MACHINE LEARNING & THE INTELLIGENT EDGE

THE STRATEGIC OPTION FOR DEVELOPING & DEPLOYING ALGORITHMS

It has been over a decade since Amazon Web Services (AWS) burst onto the scene and introduced the notion of cloud computing to the world. Cloud computing was truly transformative, as predicted. But it took a long time for companies to make sense of it. Beyond weighing the pros/cons, they needed to gain practical experience to determine *where* they should deploy their workloads: on-premise, in the cloud, or a mixture of the two.

A similar situation is happening with machine learning. Since 2015, machine learning – and more recently, deep learning (a form of machine learning) – has appeared at the peak of inflated expectations according to Gartner's Hype Cycle for Emerging Technology. Per Gartner's model, that means we're about 2-5 years away from reaching the plateau of productivity. We can see the profound potential of machine learning in business and our personal lives. However, we'll have to overcome hyped-up expectations before we reach mass-market adoption and achieve our desired business outcomes.

AN OPPORTUNITY RIPE FOR THE TAKING

According to Scott Brinker – VP platform ecosystem of HubSpot and previously CTO of ion interactive – now is the time for companies to take action. As he says, while hype rules the day at the peak of inflated expectations, a technology's actual potential is often underestimated in the next phase (the trough of disillusionment). Brinker advises that the "desync between expectations and reality is a good thing — if you know what you're doing. The gap between expectations and reality creates opportunities for a savvy company to manage to the reality while competitors chase the hype cycle."[1]

Just as businesses had to sort through where to deploy their cloud workloads, developers now face a decision as to *where* to apply machine learning algorithms. It turns out that *where* data science algorithms are trained and deployed can arguably have as much of an impact, if not more, than the actual algorithms themselves. In other words, choose wisely and your business can exploit the potential of machine learning algorithms to the fullest.

Gartner predicts that by 2022, 50% of enterprise generated data will be created and processed outside the traditional, centralized data center or cloud, up from less than 10% in 2018[1]. It will be created at the edge, such as within a factory, on an airplane or oil rig, in a retail store floor, or inside a medical device. To convert this data into actionable intelligence, machine learning must analyze it at the edge quickly.

[1] ThinkGrowth.org, One thing everybody forgets about Gartner's hype cycle: The gaps between hype and reality are opportunities

WEIGHING THREE MACHINE LEARNING ALGORITHM DEVELOPMENT & DEPLOYMENT OPTIONS: THE PROS & CONS

As shown in Figure 1, developers can develop and execute machine learning in three primary places, each with its own distinct advantages and challenges:

