SCTE·ISBE CABLE-TEC EXPO 2018 ATLANTA, GA OCTOBER 22-25

2018 Fall Technical Forum SCTE·ISBE • NCTA • CABLELABS

Computing At The Edge Still Has An Edge

A Technical Paper prepared for SCTE•ISBE by

Arun Ravisankar
Senior Engineer, Comcast Labs
Comcast Corporation
1701 JFK BLVD, Philadelphia, PA 19103
Phone:2152867558
Arun_Ravisankar@comcast.com

Table of Contents

List of Figures

List of Tables

Introduction

History is witness to the evolution of civilizations and how humans continue to discover and innovate things that would propel everyone to a newer level of technological advances, as we aspire to attain a higher intellectual state. Industrial revolutions are key indicators of how humankind continues to seek techniques that would improve lifestyles and bring advancement to civilization. The first industrial revolution was about mechanization, which involved the development of machine tools and the rise of huge factories and factory systems. The second revolution, also known as the Technological Revolution, brought about a rapid rise in industrialization, which involved increases in automation. Digitization can be seen as the third industrial revolution, where digital systems of all types saw an increase in adoption.

The fourth industrial revolution could be envisioned as a function of AI (Artificial Intelligence) and ML (Machine Learning), which are vital in building "Intelligent Machines." It follows that those "Intelligent Machines" could be referenced as "the compute edge," as opposed to "the network edge" -- in our case, usually defined as the node, where optical-to-RF conversion occurs. AI/ML technologies influence a large part of the devices and services we use on a daily basis, be it a voice assistant or vehicular parking assist, or be it an entertainment platform that understands our preferences and predicts shows and titles we may like. Apart from these examples, many AI/ML-based applications can help improve lifestyles and bring peace of mind to customers.

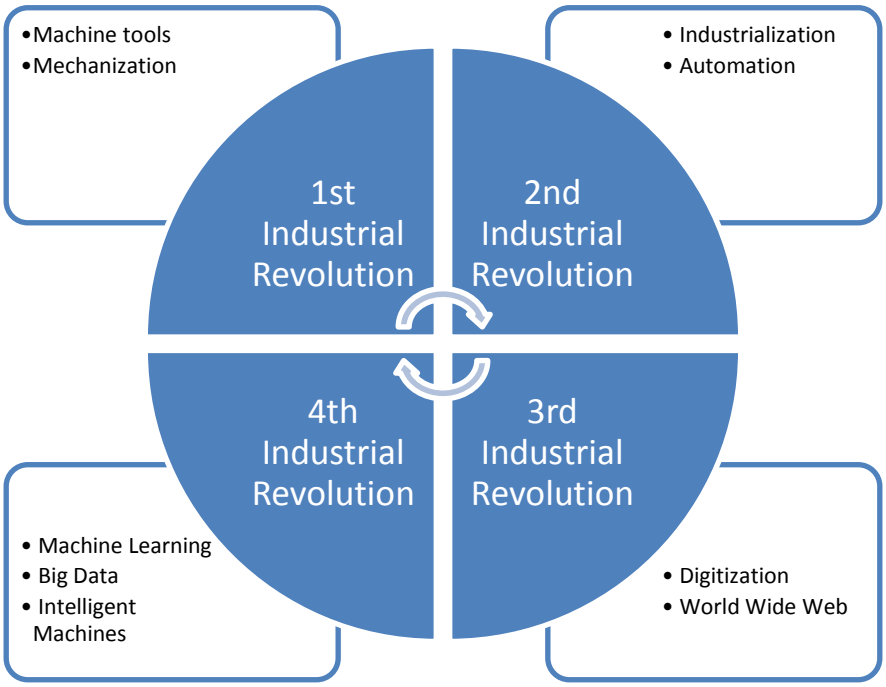

Figure 1 - Evolution of technology and its influence in the society

AI/ML plays a vital role in almost any products and services that are offered to customers now. Any application or service rendered in a customer's home, be it via a set-top box (STB), DOCSIS-based gateway, home automation gateway, or IoT device, involves multiple components working in tandem. The "compute edge" discussed in this paper is comprised of those in-home devices. Because "edges" in general vary widely, for the fourth industrial revolution -- AI and ML -- we define the "compute edge" as

the premise. That necessarily includes devices in the premises, linked to applications running on cloud servers that are racked up in a data center.

IoT applications process data from devices at the edge and are subject to a decision tree usually deployed on a cloud server. The decision tree or rules engine determines the course of action for data sent from a device. With the advent of machine learning and artificial intelligence, and given their natural fit with IoT applications, the demand for higher computing power has increased significantly. Now, with the increase in silicon capabilities that accelerate AI/ML algorithms, devices on the edge can process some information locally, which move some parts of the decision tree to the edge. This paper will discuss how edge compute could improve the delivery of IoT applications.

