so we're going to make something that's insanely cool, but it's not going to be appealing to the user."

Ben: Speaking of which, this allows me to move on to our next topic, which is data-informed product building. You mentioned product managers—I asked a good friend of mine, Ira Cohen, who's the founder of a company called Anodot, to develop a tutorial for me a few years ago. He started teaching it at the conferences I chaired at the time; the title of his tutorial was, "Herding cats: Product Management in the Machine Learning Era." The thesis there was that product managers should understand ML because all products are going to have ML moving forward. Are you starting to work with other teams at Shopify, like the product managers, to get them more informed about data engineering, data science, and ML?

Solmaz: Yes. As I said, data science and engineering is embedded in every product area, so day to day, my team probably works more with product managers than the rest of the data scientists. The product management at Shopify has always had an appreciation for data and being data-informed. It's really important to have that understanding of data before you get to machine learning, because machine learning is yet another wrinkle in how you are going to build a system. If the product manager and the data science team have worked with each other in the past, though, the product manager knows that the data scientist understands the problem, the product they're going to build, and the data scientist already knows how to work across crafts. That's very important.

Whenever we have brought machine learning into a product, which now we have across our portfolio, the thing that has helped is being very clear about what we can do and what we cannot do with the models from early on. On the other side, we need to make sure the models we're building actually have product metrics. For example, the product manager we have on order risk, from Day One, they understood the industry really well—acceptance rates for orders, what are the metrics we should care about from the industry perspective? And they guided the weight that way. On the other side, we also help them understand the times that we have to think about scenarios where the model is not going to have enough data, or it's not going to have the best optimal answer, and how to think through that.

That collaboration has worked really well. Now we have a portfolio of products—we have the Shopify Fulfillment Network—that's powered by machine learning. What has been actually amazing is having product managers that mastered the domain that we are working in and then helping them understand what machine learning can do and cannot do—because the other aspect is that

machine learning is not magic. You have to have good data and good understanding, and then there's a high likelihood of having a good model. That understanding has been useful.

Ben: You've actually hinted at this numerous times in our conversation, but in ML, there's a lot of uncertainty; it's a probabilistic sort of development process, and there are iterations that need to be done. Just like building any product, you iterate, but here there's uncertainty built into the process.

Solmaz: Exactly—and then planning for that because it is going to happen. So when it happens, how are you going to handle that experience?

Ben: To what extent are you starting to think about the issues that I've been kind of putting under the umbrella of "managing risks," which includes things like privacy and security, fairness and bias, explainability, and reliability and safety? How have you folks built that into your ML development process?

Solmaz: That's a very good question. One thing I want to start with is that as a company, one of our values is that we are successful only when our merchants are successful. And that value has translated really well in how we make decisions day to day. The fact that we are here to support our merchants' growth means that whatever product metrics we put into our models support that, so there are less conflicting criteria we are trying to meet that don't agree with each other. That on its own is a very good starting point to think about. In terms of privacy, we ensure that all of our models and all the data we use are privacy-compliant. GDPR wasn't actually that hard for us because it was very close to the way we've been collecting data and thinking about it.

In terms of fairness and explainability, you have very good points. We do a lot before a model sees the light of day and is part of the product. One of those aspects is going back to the features and doing analysis to see if there is any bias in any of them, if they're favoring one group of users versus another. The other part is that we do a lot of backtesting before putting anything into production so we can see if there are groups of users whose experience is going to change without that being intentional, or to find out if there are other underlying factors that might cause that. We do causal inference on features to understand if they're pointing to something else that we don't want to be part of the features.

In terms of explainability, that's also important. As you know, we're a public company, and one of the first data products we built was Shopify Capital. Shopify Capital allows us to give cash to our merchants to help them grow their business, and they only remit money back to Shopify when they make sales. It was an interesting first task to take on as a team. We said, "OK, we want to see if machine learning has the capacity to help here." For that product to grow in the beginning, we said, "You know what? We want to only take a small portion of the portfolio and do that with machine learning." So every week for that part of the portfolio, our data science team would bring all the features, their distributions, the values for each of the offers, and we sat with our product manager and our head of risk and treasury, and we'd go one by one, helping them understand what was

going on in the models and in the features.

We also use very explainable models. We used random forest, and then we moved to boosted trees—we have actually a blog post around that. We put a lot of emphasis on wanting the people who are going to make decisions based on these models to be comfortable in how we are making these decisions. We also wanted the models to be explainable and reproducible in a way that we can stand by them over time. By building that into our systems and into our workflows, after a very short amount of time, the whole portfolio is now machine learning-driven, and it has been for the past five years. To date, we have extended over $800 million in capital to merchants. We really take transparency and reproducibility and fairness to a high degree.

Ben: Wow. So, let's close by having you make a prediction. Obviously you work constantly with and talk constantly to people who are responsible for building some of the data products that automate a lot of things. As you look at the data engineering, data science, and ML space, which parts of data science and data engineering do you think will be automated over the next two years?

Solmaz: There is a lot of opportunity to automate parts of model governance as more teams and more industries build these things and the best practices become known, they're being shared. Building multiple platforms that can allow you to do model governance and model monitoring becomes very important. With the speed that academia is growing, I think a lot of really complex algorithms that we can all use are going to be available out of the box. The focus and value should be shifting to how successful are you in building these things and scaling them to be used day to day? My hope is that as an industry and as a field, we invest a lot in those things, into before and after the models, not necessarily just in the model itself. Even if we make that 5% better, then there's 5% more models that would see the light of day. That would create better experiences for users and would increase the trust of society on machine learning-driven products.

Attracting and Retaining AI Talent

At the time of this interview, Bahman Bahmani was the VP of Data Science and Engineering at Rakuten, a large Japanese ecommerce and online retail company. In late 2019, I heard him give a presentation on how data science and data engineering were enabling new innovations and products at Rakuten. The focus of our conversation was on how the company structures its data teams, its approach to data and AI, and culture and organizational structure.

Highlights include:

- Rakuten has businesses that are essentially completely driven by AI.
- They have four main roles in their AI team: product managers, data scientists, machine learning engineers, and data wranglers. Data wranglers are people who help prepare data for models and evaluate the results of the models.
- They use a semi-centralized structure for their AI teams. They form teams that have end-to-end responsibility to deliver AI products into production, and they make sure the teams have the resources they need to do so. These teams are measured and evaluated based on business metrics.
- They look for AI talent globally and have geographically dispersed teams. Team members work on the same types of things regardless of their geographic location, and they collaborate very closely.

This conversation has been edited for clarity.

Ben: So Bahman, before we dive into how you do AI technologies and implementation in Rakuten, at a very broad level, how would you describe the impact of AI and machine learning technologies within Rakuten?

Bahman: It's been very significant. We have businesses that are essentially completely driven by AI. For example, we have a suite of products that we call Rakuten intelligence. It is in the alternative data business for financial companies, such as hedge funds, and it is completely driven by AI. We drive insights around Ecommerce purchases that the population is making, and what brands and companies are getting traction, and we provide these insights to investors, who make significant investments based on them. Rakuten itself also makes investments based on them. Some of our largest Rakuten investments that you may have heard about were based on these insights.

So, we have businesses that are completely based on AI, and AI has percolated throughout Rakuten in a lot of different lines of business that we have. We have a large fintech business that heavily relies on AI—for instance, we have a very large insurance business. We have Ecommerce, which heavily uses AI. We have a mobile network now, and we are working heavily on AI for mobile networks. We have a medical business called Rakuten Medical, which is aiming to cure certain types of cancer, specifically head and neck cancers. And we use computer vision to detect cancer cells. So, it's really all over the place; we use it in a lot of different places. As I mentioned, some of our businesses are completely driven by it.

Ben: So, let's get into the nuts and bolts. You run teams within Rakuten focused on data science

and AI. What are some of the key roles and job roles within your AI team?

Bahman: We have four categories of people in our AI work. The first type is product managers, who essentially drive requirements for the products we build. We have data scientists who develop the models. We have machine learning engineers who essentially help implement the models and the required pipelines for them.

We also have an additional role that we call data wranglers. Data wranglers are people who help, at a very basic level, with preparing data for the models and evaluating the results of the models. So, they feed and evaluate the models developed by data scientists. We can go into detail on any of these roles if you like.

