Chapter 2. People of MLOps

Even though machine learning models are primarily built by data scientists, it's a misconception that only data scientists can benefit from robust MLOps processes and systems. In fact, MLOps is an essential piece of enterprise AI strategy and affects everyone working on, or benefiting from, the machine learning model life cycle.

This chapter covers the roles each of these people plays in the machine learning life cycle, who they should ideally be connected and working together with under a top-notch MLOps program to achieve the best possible results from machine learning efforts, and what MLOps requirements they may have.

It's important to note that this field is constantly evolving, bringing with it many new job titles that may not be listed here and presenting new challenges (or overlaps) in MLOps responsibilities.

Before we dive into the details, let's look at the following table, which provides an overview:

Role	Role in machine learning model life cycle	MLOps requirements
Subject matter experts	• Provide business questions, goals, or KPIs around which ML models should be framed. • Continually evaluate and ensure that model performance aligns with or resolves the initial need.	• Easy way to understand deployed model performance in business terms. • Mechanism or feedback loop for flagging model results that don't align with business expectations.
Data scientists	• Build models that address the business question or needs brought by subject matter experts. • Deliver operationalizable models so that they can be properly used in the production environment and with production data. • Assess model quality (of both original and tests) in tandem with subject matter experts	• Automated model packaging and delivery for quick and easy (yet safe) deployment to production. • Ability to develop tests to determine the quality of deployed models and to make contin-

Role	Role in machine learning model life cycle	MLOps requirements
	to ensure they answer initial business questions or needs.	ual improvements. • Visibility into the performance of all deployed models (including side-by-side for tests) from one central location. • Ability to investigate data pipelines of each model to make quick assessments and adjustments regardless of who originally built the model.

Role	Role in machine learning model life cycle	MLOps requirements
Data engineers	• Optimize the retrieval and use of data to power ML models.	• Visibility into performance of all deployed models. • Ability to see the full details of individual data pipelines to address underlying data plumbing issues.
Software engineers	• Integrate ML models in the company's applications and systems. • Ensure that ML models work seamlessly with other non-machine-learning-based applications.	• Versioning and automatic tests. • The ability to work in parallel on the same application.

Role	Role in machine learning model life cycle	MLOps requirements
DevOps	• Conduct and build operational systems and test for security, performance, availability. • Continuous Integration/Continuous Delivery (CI/CD) pipeline management.	• Seamless integration of MLOps into the larger DevOps strategy of the enterprise. • Seamless deployment pipeline.
Model risk managers/auditors	• Minimize overall risk to the company as a result of ML models in production. • Ensure compliance with internal and external requirements before pushing ML models to production.	• Robust, likely automated, reporting tools on all models (currently or ever in production), including data lineage.

Role	Role in machine learning model life cycle	MLOps requirements
Machine learning architects	• Ensure a scalable and flexible environment for ML model pipelines, from design to development and monitoring. • Introduce new technologies when appropriate that improve ML model performance in production.	• High-level overview of models and their resources consumed. • Ability to drill down into data pipelines to assess and adjust infrastructure needs.

Subject Matter Experts

The first profile to consider as part of MLOps efforts is the subject matter experts (SMEs); after all, the ML model life cycle starts and ends with them. While the data-oriented profiles (data scientist, engineer, architect, etc.) have expertise across many areas, they tend to lack a deep understanding of the business and the problems or questions that need to be addressed using machine learning.

Subject matter experts usually come to the table—or, at least, they *should* come to the table—with clearly defined goals, business questions, and/or key performance indicators (KPIs) that they want to achieve or address. In some cases, they might be extremely well defined (e.g., “To hit our numbers for the quarter, we need to reduce customer churn by 10%” or “We’re losing $N per quarter due to unscheduled maintenance; how can we better predict downtime?”). In other cases, the goals and questions may be less well defined (e.g., “Our service staff needs to better understand our customers to upsell them" or "How can we get people to buy more widgets?").

In organizations with healthy processes, starting the machine learning model life cycle with a more defined business question isn't necessarily always an imperative, or even an ideal, scenario. Working with a less defined business goal can be a good opportunity for subject matter experts to work directly with data scientists up front to better frame the problem and brainstorm possible solutions before even beginning any data exploration or model experimentation.

Without this critical starting point from subject matter experts, other data professionals (particularly data scientists) risk starting the machine learning life cycle process trying to solve problems or provide solutions that don't serve the larger business. Ultimately, this is detrimental not only to the subject matter experts who need to partner with data scientists and other data experts to build solutions, but to data scientists themselves who might struggle to provide larger value.