Machine Learning Overview

Machine Learning techniques involve statistical algorithms that give computers the ability to learn. This helps machines to progressively improve their performance on a specific task. The learning process is automatic relative to the data being gathered, and does not involve explicit programming. Most Machine Learning algorithms could be grouped into the following classes (see Table 1). These algorithmic classes add value to service providers:

1. Classifiers
2. Clustering Algorithms
3. Recommender Systems
4. Anomaly Detection Algorithms
5. Linear Regression

The table below shows a basic description and examples of each of the above algorithms.

Table 1 - Machine learning algorithms and examples

Class of Algorithms	Description	Technology Examples	Applications
Classifiers	Assigns new inputs to one or more classes, based on similarity to other data	Neural Networks	Image Classification, Spam filtering
Clustering Algorithms	Groups similar data into clusters	K-Means	User Profiles and anomaly detection
Recommender Systems	Makes recommendations based on historical data	Filtering	Product recommendations
Anomaly Detection	Detects rare events, usually not normal	Joint Probabilistic modelling	Fraud detection, Home Security and healthcare use cases
Linear Regression	Predicts values for continuous variables	Linear Regression	Churn Rate Prediction

Machine Learning application development usually involves two parallel, yet connected, processes. One is a modelling workstream, and the other workstream involves deployment of the models. The first workstream, as its name indicates, is more centered on the modeling effort. The second workstream focuses on ensuring that there is a path to deployment for the models being developed. The two efforts

are viewed as happening concurrently, because of the complex nature of deploying a machine learning solution in a cable system operator's production environment.

Figure 2 shows the process in a typical machine learning-based application. The aspects shown in Figure 2 can be categorized into two tracks. One track is of model development and other track would be of deployment and integration.

Figure 2 - Machine Learning Process and Events involved

The events and steps shown in Figure 2 could be split into model development and deployment. Model *development* includes the following characteristics:

- Business/Data Understanding
- Data Preparation
- Modelling

Model *deployment* includes the following characteristics:

- Evaluation
- Deployment and Integration

Model development involves the "learning" process, where training data is used to build models. Learning processes are usually done on high performance systems and are resource intensive, as learning process involves processing a large amount of data to prepare models.

The model is deployed and executed based on the data that is received by the system. The execution depends on the use case; the models are built based on those use cases.

Computing at the Edge

Once the models are developed, they are deployed on devices that execute these models, on test data, and arrive at conclusions that depend on the particular use case. Figure 3 shows a simple depiction of the learning process, where systems work on the data to create models. These systems are compute-intense, and require necessary infrastructure to be set up.

