TechTarget
Custom Media
White Paper

AI Engineering: A New Discipline for Artificial Intelligence and Machine Learning

Having gained some initial experience with AI development, many organizations which have used cloud tools are now looking for ways to operationalize AI as a development discipline. And for good reason. AI has delivered outstanding results in such areas as conversational chatbots, fraud detection, facial recognition, natural language processing, and intelligent document processing. Among the payoffs AI is already delivering to enterprises are improved productivity, better customer experience, higher employee satisfaction levels, and more data-driven decision making.

However, AI workloads differ from typical applications in some fundamental ways, and that creates complexity when deploying them on traditional IT infrastructure. Integrating AI applications into the organization on a piecemeal basis can create havoc because their unique processing power, storage, and networking requirements are so different from conventional workloads and often require special accommodations.

One-off deployment creates pressure on IT organizations to accommodate unconventional processing loads using infrastructure that is not

NVIDIA. | ddn

optimized for the task. A piecemeal approach can also prompt business users to develop models on their own without IT's knowledge or oversight, creating poorly governed islands of automation that the IT organization often has to take over and fix without the necessary background or business context.

What's different about AI workloads

Let's look at some of the key differences between traditional enterprise workloads and AI applications.

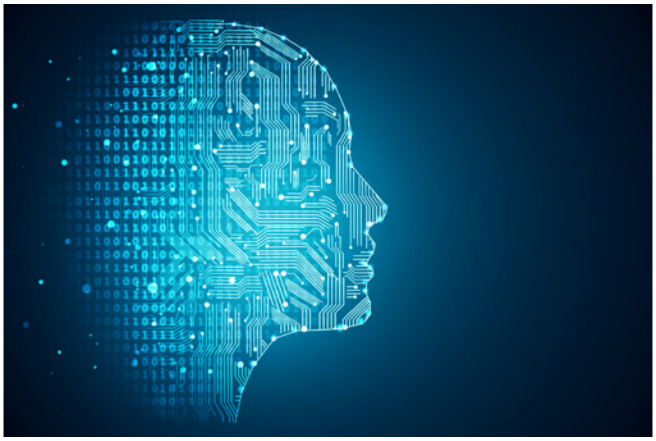
Conventional applications are usually intended to automate manual processes at scale, are purpose-built for a task, and presume a predictable set of inputs and outputs. Resource requirements are predictable and scale linearly, and the application changes little over time. Change is undesirable unless the organization specifically requires it.

Timeframes are predictable, allowing development projects to be integrated into a master schedule. In most cases, the development process follows a structured and time-tested path from defining requirements to prototyping, testing, and delivery. Variations in data quality do not influence the effectiveness of the application; good data is processed, and bad data is flagged for attention. Program logic, data flows, and processes are easily documented and well-understood.

AI applications differ in some significant ways, starting with the goals they are meant to achieve. AI is often used to derive insights that are beyond the capabilities of human operators or to arrive at conclusions that don't follow tightly scripted logical constraints. Surprises are common and even desirable.

Inputs and outputs are typically semi-structured or unstructured, often involving large numbers of small files. AI applications in areas such as medical diagnosis may deal with a wide variety of input data types and formats that the machine must interpret. Training data is typically both read- and write-intensive and is processed in parallel on GPUs rather than CPUs. That can create performance challenges that organizations haven't seen before and that they lack the tools to diagnose.

AI applications learn through a process of initial training followed by continuous adjustments and feedback from automated and human operators. Models are meant to change over time, and change is something to be desired rather than avoided.



Changeability also involves risk, though. While model evolution improves output quality over time, the introduction of biased or corrupted data may cause the quality of results to deteriorate. This was the case with Microsoft's famous Tay chatbot, which was intended to engage in benign conversations on Twitter but which quickly took a dark turn after a few mischievous users stoked it with racist rhetoric.

AI models can also be opaque, making it difficult for business users and even developers to understand why the system delivers the results it does. The same set of processes and inputs may deliver different results at different times based on the maturity level of the learning model, rules, and the training data used.