Another negative outcome when SMEs are not involved in the ML life cycle is that, without real business outcomes, data teams subsequently struggle to gain traction and additional budget or support to continue advanced analytics initiatives. Ultimately, this is bad for data teams, for SMEs, and for the business as a whole.

To add more structure around SME involvement, business decision modeling methodologies can be applied to formalize the business problems to be solved and frame the role of machine learning in the solution.

BUSINESS DECISION MODELING

Decision modeling creates a business blueprint of the decision-making process, allowing subject matter experts to directly structure and describe their needs. Decision models can be helpful because they put machine learning in context for subject matter experts. This allows the models to be integrated with the business rules, as well as helps the SMEs to fully understand decision contexts and the potential impact of model changes.

MLOps strategies that include a component of business decision modeling for subject matter experts can be an effective tool for ensuring that real-world machine learning model results are properly contextualized for those who don't have deep knowledge of how the underlying models themselves work.[1]

Subject matter experts have a role to play not only at the beginning of the ML model life cycle, but at the end (post-production) as well. Oftentimes, to understand if an ML model is performing well or as expected, data scientists need subject matter experts to close the feedback loop because traditional metrics (accuracy, precision, recall, etc.) are not enough.

For example, data scientists could build a simple churn prediction model that has very high accuracy in a production environment; however, marketing does not manage to prevent anyone from churning. From a business perspective, that means the model didn't work, and that's important information that needs to make its way back to those building the ML model so that they can find another possible solution, such as introducing uplift modeling that helps marketing better target potential churners who might be receptive to marketing messaging.

Given the role of SMEs in the ML model life cycle, it's critical when building MLOps processes to have an easy way for them to understand deployed model performance in business terms. That is, they need to understand not just model accuracy, precision, and recall, but the results or impact of the model on the business process identified up front. In addition, when there are unexpected shifts in performance, subject matter experts need a scalable way, through MLOps processes, to flag model results that don't align with business expectations.

On top of these explicit feedback mechanisms, more generally, MLOps should be built in a way that increases transparency for subject matter experts. That is, they should be able to use MLOps processes as a jumping-off point for exploring the data pipelines behind the models, understanding what data is being used, how it's being transformed and enhanced, and what kind of machine learning techniques are being applied.

For subject matter experts who are also concerned with compliance of machine learning models with internal or external regulations, MLOps serves as an additional way to bring transparency and understanding to these processes. This includes being able to dig into individual decisions made by a model to understand why the model came to that decision. This should be complementary to statistical and aggregated feedback.

Ultimately, MLOps is most relevant for subject matter experts as a feedback mechanism and a platform for communication with data scientists about the models they are building. However, there are other MLOps needs as well—specifically around transparency, which ties into Responsible AI—that are relevant for subject matter experts and make them an important part of the MLOps picture.

Data Scientists

The needs of data scientists are the most critical ones to consider when building an MLOps strategy. To be sure, they have a lot to gain; data scientists at most organizations today often deal with siloed data, processes, and tools, making it difficult to effectively scale their efforts. MLOps is well positioned to change this.

Though most see data scientists' role in the ML model life cycle as strictly the model building portion, it is—or at least, it should be—much wider. From the very beginning, data scientists need to be involved with subject matter experts, understanding and helping to frame business problems in such a way that they can build a viable machine learning solution.

The reality is that this very first, critical step in the ML model life cycle is often the hardest. It's challenging particularly for data scientists because it's not where their training lies. Both formal and informal data science programs in universities and online heavily emphasize technical skills and not necessarily skills for communicating effectively with subject matter experts from the business side of the house, who usually are not intimately familiar with machine learning techniques. Once again, business decision modeling techniques can help here.

It's also a challenge because it can take time. For data scientists who want to dive in and get their hands dirty, spending weeks framing and outlining the problem before getting started on solving it can be torture. To top it off, data scientists are often siloed (physically, culturally, or both) from the core of the business and from subject matter experts, so they simply don't have access to an organizational infrastructure that facilitates easy collaboration between these profiles. Robust MLOps systems can help address some of these challenges.

After overcoming the first hurdle, depending on the organization, the project might get handed off to either data engineers or analysts to do some of the initial data gathering, preparation, and exploration. In some cases, data scientists themselves manage these parts of the ML model life cycle. But in any case, data scientists step back in when it comes time to build, test, robustify, and then deploy the model.