Data wranglers are people who help, at a very basic level, with preparing data for the models and evaluating the results of the models. They feed and evaluate the models developed by data scientists.

Ben: First off, the distinction between data scientists and machine learning engineers—do you think that, over time, this distinction will go away, assuming that the tools for deploying models to production become more tightly integrated with the start of the pipeline and the model?

Bahman: Yes, that's a very good point. Definitely. The distinction between data scientists and machine learning engineers is a spectrum. Data scientists, as you would expect, are people who know statistics, machine learning and algorithms, and design and develop the AI models. The role of machine learning engineer is a little bit more of an emerging role. It's a mix of data engineer and DevOps, with some basic understanding of data science. As I mentioned, they build scalable pipelines and infrastructures, and help the data scientists who scale and operationalize their models.

Now, there are two trends. One is that as time goes on, people become more sophisticated in their skills: machine learning engineers will probably learn more about data science, and data scientists may learn more about engineering. Also, as you mentioned, there are technologies and platforms that are helping data scientists to, for instance, deploy their models more easily. On the other side, there are technologies such as AutoML that allow an engineer to essentially develop models with minimal knowledge of data science. So, there will be a convergence between these roles. But it's a matter of time before that happens—right now, they're still two distinct roles, to some extent.

Ben: You mentioned a job role called data wrangler, which sounds to me like someone who helps at the early stage of the process—getting the data ready and things like this—but also has a role toward the latter stage of the process, with model testing and validation. Is that right?

Bahman: Yes, definitely. Look at the steps it takes to develop an AI product. The data preparation step, preparing the data for the models, and then evaluation of the models, these steps actually do not require a lot of technical sophistication. But they take a very large percentage, in some cases in some projects even up to 80%, of a data scientist's time for that project. At Rakuten, we hire people at very junior levels for the role of data wrangler, and we typically hire them in offshore regions. We provide them with just enough training to be able to do these steps—the data understanding, data prep, and evaluation.

In these steps, essentially what they do are things like data discovery, selection, data pre-processing, transformation, normalizing the data, and labeling and cleaning it to prepare the data for data scientists for modeling. After the modeling, they also do the model evaluation, and interpretation and error analysis. There are three major benefits to this. First is that it significantly frees up data scientists' time to focus on the core modeling challenges, which not only makes the data scientists happier and reduces their churn, but also reduces the number of data scientists you require. This is great because among the four types of roles we discussed, data scientists are typically the bottleneck and the hardest to hire and retain.

The second benefit is that data wranglers are significantly less costly than data scientists, so this way of structuring the resources has significant financial benefits as well. The third benefit is that as data wranglers get more experienced during the projects by working alongside more senior people, they become more sophisticated in their technical skills. We've seen that they tend to grow into one of the other three roles—data scientist, ML engineer and product manager.

The first thing I recommend, even before getting into the organizational structure, is putting people into teams that have end-to-end responsibility. This way, they can make sure the teams have the resources they need to deliver AI products into production. This also allows teams to be measured and evaluated on business metrics.

As they grow, they can then train their own successors—the next batch of data wranglers. And this provides very strong economies of scale for the team. Notice, that same thing does not hold, for instance, for Ph.D.-level data scientists: Ph.D.-level data scientists cannot train the next batch of Ph.D.-level data scientists while they are working at the business. This is something specific to the data wrangler role. We have been heavily leveraging three benefits that come with this role in how we set up our teams: reducing the need for data scientists and the churn of data scientists, the financial benefits, and the economies of scale.

Ben: This is very interesting for a couple of reasons. The first is, having the data wrangler do the

model evaluation and testing is actually great because, in general, you don't want the person who developed the model to test their own model. In software engineering, for instance, you have someone doing QA.

Secondly, as you mentioned, it allows people a growth path, which is so important for talent retention reasons. So, this notion of data wrangler, as you described it, was that something you came up with or something that was already in place when you joined Rakuten?

Bahman: I wouldn't take 100% of the credit for myself because this was something that was developed between me and some of my peers. But yes, it was developed after I joined, especially this way of structuring our resources and mapping the resources and coming up with crystal clear role definitions for who is doing which part of the process, which steps—this was something that we developed after I joined Rakuten. It's now become a pretty nice practice.

Ben: So, we talked about the roles. The next question is around organizational options. There's the notion of having people decentralized, decentralized themes, or centralized data organization or semi-centralized. Which one do you use?

Bahman: That's a very good question. Before we get into the organizational structure, we should discuss how we structure these resources into teams. I have seen in a lot of companies, the four roles we talked about are structured in separate teams. So, for instance, you have data scientists in one team, data engineers or ML engineers in another team, and maybe ML operations people in yet another team. Essentially, then, the working model is that you throw things over the wall. The data scientists develop some prototype or model; they throw it over the wall to the engineers, who implement it; and then they throw it over the wall to the ops people, who productionize it and manage it in production.

This is a common practice, but we do not do it. There are very significant drawbacks to it that people should really think about if they decide to go with this, and I would recommend against it, basically. The first drawback is that it is extremely inefficient. Each time things are thrown over the wall, there is information loss. This can create a gap, for instance, between the prototype that's been developed by the data scientist and the implementation that's been done by the engineers. These gaps can make things very difficult to iterate; that's the first drawback.

The second drawback is that the ownership of the AI models is not very clear when you're throwing things over the wall. Models require ongoing care and maintenance. Typically in this model, the operational people end up having to own the models, but they do not really have the capabilities and skills to do that. They may be able to tell you if the model is running and if it is meeting SLA's on latency, but if there is data drift and the model starts doing the wrong thing, the ops people do not have the capabilities to tell you about that, and that can create significant risks for the business. The third problem is that, because these people are in different teams and they're measured differently—for instance, data scientists are measured based on the accuracy of their models,

whereas the engineers are measured based on the efficiency of the implementations that they have done—if a problem comes up, it can easily turn into a blame game, and nobody will take responsibility for it. For instance, in one of the projects we had in my own organization, our data scientists had developed an algorithm that was getting really good accuracy metrics, definitely meeting the accuracy metrics we needed, but when we ran it at scale, it was using so many resources, so much memory specifically, that it just crashed instances.

If you have people in different orgs, it's very easy for the data scientist to say, "Well, I'm meeting my accuracy measurements; it's that engineer's problem to solve it." And the engineer will in turn say, "Well, I cannot run this thing; the data scientist has to go back and fix the algorithm." And nobody takes responsibility in this case. But if they are on the same team, it becomes a collaborative effort instead of a blame game.

So, the first thing I recommend, even before getting into the organizational structure, is putting people into teams that have end-to-end responsibility to deliver AI products into production and making sure the teams have all the resources they need to do that, and then measure and evaluate the teams based on business metrics. So, the members of these teams will be part of the same functional unit, and they'll come together around a specific project with concrete business deliverables. After the project is done, the team members can move on to other projects—that is, these teams have an ephemeral nature. As part of your AI organization, you can have multiple such teams, which allows you to scale. Also, as these ephemeral teams perform these projects, if they develop reusable materials, like continuous integration and continuous deployment (CI/CD) pipeline features for a feature store, they can contribute them back to a central repository so future teams can also take advantage of them.

Ben: That latter point assumes there's a mechanism for making that contribution.

Bahman: Definitely. If your AI organization starts to scale, then you can have a platform sub-team that creates some of the common platforms that are needed. The one thing you need to be very careful about when you do that is to make sure the platform development does not turn into a pure engineering exercise that is not tied to any concrete business metrics. But yes, you may have a platform team that is developing the CI/CD pipelines and developing the features store where the features will be shared.

So, that is about the team. Now, when we go into a large company and we want to look into the organizational structure, as you mentioned, there are three options: decentralized organization, centralized, or semi-centralized, let's call it. What we now have in Rakuten is a semi-centralized organization. That's really the solution most enterprises will probably be looking for. But let's go through the other two options quickly.

Ben: Let's just jump into the semi-centralized, because I'm curious.

Bahman: Definitely. The idea is that you have a centralized AI organization, which controls all the resources, the people in those four roles we discussed, but then assigns end-to-end teams of people to projects within different business units or within different functional units. This allows for direct business interaction for these teams, which facilitates business understanding and resolves delivery issues. This is one of the issues that comes up with a pure centralized organization—the business understanding a lot the time is missing because they are not working closely with the business.

