This article has been accepted for inclusion in a future issue of this magazine. Content is final as presented, with the exception of pagination.

Invited Article
Challenges and Opportunities for Data Science and Machine Learning in IoT Systems – A Timely Debate: Part 1

Sumi Helal, Flavia C. Delicato, Cintia B. Margi, Satyajayant Misra, and Markus Endler

Abstract

This position paper summarizes the main visions, opinions, and arguments of four experienced and well known researchers in the area of Internet of Things (IoT) and its relation to Data Science and Machine Learning (ML) as IoT permeates the globe and becomes "very large". These visions were raised in an enthusiastic discussion panel held during the Third International Workshop on Very Large Internet of Things Systems (VLIoT 2019), in conjunction with VLDB 2019, in Los Angeles, USA. Each panelist delivered a vision statement before the floor was opened for questions and comments from the audience. Instead of reproducing ipsis literis each of the speeches, questions and replies, we decided to structure a two-part paper summarizing in-depth the panel opinions and discussions. In this first installment, we present the panelists' opening statements and views on issues related to IoT infrastructure and how it can support the growing demands for integrated intelligence, including communication, coordination and distribution challenges and how such challenges can be faced in the new generation of IoT systems.

Introduction

Markus Endler moderated the panel which he kicked off by introducing the panelists: Flavia, Sumi, Cintia and Satyajayant ("Jay") (see Fig. 1 and the biographies at end of article), and by asking them to present, as opening statements, their visions and opinions regarding the following general question:

> *If the IoT infrastructure is the nervous system of a cyber-physical system, Data Science is the knowledge construction, and Machine Learning is the brain, how can we be sure that we are collecting and processing all bits of information to build really smart, adaptive and human-friendly systems?*

The question was designed to address the complex interplay between three emerging and interrelated fields. Understanding and optimizing the interplay and knowing what is working and what is missing should provide urgently needed guidance into the research directions of such "glue" that can create a smart IoT, cyber-physical systems and data science cohesion.

After each of the panelist responded to this general question in their opening statements, Markus directed a series of questions to the panelists on infrastructure issues in such an ultra large-scale distributed IoT systems. He introduced his questions by stating that before we can take advantage and leverage the synergy between IoT, ML and Data Science, we need to provide solutions for scalable communication, coordination, and distribution, typically addressed by middleware systems. He noted that although these issues are already extensively investigated in the context of traditional distributed systems, they take on a new and more complex dimension when reaching the envisaged scale for IoT. Responses to questions stirred an interesting discussion by the panelists, which we summarize in this installment after presenting the opening statements.

In the next installment, immediately following in this issue, we will present the panel's views on issues of security, information and event processing, and the needed business models that are anticipated to emerge to affect a value chain from the connected smart devices and the data generated by them. We hope you enjoy the panel coverage in both installments and welcome your comments and email questions to the panelists.

Digital Object Identifier: 10.1109/IOTM.0011.2000002

FIGURE 1. Animated discussion among the panelists Delicato, Helal, Margi and Misra (left to right).

Opening Statements

Flavia: The Internet of Things (IoT) aims to leverage Internet technology to the next level, by connecting an unprecedented number of devices, generating a swarm of heterogeneous sensors and actuators that can interact with the physical environment, collect several types of variables, and support dynamic decision-making processes across multiple application domains. It is envisioned that this myriad of connected devices is "smart", continually learning from behavioral patterns of humans and other devices, and then autonomously adapting to changes at runtime. Such ability is based on the implicit assumption that IoT systems can make real-time decisions about data, usually on the move. Ultimately, the great potential of IoT is not about getting data, but about extracting valuable knowledge from that data. In this context, data science can make a great contribution to make IoT systems more intelligent. Data science is the combination of different fields of sciences that uses data mining, machine learning (ML) and other techniques to find patterns and new insights from data. These techniques include a broad range of algorithms applicable in different domains. However, the intrinsic features of IoT, such as the high heterogeneity, velocity, volume, dynamism, volatility and uncertainty of the sensor generated data, makes applying ML techniques very challenging. The IoT requires a new generation of distributed algorithms based on lightweight, online and incremental learning. On the other hand, ML techniques require large

Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on January 12,2021 at 18:12:38 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this magazine. Content is final as presented, with the exception of pagination.

computational capabilities such as on powerful servers, and thus are not suitable for execution on small and resource constrained IoT devices. Thus, IoT should leverage the capabilities of resources at the edge of the network (edge nodes) and the collaboration among nodes. Horizontal collaboration among edge nodes should be promoted with the goal of sharing data, information and inferred knowledge about the environment. Moreover, the logical topology among the processing nodes in the IoT-edge system should be dynamic so as to facilitate the data flow between the data analyzing components, according to the Machine Learning workflow. This would be a data-driven topology. However, before one can build a value chain for knowledge generation, the starting points are the most varied, and often tiny, objects empowered by IoT's technological capabilities: it all starts with the things.

"It seems we have not focused adequately on the architectural aspects of things in our pervasive computing journey. To get there, we will have to walk before we run, that is, we have to realize thing before we are able to realize the Internet of Things."