(1) In the cloud

(2) At the edge, in the device/gateway

(3) At the edge, near the device

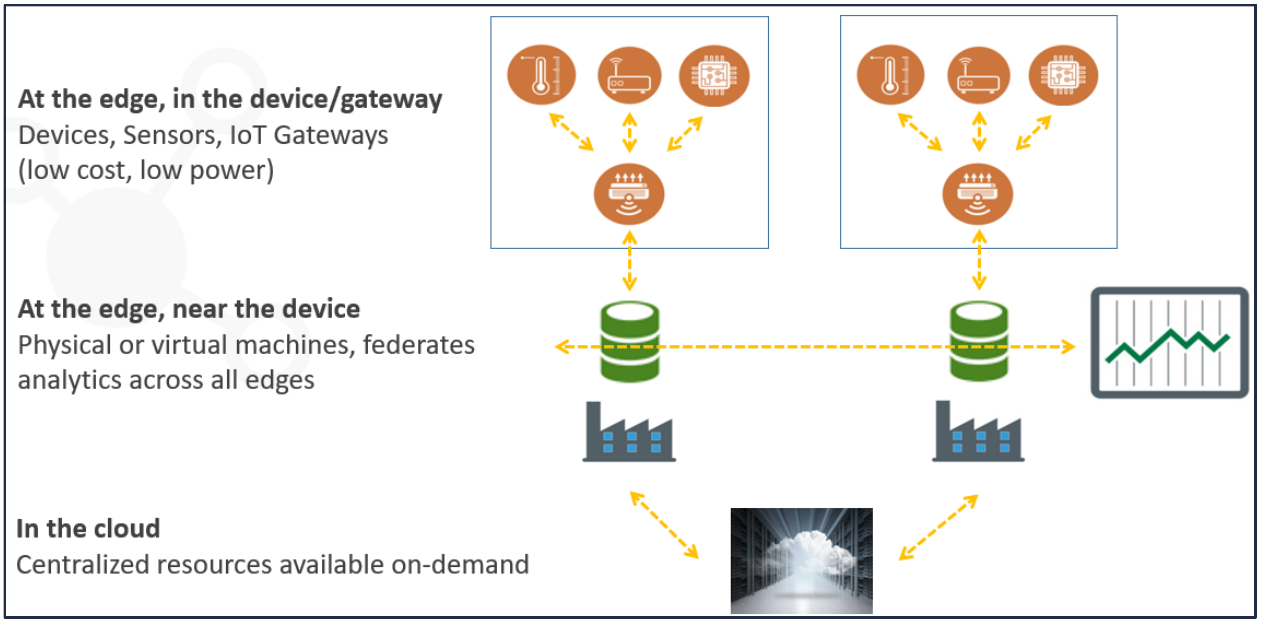

At the edge, in the device/gateway
Devices, Sensors, IoT Gateways
(low cost, low power)
At the edge, near the device
Physical or virtual machines, federates
analytics across all edges
In the cloud
Centralized resources available on-demand

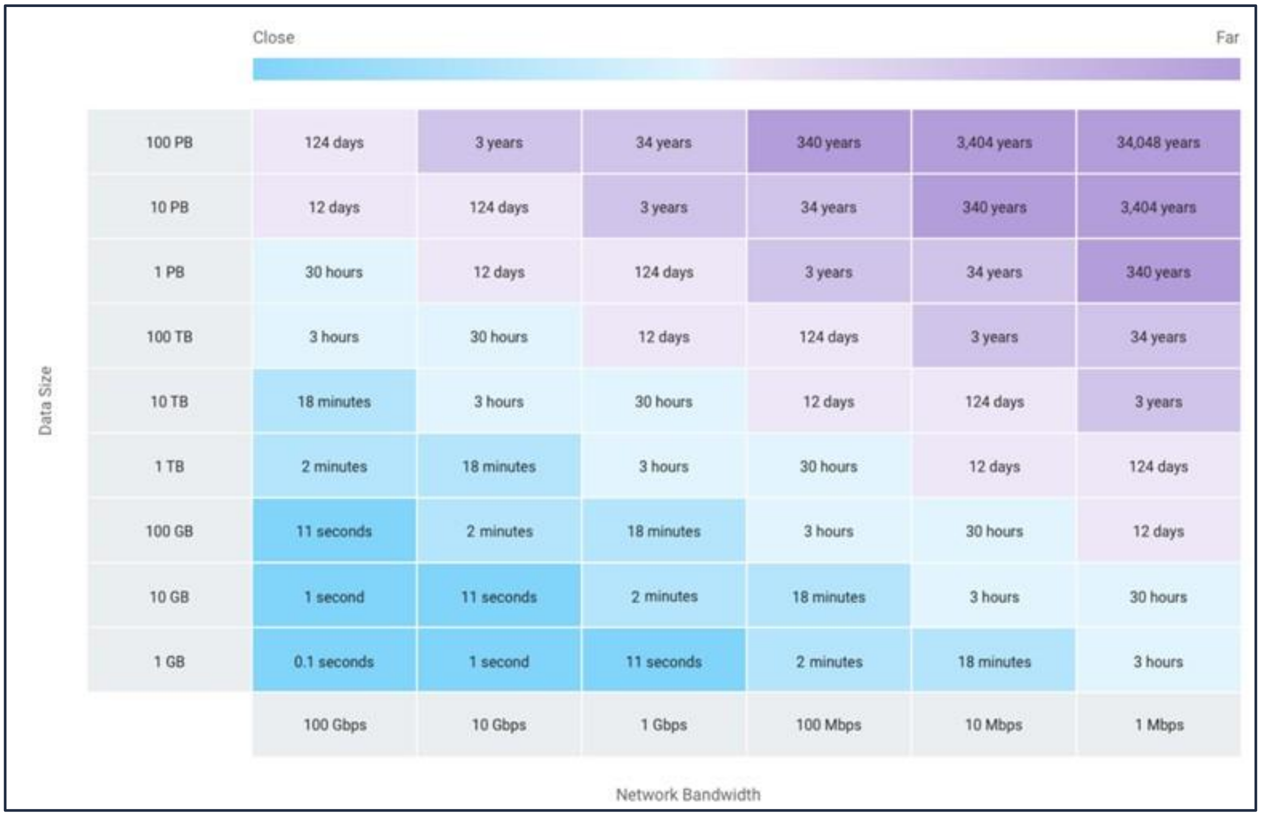
Figure 1. Three places to develop and execute machine learning

1. MACHINE LEARNING APPLIED IN THE CLOUD

Performing machine learning in the cloud is the prominent and familiar method today. Large cloud platform providers such as AWS, Azure and Google offer machine learning services. Amazon SageMaker, Azure Machine Learning Studio and Cloud Datalab from Google enable data scientists and developers to quickly and easily build, train and deploy machine-learning models. Support for deep-learning frameworks such as TensorFlow enables an open, flexible environment. It’s easy to prepare and load data for machine learning directly from cloud-based storage and data warehouse services offered by the cloud provider.