Ben: So, Bahman, a quick question. In this case, let's say I'm part of an end-to-end team working on churn. Is it possible for me to also be part of another team simultaneously working on recommenders?

Bahman: It depends on the scale of the project. A lot of times, the context switch is very difficult for people. You can do that over time. That's actually one of the benefits of the semi-centralized organization: you can have a rotation program. One of the drawbacks of the decentralized organization is that people work on the problems from within one functional unit or one business.

AI is a global phenomenon right now, and there's AI talent all over the world. You don't need to limit yourself to looking for talent in your immediate geographic vicinity.

Ben: Again and again.

Bahman: Exactly. They get bored and they leave. But if you have a semi-centralized organization, you can actually create rotational programs where, yes, you allow a person who was working on a churn project for six months to move on to another project for a different business unit and work on recommendation systems. You may not want to do that at the same time, because it becomes a lot of mental overhead to switch context. Also, there's a lot of institutional knowledge that goes into any of these projects. You need to know the use case, you need to know the business requirements. So, it may be difficult to work on two projects at the same time for, let's say, any given data scientists, but yes, over time they can definitely do that. That's one of the significant benefits of a semi-centralized organization.

Ben: AI, data, and machine learning are all hot topics. Companies are chasing after a limited but growing talent pool. What are some of your tips as far as attracting and retaining talent?

Bahman: That's a very good point. The talent gap is a major obstacle for most enterprises on their path to AI adoption. A typical AI team member tenure is less than two years, and it takes around three months to ramp them up. So if you cannot solve the talent gap, it is very difficult to execute on AI initiatives. Another challenge, as you mentioned, is that AI is a very hot topic right now. When you start recruiting, you receive a lot of resumes, but typically there is not much quality in that pile of resumes.

I can share three recommendations here. The first one is to hire for the role that you actually need. We discussed the four roles and how they map to the different steps of what it takes to develop an AI product. Make sure you're identifying the role that you actually need and hire for that; do not try to hire a "unicorn" who can do everything, because most likely you will not succeed. So that's number one, being very targeted.

The second recommendation is that AI talent right now has a lot of options. Businesses need to differentiate themselves. For instance, if you have a particularly interesting and challenging project that you would have these people work on, you can pitch that. Or if you have career development—take career development for your folks very seriously, and if there's career development that they can achieve in your company, you can also pitch that. One of the things we do, for instance, is explicitly ask our candidates where they want to be in two years, sometimes even in a very direct way. We might ask them, what interview do you want to be able to pass in two years? A lot of candidates actually find this very surprising. So, we'll make sure to help them pass that interview. If they join, we actually do try to help them get virtually anywhere they want. Or, if your company has a particularly compelling social mission, you can also use that to differentiate yourself.

The third recommendation is around where to find talent. You can find talent internally in your own company by training your existing quantitatively-oriented employees for the roles that you need in your AI department. You can also find talent externally. In that case, my recommendation would be to make sure you have a global perspective. AI is a global phenomenon right now, and there's AI talent all over the world. You don't need to limit yourself to looking for talent in your immediate geographic vicinity, especially if you are somewhere like Silicon Valley; you may find it very difficult to hire top talent.

Ben: This is related to a question I wanted to ask you around this topic—what's your position on remote workers and remote teams?

Bahman: We definitely have a very global perspective, a very global view into our hiring. My own teams right now are located on three different continents. We're actually growing in terms of the number of locations we have. I would definitely recommend having a global perspective and building a team in a globally distributed fashion. There are three things you need to consider, though, if you're doing that. Number one is that you need to make sure you have the right framework for collaboration and coordination of resources, and also capable management who can manage a distributed team. That's number one. The second thing you need to keep in mind is to make sure to treat everyone as one team, not as first class and second class based on their geographic location.

Ben: Or say, this particular set of narrow tasks are for our team in India.

Bahman: Exactly. We don't do that. We really treat everybody in my organization as part of the

same team. They work on the same types of things; they collaborate very closely. It's not like, as you mentioned, we gave some team grunt work, or that we look at them as an outsource resource. We really view everybody as one team. That's the right mentality to have.

The third thing is that, to the degree that you can, have periodic face-to-face meetings for your team members. That will help a lot with team dynamics. One of the things I do is make sure to visit my teams fairly frequently. It means a lot of traveling, but it helps a lot. If you have your team members meet each other in person, that helps a lot, too. The benefits far outweigh the challenges.

Ben: Our mutual friend, Paco Nathan, and I have done a series of surveys. One of them was about enterprise adoption of AI. We found, actually, for companies just getting started with AI, one of the main bottlenecks is finding use cases, and also convincing the rest of the organization that AI is worth investing in. I'm assuming, since you guys are further along, this is no longer a problem; obviously, you guys are convinced that AI is useful and that you have a whole portfolio of use cases to attack, correct?

Bahman: Yes, that is true. One of the big challenges we have seen is that, in a lot of cases, companies start from the latest tools or latest technologies. The question they ask is, what can we do with deep learning or what can we do with TensorFlow? That is really the wrong place to start. Or, in a lot of cases, they try to imitate some of the largest, let's say, cloud service providers, such as Google. They say, Google is doing this, so we should be doing the same. In a lot of cases, the answer is, no, you shouldn't. That's one quick thing about that: think about a chart with an X axis and Y axis. On the X axis, you could look at the stages of developing an AI product from developing the kind of infrastructure that you need for it, and then doing some simple machine learning. And then the third stage, let's say, developing advanced AI models. And on the Y axis, if you look at the value that you get, in a lot of cases, the curve you would have is an S shaped curve, or for more technical audiences, it looks like a sigmoid function. If you are a company like Google, that last bit of improvement you get in terms of the value you get can far outweigh the costs. If you are a company like Google and you improve your click-through rate on online ads, that's a lot of money. That does justify the cost to build the resources to do that.

For a lot of businesses, that actually does not. What I recommend as an alternative is to start from your own business, look at how your business creates value and how it differentiates itself from its competition. We really have a strategic view. If you think about it, in any business, there are three ways to create competitive differentiation. The first one is through operational excellence, which is essentially developing and delivering products and services more efficiently. As a result, at a lower cost point than any competitor. AI can help with this strategy, if this is how your business creates value. AI helps with this strategy in various ways—for instance, automation and augmentation, or better forecasting of inventory and demand, or improving your logistics, or doing better QA to reduce waste. If this is how your business creates differentiation and competitive differentiation, these are the kinds of things you want to look at. We have some projects in Rakuten where we have largely automated parts of the tasks that we were doing in a fairly manual fashion.

The second strategic approach for businesses to create differentiation is through product leadership, also known as performance superiority, which is simply having the best and most performing product on the market. An example of where this shows its importance is in risk models in insurance or banking. If this is how your business operates and how it creates value, AI can help with this strategy through what is known as the virtuous cycle of AI—namely, the data captured from product usage is used to improve AI models; better AI models result in better products. Then these better products attract more customers. And then more customers create more data, which then improves the AI models even further. And then the cycle builds on itself.

These things typically result in winner-take-all markets. This is how a product such as the Google search engine has reached impenetrable dominance. If you want to go this route, one thing you need to think about, like the virtuous cycle of AI, is your cold-start strategy—that is, how you're going to bootstrap the cycle, especially if you're launching a new product. One of the best ways to do that is through human-AI integration. Essentially what that means is that in the early phases of your product, you can leverage human intelligence instead of AI. As the product gets more traction and more data is collected through it, then you can increasingly automate aspects of the human involvement in the product or service. Overall, this is the second strategy, the product leadership.

The third one is customer intimacy—simply knowing your customer very well and better than anybody else, and utilizing that knowledge to serve them better than any other competitor. AI can help in this direction as well. For instance, through things like churn or lifetime value (LTV) prediction, if your AI model indicates that a high LTV customer is about to churn, you may want to provide them with promotional offers, or if they have reached customer service, you may want to escalate their query to a manager who's better equipped to serve them.

One last thing is, apart from these competitive strategies, any business also has to meet various regulations, and AI can help with regulatory risk reduction as well. As an example, my company worked on personally identifiable information (PII) detection and anonymization in various documents that you may have all over your business to meet privacy regulations, such as GDPR and so on.