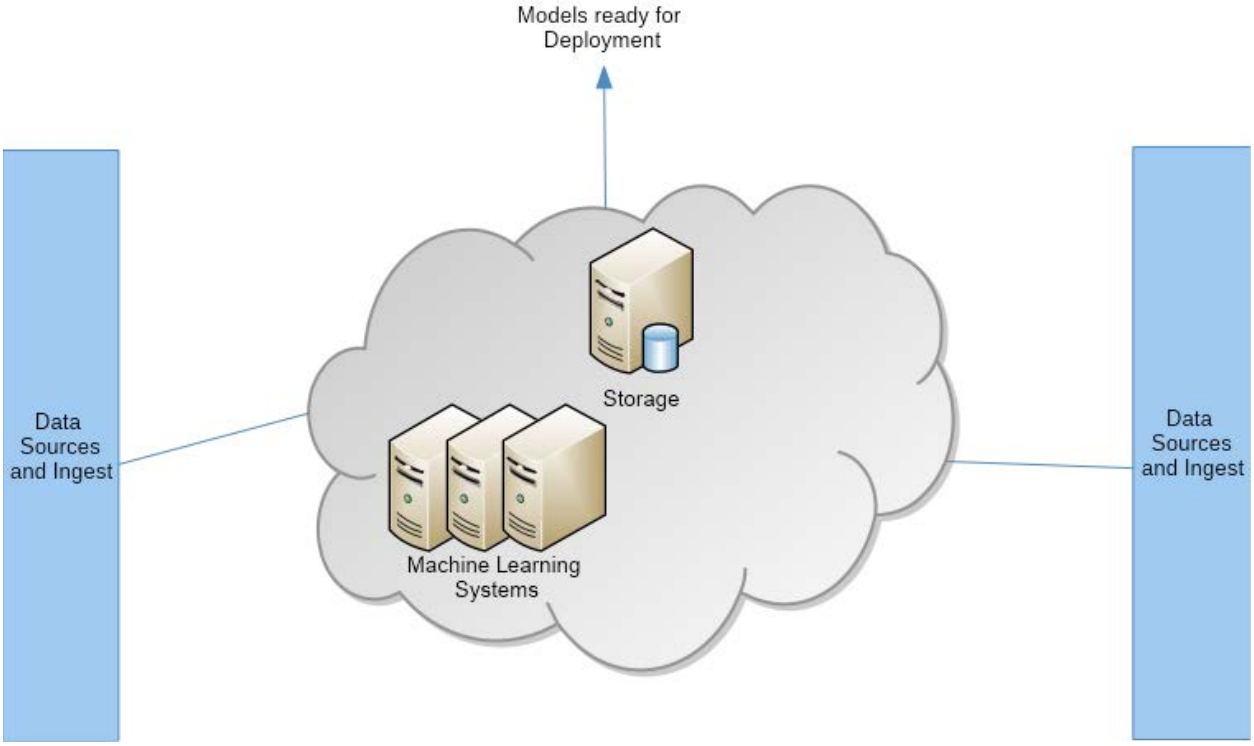

Figure 3 - Machine Learning Systems

Once the models are ready for deployment, they are deployed on systems that can apply them to the live data coming in from the various sources. For example, a video analytics-based ML application would use models that were trained using images and videos. Once the model is trained, images and video from a camera are analyzed by applying these models. In this specific example, the inferencing engine needs to process the video signals and then apply the model as deployed. The use cases could vary between, say, monitoring an area to monitoring facial expressions. Hence the inferencing engine would also require high performance computing in order to provide results accurately, with minimum latency. It would be a stretch for the customer premises equipment presently deployed to meet these compute requirements. Because of the need for high levels of processing, inferencing engines are often deployed in a cloud infrastructure, where units could be racked to meet the compute and power requirements.

Figure 4 shows how a cloud-based inferencing engine would operate on the data being ingested to provide services to the consumer by executing the rules defined by the models.