Following deployment, data scientists' roles include constantly assessing model quality to ensure the way it's working in production answers initial business questions or needs. The underlying question in many organizations is often whether data scientists monitor only the models they have had a hand in building or whether one person handles all monitoring. In the former scenario, what happens when there is staff turnover? In the latter scenario, building good MLOps practices is critical, as the person monitoring also needs to be able to quickly jump in and take action should the model drift and start negatively affecting the business. If they weren't the ones who built it, how can MLOps make this process seamless?

OPERATIONALIZATION AND MLOPS

Throughout 2018 and the beginning of 2019, operationalization was the key buzzword when it came to ML model life cycles and AI in the enterprise. Put simply, operationalization of data science is the process of pushing models to production and measuring their performance against business goals. So how does operationalization fit into the MLOps story? MLOps takes operationalization one step further, encompassing not just the push to production but the maintenance of those models—and the entire data pipeline—in production.

Though they are distinct, MLOps might be considered the new operationalization. That is, where many of the major hurdles for businesses to operationalize have disappeared, MLOps is the next frontier and presents the next big challenge for machine learning efforts in the enterprise.

All of the questions in the previous section lead directly here: data scientists' needs when it comes to MLOps. Starting from the end of the process and working backward, MLOps must provide data scientists with visibility into the performance of all deployed models as well as any models being A/B tested. But taking that one step further, it's not just about monitoring—it's also about action. Top-notch MLOps should allow data scientists the flexibility to select winning models from tests and easily deploy them.

Transparency is an overarching theme in MLOps, so it's no surprise that it's also a key need for data scientists. The ability to drill down into data pipelines and make quick assessments and adjustments (regardless of who originally built the model) is critical. Automated model packaging and delivery for quick and easy (yet safe) deployment to production is another important point for transparency, and it's a crucial component of MLOps, especially to bring data scientists together to a place of trust with software engineers and DevOps teams.

In addition to transparency, another theme for mastering MLOps—especially when it comes to meeting the needs of data scientists—is pure efficiency. In an enterprise setting, agility and speed matter. It's true for DevOps, and the story for MLOps is no different. Of course, data scientists can deploy, test, and monitor models in an ad hoc fashion. But they will spend enormous amounts of time reinventing the wheel with every single ML model, and that will never add up to scalable ML processes for the organization.

Data Engineers

Data pipelines are at the core of the ML model life cycle, and data engineers are, in turn, at the core of data pipelines. Because data pipelines can be abstract and complex, data engineers have a lot of efficiencies to gain from MLOps.

In large organizations, managing the flow of data, outside of the application of ML models, is a full-time job. Depending on the technical stack and organizational structure of the enterprise, data engineers might, therefore, be more focused on the databases themselves than on pipelines (especially if the company is leveraging data science and ML platforms that facilitate the visual building of pipelines by other data practitioners, like business analysts).

Ultimately, despite these slight variations in the role by an organization, the role of data engineers in the life cycle is to optimize the retrieval and use of data to eventually power ML models. Generally, this means working closely with business teams, particularly subject matter experts, to identify the right data for the project at hand and possibly also prepare it for use. On the other end, they work closely with data scientists to resolve any data plumbing issues that might cause a model to behave undesirably in production.

Given data engineers' central role in the ML model life cycle, underpinning both the building and monitoring portions, MLOps can bring significant efficiency gains. Data engineers require not only visibility into the performance of all models deployed in production, but the ability to take it one step further and directly drill down into individual data pipelines to address any underlying issues.

Ideally, for maximum efficiency for the data engineer profile (and for others as well, including data scientists), MLOps must not consist of simple monitoring, but be a bridge to underlying systems for investigating and tweaking ML models.

Software Engineers

It would be easy to exclude classical software engineers from MLOps consideration, but it is crucial from a wider organizational perspective to consider their needs to build a cohesive enterprise-wide strategy for machine learning.

Software engineers don't usually build ML models, but, on the other hand, most organizations are not *only* producing ML models, but classic software and applications as well. It's important that software engineers and data scientists work together to ensure the functioning of the larger system. After all, ML models aren't just standalone experiments; the machine learning code, training, testing, and deployment have to fit into the Continuous Integration/Continuous Delivery (CI/CD) pipelines that the rest of the software is using.

For example, consider a retail company that has built an ML-based recommendation engine for their website. The ML model was built by the data scientist, but to integrate it into the larger functioning of the site, software engineers will necessarily need to be involved. Similarly, software engineers are responsible for the maintenance of the website as a whole, and a large part of that includes the functioning of the ML models in production.

Given this interplay, software engineers need MLOps to provide them with model performance details as part of a larger picture of software application performance for the enterprise. MLOps is a way for data scientists and software engineers to speak the same language and have the same baseline understanding of how different models deployed across the silos of the enterprise are working together in production.