You start from the needs of the business. With any of these broad strategic approaches, what you're looking for are problems for which the business logic is not easily specified. But you either have lots of legal training data lying around, or you can efficiently create lots of legal training data for it.

Ben: This could be a simple checklist—I don't know if you agree with this: so, the first thing you ask is, is the task I'm talking about here data-driven? And then secondly, do I have the data to support the automation of this task? And then finally, do I really have the scale to justify automation?

Bahman: Yes, that's a really good point. That's a very nice checklist. Definitely. Those are all things

that need to be considered as you're deciding what to work on.

Ben: I love your framing for the strategic view for AI. Another possible way to frame this to be even more specific for applications of AI—AI might be able to help you improve your decision-making or your operational efficiency. It might be able to generate revenue or add to your existing revenue, or help you with fraud and forecasting to predict or prevent fraud or forecast or minimize risks at a high level. Those are some classic use cases.

So, I have one final question. You're different from some of the other AI and data science leaders in the enterprise in that you were deeply technical before. Let's take a hypothetical situation: there's a new development in AI but you're not quite as familiar with it anymore because, obviously, you're not as hands-on anymore. What level of understanding and trust in your people do you have to have in order to green light something that you yourself may not be familiar with anymore?

Bahman: Yes, that's true. In my case, fortunately, so far that has not been a problem. If it happens in a number of years, maybe. But so far it has not been a problem. It's ideal to avoid that situation in the first place. If companies, for instance, are hiring leaders for their AI organizations, ideally they would like to have someone who does have that technical expertise. It may be very difficult to hire these kinds of folks, who do have deep technical expertise and also are solid on the management and leadership fronts. But if a company can do that, that's ideal. If that is not the case, then yes, you would need to rely on your technical folks and your technical resources. In that case, the way that you manage it is by key performance indicators (KPIs).

You don't need to know exactly how a specific algorithm operates if you know what kinds of KPIs you're looking for at the end of the day. In a lot of cases those KPIs are business KPIs, not just technical KPIs—and that's actually preferable. If you have those business KPIs, then the internals of the algorithm may not matter as much. If your folks develop those business KPIs, if they reach those business KPIs, then you may not need to know the internals. It does become a little bit more difficult. For instance, if you are looking for some level of accuracy that your product needs, but you are not getting there, it may be a little bit more difficult if you don't have that deeper understanding about whether or not it's even achievable. You would then have to rely on your technical resources. In that case,the only thing you can do is make sure you are taking care of that in your hiring process, and hire people who have a strong track record of delivery, and then you need to trust them and let them lead the way.

Transparency and Explainability in Machine Learning

Krishna Gade is the founder and CEO at Fiddler Labs, a startup focused on helping companies build trustworthy and understandable AI solutions. Prior to forming Fiddler, he led engineering teams at Pinterest and Facebook. As machine learning gets embedded in more applications and products, the need for trust, transparency, and explainability take center stage. Multiple stakeholders—users, developers and engineers, and regulators—need to be able to trust, operationalize, and oversee these complex systems. Each of the previous chapters already touched on the importance of model explainability. Gade closes this volume with a tour through the world of explainable and trustworthy AI.

Highlights include:

- The biggest problem today with machine learning is how to operationalize it. Beyond regulatory requirements, trust and explainability are critical to AI adoption.
- Healthcare and financial services have been at the forefront of developing and adopting tools for model explainability. But Gade cites other domains that are starting to develop similar tools and requirements.
- Gade presents a set of guidelines for companies that want to develop more explainable and transparent machine learning models.
- He also cites emerging tools and trends (including job roles focused on model risk), as well as a growing awareness among companies that model accuracy alone is no longer sufficient.

This conversation has been edited for clarity.

Ben Lorica: You're a long-standing data engineer, and there are many aspects in an end-to-end machine learning (ML) platform: what made you decide to focus on explainability?

Krishna Gade: That's a great question. Machine learning has been part of technology companies for the past two decades. When I was a search engineer at Bing, I remember they were one of the first teams to productize search ranking. So, machine learning has been in play for companies that had large amounts of data, had compute power, and could devise algorithms to process this

data. However, in the last few years, it has broken through into general enterprise. You can see companies in finance, healthcare, oil and gas, and other sectors trying to deploy machine learning.

The biggest problem today with respect to machine learning is how to operationalize it, and how to take this technology and affect the business process workflow. There is a big gap between how machine learning technology works and how business owners and business leaders operate today. There's a gap in terms of literacy. There's a gap in terms of trust. "How do I trust this machine learning decision?" Then there's an aspect of regulation coming up because of increasing reports around bias in machine learning.

The biggest problem today with machine learning is how to operationalize it.

Companies need to be able to look at how machine learning is being built and how the models are actually working when they're deployed. Therefore, explainability becomes a lens to look at what's going on; it provides visibility and layers of insight so you can build trust with AI. We feel this is the last link, the missing link, for AI adoption. If we can crack it, then we can see AI adoption perforate across the board in enterprise. That's why I've been working on explainability in AI.

Ben: I imagine you're someone who's very thorough and really did a deep dive into the ML landscape. Obviously, when you decided to become an entrepreneur, you probably examined the ML ecosystem of tools, companies, and open source projects. What, in particular, about explainability convinced you that it was the right direction?

Krishna: I wanted to work in the AI space. I wanted to build a company and enterprise in the AI space. We were looking at different problems in the AI space, but we noticed there were a lot of enterprises trying to productionize AI. They had been trying to build models, but the models were sitting in labs, so they were not being operationalized. We found that the missing link preventing operationalization was a lack of transparency and a lack of trust with machine decisions. My work at Facebook, fortuitously, was actually on building explainability for the News Feed models internally. We found there is a nice match between the work I did at Facebook and the need in the AI industry, to really operationalize AI to affect the businesses.

Then there was also a regulatory pressure companies started having to deal with. There was the Algorithmic Accountability Act that was introduced in Congress this year. There was GDPR a couple of years ago, requiring that companies provide transparency to customers for automated decisions, a sort of trust building from your machine decisions, and so on.

All of these things culminated into a problem that we felt had a really big technical challenge to solve—businesses needed tools and technologies to actually operationalize AI. There was also a societal impact that we could make. If we could help companies build AI that is bias-free, then

that would be a net benefit for society, too. You don't really come across these problems often. We jumped at it and wanted to start the company in this space.

Ben: I've chaired many conferences in data science, big data, and AI over the years. And, to confirm what you're talking about, I find that talks on explainability are well received and well attended. Particularly when the talks are presented by speakers from highly regulated industries, like healthcare and finance. So, there are companies like SAS, Fair Isaac, and banks themselves that give talks. It seems the audience really likes their perspective. Now that you're deeper into the world of model explainability, besides healthcare and finance, what other industries have been aggressively talking about and trying to tackle explainability?

Krishna: Finance industries definitely are the foremost, because they see a huge need and value for AI. There are increasingly diverse data sources that are available for them to take advantage of to increase their top line, to provide credit access across the board or reduce fraud, for example. For them, explainability is important because they already have pretty well-defined regulations like SR 11-7, the Fair Credit Reporting Act, OCC guidelines, and so on. For them, it's a natural move from old school quantitative models to machine learning models. They need platforms that can actually minimize the risk of using AI, and that can provide governance capabilities and continuous monitoring around the area. For us, it felt like a natural fit to focus on that industry.

It's the same with healthcare. Healthcare is a little bit behind finance in terms of AI adoption, but we see companies trying to use machine learning to cure diseases and they need to explain to physicians how the technology works.

Apart from finance and healthcare, we also see machine learning explainability use cases across the board in industries like oil and gas, for example. I was talking to a couple of companies in this space where they invest millions of dollars in deciding when and where to dig an oil well and how to decide where to explore for oil. They are trying to use deep learning to figure this out. You can't actually make and execute on a decision delivered by a machine, though, without having a geologist review. So, there is a human review of the machine decisions required because of the risk of mistake. That's where the benefit of explainability becomes obvious.

Another domain we are looking into is human resources. Companies are trying to automate recruiting and HR interviewing. They need to explain to candidates how they're using the automated AI system to interview or screen, how are they selecting them from job recommendations, and so on. There's a huge aspect of fairness here as well. This is another industry where explainability is super important.

Ben: For people who don't follow these things very closely—many people have heard of GDPR and now the California Privacy Act. Both of these sets of regulations have some sort of aspiration as far as model explainability as well, correct?