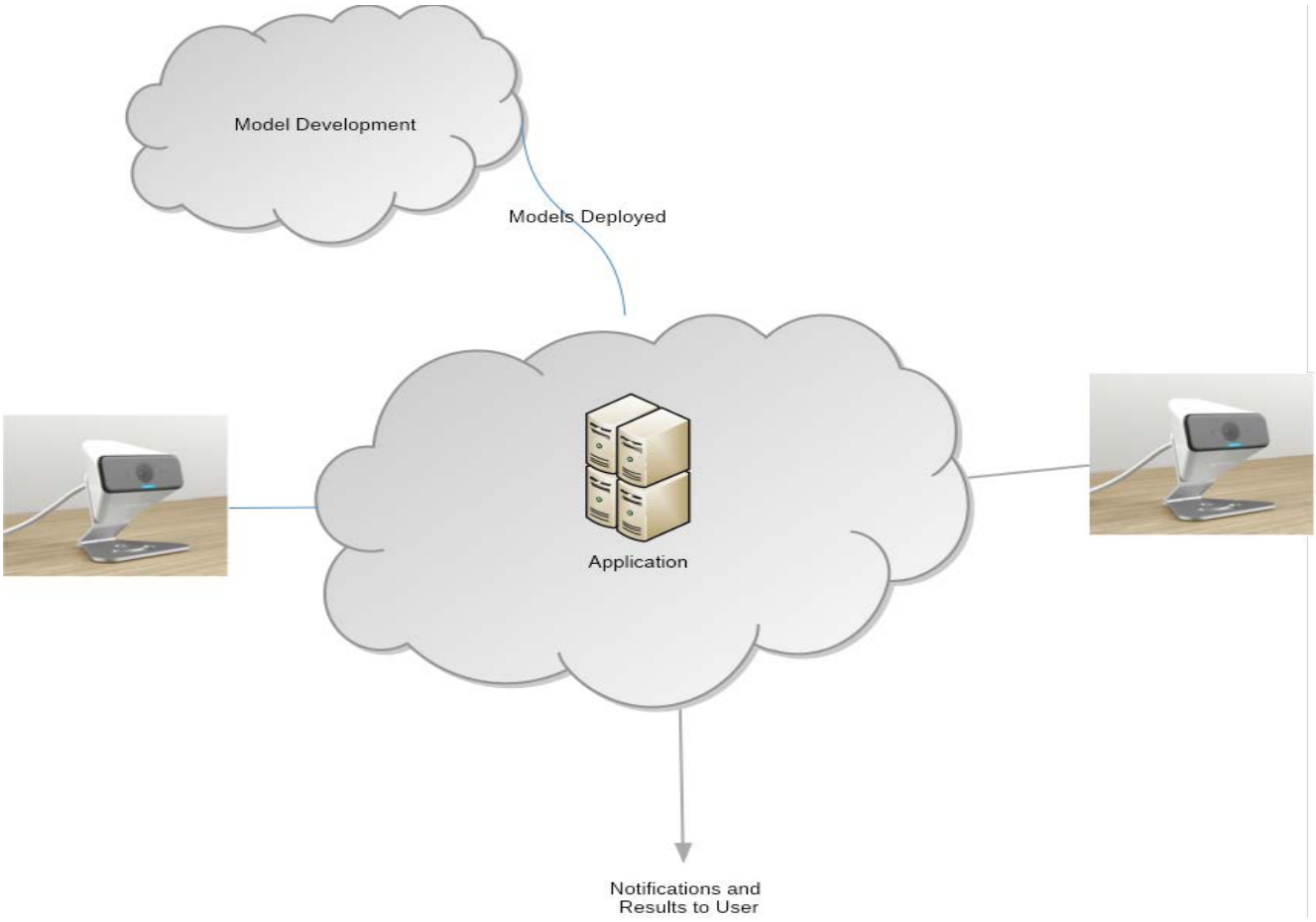

Figure 4 - Cloud-based Inferencing Engine

Newer, advanced hardware platforms offer higher compute performance, while requiring less power and memory. These hardware acceleration platforms provide the basis to run AI/ML-based algorithms. Compute power throughput is measured in Tera Operations Per Second (TOPS) or Tera Floating-point Operations Per Second(TFLOPS). Most AI/ML-based applications run on platforms that offer about 0.5 to 1 TFLOPS. Another important metric is the efficiency of the processor architecture, and is measured in GFLOPS/W, which translates to Giga Floating-point Operations Per Second per Watt of energy consumed.

The efficiency factor determines if the system is best deployed in a rack at a location and services are accessed through cloud, or if the system could be deployed at the edge, again meaning the premise.

Major and sustained advancements in silicon manufacturing have led to the development of high efficiency processors that can support a throughput that is comparable to most of the high-performance CPUs that are deployed. Next, we will look at the use cases best suited for these processors, in terms of improving the overall experience with ML/AI.

One of the major advantages of computing at the edge is the improvement in latency of the system, because the data is processed at the premises, rather than being sent over a network to a server-based processing engine. Hence, applications that need fast response times tend to require edge compute resources.

1. Use Cases

Automotive Applications: Most cars now offer several driver-assist features that use a variety of sensors. The data coming in from these sensors needs to be processed in real-time, so that alerts or actions can be executed. This involves processing a lot of data, and the processing needs minimal latency. Apart from driver-assist features, as shown in Figure 5, there is an increased level of interest in the automobile industry to build self-driving cars. These cars function similarly to airplane auto-pilot mechanisms, where human intervention is required in specific circumstances. Imagine the amount of computing involved, if we need to match the sophistication that is equivalent to an airplane! This needs a prohibitively large amount of computing -- and the computing has to happen in real-time. In such cases, most and in fact all of the computing needs to happen at the edge (in this case, the on-board computer of the automobile). The system can then process signals from various sensors and initiate appropriate actions.