Sumi: While we all share the great IoT vision and foretell of impressive IoT scenarios and possibilities enthusiastically, we do not yet have a clear pathway to realizing this vision at a wide scale. In fact, it can be argued that the focus on vision and abstracting away many details, including about "things" themselves, was intentional to productively bolster our imagination, but this approach has now run its course. Ignoring the details and staying abstract will be counterproductive at this stage. Indeed, the success of the IoT will largely depend on how its main ingredient, the thing, is architected and prepared to match the high expectations and to fulfill the big heroic role that will magically make blue-sky visions a reality. Unfortunately, it seems we have not focused adequately on the architectural aspects of things in our pervasive computing journey [1]. To get there, we will have to walk before we run, that is, we have to realize thing before we are able to realize the Internet of Things. Just because thing can communicate does not mean thing is smart or that thing is ready to realize IoT. What seems to be needed is an explicit thing architecture that captures requisite requirements drawn from the blue-sky thinking. My research team has been trying a few ideas [2, 3], but more concentrated and coordinated efforts are needed. If we somehow find these requirements and if we manage to create such architecture, we may succeed in creating proper things for the IoT, things of high IoT utility (high IoTility as I attempted to define it in [4]) in terms of the multitude of scenarios made possible and programmable. Minimal hardware and networking to enable reasoning and interactions will be one basic requirement. But this will not be enough or we end up with an Mbed-OS-like architecture [5], which is a good and efficient architecture but falls short of elements that could enable blue-sky thinking and aspiration. We may need to explore the ability to chat, ability to socialize, ability to establish meaningful calculated interactions, ability to self-API, and even the ability of a thing to create apps or parts of apps. My view is IoT is not just a massive data generating infrastructure, but also an intelligent active infrastructure that consumes its data in place through reasoning and embedded intelligence. Such infrastructure will be the brick and mortar of our smart living spaces and future smart cities. We will have it both ways, as a data and knowledge generator, and as an intelligent and active environment if we succeed to architect things properly for both purposes. This would necessitate architectural elements that render IoT to be a programmable machine, which will accelerate a shift in thinking from IoT concepts to IoT apps.

Jay: The long term vision of IoT is that of ubiquitous things, sensing their environment using different modes and sending information to a brain (either distributed or centralized), which helps make intelligent decisions using ML or data mining techniques. This creates two challenges: a) How do we transport all relevant bits of information to the brain, while meeting the needs of the application in terms of latency, bandwidth, and data reliability? b) How do we ensure that the information is accurate, every piece of information has provenance, and the data transmission, storage, and utilization meets the security and privacy needs of the applications and end-users. Answering these two questions satisfactorily is absolutely essential for the success and application of VLIoT applications, such as smart cities, autonomous driving, and the smart grid. The Internet will be the conduit for the massive upload of data from the forecast billions of IoT devices at the Internet's edge. However, it was built (is still being upgraded with the same principle) for massive data downloads from the servers to mostly-passive clients. We have fat pipes at the core of the network that push data to the edges (with content delivery networks, content are close to the core). However, the reverse is not true. We do not have fat pipes going from the Internet's edge to the core, essential for moving the massive amounts of data generated by the billions of devices that will make up the VLIoT constituents. This has called for a rethink of the design of the Internet's network architecture to address the challenges of VLIoT, which resulted in efforts around the world including the U.S. National Science Foundation's Future Internet Architecture (FIA) [6] effort and the European Union and Japan ICN2020 project [7], which started thinking of a new design paradigm, information-centric networking (ICN), for the Internet. The idea of ICN is to leverage data names at the network layer of the TCP/IP stack instead of IP addresses for transferring data in the network, which enables requesting for content using names of locations, events, etc., and also provides the intermediate nodes forwarding nodes to use the names to provide specialized service to different data packets. Further, ICN makes signature of the data by the source/originator of the data mandatory. This provides provenance for the data. To handle the data volumes emanating from the edge there have also been proposals such as edge computing, i.e., computing on the data near the source of its generation to draw inferences and also to substitute the transmission of the data with the transmission of the model or the insights. Machine learning will serve as an important enabler in performing this volumetric reduction. For instance, federated learning will help learning to be performed at the edge or fog computing servers with the insights/partial models being transmitted to the brain, which in turn does the aggregation of the partial models into a holistic global model, which can be sent back to the local edge servers. These enablers will help move the data bits (or the model created with the learning of the bits) from all the end devices to the brain of the IoT network.

Cintia: IoT is a heterogeneous environment, composed of several different devices (some with very constrained capacity in terms of processing and memory), with different types of sensors (both in terms of what they sense and their accuracy) and communication technologies (different data rates and coverage, to say the least). To some extent, this is very similar to the Wireless Sensor Networks (WSN) environment. Research in WSN has been ongoing for over 20 years, and while the research community has made significant contributions and supported the development of several standards, WSN did not achieve the much awaited impact as was expected in terms of application. So what makes IoT different from WSN? IoT is related to several different applications leveraging from the sensed data, achieved by instrumenting the real-world objects, so it all starts with the things. Looking back at WSN, most deployments targeted a specific application, and devices and protocol selections were driven by that, so infrastructure was not shared among different applications. Therefore, IoT can leverage from WSN, but we

Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on January 12,2021 at 18:12:38 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this magazine. Content is final as presented, with the exception of pagination.

