White Paper
September 2021

IT@Intel: Push-button Productization of AI Models

Learn how Intel IT deploys AI faster and enables self-maintaining, cost-effective AI services in production at scale

Author

Moty Fania
Principal Engineer, CTO and Head of Machine Learning Engineering

Table of Contents

Executive Summary

Intel IT has a large AI group that works across Intel to transform critical work, optimize processes, eliminate scalability bottlenecks and generate significant business value (more than USD 1.5B return on investment in 2020). Our efforts unlock the power of data to make Intel's business processes smarter, faster and more innovative, from product design to manufacturing to sales and pricing.

To enable this operation at scale, we developed Microraptor: a set of MLOps capabilities that are reused in all of our AI platforms. Microraptor enables world-class MLOps to accelerate and automate the development, deployment and maintenance of machine-learning models. Our approach to model productization avoids the typical logistical hurdles that often prevent other companies' AI projects from reaching production. Our MLOps methodology enables us to deploy AI models to production at scale through continuous integration/continuous delivery, automation, reuse of building blocks and business process integration.

Our MLOps methodology provides many advantages:

- The AI platforms abstract deployment details and business process integration so that data scientists can concentrate on model development.
- We can deploy a new model in less than half an hour, compared to days or weeks without MLOps.
- Our systematic quality metrics minimize the cost and effort required to maintain the hundreds of models we have in production.

Background

Intel IT's AI group includes over 200 data scientists, machine-learning (ML) engineers and AI product experts. We systematically work across Intel's core activities to deliver AI solutions that optimize processes and eliminate various scalability bottlenecks. We use AI to deliver high business impact and transform Intel's internal operations, including engineering, manufacturing, hardware validation, sales, performance and Mobileye (see Figure 1). Over the last decade, we have deployed over 500 ML models to production—more than 100 models were deployed just during the last year.

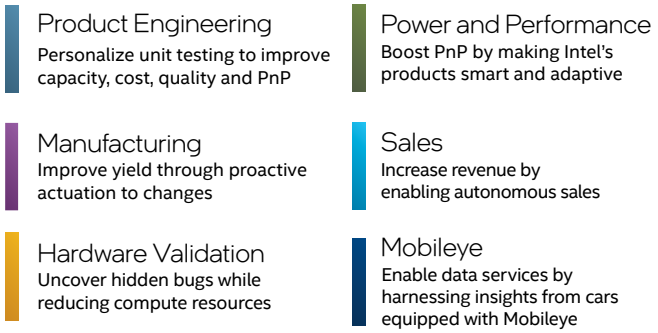

Figure 1. Our AI efforts support many areas of Intel's business.

By embedding and enabling AI across Intel's critical business processes, we delivered more than USD 1.56B in value in 2020. Here are some examples:

- We automated and enhanced our product validation capabilities, and created more intelligent products that reduced power and improved both performance and battery life.
- We provided data insights that enabled Intel's sales force to better target customers with relevant and valuable solutions, and helped determine the right product performance and pricing structure for customers.
- We helped improve factory yield by an extra 3.7 million units (USD 372 million) for a combination of several Intel® processors.

While many companies' AI journeys are in their infancy—many are still conducting small proofs of concept—Intel is reaping substantial business value from AI across the entire enterprise. One thing we've learned along our AI journey is that treating each AI project as a separate entity cannot lead to long-term production-level benefits. To truly reap the benefits of AI, we must fully understand the AI lifecycle and solve the challenges of creating and maintaining AI algorithms at scale.

AI Lifecycle

As illustrated in Figure 2, the lifecycle of AI consists of three steps.

Step 1. The first, essential step is to develop a high-quality ML model that meets specific business requirements and is proven to address the problem we are attempting to solve. However, simply producing the model is not sufficient.

Step 2. Once the model is ready, we would ideally productize it as soon as possible so it can realize its value. By "productize," we mean that we put an AI solution to work transforming data into business insights for a specific problem or use case.

Step 3. Once the model is deployed in production—theoretically generating good business value—we cannot rest. Models degrade over time. Without proactive monitoring and tracking of model health and taking appropriate maintenance actions (such as retraining), the model may degrade to a point that it provides incorrect results and even cause real damage.

