2022-03

Expect AI/MLOps Complexity

Kolsch, Mathias

U.S. Department of the Navy

http://hdl.handle.net/10945/70476

CHIPS THE DEPARTMENT OF THE NAVY'S INFORMATION TECHNOLOGY MAGAZINE

Notify Me of New Issue

CURRENT ISSUE BACK ISSUES AUTHOR INDEX BROWSE TAGS ABOUT CHIPS GO

Email

Expect AI/MLOps Complexity

By Mathias Kolsch, Naval Postgraduate School - January-March 2022

Artificial intelligence and machine learning (AI/ML)-involved projects require a well-defined process to successfully mature an idea or requirement into an operational system, and such systems require great care and feeding during operations. This is true for AI/ML systems in any sector, but the DoD faces additional challenges getting this right due to its funding mechanisms, workforce structure and the need for well-defined outcomes. The payoff for successful deployment is increased force readiness, availability and agility to dominate in an ever-changing battlespace.

DevOps (Development and Operations) is a modern software development life cycle (SDLC) that entails the automation and monitoring at all steps of software system construction and operation: coding and implementation, component integration, testing and quality assurance (QA), deployment, and infrastructure management, monitoring and scaling. This automation permits the DevOps SDLC to iterate rapidly and continuously, delivering value incrementally and rapidly. DevOps enables agile development; together, they bear many advantages over the traditional Waterfall SDLC model, which essentially is a single sequence from requirements specification to delivery. QA tests, releases, customer validation and the like are shifted left, that is, performed earlier, owing to being part of every iteration.

AI/MLOps, or DevOps for AI/ML-involved systems, is difficult and complex and not merely an additional step in the life cycle. To successfully evolve an AI/ML-involved project from concept to application, you should expect more!

- More co-evolving components;
- more iterations to arrive at the target;
- more diverse project teams;
- more demands on the data;
- more tooling and engineers to maintain the tools and automation;
- more and unending testing and validation; and
- more dynamic system adaptation.

Iterations: The first challenge is typically to define the exact requirements for project success. While the overall objective is clear, this often does not directly translate into data requirements or sufficiently actionable AI/ML requirements for the data scientist. For example, the objective might be to save on unnecessary aircraft maintenance costs based on flight sensor data. However, it is unclear which data exactly should be collected, what the desired tradeoff is between type 1 and type 2 errors, or what the ultimate metric to measure quality should be. Iterations create a closed-loop feedback cycle with the end user and thereby permit rapid requirement refinement, early-stage course corrections and comprehensive stakeholder engagement.

Diverse teams: A goal-driven project team has subject matter experts (SMEs) embedded with data scientists, ML engineers and other developers. SMEs that are absent for more than a week may cause focus drift, unanswered questions to block an increasing number of tasks and incorrect assumptions to creep into the product. The most consistently successful teams have developers working in close proximity to the end users – if not physically close, then with a low-friction communication channel such as Microsoft Teams. These cross-functional AI/ML project teams should regularly consult with an external reviewer who is competent in AI ethics.

Data: There is never enough good data: labeled with "ground truth" and as close as possible to the data that the operational system will encounter. Equally important are data traceability or pedigree (where did this data come from?), to answer questions as to why the system made certain prediction and to assess data quality even in the future.

Tooling and engineers: Software with AI/ML has more components, building it involves more steps, more artifacts, more tests, more validation, more configuration, more co-evolving asynchronous processes that must be coordinated. Some tools are available, but they need people to configure and maintain them. AI/ML and its models and necessary data are not just an additional software component, they are more akin to adding wings to a car than to installing a more powerful engine. A longer DevOps pipeline for AI/MLOps stresses the importance of automation and testing as far "left" – early on – as possible. For example, some security compliance checks and vulnerability scans can and should be automated in this pipeline.

Related CHIPS Articles

CHIPS Act Advances DoD's Emphasis on Microelectronics

DISA's New Collaborative Lab Environment Fosters IT Innovation

Boldly Choosing to Measure Innovation as an Accelerator to Digital Transformation

CISA Releases Second Version of Guidance for Secure Migration to the Cloud

DON Names NAVWAR Employees as Dr. Delores M. Etter Award Winners

Related DON CIO News

DON 5G Newsletter - August 2022

DON 5G Newsletter - July 2022

DON DevSecOps Newsletter - July 2022

DON 5G Newsletter - June 2022

DON Digital Workplace Newsletter - June 2022

Testing and security: AI/ML systems are bound to receive data input from evolving sources over their lifetime, be it due to a different deployment environment than where the training data came from or due to an upgrade of the data source: camera, sensor or written text origin, for example. The system might even be dynamic and learning on the fly. AI/ML systems need continuous validation that their output is not only correct – verification – but that it is within the semantic bounds of the application.

Examples of systems gone rogue abound; a well-known instance of this is the Microsoft Tay chat bot. The increasing reliance on open-source software to keep up with the rapid pace of industry and adverse actors means that the software evolves ever faster. This can decrease the mean-time-to-repair for vulnerability fixes to combat zero-day exploits but requires a maintained pipeline and AI/MLOps processes.

A common picture and best practices are emerging: What has worked well is to create persistent teams and task forces that can progress a system through its iterations for many years such as the Digital Integrated Maintenance Environment (DIME). Sustained funding permits iterations. A closely collaborating cross-functional team brings experts to the same table in tools, data, science, the SMEs and end user for validation. The Defense Innovation Unit (DIU) has developed a set of guidelines for the successful program management of AI/ML projects from planning through development to deployment.

The Naval Postgraduate School (NPS) has delivered targeted education offerings for the Joint AI Center that with custom crafted content to give DoD personnel the insights and tools to succeed in their AI/ML system efforts.

Systems that involve AI/ML are more complex in almost every aspect, as illustrated above. The solution is to educate stakeholders to expect this complexity, for program managers to expect more AI/MLOps and automation needs, and to insist on using best-practices; mature code; proven pipelines; and accepted software development principles.

The opinions expressed here are solely those of the author, and do not necessarily reflect those of the Department of the Navy, Department of Defense or the United States government.

TAGS: CDIO: AI, Data Strategy, Digital Workplace/O365, Emerging Tech, DEVSECOPS, Infrastructure

CHIPS is an official U.S. Navy website sponsored by the Department of the Navy (DON) Chief Information Officer in partnership with the Naval Information Warfare Center (NIWC) Atlantic.

Online ISSN 2154-1779; Print ISSN 1047-9988
Hyperlink Disclaimer