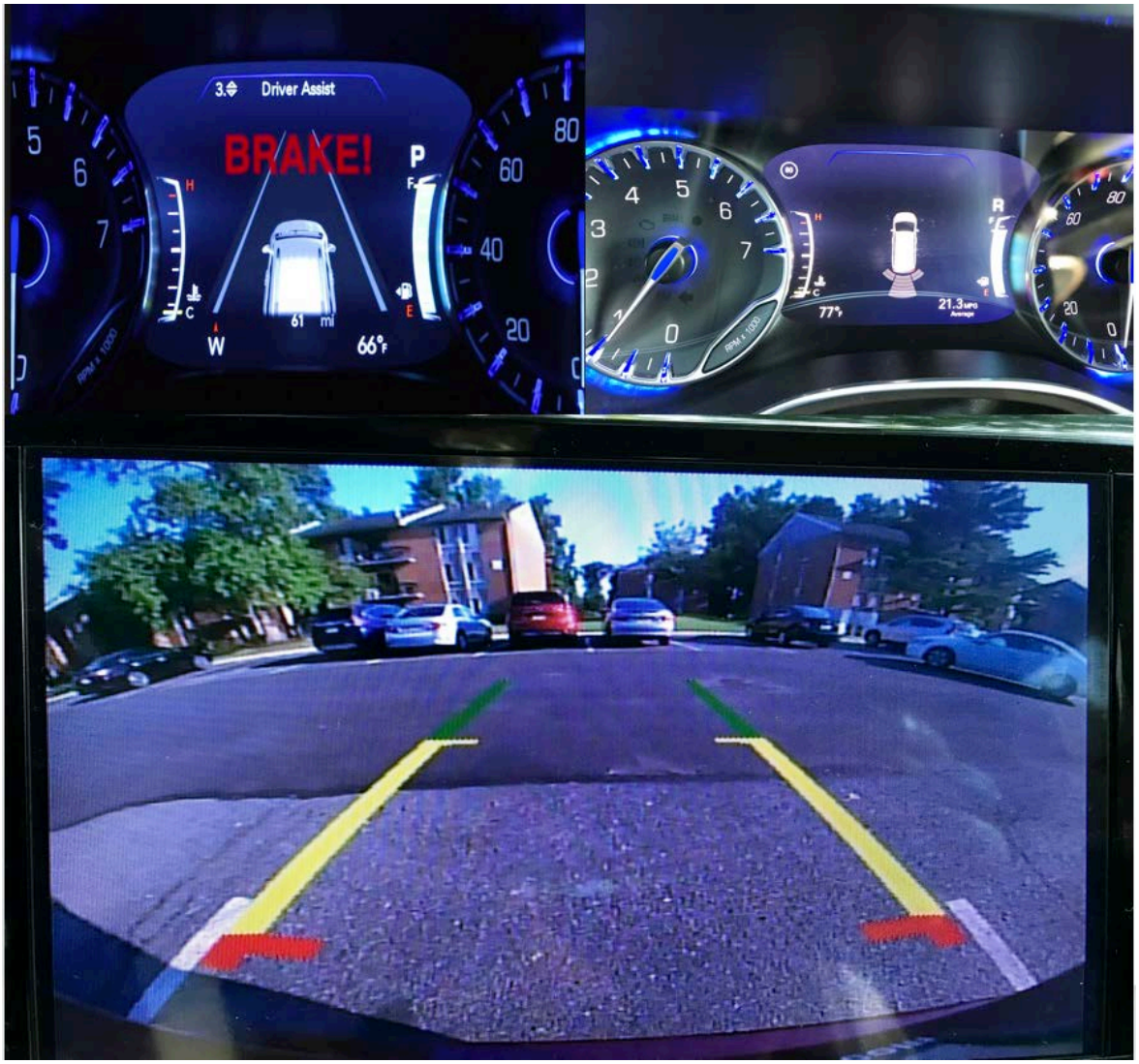

Figure 5 - Driver-assist features in a car

IoT Applications: Beyond how edge computing technology drives the development of next-gen automobiles, these systems could also be leveraged in IoT (Internet of Things) applications, like home monitoring, security, and healthcare, to name a few. Many IoT applications involve anomaly detection, in which the application processes data coming in from various sensors. These applications deploy machine learning models, that process data from various sensors -- for example, an application that detect events based on video feeds from security cameras, which use computer-vision based models to analyze the current situation. This also involves a significant amount of processing, for both video and AI.

Figure 6 - Example analyses of video from a security camera

Figure 6 depicts examples of events that homeowners tend to be interested in knowing: Who's at the door, what's that truck parked out front, and so on. In a camera-based application, AI/ML based models are deployed on high performance inferencing engines to analyze data derived from the camera feed. These models can be deployed on processors that provide acceleration to AI/ML models to perform tasks that are time-critical at the premise, while further processing and learning could be carried out in cloud-based servers.

If the camera (or any premises equipment) is built with silicon that provides hardware-based acceleration to run AI/ML models and algorithms, there could be significant improvement in latency of the system. Apart from latency, there would also be an improved sense of privacy (in a camera-based application), because the images are being analyzed at the premises and might not ever leave the premises.

Healthcare: IoT technologies contribute immensely to connected healthcare applications. With the use of IoT and AI/ML technologies, monitoring health and wellness could be significantly expanded to provide peace of mind to people who care for family members and patients. Eldercare is a classic example: A combination of IoT and AI/ML technologies can be used to monitor daily activities of the elderly, and notify either the care provider or the family member in the event of perceived abnormalities. Solutions can be built that detect falls, or analyze gaits and alert the appropriate caregiver. Using computer vision, itself a subsystem of AI/ML, such systems can identify both objects and people, and can determine activities, detect falls and otherwise inform a healthcare application. Figure 7 shows an application that can determine the activity of individuals using video analytics.