Krishna: That's correct. They don't go into the details because these are high-level guidelines. They basically illustrate that if you're making an automated decision, then you need to provide transparency for the decision. For example, in the case of GDPR, it's quite clear that Article 22 states that if a company is making an automated decision, let's say issuing a credit card or rejecting a loan, they have to provide an explanation for the customer as to why they made the decision they made. In certain domains, for example, in the financial industry, some of this already exists. For example, there is this concept of adverse action notice that notifies a customer that you're unable to approve a loan or a request in terms of increasing the credit line. But the GDPR is generalizing this across the board, requiring industries to provide transparency to automated decisions. A lot of the regulations the United States has created are inspired by GDPR, whether that's California Consumer Privacy Act, or the three bills that were introduced into Massachusetts, Washington, and Illinois recently, and then there's the Algorithmic Accountability Act that was introduced in Congress, all of which speak the same language.

Ben: As you point out, there are certain industries that are more aggressively pursuing model explainability, like finance and healthcare. Based on what you know about these regulations, at some point in the future, will model explainability be something that anyone in any industry has to be aware of?

Krishna: Absolutely. It's not just regulation—we don't just believe that explainability is a tick box in a compliance checklist. We believe that if you don't actually pursue explainability, then companies could hurt their brand and could actually get into customer mistrust issues.

We have seen this happen recently in the alleged gender bias case with respect to Apple Card. We don't know the actual underlying issue because there's a regulatory probe going on, but we know there was an incident where a customer applied for the Apple Card, and he and his wife had a large discrepancy in their credit limits. When they approached customer support, they didn't get a proper response; the response was: "It's just the algorithm."

We believe that if you don't actually pursue explainability, then companies could hurt their brand and could actually get into customer mistrust issues.

When incidents like this happen, customers lose trust and brand reputation erodes. Companies need to care not just about meeting regulations, but also making sure they are delivering the products and they're actually keeping the customer's trust in place. We believe it's much more important. Therefore, explainability is a must-have. For any company thinking about AI, that's going to affect their customers in the future.

Ben: We've established that explainability needs to be a high priority item for anyone or any company serious about machine learning. What are your top three guidelines for companies that

are just beginning to tackle explainability?

Krishna: I would start by defining the problem you're trying to solve. Clearly define the problem for the machine learning task at hand to see if you really need machine learning for that. There are cases where machine learning may not be suitable, especially when you don't have the right amount of data or there is inherent bias in data. A good way to start is to explore data. We call it pre-model explainability. Even before you build the model, you can use explainability techniques to understand the data, to explore data, to see potential bias in data, and to see whether you can actually use the training dataset to build a model. Machine learning is garbage in, garbage out. If you feed in biased datasets, then you will build biased models. You can use explainability techniques to look into your dataset, explore it, and see the value it's providing to your model, and whether there's bias in the data. That's number one.

Number two is, when it comes to model building, there is a decision one needs to make. Do you really need to build black box models, also known as deep learning models, for your use case, or can you get away with "explainable" models? Whether they're simpler models like logistic regression or more explainable, inherently interpretable models like GAMS and GA2Ms, if you can actually build an inherently interpretable model, it's probably a good way to go about it. If you don't, but you need to capture unstructured data, or you need to process image data, or you need to bring together diverse data sources, or you need to achieve a certain level of model performance, maybe you need to go into advanced modeling techniques like gradient boosted trees, random forests, and deep neural networks.

When you actually do these things, then you need what we call a post-model explainability, which means now you have built a black box model and you cannot really know how it is working in detail. You need post-model explainability techniques so you can explain the individual decisions of the model to your technical users and to your business users, so they can ask questions like, "Why is the model denying this loan for me? Why is the model diagnosing my patient this way? Why is the model telling me there's an oil well here?" or whatnot. These are the questions you can actually answer. I would say, think about it and holistically start with data explainability, and then think about explainable modeling if that fits your use case. If you have to build black box models, then definitely think about post-model explainability.

Ben: These are great tips. Obviously, tools like what you folks are building at Fiddler are going to be very helpful and very convenient for many companies. What about on the staffing side of the house? Let's say, there's a team of people who are building the models, then there's a team of people who are auditing and validating the models, and auditors and validators that also include explainability as part of their checklist. So, are there staffing changes that companies also have to consider?

Krishna: Absolutely. That's a great question. If you use this "jobs-to-be-done" framework, identifying who will actually be doing this job in the future, we believe there is a role for some sort of

a model ops or model QA that is responsible for monitoring the model performance, understanding the model, testing the model before launch, validating for bias, creating compliance reports for regulators if there is a need for the industry, and so and so forth. On the "old school" systems, we have software testing and software QA people, or DevOps people who are responsible for testing the software and making sure it runs reliably. I would imagine in the future we will have these kinds of dedicated people for ML ops and ML QA jobs that deal with explaining models, monitoring models, and validating models.

I imagine in the future we will have dedicated people for ML ops and ML QA jobs who deal with explaining models, monitoring models, and validating models.

There's already an example of this in the financial industry. For example, let's say a bank wants to launch a new credit risk model. They cannot just build the model and launch it, as a tech company might. They have a very strict process to launch models, even if the model is not a quantitative model, or it's not necessarily machine learning. When a business unit wants to launch a new credit risk model, they create a new quantitative model and they pass it onto a model risk management team. This centralized model risk management team makes sure they are launching the model in compliance with regulations, and they also make sure they are reducing the risk around this market, reducing operational risk, reducing market risk, and reducing credit risk from the bank's point of view.

When they do a whole bunch of testing, essentially, they're trying to break the model and/or come up with an alternative model. Then they provide feedback to the business unit to improve the model. After this happens, the model goes through a third line of defense, which is the model audit, which looks into the reports generated by the risk management team and then approves for launch. So, there's a pretty elaborate, well-defined workflow that exists, which the rest of the industry could adopt in the future to have a holistic model risk management in their system.

Ben: Model explainability is a hot research area for people in the machine learning world. There's a lot of research and development in the model explainability side. At a high level, are there any exciting developments in model explainability that our audience should pay attention to? For example, you already alluded to black box and really complex, large, deep neural network architectures. Is there equivalent progress in model explainability that our nontechnical audience should be aware of?

Krishna: Absolutely. In the last few years, the model explainability research has really taken off. If you saw the last NeurIPS conference, there were a ton of papers and workshops about model explainability. This year, we actually added a four-hour tutorial working with the LinkedIn Critical AI team at KDD, in Alaska, about explainable AI. They're doing one at ACM FAT* next year, that's a fairness and transparency conference, and another one in AAAI. There's a lot of interest in this topic.

If you think about model explainability algorithms, you can broadly classify them and take the algorithm model as a black box and try to explain it. For example, take an algorithm trying to build a proxy model, or a surrogate model, for a local set of data points, and then try to explain it. Essentially, you're trying to rely on the surrogate algorithm to almost reverse engineer what the original algorithm might be doing. In another case of algorithms that are trying to take advantage of game theory techniques, inspired by a classical economics paper on the Shapley value that came out in the '50s—there are a bunch of algorithms being developed using this notion of Shapley value.

Ben: The main question is: are there models that are still hard for these model explainability techniques to shed light upon? Or is the progress on the modeling side and the model explainability side pretty much moving in parallel?

Krishna: Well, research is ongoing. You cannot just say today that every model in the world is explainable. That's a very tall claim to make. There's a level of explainability that you can make.

Ben: Are there enough models that can be explained that the typical enterprise can use?

Krishna: Absolutely. A typical enterprise might be using boosted trees or random forest when it comes to traditional machine learning models, or they may be using LSTMs, or convolutional neural networks when it comes to deep neural networks. They can be explainable using these algorithms. For example, you can use Shapley value techniques to explain, say, the traditional boosted trees algorithms, or you can use newer techniques, like integrated gradients, to explain some of the deep networks. These techniques are available, but obviously, new research is continuously improving and updating these algorithms.

Ben: Also, it seems, as far as tools and products, there are more products on building models rather than explaining models at this point, correct?

Krishna: That's correct. There has been a lot of effort in the last five or six years in automating the training process. There are automated machine learning platforms that customers are using to increase productivity, trying several models in parallel to automate the whole process of training a model. That's become a well-studied problem, with a variety of products customers can select in the vendor landscape today. However, there is not enough in the model explainability world, to explain issues like, "Now that I have a model, how do I operationalize it? How do I explain it? What's the interface explainability?"