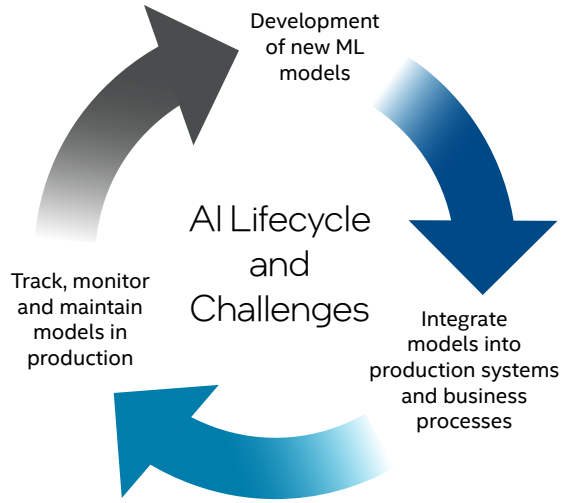

Figure 2. The AI lifecycle has three steps—all of which must be streamlined for AI to produce business value.

While maintaining a few models in production is not hard, at larger scale—we have hundreds of models in production—it becomes a bigger challenge. Our main concern is that our AI experts may become so busy with maintaining existing ML models that they no longer have bandwidth to solve new problems with AI and create new value and growth. Therefore, we believe it is imperative to perform model maintenance efficiently so that most AI resources can focus on new projects and new business problems. We want to avoid continually dealing with issues related to a multitude of already-deployed models.

Our IT AI group's goal is to productize models in minutes using full automation along with an adaptation of the continuous integration/continuous delivery (CI/CD) concept common in the software world. We have adapted our DevOps infrastructure—which we typically use for fully automated releases and deployment processes for traditional software—to apply the same methodology in the world of AI.

Once in production, our goal is to efficiently and proactively manage and maintain models. We set an ambitious goal of investing 10 percent or less of total resources in maintenance efforts. For the past few years, we have sustained this 10 percent investment level and have continuously reduced its cost, despite the growing scale and hundreds of new models deployed annually.

AI Productization Challenges

In the real world, productizing AI models typically requires substantial time and effort. In many cases, a model may never make it to production. In fact, productizing ML is one of the biggest challenges in AI practices today. About 87 percent of AI projects languish in the lab,[1] and "launching pilots is deceptively easy but deploying them into production is notoriously challenging."[2]

Creating custom ML applications is a primary hurdle to productization of ML models. It is tempting to build a dedicated wrapper application for each model using modern compute platforms, tools and open-source code—these offer excellent building blocks for implementing any application, system or AI solution. With this approach, each AI solution or model is built as a small, isolated application and deployed to production. The custom wrapper application handles data extraction and data preparation. It also exposes the model's results for consumption through a web service, a file or a report. It can take two to three weeks to develop a wrapper application and an additional week to test, so a model can be productized in a few weeks.

However, based on our experience, the custom wrapper approach is not scalable. It can easily lead to a chaotic situation of hundreds of unmanaged AI applications. The following list depicts the primary drawbacks of the custom wrapper approach:

- **Neither easy nor rapid.** A few weeks of effort use considerable resources, which is far longer than the CI/CD goal of productization in mere minutes.
- **Limited or no reuse.** This approach promotes redundancies instead of reuse. Different applications may be performing similar tasks that potentially could have been reused.
- **No separation of concerns (tasks).** The data scientist developing the model may be also required to develop the wrapper, which usually requires code that has nothing to do with data science. Even if a software developer writes the wrapper, it is likely that the developer and the data scientist may need to engage in substantial communication and rework.
- **High cost to maintain models.** With hundreds of unmanaged AI applications scattered across the enterprise, model management becomes a problem. It is difficult to monitor and track the health of so many applications; as models inevitably degrade, issues and escalations are also inevitable.

[1] VB, "Why Do 87% of data science projects never make it into production?" venturebeat.com/2019/07/19/why-do-87-of-data-science-projects-never-make-it-into-production

[2] Gartner, "I&O leaders need to strategically leverage AI as a core accelerant to digital business initiatives," gartner.com/smarterwithgartner/gartner-predicts-the-future-of-ai-technologies

What Is MLOps?