must look into ways of turning this WSN infrastructure shared among different applications. Concerning the IoT applications, what are the requirements concerning: (i) sensing/data collection, (ii) local and in-network processing, and (iii) communications, such as delay and packet delivery rate (or acceptable packet loss rate)? These different requirements impact the device selection and infrastructure construction, concerning communication protocols and node placement. Software-Defined Networking (SDN) has been presented as an approach that can benefit WSN and IoT, since it brings flexibility in the network configuration, and enables improved management, resource sharing and reuse (i.e., sensor nodes and communication infrastructure) [8]. In order to evaluate this approach, we designed and developed a Software-Defined WSN framework: IT-SDN [9]. The design requirements included: resource constrained devices, IEEE 802.15.4 as MAC layer, and in-band control. The framework is composed by: southbound, neighbor and controller discovery protocols; northbound API; and network monitoring features. Experimental evaluation results, considering several different scenarios and network sizes, indicate that SDN is feasible for WSN, presenting a competitive data delivery ratio while saving energy in comparison to RPL, the Routing Protocol for low-power and lossy networks [10]. In summary, I consider that the SDN approach will enable an IoT communication infrastructure able to meet requirements for data exchange with the brain of the IoT network (what was already very well explained by my colleagues), to support different data collection applications and communication patterns, and to enable in-network processing. Furthermore, the SDN central points could be used to facilitate applications, such as to apply machine learning algorithms to improve infrastructure, as well as to improve applications.

"The traditional Internet has only become what it is, a global network for interconnecting computers, thanks to the TCP/IP stack. In contrast, there is not yet a standard protocol stack for IoT, although some protocols such as IEEE 802.15.4 are emerging as a trend."

Communication Issues

Markus: IoT is ultimately about interaction among smart devices, and of smart devices with back-end services in the cloud. Since all this involves massive communication over wireless and wired links, how does it affect the QoS of the exchange of data, information and shared knowledge?

Flavia: Certainly, communication is at the bottom of the pyramid to generate knowledge from distributed sources like IoT. There are two important aspects regarding communication in the IoT. The first aspect concerns heterogeneity. The second aspect concerns the question that Markus is bringing up: QoS provisioning. Naturally, since we are dealing with the Internet of everything, there is of course a huge variety of devices, of the respective generated data and of the communication protocols adopted. The traditional Internet has only become what it is, a global network for interconnecting computers, thanks to the TCP/IP stack. In contrast, there is not yet a standard protocol stack for IoT, although some protocols such as IEEE 802.15.4 are emerging as a trend. Therefore, for IoT to reach its full potential as the basis for building advanced information exchange applications, one possible way is to invest in standardization efforts. There are a number of candidate protocols already in use, and standardization bodies should continue to work to create certifications and compliance rules to converge toward the adoption of standards that favor interoperability [11]. However, there is an important difference here between the traditional Internet and IoT regarding application-specific characteristics. IoT heterogeneity also encompasses applications, and while the Internet was built as an application agnostic network, or to meet the needs of simple applications for document exchange, IoT has already been designed to meet the requirements of intelligent applications, with different QoS requirements. The Internet was created as a best-effort network where applications with QoS requirements were not the target. In the IoT, the network infrastructure must contribute to meeting different application requirements. For some of them, low latency is critical, and some protocols best contribute to meet this requirement. For others, high throughput is the target requirement, while for still others the data accuracy is essential, and this is usually achieved over a low latency, favored by a different type of protocol. Therefore, it is difficult to find a one-fits-all solution for the communication protocols in IoT. Instead, an alternative way to standardization is to exploit edge devices to translate between different data formats and protocols. While acting as bridges to address the heterogeneity of communication protocols, such devices will be an integral part of the IoT system's intelligence generation process. Regarding network QoS provision, one aspect that has been explored recently concerns the virtualization of network functions and the vision of software-defined networks.

Markus: In fact, there is some divergence between researchers regarding converging toward the adoption of global/universal standards for IoT protocols and middleware, and the vision of relying on multi-protocol gateways for a conversion and translation between different protocols, each of which is best suited to meet the specific requirements of the applications and the autonomic control of its IoT devices. What are your opinions on this, Cintia and Jay?

Cintia: 6LowPAN, RPL and CoAP [12] are examples of the efforts taken by IETF working groups to incorporate low power and lossy networks to the Internet. CoAP and 6LowPAN are important standardization efforts to support end-to-end application communication from the IoT devices to the cloud. On the other hand, RPL addresses routing in the constrained devices networks, but do not consider QoS requirements. Software-Defined WSN approaches could leverage from the controller view of the network and use information provided by application managers to use different objective functions (i.e., routing metrics) to select the network paths for each different application. For instance, if delay is a requirement, the controller could use information from the network monitoring module to determine the route with the least accumulated delay. On the other hand, if data loss is an issue, the controller could select a route with minimum data packet loss, even if the delay is larger. Furthermore, the centralized view the controller has and updated information from the network status could be used to determine if incoming applications with hard requirements would be able to run on the infrastructure.

Jay: VLIoT will consist of diverse devices, using heterogeneous wireless communication technologies, such as WiFi, LiFi, and Bluetooth Low Energy. In addition, new protocols, such as CoAP, 6LoWPAN, RPL, Message Queue Telemetry Transport Protocol (MQTT), and Advanced Message Queuing Protocol (AMQP) will be in use in the devices. This will make effective communication between the devices a challenge, particularly when the devices are autonomous, independent and need to coordinate. This will call for the capability of the IoT devices to perform translation across multiple communication technologies as well as different protocols. There has been work on multi-protocol gateways in IoT [13, 14], but they have been initial and more needs to be done. Further, the sheer number of devices potentially communicating even on diverse bands, namely 300 MHz, 2.5 GHZ, 5 GHz, and 60 GHz, the number of devices and their differently capable MIMO (multiple-input multiple output) antennas will generate such large amounts

Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on January 12,2021 at 18:12:38 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this magazine. Content is final as presented, with the exception of pagination.

of data that it will require scheduling of the devices to be able to use the scarce spectrum, which will make the problem challenging. In addition, differentiated QoS for different applications will become a necessity, particularly with the widening of the adoption of 5G, which will support low-latency, high bandwidth applications with potential device-to-device communications to achieve the results.