The other challenge in explainability is not just picking the right algorithm, but what is the interface? Because the explainability could differ. When I was working at Facebook, when we built the explainability platform, the insights we showed to developers would differ from the interface we would actually show to the product managers or the business operations folks, and that differed

from the interface we would show to the end users—because everyone's needs are different. A developer might be interested in debugging the model, finding out why certain predictions are the way they are, and finding the root cause of why something happened. Whereas a product manager or a business operations person might be looking at the strategic questions, like "How can I improve the model? Why is the model behaving this way?" For example, in the case of a credit lending model, an analyst might be asking, "How do I develop a better credit strategy? How do I improve loan processing for these sorts of loans?"

Then from an end user perspective, they care about transparency. They don't really care about how the model is working. They want to know how they can get a better decision. If I'm refused a loan, I want to know how to get a loan approval in the future. There are various needs, so explainability services need to cater to these needs in order to be functional.

Ben: That was the easy part of the interview. So, now we're going into the harder part, which is basically to put you on the spot and make you prognosticate and predict two to four years ahead. If you look two to four years ahead, what will the state of model explainability in a typical enterprise look like?

Krishna: If you take four years out, our vision and our belief is that explainability will be big throughout the AI world. It won't be an afterthought. For example, when we started working on explainability at Facebook, it was kind of an afterthought. We basically built all these complex models for News Feed, boosted trees and the other networks. Then we came to a point where News Feed became a very big system that was hard to debug. So we had to build these tools.

A modern enterprise that is trying to introduce AI will probably think about explainability-first approaches, where they bake in explainability throughout the workflow. So what does that mean? It means using explainability during the training process to debug the model; using explainability to make sure your dataset is clean and bias-free; using explainable as a justification mechanism during prediction so that you're always not just providing decisions but also providing justifications; using explainability as a way to QA models, like to compliance test the models around fairness and around all kinds of regulations that one might need to satisfy; and then using explainability to slice and dice models, to analyze them and figure out the root cause of a certain incident. Then monitor the models, understand the problems within monitoring and explain why these model predictions are outliers, why there is a data drift happening, or why model predictions are drifting. Throughout the workflow, there needs to be explainability baked in. That's the future we foresee that every enterprise will have to do in the next four years.

Ben: What about the incentive structure? Let's say I'm a deep learning engineer or a machine learning engineer, and I am able to productionize a new model that produces impressive ROI, then, obviously, I get recognized and maybe even get rewarded. But what is the incentive structure for the model explainer?

Krishna: That's a great question. Firstly, I think businesses are realizing that accuracy alone is not important. For example, if you're a machine learning engineer and you get very high accuracy during training time, you might get really suspicious about the model because you don't really know if the model is overfit when it performs well when you deploy. Accuracy on its own is not the only metric that developers have to consider. Their main goal is to solve the business problem using machine learning, which means providing the best possible model that can work in the wild and providing a base so they can build trust with their business users or end users. It's no longer the case that, "I have built the most accurate model and I'm done with it." Is the model actually working? Is the model actually affecting the business metric? Is the model actually serving the user the right way? Are your business users on the other side of the table able to trust the models enough to use them in their day-to-day work?

Businesses are realizing that accuracy alone is not important. Their main goal is to solve business problems using machine learning, which means providing the best possible model that can work in the wild, and providing a base on which to build trust with their end users.

Increasingly, when machine learning is being used in a way that there's an end user who's consuming the outputs, it's almost on the developers to build the trust, to actually realize this huge impact, even though today they may not see the value right away. In terms of explainability from the developer's point of view, they will see that to have the largest possible impact when it comes to their machine learning work, they need to think about explainability.

We already see that the top-level executives and business leaders are getting it, because they want to make sure they are de-risking their AI workflow and products in the company, and they're making sure there's transparency built into their workflow so they're able to get insight and visibility throughout the business. There's obviously a top-down sort of a pressure coming, where the execs start saying, "We need to actually increase visibility." Developers are moving in that direction, but in the future there will also be a bottoms-up interest to think about explainability from the get-go.

Ben: This is more of a sci-fi question on my part, but is there an overlap between model explainability and adversarial ML? Let's say, if I have the techniques to explain the model, that means I understand the model, and then maybe that gives me a way to attack the model.

Krishna: That's a very interesting question. Now, it comes down to the policy of the company, how much transparency you want to create, and how much you want to divulge to the end users. Of course, if you were to divulge a lot of information, then obviously it can lead to vulnerability and attacks. Already, attackers, whether you show explainability or not, they will try to attack.

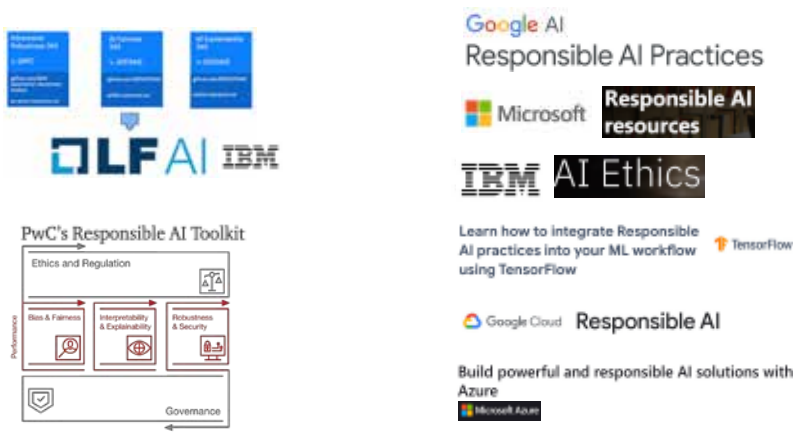
Navigate the road to Responsible AI

Deploying AI ethically and responsibly will involve cross-functional team collaboration, new tools and processes, and proper support from key stakeholders.

The use of machine learning (ML) applications has moved beyond the domains of academia and research into mainstream product development across industries looking to add artificial intelligence (AI) capabilities. Along with the increase in AI and ML applications is a growing interest in principles, tools, and best practices for deploying AI ethically and responsibly.

In efforts to organize ethical, responsible tools and processes around a common collective, a number of names have been bandied about, including Ethical AI, Human Centered AI, and Responsible AI. Based on what we've seen in industry, several companies, including some major cloud providers, have focused on the term Responsible AI, and we'll do the same in this post.

Responsible AI: Emergence of Frameworks, Tools



The term "Responsible AI" is emerging across industries to describe the ethical, responsible deployment of AI applications. Source: GradientFlow.

It's important to note that the practice of Responsible AI encompasses more than just privacy and security; those aspects are important, of course, and are perhaps covered more in mainstream media, but Responsible AI also includes concerns around safety and reliability, fairness, and transparency and accountability. Given the breadth and depth of domain knowledge required to address those disparate areas, it is clear that the pursuit of Responsible AI is a team sport. Deploying AI ethically and responsibly will involve cross-functional team collaboration, new tools and processes, and proper support from key stakeholders.