Machine-learning operations (MLOps) is the practice of efficiently developing, testing, deploying and maintaining ML in production. It automates and monitors the entire ML lifecycle and enables seamless collaboration across teams, resulting in faster time to production and reproducible results.

Intel IT's MLOps Solution

Intel IT has been using DevOps for years to streamline project and software delivery. To avoid the problems associated with the custom wrapper approach to productization of AI, we applied our DevOps learnings and infrastructure to enable and improve machine-learning operations (MLOps). To enable MLOps, we build an AI productization platform for each business domain that we work with, such as sales or manufacturing (see Figure 3). Models and AI services are all delivered, deployed, managed and maintained on top of the AI platforms.

Our ML engineers build the ML workflows and infrastructure necessary to productize AI models and enable MLOps. These engineers have the following primary duties:

- Extract and organize data for ML at scale.
- Build inference interfaces for ML models.
- Enable MLOps for CI/CD and automation of ML pipelines.
- Build and sustain AI platforms.

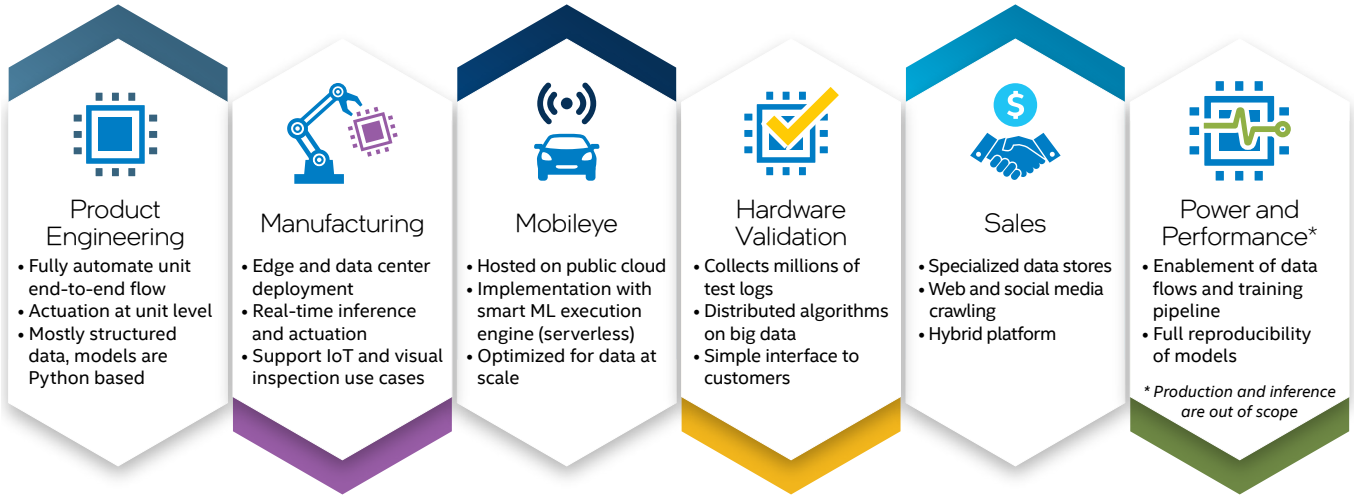

Figure 3. AI platforms for each business domain have unique features.

A Closer Look at Our AI Platforms

AI platforms have four main tasks:

- Enable an easy way to rapidly deploy models to production, with full CI/CD.
- Acquire data that is relevant to all models in that business domain, with support for both streaming and batch processing. The platform extracts data that could be used by many models.
- Enable a closed feedback loop at scale and in a timely manner. The platform offers deep integration with business processes. These integration points can be used for many models and the creator of the model does not need to worry about how to integrate the model's results into the business workflow.
- Maintain models. The platform handles all manageability aspects such as retraining models, collecting logs and tracking model health indicators.