The primary limitation of this approach is the challenge of moving the data from the edge to a cloud data center so it can be used to prepare and develop machine-learning models. Numerous latency and cost issues make it impractical to move large volumes of data to a centralized data center. Consider the table

Google created to help companies get a sense of how much time it may take based on available network bandwidth, size of the data to be transferred, and closeness to Google Cloud Platform. Note that Google's own analysis shows it can take days, months, or even *years* to ship large volumes of data across geographies (see Figure 2).


Close
Far
Data Size
Network Bandwidth

100 PB	124 days	3 years	34 years	340 years	3,404 years	34,048 years
10 PB	12 days	124 days	3 years	34 years	340 years	3,404 years
1 PB	30 hours	12 days	124 days	3 years	34 years	340 years
100 TB	3 hours	30 hours	12 days	124 days	3 years	34 years
10 TB	18 minutes	3 hours	30 hours	12 days	124 days	3 years
1 TB	2 minutes	18 minutes	3 hours	30 hours	12 days	124 days
100 GB	11 seconds	2 minutes	18 minutes	3 hours	30 hours	12 days
10 GB	1 second	11 seconds	2 minutes	18 minutes	3 hours	30 hours
1 GB	0.1 seconds	1 second	11 seconds	2 minutes	18 minutes	3 hours
	100 Gbps	10 Gbps	1 Gbps	100 Mbps	10 Mbps	1 Mbps

Figure 2. Time to transfer big data sets to Google Cloud Platform

Source: Google, Transferring Big Data Sets, https://cloud.google.com/solutions/transferring-big-data-sets-to-gcp

Even with a 100 Mbps Internet connection, Google estimates it will take 12 days to ship 10TB of data to the cloud. In a world where many connected devices generate hundreds of megabytes or even terabytes of data every day, this latency is unacceptable.

Since machine learning accuracy is only as good as the data itself, companies face an unwanted tradeoff. They must select "sample" data to transfer to the cloud to train and continuously refine their machine-learning algorithm because it is often impractical to move all their data. The drawbacks don't end there: Data scientists are forced to wait for the most recent set of newly generated data to be shipped to the cloud. Simply put, businesses are delayed gaining actionable insight from newly generated data and are hamstrung in their efforts to analyze a complete dataset combining the newest data and historical data.

Pros and Cons of Machine Learning in the Cloud

PROS: Ease; flexibility to develop algorithms within major cloud platforms; and abundance of CPU/GPU resources required for deep analysis.
CONS: Slow transfer and loading of large data volumes; smart sampling of data required to train and refine machine-learning models; latency due to distance from data center to devices; privacy concerns associated with moving data from one geography to another.

2. MACHINE LEARNING PERFORMED AT THE EDGE, IN THE DEVICE

Data volumes are rapidly rising with as more machines and sensors generate hundreds of terabytes, or even petabytes, of data. It's increasingly impractical to transfer such large volumes of data to the cloud. With the growing need to take action in real – or near-real –time, it becomes increasingly important to shift aspects of machine learning to the edge, instead of or in combination with the cloud.

Machine-learning algorithms benefit from "intelligent edge" software embedded within a device that fits into one of the following approaches:

- *Developed/trained in the cloud and uploaded into the device to execute close to where data is generated.* This is the simplest method as it preserves the ease and flexibility of developing machine learning in the cloud but executes the algorithm close to the data source. As a result, it's possible to apply the algorithm to incoming data in real time. Small amounts of data might need to be transmitted to the cloud when additional processing resources are required to augment a capacity-constrained device. The same applies when data scientists need to refine the machine-learning model with summarized training updates using novel methods such as those proposed by Google with Federated Learning.