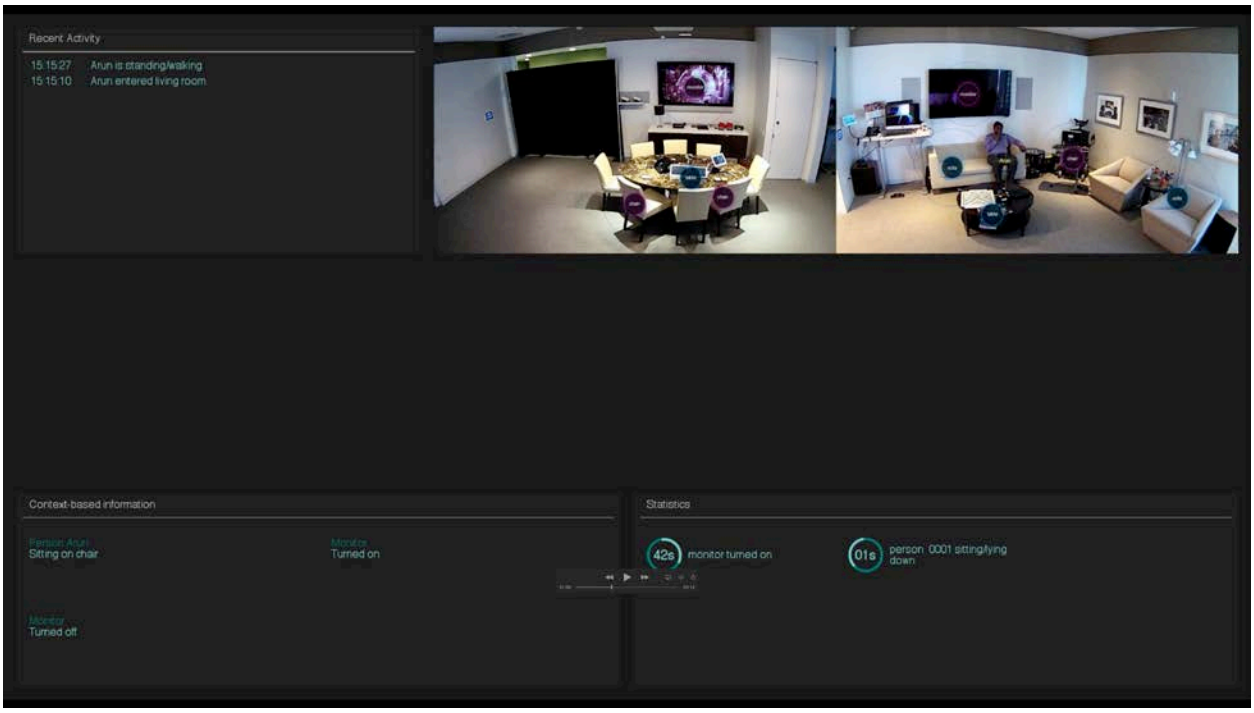

Figure 7 - Activity Determination using Machine Learning

The application depicted here can recognize common household objects like tables, chairs, couches, and TVs. The application can also determine if the TV is on/off, and can identify a person, in a way that is differentiated from a visitor coming into the home. This can be seen in Figure 9, which shows how the objects in the room are identified. Figure 8 explains the process in which an edge compute system could operate. The example chosen is a computer-vision based system, where the models are developed in a cloud-based system by analyzing a vast amount of training data. These models are deployed on the edge system (premises) and the software on the edge uses the on-board AI/ML acceleration features to perform inferencing and display results to the user.

In this example, the camera is used as a sensor. Similar applications could be built using other sensors deployed in home, for example, motion sensors, or door/window sensors. We can also look at the potential of RF sensing for these use cases (including both WiFi and RADAR). The type of sensors used determines the data format and hence the models that are created. Appropriate models have to be developed to work on the sensor, such that it meets the application's requirements. A computer vision-based application was an easier choice for a proof of concept, because of the vast sets of training data available, and because the training data can be continuously generated using a camera.