Our MLOps approach offers a good separation of concerns, because data scientists can develop models without bothering with the engineering aspects that are not part of their expertise. The AI platforms result in less code overall, more predictability enabled by full automation and more opportunities for code reuse. In addition, the AI platforms include innovative manageability features that help track and maintain models in production and reduce their total cost of ownership. The manageability features include applicative monitoring, built-in health checks, system tests and retraining of models.

Our AI platforms are built using various open-source technologies combined with rich internal intellectual property and in-house resources. In most cases, the platforms are based on a streaming, microservice-architecture that is optimized to be easily deployed with Docker containers and Kubernetes either on-premises or in the public cloud.

AI platforms significantly accelerate time to market and reduce the total cost of ownership of AI solutions. They promote reuse, help AI practitioners become more independent in their work, and generally improve the quality and reliability of AI deployments.

Common Aspects of All AI Platforms

Each business domain's AI platform works with different types of data, different data extraction processes and different integration points into the relevant business workflow. However, our AI platforms encourage reuse because they share several objectives:

- Enable experiment tracking, model registry and training reproducibility.
- Achieve separation of concerns and require minimal hand-offs or rework between data scientists and ML engineers.
- Enable CI/CD for model deployment.
- Enable scalable and flexible inference system(s).
- Enable maintainability and quality of models in production.

To satisfy these common requirements, we built Microraptor: a set of reusable MLOps capabilities that are reused in all of our AI platforms. Microraptor was developed to enable a fully automated CI/CD process for models, combined with systematic measures to minimize the cost and effort required to maintain hundreds of models in production. Among many capabilities, Microraptor enables deployment of complex models to production in minutes (push-button deployments); while in production, it runs the AI models at scale, tracks their quality metrics with a built-in model quality indicator system and enables advanced retrain capabilities.

Overview of Open-source Microraptor Components

As illustrated in Figure 4, Microraptor uses many open-source technologies to enable the full MLOps lifecycle while abstracting the complexity of these technologies from the data scientists. Data scientists do not have to know anything about Kubernetes, Helm, Seldon, Jenkins or Elasticsearch. They can focus their efforts on finding or developing the best ML model.

Once the model is ready, a data scientist can simply register the model to MLflow (an open-source platform for managing the end-to-end ML lifecycle) while complying with some basic coding standards. Everything else—from building to testing to deploying—happens automatically. The model is first deployed as a release candidate that can be later activated with another push of a button into the relevant business domain's AI platform.

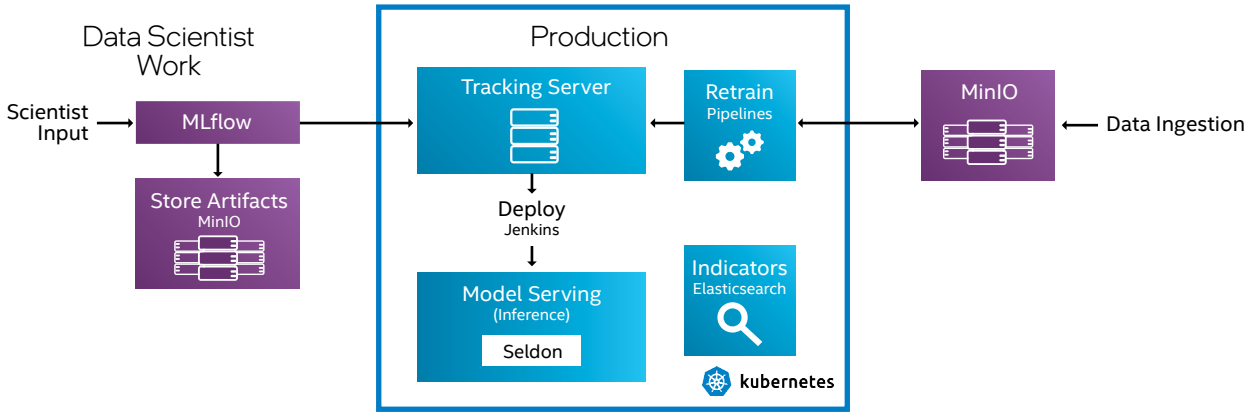

Figure 4. The Microraptor architecture uses a variety of open-source components, some of which we have customized.