- *Developed/trained exclusively at the edge, in the device, based on the data that flows through it.* This approach leverages intelligent-edge software where embedded system design or operational technology (OT)-friendly GUI environments serve as the foundation for development environments. To accomplish this fully at the edge, developers must often rely on a digital twin, or virtual model, that is updated regularly with newly generated data. Low-code platforms make it faster and easier to develop intelligent edge software and apply machine learning for embedded use within low-cost devices and gateways.

While these two edge approaches help overcome latency concerns, they can suffer from accuracy issues. Cloud-developed machine learning brought to the edge is based on a sampling of the data. Moreover, with the first approach above, developers will find it difficult to develop highly accurate machine-learning models exclusively within the edge devices. These machines and sensors are optimized for low cost and low power. As a result, in the best-case scenario, they can retain several hours of data and apply machine learning on a limited basis for select use cases. Plus, the device does not retain historical data due to capacity limitations. Algorithm refinement is unique to each device and can't easily learn from data flowing through other devices.

In the second approach, machine learning in the device doesn't solve the need for long-term data retention and federated analytics. In other words, this approach doesn't simplify efforts to apply newly generated and historical data individually, in groups, or in aggregate as part of training and refining machine learning algorithms.

Pros and Cons of Machine Learning at the Edge, in the Device

PROS: Ease of moving machine-learning algorithms developed in the cloud to the edge; ability to train machine-learning algorithms at the edge based on real-time data; and low-latency, edge processing.
CONS: Cannot support a complete dataset for modeling; unable to retain historical data for analysis; and each edge device cannot learn easily from other similar devices, forcing reliance on digital twin modeling.

3. MACHINE LEARNING PERFORMED AT THE EDGE, NEAR THE DEVICE

While the least prominent and least familiar method, machine learning at the edge near the device addresses the limitations of latency as well as modeling accuracy. This approach incorporates an edge aggregation layer with edge resources that reside near – but not within – the device. This layer can aggregate data from thousands and even millions of individual devices, scales to petabytes, and federates across geographies. As a result, it brings the attributes of a big data warehouse to edge computing environments while applying machine learning on distributed data from all edges/devices simultaneously.

The primary benefit to this approach is that it overcomes the latency and cost issues associated with transferring data to the cloud. Yet this approach enables highly accurate modeling at the edge – something not possible in the second approach due to the limited analytical capabilities within each device. That's because machine learning at the edge near the device can leverage a complete dataset, across geographies, using data of any age – whether it is seconds, months or even years old. This approach is also ideal when data privacy or compliance is a top concern and the data can't be moved off-premise to another location. Moreover, developers can use this approach to train and refine machine learning models at the edge, and complement in the in-device approach if models need to run "closed loop", or autonomously, within devices.

The Edge Intelligence approach provides intelligent edge software that serves as a distributed, federated analytics platform. This platform is based on an SQL architecture that incorporates edge aggregation near, but not within, the device. As a result, it enables data analytics, near real-time stream processing and machine-learning algorithm training and refinement to operate on the data locally, directly within a geographically distributed database. This eliminates the need to transfer distributed data across geographies. In addition, unlike cloud-based approaches, the Edge Intelligence approach data obviates the need for organizations to load their data from its original storage repository into a separate system to perform machine-learning analysis.

Apache MADlib serves one means for performing geographically distributed machine learning. This allows machine learning to be performed in a massively parallel manner across a distributed set of edge locations. Developers and data scientists can start quickly by leveraging open source algorithms. Data scientists can perform supervised learning, unsupervised learning, time series, nearest neighbors and other methods on a system that scales to petabytes of data stored. Developers and data scientists who know R but very little SQL also benefit from the performance and scalability benefits of MADlib. The system translates R model formulas into corresponding SQL statements, executes these statements in the database, and returns the summarized model output to R.

Other emerging approaches are directionally similar, but based on different intelligent edge software and underlying database. For example, Dell World Wide Herd (WWH) calls upon a global network of Apache Hadoop instances to push analytics where the data resides. Other vendors including SAS are looking at ways to overcome the problems of centralized data while moving event stream processing capabilities usually associated with a centralized data warehouse to a distributed, edge architecture.

Hewlett Packard Enterprise plans to invest $4 billion in Intelligent Edge technologies and services by 2022 to help customers turn their data – from every edge to any cloud – into intelligence.[2] Microsoft is investing $5 billion in IoT, which it says “is ultimately evolving to be the new intelligent edge.”[3] And in August 2018, VMware unveiled its extended edge computing strategy, sharing plans to develop a framework that extends the VMware hybrid and multi-cloud environments to the edge.[4]

Pros and Cons of Machine Learning at the Edge, Near the Device

PROS: Can develop highly accurate models based on the most complete set of data; can address privacy/compliance concerns; and can conduct near real-time decision-making due to low-latency edge processing (e.g., SQL stream processing).
CONS: Limited flexibility for data scientists and developers, although improving as intelligent edge software matures and gains additional awareness, investment and supporting ecosystem.