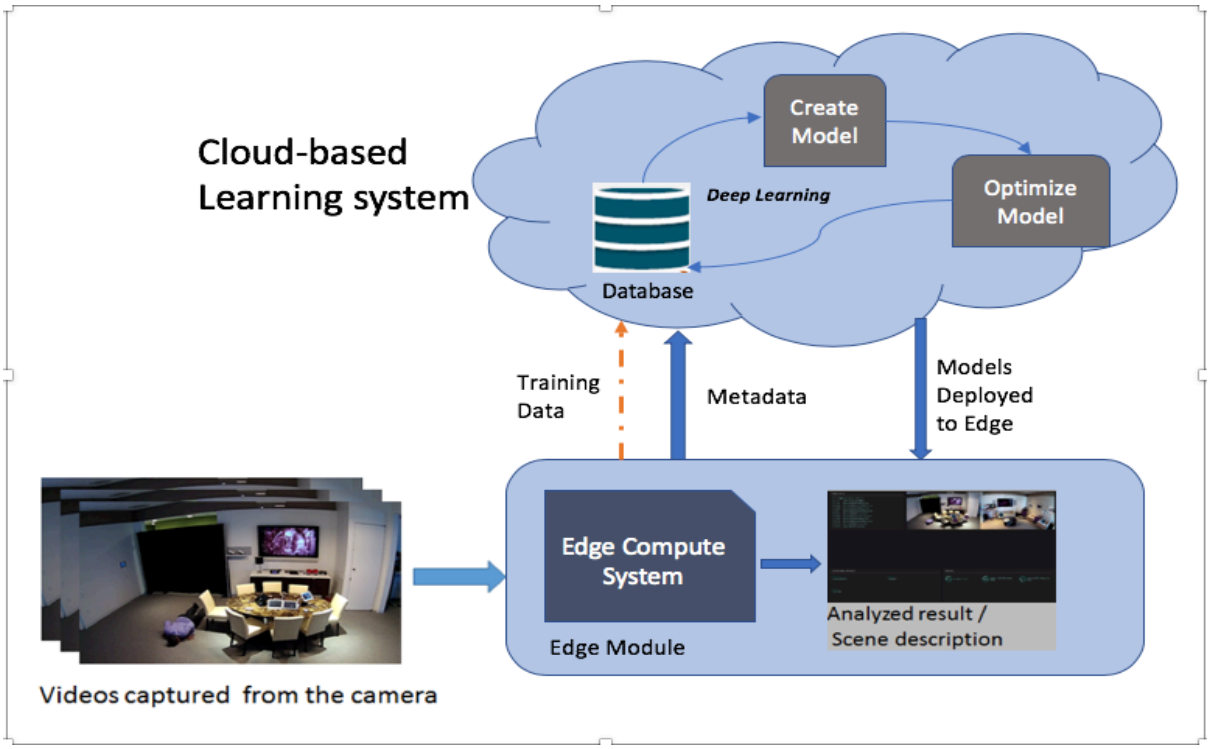

Figure 8 - Edge compute process example

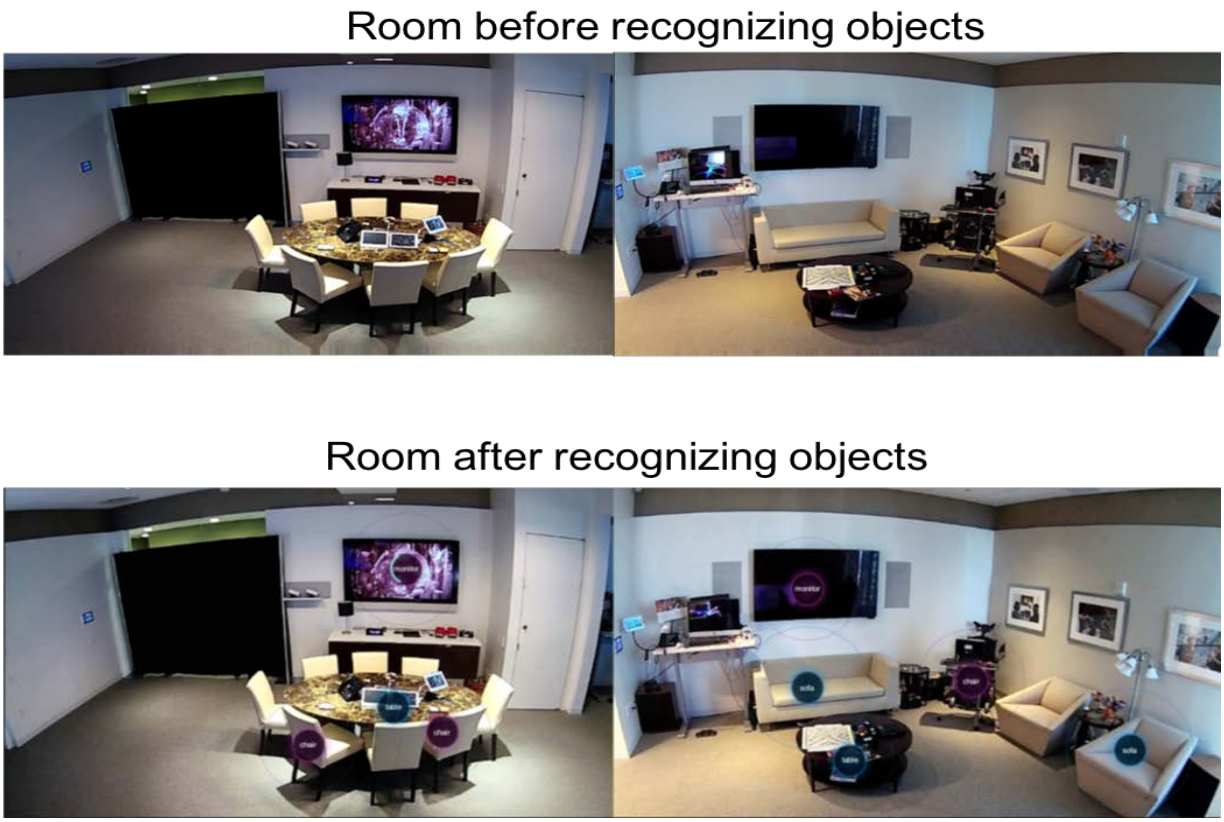

Figure 9 - Object recognition using Machine Learning

Home Automation/Smart Assistants: Another relevant aspect of everyday life is the role smart-assistants and home automation can play. Interaction with these devices is gradually increasing. Devices like Alexa and Google Home have become an everyday lifestyle tool for many people. Be it "Alexa, where are my keys?" or "Hey Google play my favorite radio station", we use these smart assistants for a variety of purposes. These systems, including the voice remote, which is seeing steadily increased usage, are based on machine learning applications. They have to process speech and the language being spoken by the user. For this they use Natural Language Processing (NLP) algorithms. In most cases, a microphone lists the words spoken by the user (upon trigger/wake word) and then sends the corresponding audio packets to a system that converts speech into text, so that a computer can decipher the contents. Once converted to text, the data is processed using NLP to understand the requests from the user, and advanced AI methods are applied to understand the context and intent factor, so that the response to a query is as accurate as possible.