Markus: Now Sumi, I am just curious about your opinion about this.

Sumi: Well, Markus, let me first say that the IoT communication issue is a very important one because we are talking here about a highly fragmented market of devices in the shaping, and multiple ecosystems and proposed standards. How can IoT achieve friction-free inter-thing interactions despite such fragmentation? I agree with Flavia and other panelists that this is a big challenge. On the one hand, we cannot discourage or fully convict closed ecosystems (and IoT Platforms) such as Samsung's SmartThing, Microsoft's Azure IoT, Apple's Home Kit, and Amazon's AWS IoT, as they drive innovation and build the market. Perhaps learning from the past and applying heterogeneous or federated database system integration concepts may be helpful given the fragmentation. In a recent work [15], we introduced an interoperable communication framework for bridging RESTful and topic-based communication in IoT. The framework can be implemented as a cloud or edge service, or even as dedicated IoT things embedded in a smart space to achieve such translation and interoperability. Another major issue in IoT communication arises in large-scale IoT deployments (e.g., smart cities) in which the IoT things must interact with cloud hosted and provisioned IoT applications or services. As the smart city grows, demand on cloud services spikes and the dimensionality of the cloud becomes cost prohibitive. Also, as more smart city applications are added, energy concerns arise if access to the IoT things by a large number of applications is uncoordinated. To solve this problem, we need to use the edge creatively. In fact, I would go as far as to suggest that opposite to the original cloudlet approach in which an edge is utilized to bring the cloud and its benefits closer to the applications (often mobile apps), in cloud-and edge-connected IoT systems where the applications are deployed and run in the cloud, we should exploit the edge differently, either by bringing the IoT's physical world and its data up closer to the edge or even the cloud and its applications and services, or by caching parts of the various applications down closer to the physical world, maybe at the edge or even beneath at the IoT itself. We have attempted to lay down a theoretical foundation for such inverted use of the edge in [16] but much more needs to be done to explore this concept.

"As the smart city grows, demand on cloud services spikes and the dimensionality of the cloud becomes cost prohibitive. Also, as more smart city applications are added, energy concerns arise if access to the IoT things by a large number of applications is uncoordinated."

Coordination Issues

Markus: One of the uses of IoT is also for monitoring and automating processes in the physical world, such as in Smart Buildings, Industry 4.0, precision agriculture or healthcare. In several such cases of complex automation, the IoT smart device's functions and actions must be coordinated, both in time and space and in the actions, as for example for swarms of robots/UAVs [17]. But to me coordination appears as a very hard problem due to the potentially variable/dynamic set of interacting devices, and due to the heterogeneous capacity, provided QoS and fault-resilience of the IoT devices and (wireless) communication links, as well as the high latency to/from the backstage cloud services. So, how do you see the means to ensure correct coordination among the IoT devices in such a dynamic and heterogeneous environment?

Flavia: In this regard, once the heterogeneity issue has been overcome, the great challenge is dealing with the massive scale of IoT. Centralization is often an easier option for coordination than distribution, but again the scale factor makes centralization not a sustainable solution. Therefore, one possible way is to adopt hybrid solutions. Coordination between parties involved in data processing to generate intelligence in the IoT systems could be hierarchical, with the global view (provided by a node in the cloud) being adopted only when strictly necessary, for example to optimize some process, while localized decisions would be the most common practice for coordination. A promising option is to adopt a hierarchical edge node topology and clustering IoT (end) devices using distance-based strategies to associate them with the edge nodes that would be responsible for their coordination. An example of such an approach is proposed in [18]. Each edge node would coordinate its subordinate nodes but would eventually resort to the cloud to update models or adjust parameters from the global view of the system. At the same time, groups edge node could collaborate horizontally, in a peer to peer fashion to share data and tasks, as proposed in [19]. In this view, coordination is fully distributed at the edge tier, but again the cloud can be triggered when needed. To sum up, IoT coordination should be distributed, possibly hierarchical, adaptive and context-aware.

Cintia: Standards will enable the communication between heterogeneous devices, but the massive amount of devices and data creates challenges for coordination. I believe global coordination among IoT devices is not feasible either with distributed or centralized approaches, so a hybrid and hierarchical approach should be used. Different application domains will require different types of coordination as well. Thus, local coordination should be used to address such requirements. Given local coordination concerns a smaller amount of devices, it could take advantage of a centralized view (for instance using a Software-Defined approach). On the other hand, a global general coordination could provide requirements and information to support the local coordination.

Sumi: Markus again asks the right and important question here. Indeed, coordination in an IoT is of paramount importance. Interference and conflicting operations in a smart space could create faults and even hazards. Compounded with promises to deliver intimate and convenient services surrounding our daily lives, the IoT vision poses imminent concerns and raises additional requirements for safety. As sensors and actuators extend the capability of computer systems into effecting the physical realm, false logic and erroneous executions of IoT-based pervasive systems also implicate not only data loss or software crash, but also real dangers and physical harms. To manage safety in an open and constantly evolving smart space, a deliberate and systematic approach is required to accurately model the cyber-physical and inter-IoT interactions, to effectively manage and regulate these interactions, prevent conflicts and interferences, and enforce safety constraints at run time. My team is currently working on IoT Transactions (IoTXN) as a means to limit access to the smart space where all interactions must be through IoTXNs. We borrow from serializability theory but also see the need for new-look concepts, protocols and algorithms [20, 21]. We certainly need to look at safety-oriented programming models and language constructs for IoT. For instance, exception handling needs to be redone within an IoT.