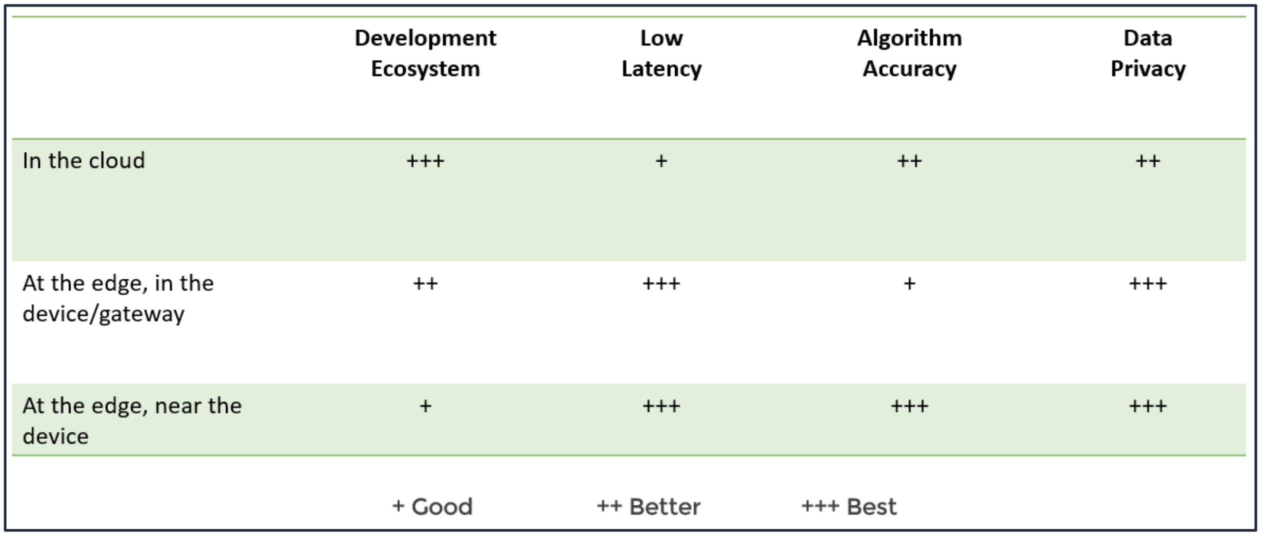
[2] LightReading, HPE to Make 4-Year $4B Intelligent Edge Investment, 6/19/2018
[3] Microsoft Internet of Things blog, Microsoft will invest $5 billion in IoT. Here’s why.
[4] VMware, VMware Extends Digital Foundation to Edge and IoT, August 28, 2018

LOOKING FORWARD: THE PROMISE OF INTELLIGENT EDGE SOFTWARE

Choosing where to develop and execute machine-learning algorithms is a game-changing decision. The final choice will greatly impact how well and how accurately applications can make decisions and how quickly they can respond to events.

Some assume that the deeper the learning, the more pressing the need it to move it to the cloud. However, intelligent edge software is making it possible for companies to develop and deploy sophisticated machine learning entirely at the edge. Such an approach provides the advantages of ample CPU/GPU processing power, and a federated-edge aggregation layer capable of retaining petabytes of data and providing instant access to data whether it is seconds or years old.

The next few years will be fascinating as companies consider the best approach for solving the use cases to which they are applying machine learning. What will be most telling is their choice of *where* to develop and execute their models to ensure development ease and flexibility while addressing data latency, model accuracy and data privacy. While no single approach applies to all use cases, rapidly evolving intelligent edge software technology is a promising option since it addresses gaps caused by transferring data to centralized resources on-premise and in the cloud.



	Development Ecosystem	Low Latency	Algorithm Accuracy	Data Privacy
In the cloud	+++	+	++	++
At the edge, in the device/gateway	++	+++	+	+++
At the edge, near the device	+	+++	+++	+++

+ Good ++ Better +++ Best

Figure 3. Summary of machine learning deployment options

If you'd like to see our unique approach to powerful SQL-based analytics and in-database machine learning applied instantly to petabytes of globally distributed data – without the need to move the data – please request a no-obligation demo.