Most of the AI/ML processing is done on high performance systems, and for that reason, the compute edge can play an important role to augment the processing by performing some analysis on the edge, while more complex analysis and learning is done in cloud. This improves latency and is additionally useful in scenarios where network connectivity is poor or lost.

As shown in Figure 10, the processing of speech and analysis mostly happens in a cloud-based system, yet sometimes, the speech to text could be done on the edge, with assistance on NLP and AI in the cloud before the response is sent back to the user.

With the continuing advances in silicon and processor architectures, part of the processing, including the NLP, could move to the compute edge. This would both improve response times and help in situations where the network connectivity is poor or suffering an outage.

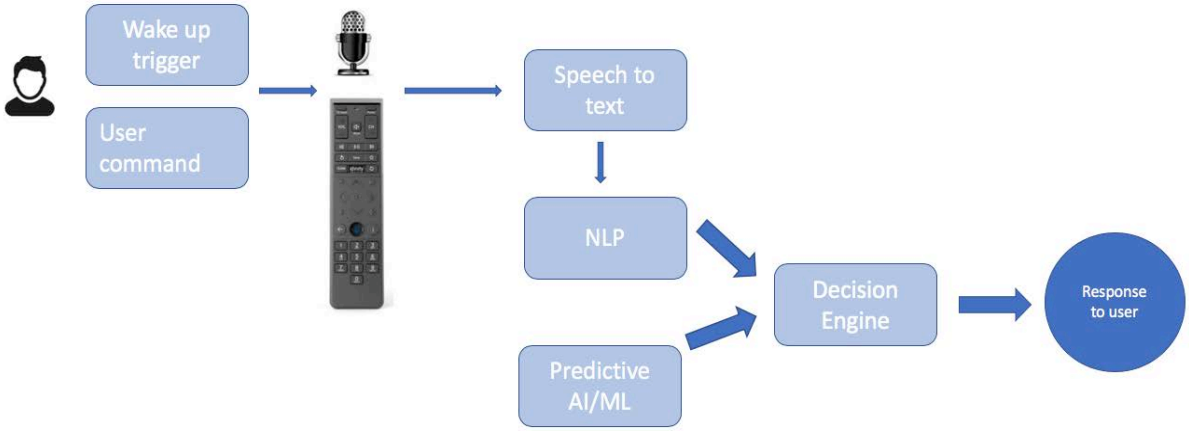

Figure 10 - Sample flow in a Voice command system

Customer Experience: AI/ML applications are playing an important role in improving the customer experience. Predictive AI is an example. With it, the system predicts an issue based on the data points, or can be applied to customer interfaces for quick issue resolution, using chatbots or self-healing techniques. The CPE (Customer Premises Equipment) is home to a vast set of (anonymized) telemetry and troubleshooting data that can be used as an indicator of network health and system status. These data points are used to build models, and when these models are applied to live data, the system can proactively predict issues like outages. These models are simpler than the models discussed earlier, as the data sets are usually available within the premises. These models are a good fit for compute edge use cases, because the system can analyze data and provide recommendations to the user or technician visiting the premises.

Conclusion

If the fourth industrial revolution does indeed turn out to be spawned by the swift and productive rise of AI (Artificial Intelligence) and ML (Machine Learning) technologies, both of which are vital in building "Intelligent Machines," then those same intelligent machines introduce a new "edge" to the network: The *compute* edge.

In this paper, we looked at the metrics of such compute edge platforms, the efficiency of the platforms and how newer hardware is emerging with higher performance and efficiency. We examined relevant use cases where the compute edge can improve response times and improve the sense of privacy. Also, a compute edge system can augment and complement existing cloud-based systems with more of a near-field analysis.

The edge platforms that are discussed here are not envisioned as replacing the cloud-based systems, but rather to enhance the efficiency and to better distribute the processing responsibilities. One major advantage is to be able to make minimal use of a system, when there is an outage or an intentional

sabotage. The system could provide the first level of AI capabilities and could leverage the cloud systems for further detailed analysis.

While we are excited about the compute edge platforms, we have to note that cloud-based systems are comparatively easy to maintain, because they enjoy a one-to-many relationship. It would add complexity in the system to maintain various compute edge platforms. Such challenges could be mitigated, to an extent, by using the same model structure. Suffice it to say there is still a lot of ground to cover, and such systems would need to be vetted.

Abbreviations

AI/ML	Artificial Intelligence/Machine Learning
bps	bits per second
CPE	Customer Premises Equipment
CPU	Central Processing Unit
DOCSIS	Data Over Cable Service Interface Specification
Hz	hertz
IoT	Internet of Things
ISBE	International Society of Broadband Experts
NLP	Natural Language Processing
RADAR	Radio Detectino and Ranging
SCTE	Society of Cable Telecommunications Engineers
STB	Set Top Box
TFLOPS	Tera floating point operations per second