Jay: In light of 5G coming to life and a large proportion of the devices needing to communicate with other local devices to enable the ever increasing applications at the edge, e.g.,

Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on January 12,2021 at 18:12:38 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this magazine. Content is final as presented, with the exception of pagination.

augmented reality, online games, autonomous driving, coordination will become a necessity. The applications and hence the supporting network will have to be designed in a way that local coordination is feasible, inputs from the cloud are utilized efficiently, and mobility and energy projections of the devices are used to identify the best means of coordination. Due to the highly dynamic nature of the nodes in the network, the hierarchical nature seems to be the best way to connect nodes into a hierarchical cluster for coordination. However, the challenge will be the automatic, dynamic reconfiguration of the network topology to meet the application needs. This also leads to the idea of concentrating/aggregating the data at the different cluster heads, who then send their data to their cluster head and so on, so that the data can migrate up the hierarchy as needed.

"Only recently data streams are increasingly being considered as learning objects. In IoT, in contrast, most data sources take the form of distributed streams and learning must often take place in an online fashion."

Distributed (Cloud-Edge) Computing

Markus: Now it is almost taken for granted that for an IoT system to scale in the number of supported smart things, in the required communication bandwidth, and providing the data processing and storage functions required by the applications, such a system needs to employ different kinds of mutually dependent processing functions distributed over several nodes of the IoT infrastructure at varying "distances" from the cloud-based services, i.e., at the Edge and Fog devices, such as in ContextNet [22] and many other IoT middleware systems. But while coordinated distributed computing on homogeneous network nodes is already quite complex, such distributed Cloud-Edge Computing is even more so because of the very different processing, storage and (wireless) communication capacities, as well as the very different availability and reliability profiles of these classes of machines. How do you think we should approach this distributed computing complexity?

Flavia: In my opinion, the distribution challenges in IoT must be analyzed at least from three dimensions: (i) the data itself (distributed sources); (ii) the data processing; and (iii) the management of the resources required for such processing [23]. Let's focus our discussion on the data processing point of view, in particular, what is needed for the ML lifecycle. Data processing in ML varies depending on the specific technique, but it generally takes place over a well-defined lifecycle. At the core of any ML technique is the model building, but the whole lifecycle involves several steps, from acquiring and preparing data to deploying models and putting them into production. Data acquisition in ML traditionally deals with data sources of different formats, but in IoT the degree of heterogeneity and distribution is higher. In addition, many ML techniques assume the use of file stores and data tables as the primary source of data, with learning generally occurring in persistent data at rest, with heavy use of historical data. Only recently data streams are increasingly being considered as learning objects. In IoT, in contrast, most data sources take the form of distributed streams and learning must often take place in an online fashion. Also, storing all data is not always required or possible in the constrained IoT devices. This makes it necessary, on the one hand, to delegate storage to computational nodes other than those that produced the data, and on the other hand, to decide when and which data should be stored. Once the data is acquired, there are preparation and preprocessing steps before such data is considered fit to participate in the training and model building. Data preparation and cleaning are typical examples of activities that should occur as close to the data sources as possible, either on the generating device itself or on edge nodes. Because of their proximity to the data sources and their greater resource capacity (compared to IoT devices), edge nodes are in a prime position to perform data cleaning and filtering tasks. However, depending on the amount and rate of generated data, their preprocessing may become infeasible in a single node, requiring the decentralization of this step, that can be done collaboratively among multiple edge nodes. Recent work has proposed solutions for distributed data cleaning in the context of wireless sensor networks and Big Data [24] [25] [26].

Markus: Distributed data cleaning sounds cool, but what about data analysis? Can, and should, it also be done by the edge devices? What do you think, Flavia?

Flavia: Once the data is clear and prepared, we move on to the analysis phase, whose aim is building an ML model to analyze the data using various techniques and review the outcome, evaluating the model. The initially created model is then trained to improve its performance, often measured by its accuracy during a testing phase. Once the performance is considered suitable, the model can be put into production and executed to make predictions, inferences, and other learning outcomes. The training phase requires optimizing hundreds of millions of parameters and demands a lot of processing power. Therefore, most model training typically takes place in the cloud, often in a centralized server. However, as data size increases, it becomes hard for a single server to solve large-scale ML problems. To address the issue, distributed machine learning began to be adopted early in the last decade, where a typical ML task is accomplished through the cooperation of multiple servers. Several theoretical approaches in this context have emerged [27], with the goal of making parallel originally centralized learning algorithms, and focusing on collaboration between machines, but generally assuming homogeneous and powerful computers. When applying traditional distributed ML in the IoT context, the data processing is fully performed on cloud servers. Therefore, from the point of view of our current discussion, it is still considered a fully centralized approach. The growing presence of powerful mobile devices generating massive amounts of data is contributing to change this picture. Researchers are investigating ways to decentralize the training and execution steps of ML models and to perform the entire ML life cycle inside IoT devices [27]. The idea of performing ML on mobile devices [28] has emerged mainly motivated by the proliferation of advanced applications (e.g., face recognition) that demand such techniques. In the initial approaches, an ML model was first trained on servers (in the cloud) using huge datasets and then sent to IoT devices, where inference and predictions could be made locally. This scheme concentrates the entire workload of model training on cloud servers and can be considered a partially decentralized approach. The inference process performed locally only returns a prediction result which may be used to produce a service for the user (device owner), thus wasting resources (not so negligible) available on IoT devices and incurring the traditional issues of bandwidth-intensive and high latency. However, if part of the training can be done locally on IoT devices, several benefits can be obtained. The generated trained model can become personalized, and the improved model can be used immediately, thus providing a better Quality-of-Experience to the user. By periodically updating the locally trained model to the server, location awareness can be exploited to provide valuable information for the global model. Local resources are exploited, and the bandwidth consumption is decreased. This is exactly the idea of mobile distributed machine learning [29], which was motivated by the need to make the best use of user data generated on mobile devices while protecting users' privacy. Distributed ML on mobile devices separates the learning task into sub-problems. The mobile devices solve sub-problems according to local

Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on January 12,2021 at 18:12:38 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this magazine. Content is final as presented, with the exception of pagination.

user data, eventually uploading their results to a centralized server that finally aggregates all intermediate results into a global model. Eventually, updated parameters are sent back to mobile nodes for a next round of iteration. All these stages pose their own challenges, such as decomposing the learning task into sub-tasks, compressing data for upload to the aggregator, and selecting the best frequency of uploading, balancing communication cost with model accuracy. Initial approaches for mobile distributed ML only consider end devices as performing local training. However, as the edge/fog paradigm evolved, approaches integrating MEC with mobile ML emerged, in which more powerful edge nodes perform the local training [30, 31]. In 2015, researchers from Google proposed the federated learning approach, which shares the same principles of mobile learning. For example, the authors in [32] propose an algorithm to determine the frequency of global aggregation so that the available resources are most efficiently used. Mobile distributed and federated learning are examples of fully decentralized approaches, which make the most of the high degree of distributed resources in the current IoT-edge systems while trying to cope with the inherent challenges. From 2016 to 2018 Google published several related articles [33, 34] to complement federated learning's framework.

"Ideally, we should not insist on a single standard (e.g., MQTT or other topic-based communication such as CoAP), but live with multiple strong ones. It should be possible to translate back and forth among a finite set of well-developed standards."

Markus: Going back to the original question, Sumi, what is your vision for dealing with the complexity of distribution in highly heterogeneous and dynamic IoT-Edge-Cloud systems?

Sumi: This is indeed crucial. Not only do we need to solve the heterogeneity problem, but we need to do so under constraints or the distributed system spanning the cloud, the edge and the fog may not be sustainable as the scale of the IoT grows horizontally (e.g., as IoT is deployed in and across cities and metroplexes) or vertically (as the number of applications and the application-IoT interaction demand sharply increases upon popularity). A first step is to concentrate our efforts in standardizing communication in two dimensions. In the first, peer communication (thing-to-thing, edge-to-edge, and cloud-to-cloud) should be enabled. In the second, inter-layer communication must also be enabled. The goal is to enable communication between elements belonging to different vendors. Ideally, we should not insist on a single standard (e.g., MQTT or other topic-based communication such as CoAP), but live with multiple strong ones. It should be possible to translate back and forth among a finite set of well-developed standards. Once communication is enabled, executing IoT apps and the entailed coordination and decision-making of what needs to be done, by whom, and where, will be our next frontier to tackle as researchers. But the classical distributed system question that emerged in the early 1980s as to whether we should move data to computation or move the computation to the data (which sparked mobile agent and DARPA's active networks research programs) becomes sharply present and relevant in the context of coordinating the run-time execution of multi-tiered cloud-edge-fog-thing (CEFT) architecture. I believe that once we standardize communication, we should work on online optimization that dynamically re-configures the CEFT architecture to meet all constraints, including energy savings (operational life span) of battery-powered things, and the limit in budgeted cost a jurisdiction is willing to pay cloud providers, and as of recently, also edge providers such as Telcos. The only thing we may all agree on here without much technical details is to minimize movements of all sort in the CEFT (queries coming down from the cloud applications and services, and data streaming up from the IoT and its things). We will need to overcome the barriers we created ourselves in this optimization problem where we focus so much on doing all work energy-efficiently. Now, we need to ask ourselves the hard question: Are we optimal in deciding which work needs to be done? Are we overworking unnecessarily (be it overworking energy-efficiently)? In other words, should not we try to first make sure we are "sentience-efficient" before we pursue energy efficiency? If we are able to break through and achieve unprecedented savings in movement and work, we may be able to establish the framework necessary for this type of distributed systems. I think this question by Markus is very important and evokes the need for a highly coordinated effort by the IoT and Distributed Systems' research communities.

Jay: I believe the information-centric networking (ICN) paradigm will be a very good networking architecture solution for this multi-level network structure. There are many challenges that need to be addressed, including how does a client node discover the available fog/edge/cloud resources? Once discovered, which edge node should provide the service needed by the client node? Whichever entity chooses the particular edge node to be used, how does it make that decision and with what kind of statistics? With ICN (particularly with the NDN architecture), the network is capable of providing the client information to answer its service query. In addition, each node in the network can have routes to every edge/fog/cloud node. This will allow the nodes in the network to send the service query to one or all the nodes providing a particular service. The service nodes may broadcast the utilization information in the network as well, which the routers can use to direct the requests in the network. In NDN (most ICN architectures), the data response as well the data request (if needed) can be cached in the network for future use. Additionally, due to the use of names, there is also the capability to reuse the computation results at the routers or at the edge by using simple longest-prefix based name matching operations using the names of the requests and the data [35]. ICN coupled with SDN and NFV technologies will be able to tackle the complexity inherent in the distributed computing scenario.