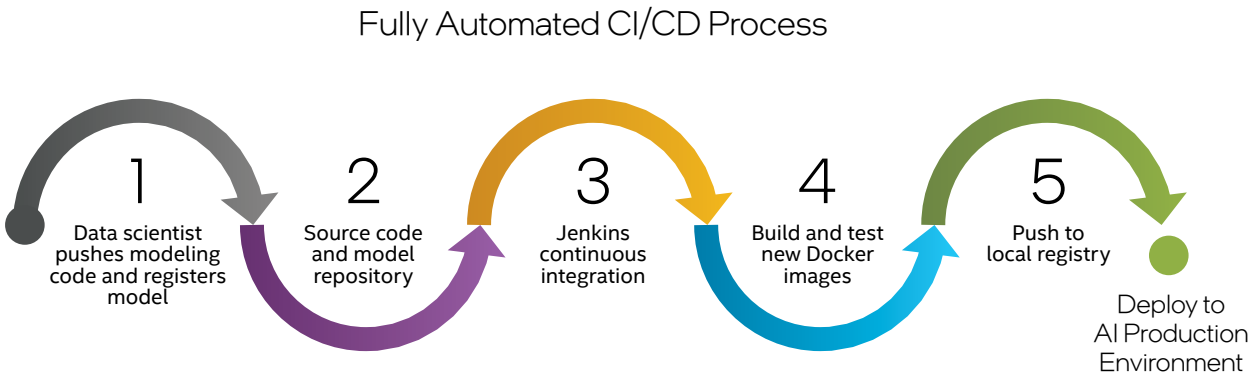

Figure 5. Microraptor supports push-button continuous delivery of models.

As Figure 5 shows, the CI/CD process is fully automated. The following sections provide more detail on the customized AI platform components (the other components are common open-source tools available online).

MLflow

MLflow is a platform to streamline ML development, including tracking experiments, packaging code into reproducible runs and sharing and deploying models. We chose MLflow because it can support many ML frameworks, which provides flexibility. We use MLflow for experiment tracking and for the model registry. In addition to these out-of-the-box features, we enabled easy instantiation, full model reproducibility and a shared storage (see Figure 6). Our approach to storage means that instead of storing experiments' artifacts and logs to the local file system, Microraptor stores everything in an object store based on MinIO, improving storage reliability and enabling sharing of results and progress at the team level.

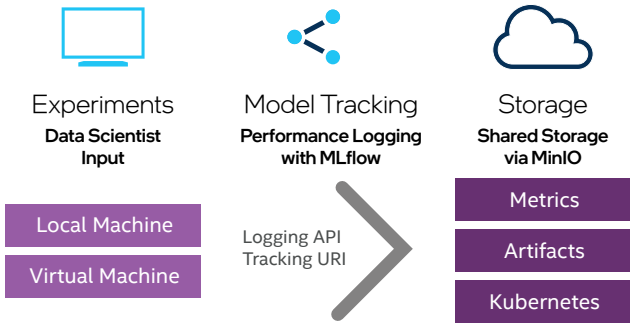

Figure 6. MLflow uses shared MinIO storage to improve reliability and shareability.

Seldon

To enable a scalable inference engine on top of Kubernetes, we use Seldon, which can convert ML models from ML frameworks like TensorFlow and PyTorch or language wrappers such as Python and Java into production REST microservices. Seldon can scale to thousands of production ML models and provides advanced ML capabilities out of the box, including Request Logging, A/B Tests, Canary testing and more. However, working with Seldon requires technical skills that are not typically part of a data scientist's skill set. Therefore, Microraptor abstracts this complexity and provides an easy way to define a Seldon Graph just with Python logic and a configuration YAML file. Data scientists can fully own the deployment process with an automated CI/CD process and without being dependent on ML engineers or having to understand the technical concepts of Kubernetes or Helm charts.

Tracking Model Quality Metrics

Once ML models are deployed to production, the quality and performance of models degrade over time. Predictions become less accurate as time passes; this degradation may even lead to real damage to the relevant business processes that rely on those predictions. Therefore, it is essential to track models in production, monitor their health and respond to issues as they arise. We have included a subsystem in Microraptor that enables the development, deployment and management of model-quality metrics and can be used for alerting and actuation to trigger an automatic retrain or if necessary, human attention.