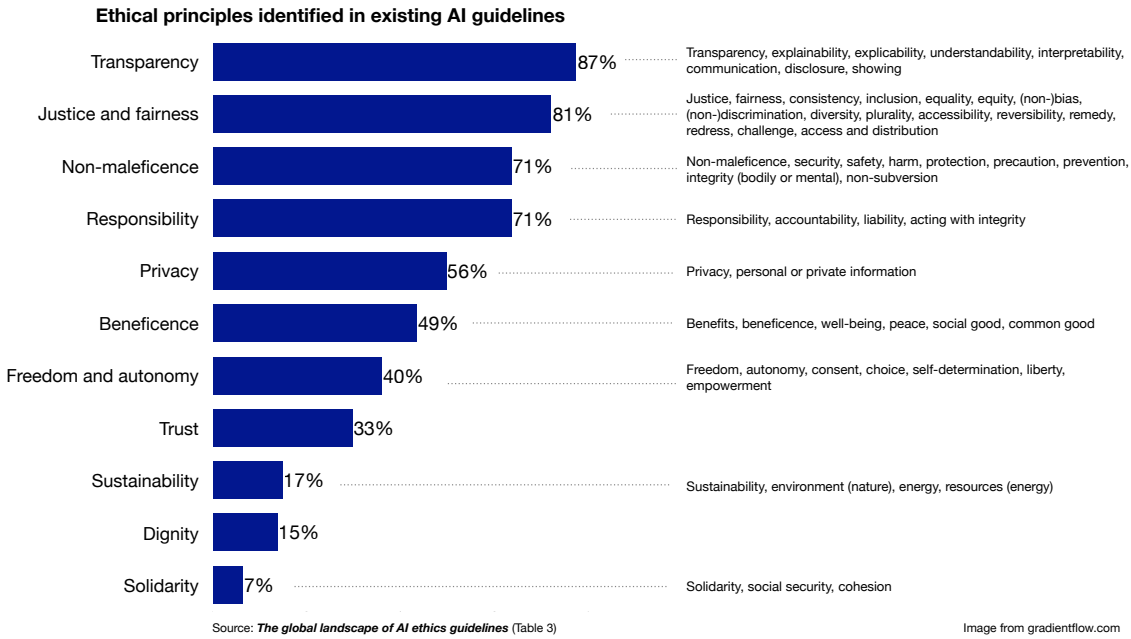
Responsible AI encompasses several areas including security and privacy, safety and reliability, fairness, and transparency and accountability. Source: GradientFlow.

In this post, we'll examine the maturity of the Responsible AI space through the lens of several recent surveys and an ethnographic study. We'll take a look at current guiding principles, what companies are doing today, and at the aspirational direction of Responsible AI practices. While companies and consumers in East Asian countries are embarking on similar pursuits—find information here, here, and here—this post, and the surveys and studies covered, focus primarily on the growth of Responsible AI in Western countries.

Guiding principles

A recent study from ETH Zurich, published in Nature Machine Intelligence, investigated whether a consensus is emerging around ethical requirements, technical standards, and best practices for deploying AI applications. The study's authors identified documents to examine by adapting a data collection protocol used for literature reviews and meta-analyses, and analyzed 84 AI ethics guidelines, written in English, German, French, Italian or Greek, issued by public and private sector organizations.

No ethical principle appeared in all 84 sources, but there were convergences around several: transparency, justice and fairness, non-maleficence, and responsibility and privacy; these principles appeared in more than half of the guidelines analyzed. Of these, transparency and justice and fairness were the most prevalent.

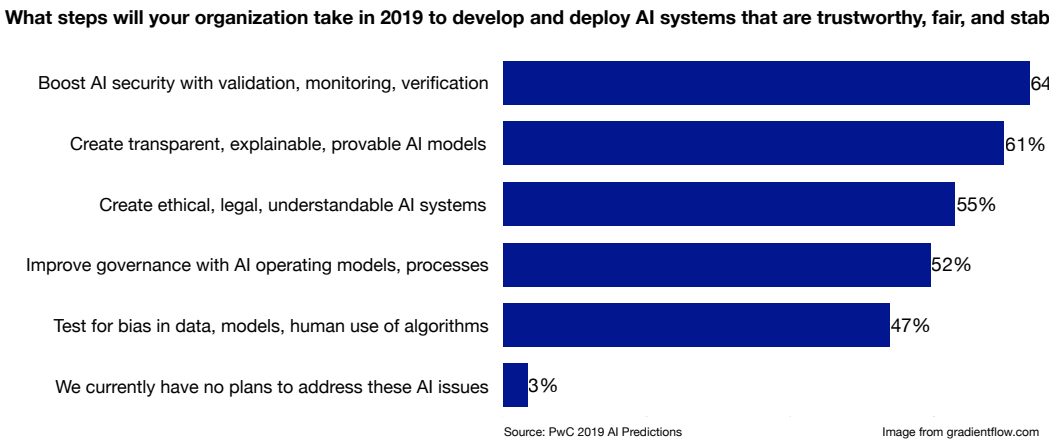


A recent study from ETH Zurich investigated whether a consensus is emerging around ethical AI principles.

What organizations are doing today

Our conversations with data scientists and machine learning professionals anecdotally agree that fairness and transparency have been the first principles they've aimed to address. The recent regulatory changes required in the General Data Protection Regulation (GDPR) and the California Consumer Privacy Act (CCPA), however, have elevated the priority of privacy, security, and transparency principles. In industry sectors like healthcare and finance that tend to be more heavily regulated or deal with more sensitive data, these principles were always a top priority, but with GDPR, CCPA and other emerging regulatory changes, they've become crucial for all organizations.

The shift in Responsible AI priorities is reflected in a 2019 survey from PwC, which surveyed 1,000 US executives. Results confirmed that security and transparency were the top two principles they intend to address; about half of the respondents also indicated that fairness—or testing for bias—has become a top priority.



A 2019 survey from PwC examined Responsible AI principles companies planned to address in the near-term.

Tools

The prioritization of principles to address is also informed by the state of tools available. Responsible AI is an emerging area, so it's difficult to make sweeping conclusions about the state of tools, but Michael Kearns, computer science professor and author of The Ethical Algorithm, recently gave a talk where he ranked the different areas of Responsible AI, based on the scientific maturity at the time of writing his book (November 2019):

- Privacy
- Fairness
- Accountability
- Interpretability
- Morality

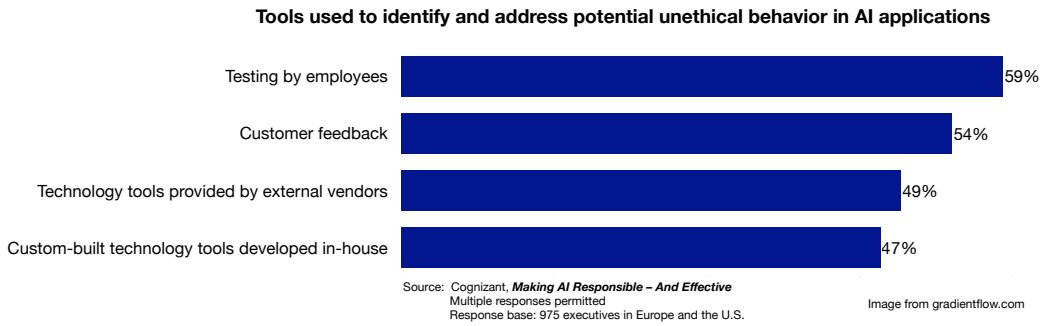
These rankings are somewhat subjective, but given that Kearns is deeply familiar with research in these fields, they likely map closely to how they manifest in real-world scenarios. They generally do agree with the findings in the surveys and reports we used for this investigation, as well as with our knowledge of available tools.

Privacy and security tools anecdotally get more coverage (see here and here, for example), which indicates they may be further along in development. Part of the challenge is that to make progress in developing tools around each of these principles is that stakeholders will need to come to agreement on precise definitions of each. All of these areas are being examined by machine learning researchers, so steady development is likely. As tools for responsible AI continue to improve, organizations face two key challenges: (1) they need to develop a clear understanding of the limitations of the tools they are using, and (2) they need to learn how to match models and techniques to their specific problems and challenges. The good news is that product and consulting

companies are beginning to provide assistance in these areas.

Identifying Responsible AI issues

Companies need to choose a method or approach to prioritize areas of Responsible AI they need to address. A 2018 report from Cognizant (pdf), based on a survey of almost 1,000 executives across the US and Europe, revealed the top two tools used to identify potential unethical behavior in AI applications were testing by employees and customer feedback.



A 2018 survey from Cognizant revealed tools used to identify Responsible AI areas that need to be addressed.