Cintia: Similar to the coordination issue, approaches that combine local and global computing should be employed. Thus, fog, edge and cloud computing would be part of this distributed computing hierarchy, and several questions will follow: (i) How to determine which data to process on each level of the computing hierarchy? (ii) Is this decision related to the critical level of the application that needs the data? (iii) Should raw data only be transmitted to the closest fog, or should raw data be sent to both to the closest fog and the next level, or to all the levels of the computing hierarchy? (iv) By sending data to multiple computing points, is this data transmitted multiple times over the IoT? To answer all these questions, which will determine how data should be routed, I think it is important to know both the network state (concerning resources and usage), as well as the application requirements. Therefore, the SDN approach could be used to support this, since the SDN controller has information about the network and could obtain information about the applications from its northbound API [36].

Conclusion

This article is the first part of a two-part article that brainstorms and debates the future of IoT architectures as well as IoT's fast-growing entanglement with the fields of Data Science and Machine Learning. We hope we were successful in coherently capturing and communicating the panelists' visions, views and opinions. Part 2, which immediatelely follows in this issue, will address the additional issues of security, information and event processing, and emerging business models. Taken together, we

Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on January 12,2021 at 18:12:38 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this magazine. Content is final as presented, with the exception of pagination.

hope both parts offer useful thoughts that can help shape an informed agenda for future R&D in these bordering areas of IoT, Machine Learning and Data Science.

Acknowledgments

We would like to thank Prof Sven Groppe, VLIoT Workshop Chair for facilitating the panel. Flavia Delicato is supported by CNPq grant number 306747/2018.9 and by FAPESP (Sao Paulo Research Foundation) grant number 2015/24144-7; Cintia Margi is supported by the ELIOT project, FAPESP (Sao Paulo Research Foundation) grant #2018/12579-7; Markus Endler is supported by CNPq grant 433183/2018-7 and INCT of the Future Internet for Smart Cities funded by CNPq, proc. 465446/2014-0, CAPES Finance Code 001, and FAPESP, proc. 2014/50937-1 and 2015/24485-9; Satyajayant "Jay" Misra is supported by U.S. NSF grant awards #1800088, #1719342, #1345232, EPSCoR Cooperative agreement OIA-1757207, and Intel grant #34627535. The opinions presented here are those of the author Misra and not the official position of the federal government and Intel Corporation.

References

[1] A. Helal, A. Khaled, and W. Lindquist, "The Importance of Being Thing, or the Trivial Role of Powering Serious IoT Scenarios," in *Proc. 39th IEEE International Conference on Distributed Computing Systems (ICDCS)*, IEEE, July 2019.

[2] A. Khaled *et al.*, "IoT-DDLDevice Description Language for the T in IoT," *IEEE Access*, vol. 6, 2018, pp. 24048–63.

[3] A. Khaled, W. Lindquist, and A. Helal, "Service-Relationship Programming Framework for the Social IoT," *Open Journal of Internet of Things (OJIOT)*, vol. 4, Aug. 2018, pp. 24048–63. Presented at the Very Large Internet of Things (VL-IoT) Workshop, in conjunction with the VLDB conference.

[4] W. Lindquist *et al.*, "IoTility: Architectural Requirements for Enabling Health IoT Ecosystems," *IEEE Trans. Emerging Topics in Computing*, Special issue on New Frontiers in Computing for Next-Generation Healthcare Systems (accepted), June 2019.

[5] ARM Mbed Operating System Architecture, https://www.arm.com/products/iot/mbed-os, 2018 (accessed Nov. 19, 2019).

[6] T. Winter *et al.*, NSF Future Internet Architecture Project, 2011.

[7] T. Winter *et al.*, NSF Future Internet Architecture Project, ICN2020: Advancing ICN Towards Real-world Deployment through Research, Innovative Applications, and Global Scale Experimentation.

[8] H. I. Kobo, A. M. Abu-Mahfouz, and G. P. Hancke, "A Survey on Software-defined Wireless Sensor Networks: Challenges and Design Requirements," *IEEE Access*, vol. 5, 2017, pp. 1872–99.

[9] R. C. A. Alves *et al.*, "The Cost of Software-defining Things: A Scalability Study of Software-defined Sensor Networks," *IEEE Access*, vol. 7, 2019, pp. 115093–108.

[10] T. Winter *et al.*, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks. RFC 6550 (Proposed Standard), March 2012.

[11] J. Dizdarevic *et al.*, "A Survey of Communication Protocols for Internet of Things and Related Challenges of Fog and Cloud Computing Integration," *ACM Comput. Surv.*, vol. 51, no. 6, Jan. 2019.

[12] T. Salman and R. Jain, "A Survey of Protocols and Standards for Internet of Things, CoRR, abs/1903.11549, 2019.

[13] P. Desai, A. Sheth, and P. Anantharam, "Semantic Gateway as a Service Architecture for IoT Interoperability," in *2015 IEEE International Conference on Mobile Services*, IEEE, 2015, pp. 313–19.

[14] M. Uddin *et al.*, "SDN-based Multi-protocol Edge Switching for IoT Service Automation," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 12, 2018, pp. 2775–86.

[15] A. Khaled and A. Helal, "Interoperable Communication Framework for Bridging RESTful and Topic-based Communication in IoT," *J. on Future Generation Computer Systems*, 2018.