Other important features for software engineers include versioning, to be sure of what they are currently dealing with; automatic tests, to be as sure as possible that what they are currently dealing with is working; and the ability to work in parallel on the same application (thanks to a system that allows branches and merges like Git).

DevOps

MLOps was born out of DevOps principles, but that doesn't mean they can be run in parallel as completely separate and siloed systems.

DevOps teams have two primary roles in the ML model life cycle. First, they are the people conducting and building operational systems as well as tests to ensure security, performance, and availability of ML models. Second, they are responsible for CI/CD pipeline management. Both of these roles require tight collaboration with data scientists, data engineers, and data architects. Tight collaboration is, of course, easier said than done, but that is where MLOps can add value.

For DevOps teams, MLOps needs to be integrated into the larger DevOps strategy of the enterprise, bridging the gap between traditional CI/CD and modern ML. That means systems that are fundamentally complementary and that allow DevOps teams to automate tests for ML just as they can automate tests for traditional software.

Model Risk Manager/Auditor

In certain industries (particularly the financial services sector), the model risk management (MRM) function is crucial for regulatory compliance. But it's not only highly regulated industries that should be concerned or that should have a similar function; MRM can protect companies in any industry from catastrophic loss introduced by poorly performing ML models. What's more, audits play a role in many industries and can be labor intensive, which is where MLOps comes into the picture.

When it comes to the ML model life cycle, model risk managers play the critical role of analyzing not just model outcomes, but the initial goal and business questions ML models seek to resolve to minimize overall risk to the company. They should be involved along with subject matter experts at the very beginning of the life cycle to ensure that an automated, ML-based approach in and of itself doesn't present risk.

And, of course, they have a role to play in monitoring—their more traditional place in the model life cycle—to ensure that risks are kept at bay once models are in production. In between conception and monitoring, MRM also is a factor post-model development and preproduction, ensuring initial compliance with internal and external requirements.

MRM professionals and teams have a lot to gain from MLOps, because their work is often painstakingly manual. As MRM and the teams with which they work often use different tools, standardization can offer a huge leg up in the speed at which auditing and risk management can occur.

When it comes to specific MLOps needs, robust reporting tools on all models (whether they are currently in production or have been in production in the past) is the primary one. This reporting should include not just performance details, but the ability to see data lineage. Automated reporting adds an extra layer of efficiency for MRM and audit teams in MLOps systems and processes.

Machine Learning Architect

Traditional data architects are responsible for understanding the overall enterprise architecture and ensuring that it meets the requirements for data needs from across the business. They generally play a role in defining how data will be stored and consumed.

Today, demands on architects are much greater, and they often have to be knowledgeable not only on the ins and outs of data storage and consumption, but on how ML models work in tandem. This adds a lot of complexity to the role and increases their responsibility in the MLOps life cycle, and it's why in this section, we have called them machine learning architects instead of the more traditional "data architect" title.

Machine learning architects play a critical role in the ML model life cycle, ensuring a scalable and flexible environment for model pipelines. In addition, data teams need their expertise to introduce new technologies (when appropriate) that improve ML model performance in production. It is for this reason that the data architect title isn't enough; they need to have an intimate understanding of machine learning, not just enterprise architecture, to play this key role in the ML model life cycle.

This role requires collaboration across the enterprise, from data scientists and engineers to DevOps and software engineers. Without a complete understanding of the needs of each of these people and teams, machine learning architects cannot properly allocate resources to ensure optimal performance of ML models in production.

When it comes to MLOps, the machine learning architects' role is about having a centralized view of resource allocation. As they have a strategic, tactical role, they need an overview of the situation to identify bottlenecks and use that information to find long-term improvements. Their role is one of pinpointing possible new technology or infrastructure for investment, not necessarily operational quick fixes that don't address the heart of the scalability of the system.

Closing Thoughts

MLOps isn't just for data scientists; a diverse group of experts across the organization has a role to play not only in the ML model life cycle, but the MLOps strategy as well. In fact, each person—from the subject matter expert on the business side to the most technical machine learning architect—plays a critical part in the maintenance of ML models in production. This is ultimately important not only to ensure the best possible results from ML models (good results generally lead to more trust in ML-based systems as well as increased budget to build more), but, perhaps more pointedly, to protect the business from the risks outlined in Chapter 1.

1 Decision requirements models are based on Decision Model and Notation, a framework for improving processes, effectively managing business rules projects, framing predictive analytics efforts, and ensuring decision support systems and dashboards are action-oriented.