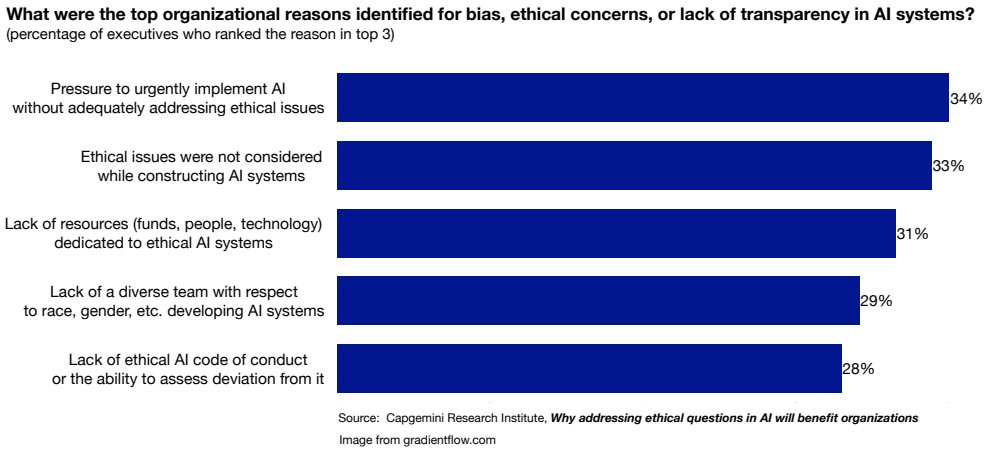
It's notable that close to two-thirds of respondents cited "Testing" as the tool used to identify areas to address—it means they have testing protocols in place that focus on areas pertaining to Responsible AI. The numbers of respondents citing "Customer feedback" are also encouraging; even if we have to generously interpret that result as companies having put tools in place to solicit such feedback (as opposed to incidental data gathering), it does indicate that companies realize the importance and usefulness of customer feedback.

Responsible AI is challenging to address

In 2018, the Capgemini Research Institute surveyed 1,580 executives in 510 organizations and over 4,400 consumers in the US and Europe to discover how they view transparency and ethics in AI-enabled applications. While the survey confirmed a strong interest in and expectation of ethical components in AI products, it also highlighted the lack of maturity in the AI landscape: only a quarter of respondents reported having fully implemented AI projects. The rest were in a pilot or proof-of-concept stage, or in the initial planning phase.

There is a growing competitive pressure to adopt new technologies like AI. Product managers and others charged with implementing AI and machine learning in products and systems are expected to focus on ROI-related aspects like business metrics and KPIs. As such, addressing Responsible AI features gets pushed down the priority list. The Capgemini survey results showed that the urgent

nature of AI production was the number one reason areas pertaining to responsible AI were not adequately addressed.



A survey by Capgemini investigated reasons why areas pertaining to Responsible AI were not adequately addressed.

Looking ahead: What do companies aspire to do?

The surveys we researched showed that companies are beginning to realize and appreciate the importance of incorporating Responsible AI principles as they produce and deploy AI applications. To discover the path forward, a group of researchers from Accenture, Spotify, and the Partnership on AI conducted an ethnographic study in 2020 based on 26 semi-structured interviews with people from 19 organizations on four continents. Their report, Where Responsible AI meets Reality, focused mainly on the Fair-ML principle, but it also assembled a great snapshot of where many organizations are today, and where they hope to be in the future. They asked participants "what they envision for the ideal future state of their work" in Fair-ML. The authors plan to conduct followup studies on other aspects of Responsible AI, but for now let's assume the results (see Table 1 below) at least partially reflect how companies are tackling areas outside of fairness.

The study results confirm several key findings from the surveys we researched. For instance most of the study participants described their current work in AI as "reactive":

> *"Practitioners embedded in product teams explained that they often need to distill what they do into standard metrics such as number of clicks, user acquisition, or churn rate, which may not apply to their work. Most commonly, interviewees reported being measured on delivering work that generates revenue."*

The cited revenue-generating measurements directly mirror results from the Capgemini survey,

where more than one-third of respondents cited, “Pressure to urgently implement AI without adequately addressing ethical issues” as a main concern.

Another interesting finding from the study highlighted a potential culture shift that will be required to provide the institutional scaffolding necessary to support the integration of Responsible AI principles:

> *“Multiple practitioners expressed that they needed to be a senior person in their organization in order to make their Fair-ML related concerns heard. Several interviewees talked about the lack of accountability across different parts of their organization, naming reputational risk as the biggest incentive their leadership sees for the work on Fair-ML.”*

The authors organized their investigation around three phases of the transition to integrating Responsible AI: the prevalent state (where organizations are now); the emerging state (what practices are being designed to move forward); and the aspirational state (the ideal framework that will support the democratization and implementation of Responsible AI).

Companies looking to establish or expand a framework to accommodate Responsible AI principles can begin by closely examining the study results around when to move forward and how to define success.

Trends in the common perspectives shared by diverse fair-ML practitioners.

	Prevalent Practices	Emerging Practices	Aspirational Future
When do we act	**Reactive:** Organizations act only when pushed by external forces (e.g. media, regulatory pressure)	**Proactive:** Organizations act proactively to address potential fair-ML issues	**Anticipatory:** Organizations have deployed frameworks that allow for anticipating risks
How do we measure success	**Performance trade-offs:** Org-level conversations about fair-ML dominated by ill-informed performance trade-offs	**Provenance:** Org-level frameworks processes are implemented to evaluate fair-ML projects	**Concrete results:** Concepts of results are redefined to include societal impact through data-informed efforts
What are the internal structures we rely on?	**Lack of accountability:** Fair-ML work falls through the cracks due to role uncertainty	**Structural support:** Scaffolding to support Fair-ML work begins to be erected on top of existing in- ternal structures	**Integrated:** Fair-ML responsibilities are integrated throughout all business processes related to product teams
How do we resolve tensions?	**Fragmented:** Misalignment between individual and team incentives and org-level mission statements	**Rigid:** Overly rigid organizational incentives demotivate addressing ethical tensions in fair-ML work	**Aligned:** Ethical tensions in work are resolved in accordance with org-level mission and values

Image from gradientflow.com

Source: **Where Responsible AI meets Reality: Practitioner Perspectives on Enablers for shifting Organizational Practices** By Bogdana Rakova, Jingying Yang, Henriette Cramer, Rumman Chowdhury

A 2020 ethnographic study investigated the practicality of integrating Responsible AI.

When to act

According to the study interviewees, most organizations today are in reactive mode when it comes to incorporating fair ML principles into product pipelines. The catalyst most often cited was negative media attention, either as a result of a public catastrophe or from a general perspective shift that the status quo is no longer sufficient. Some companies report they are beginning to proactively implement Fair-ML practices, including procedural reviews conducted across company teams.

Aspirationally, companies report that a fully proactive framework would work best to support Fair-ML initiatives. Interviewees cited clear and open transparency and communication not only internally across all company teams, but externally with customers and stakeholders. They also cited the need for proper tools to solicit specific feedback internally and externally: internally from process and product reviews, and externally from customer oversight.

The key takeaway in terms of timing is that the steps along the road to effective Responsible AI should be aimed at integrating and implementing the principles as early in the product development process as possible. The inclusion of Responsible AI principles should also be routine and part of the production culture.

How to measure success

One of the main challenges is that current methods of measuring business success don't translate to measuring the success of Responsible AI implementations. Many study interviewees noted that key performance indicators (KPIs) for business are very different from academic benchmarks, and that trying to distill academic benchmarks into traditional business KPIs was inappropriate and misleading. Interviewees also reported that they're traditionally measured against goals structured around revenue, which is tricky (if not impossible) to tie to successful Responsible AI practices.

Some interviewees reported progress at their companies in moving beyond traditional revenue metrics by establishing new frameworks for evaluating Fair-ML risks in their products and services. They outlined three drivers behind this cultural shift: establishing rewards around efforts for internal education, rewards for instigating internal investigations of potential issues, and instituting frameworks to support cross-collaboration across the company.

The key takeaway on metrics for success is that they're still under construction. Traditional quantitative business metrics aren't designed to encompass the qualitative aspects of Responsible AI principles, and, as such, aren't appropriate for measuring success in that arena. Companies will need to establish new KPIs to fit business needs in their specific contexts.

Concluding thoughts

In this post, we examined how companies are approaching Responsible AI today and what they aspire to do. Interest in Responsible AI comes at a time when companies are beginning to roll out more ML and AI models into products and services. **As companies evolve their MLOps tools and processes, they not only need to account for Responsible AI, but put infrastructure in place to integrate it early on into product development pipelines.** To realize aspirational goals around Responsible AI, companies need to cultivate a shift in perspective, starting with company leaders, that embraces the primary Responsible AI principles: privacy and security, safety and reliability, fairness, and transparency and accountability. Successful deployment of ethical, responsible AI will require collaboration between cross-functional teams, adoption of new tools and processes, and buy-in from everyone involved.