This subsystem (see Figure 7) was designed to allow development of indicators using Python skills only so data scientists can develop indicators and quality metrics independently, without the help of ML engineers.

To achieve this goal, Microraptor offers the following layers:

- A logger component keeps a record of the inputs and outputs of each inference request. This raw data is collected automatically and sent to Elasticsearch.
- On top of this logging data, the system calculates near-real-time aggregations, as required, that can create summarized data. These summaries include counts, averages, standard deviations and so on.
- The system offers an easy way to add relevant data and resources that may be required for the aggregation calculations. This additional information includes labels, training data metadata, statistical distribution and more.
- Finally, the subsystem allows data scientists to provide custom logic/rules to calculate the indicators using Python, with easy access to the data generated by previous layers, which is stored in Elasticsearch.

Once data from all these layers is available in Elasticsearch, it can be used for proactive evaluation of the health of the models. Kibana is an open user interface that visualizes the indicators and quality metrics that were stored in Elasticsearch. ElastAlert is a simple open-source framework for alerting on anomalies, spikes or other patterns of interest from data in Elasticsearch. Microraptor uses ElastAlert to define rules and more complex Python logic to calculate metrics and to trigger the required actuation.

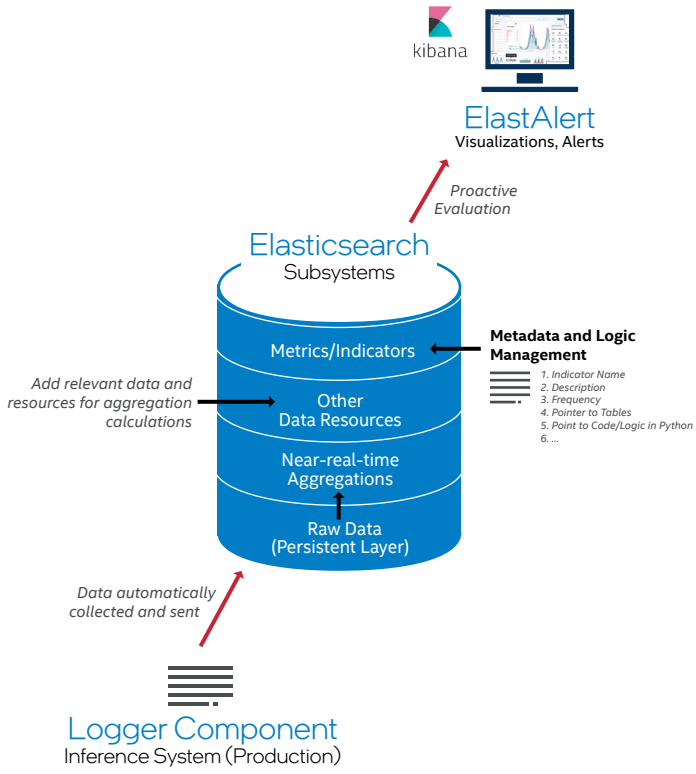

Figure 7. Microraptor includes a model-quality metrics subsystem.

Once computed, the quality metrics can be used for alerting and actuation to trigger retraining or tuning of models or human intervention. In addition to the ability to create our own indicators, we have included some industry-standard model metrics in Microraptor, which are available out-of-the-box, such as:

- **Identify concept drift.** Detect changes in the distribution of production features.
- **Check for numeric stability.** Analyze the model's prediction stability and classification classes frequency stability.
- **Skew monitoring.** For specific model types, such as ensembles, A/B and Graphs, monitor multiple ML models with production inputs and compare metrics to analyze the stability of each model.
- **Monitor predictions versus labels.** Also referred to as "offline proxy metrics," these metrics are used for models where true labels are known. Examples of metrics in this category are normalized discounted cumulative gain (nDCG), log loss, square error and confusion metrics.