Training models can be massive in size, consisting of a large number of small data sets, making auditing and troubleshooting a challenge. Small variations in data quality can have dramatic impacts on model

> **AI engineering is an emerging discipline that focuses on applying tools, systems, and processes to enable AI to perform effectively in various real-world contexts.**

performance. For example, an image recognition application trained with videos or photos that were taken on a cloudy day may perform differently when analyzing images taken in bright sunlight.

How AI Engineering Operationalizes Development and Deployment

Prototype AI applications are often built to fit specific use cases or as proofs of concept. As organizations adopt AI more broadly, they need to build a foundation of infrastructure, tools, and practices that can be reused and operationalized.

AI engineering is an emerging discipline that focuses on applying tools, systems, and processes to enable AI to perform effectively in various real-world contexts. It provides a framework and tools to design AI systems to function in complex, dynamic, and often unpredictable environments.

The objective of AI engineering is to equip AI development and engineering teams with the tools to build applications that span the full range of operating environments with performance that is observable, understandable, and, thus, trustworthy. These capabilities are needed to accommodate the unique characteristics of AI applications and workflows, as well as the optimized infrastructure they require.

As noted earlier, AI needs to be fed with huge volumes of data, which may include audio, video, images, and language syntax libraries. These data types are applied not only to build initial deep learning models but also to evolve those models through reinforcement learning. If not properly planned, the surge in data volumes can overwhelm available infrastructure, dragging down processing times and undermining model performance.

Most conventional enterprise computing and storage systems can't effectively support enterprise-class AI. As model use spreads, organizations need to build processes and pipelines that can move projects from concept to production quickly, as well as expand as needs demand.

AI engineering emphasizes the use of repeatable processes that optimize infrastructure and determine which tasks can be shared to minimize overhead. It uses an interactive approach that builds up on the data science and engineering disciplines that have emerged with the evolution of large-scale analytics.

Those same disciplines are also needed at the application level to structure and focus on the way organizations build AI applications. Development should proceed from a data-first strategy in which data needs are factored into the design phase of the project and not added as an afterthought.

Three Key Disciplines of AI Engineering

Operationalized AI is best achieved by adopting the three pillars of modern agile development.

DevOps is a widely used agile development methodology that enables innovation at both the functional and operational levels. It endows developers with control of not just the application but the environment in which it operates, enabling the rapid iteration that is required for AI model training.

DevOps adoption is essential to building enterprise-wide AI models because the discipline requires organizations to break down barriers and develop clear and accountable metrics. It creates a foundation for evaluating applications and platforms, as well as implementing cross-functional changes rapidly to meet shared goals. The result is that the entire team has a unified understanding of goals, risks, and metrics.

MLOps brings a similar focus to how data scientists innovate, evaluate, and evolve algorithms for machine learning. This collaborative function, which often includes data scientists, DevOps engineers, and IT infrastructure specialists, focuses on streamlining the process of transitioning machine learning models into production and maintaining them on an ongoing basis. It is part of the process of developing, analyzing, and evolving the models and parameters in machine learning.

DataOps is a collaborative evolution of data engineering that includes analysis and optimization of data flow between data managers and data consumers. Its goal is to create predictable delivery and change management of data, data models, and related artifacts, while establishing consistent processes for evaluating and refining data sources, ensuring data quality, and enforcing data compliance for input to an AI system.

Conclusions

To apply these principles, organizations should break down their functional requirements into the people, processes, data, and tools that will be needed. Establishing a centralized, purpose-built AI infrastructure and/or an AI center of excellence enables organizations to take a methodical approach to solving problems and finding the right people for a project. It identifies which projects are the best to host in the cloud, in the data center, or elsewhere and which elements can be reused across different projects.

Built for the unique demands of Enterprise AI – The NVIDIA DGX™ A100 and the DDN AI400X2.

Using a fully integrated AI platform that orchestrates compute, network and storage is the best way to support these evolving disciplines, providing the tools for continuous delivery and optimization of AI services. Platforms that are purpose-built for AI development, such as NVIDIA DGX™ A100 systems and DDN storage solutions, are architected to optimize execution times without impacting production processes, while accommodating scalability needs without performance degradation. Learn more from DDN and NVIDIA.

This content was commissioned by NVIDIA and DDN and produced by TechTarget Inc.

The NVIDIA DGX™ A100 system features eight NVIDIA GPUs and two 2nd Gen AMD EPYC™ processors.