[16] Y. Xu *et al.*, "Energy Savings in Very Large Cloud-IoT Systems," *Open Journal of Internet of Things (OJIOT)*, vol. 5, no. 1, 2019, pp. 6–28. Presented as the Keynote of the Very Large Internet of Things (VL-IoT) Workshop, in conjunction with the VLDB, Los Angeles.

[17] B. de Souza and M. Endler, "Coordinating Movement within Swarms of UAVs through Mobile Networks," in *IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, March 2015, pp. 154–9.

[18] I. dos Santos *et al.*, "Data-Centric Resource Management in Edge-Cloud Systems for the IoT," *Open Journal of Internet of Things (OJIOT)*, vol. 5, no. 1, 2019, pp. 29–46.

[19] M. Alves, F. Delicato, and I. Santos, "LW-CoEdge: A Lightweight Virtualization Model and Collaboration Process for Edge Computing," World Wide Web, 2019.

[20] C. Chen and S. Helal, "System-wide Support for Safety in Pervasive Spaces," *J. Ambient Intelligence and Humanized Computing*, vol. 3, no. 2, 2011, pp. 113123, 06.

[21] C. Chen *et al.*, "IoTXN: Transactions for Safer Smart Spaces," under review, 2019.

[22] M. Endler and F. S. Silva, "Past, Present and Future of the ContextNet IoMT Middleware," *OJIOT*, vol. 4, no. 1, 2018, pp. 7–23.

[23] F. Delicato, P. Pires, and T. Batista, "Resource Management for Internet of Things," *Springer Briefs in Computer Science*, Springer Publishing Company, Inc., 1st edition, 2017.

[24] X. Deng *et al.*, "An Intelligent Outlier Detection Method with One Class Support Tucker Machine and Genetic Algorithm toward Big Sensor Data in Internet of Things," *IEEE Trans. Industrial Electronics*, pp. 1–1, 08, 2018.

[25] S. Park *et al.*, "Measurement Noise Recommendation for Efficient Kalman Filtering over a Large Amount of Sensor Data," *Sensors*, vol. 7, no. 19, 2019.

[26] M. Vazquez-Olguin *et al.*, "Object Tracking over Distributed WSNs with Consensus on Estimates and Missing Data," *IEEE Access*, 2019.

[27] D. Peteiro-Barral and B. Guijarro-Berdias, "A Survey of Methods for Distributed Machine Learning," *Progress in Artificial Intelligence*, vol. 2, no. 3, 2012.

[28] R. Gu, S. Yang, and F. Wu, "Distributed Machine Learning on Mobile Devices: A Survey," 2019.

[29] X. Zeng, K. Cao, and M. Zhang, "MobileDeepPill: A Small-Footprint Mobile Deep Learning System for Recognizing Unconstrained Pill Images," in *MobiSys17: Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*, 06 2017, pp. 56–67.

[30] X. Wang *et al.*, "In-edge AI: Intelligentizing Mobile Edge Computing, Caching and Communication by Federated Learning," *IEEE Network*, 2019, pp. 156–165.

[31] E. Li, Z. Zhou, and X. Chen, "Edge Intelligence: On-demand Deep Learning Model Co-inference with Device-edge Synergy," in *Proc. 2018 Workshop on Mobile Edge Communications*, 2018, pp. 3136.

[32] S. Wang *et al.*, "When Edge Meets Learning: Adaptive Control for Resource-constrained Distributed Machine Learning," CoRR, abs/1804.05271, 2018.

[33] H. McMahan *et al.*, "Communication-efficient Learning of Deep Networks from Decentralized Data," in AISTATS, 2017.

[34] J. Konecny *et al.*, "Federated Learning: Strategies for Improving Communication Efficiency," ArXiv, abs/1610.05492, 2016.

[35] S. Mastorakis *et al.*, "Icedge: When Edge Computing Meets Information-centric Networking," *IEEE Internet of Things Journal*, 2020.

[36] E. Haleplidis *et al.*, "Software-Defined Networking (SDN): Layers and Architecture Terminology, RFC 7426, Jan. 2015.

Biographies

Abdelsalam (Sumi) Helal is a professor and Chair of Digital Health in the School of Computing and Communication at Lancaster University. He is also a professor at the University of Florida. He has made significant contributions in the areas of digital health, pervasive and mobile computing, distributed databases and the Internet of Things.

Flavia Delicato is an associate professor at Universidade Federal Fluminense and also a collaborator researcher at the Centre for Distributed and High-Performance Computing (University of Sydney, Australia). Her primary research interests are IoT, WSN, middleware and Edge computing.

Cintia Borges Margi is an associate professor in the Computer and Digital Systems Engineering Department at Escola Politécnica, Universidade de São Paulo. She does research in Wireless Sensor Networks and Software Defined Networking.

Satyajayant "Jay" Misra is a professor in computer science at New Mexico State University. He has contributed to protocol design for anonymity, security, and survivable systems of the Future Internet, super-computing, and IoT/Cyber-Physical Systems architectures.

Markus Endler obtained his Dr. rer. nat. degree (Technical University of Berlin) in 1992, and has been a professor livre-docente (University of So Paulo) since 2001, and a CNPq Researcher. In 2001 he joined the Department of Informatics at Pontifcia Universidade Catlica in Rio de Janeiro (PUC-Rio), where he is currently an associate professor. His main research interests include mobile computing, distributed pervasive systems, context awareness and Internet of Mobile Things.

Authorized licensed use limited to: UNIVERSITY OF NEW MEXICO. Downloaded on January 12,2021 at 18:12:38 UTC from IEEE Xplore. Restrictions apply.