Speeding Model Development with DSraptor

Initially, we focused on Microraptor as an example of how to easily deploy models to and maintain them in production. However, it is equally important to reduce the time to market for creating new ML models. To this end, we recently created DSraptor, an AI custom-built platform component that accelerates the time that it takes to develop new ML models, while improving productivity and quality. It is based on Kubeflow pipelines and Python Domain-specific Language for Argo Workflows. We enhanced these elements by adding capabilities, including the following:

- A syntax more similar to native Python
- Ability to work with the tool seamlessly, whether locally or remotely
- Global storage
- Reusable components
- An interactive mode
- An extensive tracking metastore

Both data scientists and ML engineers use DSraptor to accelerate the development of new models, create auto-modeling processes and to enable automated model retraining. Among its many benefits, DSraptor reduces the common rework associated with productizing ML pipelines. The pipelines that are produced by data scientists can be easily integrated into bigger production pipelines more quickly compared to traditional methodologies. In addition, DSraptor makes it much simpler to deploy extensive model retrain jobs as DSraptor pipelines. These retrain pipelines can be based, to an extent, on the work of the data scientist who produced the initial model(s). Other benefits include code modularity and quality; easy parallelism of exploration work on a larger compute cluster; and extensive reuse by allowing the creation of shared components or pipelines that can be seamlessly imported into new pipelines. We plan to provide more details on DSraptor in a future IT@Intel white paper.

Results

Our AI platforms, equipped with advanced, reusable MLOps capabilities through Microraptor, have proven extremely successful in automating and accelerating deployments of AI models to production. Deployment of new models and quality metrics is now fully automated and takes a fraction of the time than previously. In many cases, production deployment can be achieved by the data scientist independently, eliminating rework and unnecessary hand-offs. These new capabilities are now being actively used in various AI projects in the domains of sales AI, power and production, manufacturing AI, Mobileye, sales and more.

It used to take several days to several weeks to deploy a single model. With Microraptor, in 30 days we released more than 200 models (both new models and revisions) with an average time of about 25 minutes per deployment. In addition, the majority of the deployed models are now subject to proactive monitoring using Microraptor's indicator system to track model health and react proactively to model degradation.

Conclusion

Data is a transformational force, and we are using AI to drive product innovation and improve Intel's business process execution. Our Microraptor MLOps solution has enabled us to develop, deploy and maintain hundreds of ML models with low cost and effort. Our AI platforms enable us to deploy a new model in about half an hour. And because we have significantly automated model maintenance, the majority of our AI group can concentrate on developing new AI solutions, while only 10 percent of our total resources are devoted to model maintenance.

We continue to improve Microraptor, and we will continue to work with Intel's business units and design teams to put automation, AI and data to work to support Intel's growth.

For more information on Intel IT best practices, visit intel.com/IT.

Related Content

If you liked this paper, you may also be interested in these related stories:

- Building an AI Center of Excellence blog
- Improving Sales Account Coverage with Artificial Intelligence white paper
- Faster, More Accurate Defect Classification Using Machine Vision white paper
- Artificial Intelligence Reduces Costs and Accelerates Time to Market white paper
- Streamline Deep-learning Integration into Auto Defect Classification white paper
- Data Center Strategy Leading Intel's Business Transformation white paper
- IT Collaboration Leads to Unique Product Innovation blog

Microraptor and DSraptor Contributors

Eran Avidan, Amir Chanovsky, Sergei Kom, Vlad Vetshtein
Senior Machine Learning Engineers

Keren Mann Derey
AI Continuous Delivery Manager and Product Owner

Elhay Efrat
Lead DevOps

Acronyms

CI/CD	continuous integration/continuous delivery
ML	machine learning
MLOps	machine-learning operations

IT@Intel

We connect IT professionals with their IT peers inside Intel. Our IT department solves some of today's most demanding and complex technology issues, and we want to share these lessons directly with our fellow IT professionals in an open peer-to-peer forum.

Our goal is simple: improve efficiency throughout the organization and enhance the business value of IT investments.

Follow us and join the conversation:

- Twitter
- LinkedIn
- #IntelIT
- IT Peer Network

Visit us today at intel.com/IT or contact your local Intel representative if you would like to learn more.