ORCA: Enabling an Owner-Centric and Data-Driven Management Paradigm for Future Heterogeneous Edge-IoT Systems

Jianli Pan, Jianyu Wang, Ismail AlQerm, Yuanni Liu, and Zhicheng Yang

The authors present a novel owner-centric management paradigm named ORCA to address the gaps left by the owner-centric paradigm and empower IoT asset owners to effectively identify and mitigate potential issues in their own network premises, regardless of vendors'/SPs' situations. ORCA aims to be scalable and extensible in assisting IoT owners to perform intelligent management through a behavior-oriented and data-driven approach.

Abstract

Integrating the Internet of Things (IoT) and edge computing in Edge-IoT systems, converged with machine intelligence, has the potential to enable a wide range of applications in smart homes, factories, and cities. Edge-IoT can connect many diverse devices, and IoT asset owners can run heterogeneous IoT systems supported by various vendors or service providers (SPs), using either cloud or local edge computing (or both) for resource assistance. The existing methods typically manage the systems as separate vertical silos, or in a vendor-/SP-centric way, which suffers from significant challenges. In this article, we present a novel owner-centric management paradigm named ORCA to address the gaps left by the owner-centric paradigm and empower IoT asset owners to effectively identify and mitigate potential issues in their own network premises, regardless of vendors'/SPs' situations. ORCA aims to be scalable and extensible in assisting IoT owners to perform intelligent management through a behavior-oriented and data-driven approach. ORCA is an ongoing project, and the preliminary results indicate that it can significantly empower IoT system owners to better manage their IoT assets.

Introduction

The National Academy of Engineering (NAE) identified 14 grand challenges our society faces [1], including virtual reality, health informatics, secure cyberspace, clean water, and urban infrastructure. They can directly benefit from integrating artificial intelligence (AI), machine intelligence, the Internet of Things (IoT), edge computing, and 5G to closely work for citizens, businesses, and the whole society. Future smart homes, factories, communities, and cities will also be empowered. We envision a future Edge-IoT environment [2] converged with machine intelligence and data-driven approaches to better serve people and businesses. Edge-IoT can connect massive numbers of smart devices, and IoT asset owners can run heterogeneous IoT systems supported by various vendors' or service providers' (SPs') platforms, and can use either cloud or local edge computing (or both) for resource assistance. However, it is a significant challenge to scalably and effectively manage such a dramatic number and variety of devices, and heterogeneous Edge-IoT systems. Poor management partially contributes to large-scale botnet attacks and significant financial loss [3]. Specifically, current IoT systems are typically managed by different vendors/SPs as separate vertical silos [4]. The vendor-/SP-centric management overly relies on vendors' uneven capabilities, and lacks transparency and cross-subsystem insights for the owners. The owners are also at risk of losing basic management capabilities when vendors/SPs run into abnormal situations or go out of business.

In this article, we envision building a novel owner-centric management paradigm to fill the existing gaps and empower IoT owners to manage across subsystems, which the vendors/SPs are currently not able to do. The owners are in the most capable and suitable position of in-premises edge networks (not cloud) to effectively identify and mitigate potential issues. The significance of the new paradigm is multifold. First, it empowers owners to manage diverse devices and complex behavior, and can greatly reduce financial loss due to management failure. Second, it enables owners to manage across subsystems when separate silos are not fully interoperable and standardization falls behind. Third, it enables owners to continue managing their assets even when vendors/SPs stop support or are out of business.

However, there are significant technical barriers to enabling this new paradigm. First, existing methods in industry and academia either only manage small device variety in dedicated silos and only consider simple behavior, or use limited data source such as network traffic. In such tasks, simple statistic or machine learning methods suffice, and they can afford relatively expensive sample labeling. But the owners may have to manage a large device variety and complex behavior patterns, and large-scale sample labeling also becomes economically infeasible. Second, the existing methods do not account for scalability and extensibility to accommodate owners' growing management interests. Target behavior may also have different complexity, and the modeling approaches should be customizable to balance between performance and cost. Third, the existing data-driven management methods in both industry and academia focus on small scopes, and they

Digital Object Identifier: 10.1109/MCOM.001.2000237

Jianli Pan, Jianyu Wang, and Imail AlQerm are with the University of Missouri; Yuanni Liu is with ChongQing University of Posts and Telecommunications; Zhicheng Yang is with PAII Inc.

0163-6804/21/$25.00 © 2021 IEEE

lack a holistic full-cycle data-driven approach to empower the IoT owners across the whole management cycle of "observing, synthesizing, and responding."

In this article, we aim to present a scalable and extensible owner-centric management framework named ORCA to assist IoT owners to perform intelligent management for diverse devices and heterogeneous Edge-IoT systems through a full-cycle data-driven approach. Specifically, ORCA holistically addresses the above technical barriers via a series of unique designs and contributions. First, it adopts a unique behavior-oriented and data-driven approach to allow owners to model complex behavior of diverse devices and heterogeneous Edge-IoT systems utilizing various data sources. Second, ORCA allows the owners to scalably and extensibly define and deploy multi-level observable "behavior" models (output as "insights"), identify suitable modeling approaches based on behavior complexity and data features, and balance performance and cost. Third, ORCA provides full-cycle customized data-driven toolsets for the IoT owners to model device behavior, synthesize cross-silo group behavior, and make intelligent management decisions without being required to have deep technical expertise. Fourth, ORCA runs at edge premises instead of in the cloud, avoids excessive data transmission and delay, and can manage when offline. It is run by owners, independent of the existing functions in silos, and can continue managing when vendors stop support or even go out of business.

The rest of the article is organized as follows. The following section is related work and the current vendor/SP-centric paradigm. The ORCA rationale is discussed next. The ORCA architecture and the data-driven 3-step IoT management are presented. Next, we present some preliminary evaluation and discussions. The conclusions follow.

Current IoT Management and Vendor/SP-Centric Paradigm

In this section, we discuss the current related work and issues on IoT management.

Related Work

IoT management has been studied on individual aspects such as trust, resource, energy/power, data, and privacy management. Management has also been closely tied to specialized devices in industrial factory machinery, power grid, water network, and supply chain. For example, prognostics and health management (PHM) [5] for industrial machinery health diagnostics has been heavily researched using recent statistical and machine learning methods. Such industrial applications have recently been moved to the cloud and managed by vendors/SPs as separate silos [4]. Some examples include Amazon AWS IoT, IBM Watson IoT, Google Cloud IoT, and small vendors renting cloud space to provide support. Inside each silo, machine-learning-based IoT analytics are performed for predictive maintenance, big data inference, and anomaly identification. Mobile device management (MDM) [6] deals with smartphone management with limited types of devices and operating systems (OSs). Its goals and scopes are different from IoT management. Another category of works use machine learning techniques over data traffic for device fingerprinting, behavior analysis, and intrusion detection. Typical commercial products include Extreme IoT Defender, Zingbox, and Cisco Appdynamics. However, using only traffic analysis is limited. The used machine learning methods require expensive labeling, and are inadequate to model very diverse devices and growing management interests. In addition, "horizontal" efforts [7] aim for better interoperability among silos. Example efforts include standardization, industry alliances, IoT ontologies [8], and market convergence. The horizontal process is relatively slow, and by itself it cannot lead to owner-centric management.

Various Management Modes and the Vendor/SP-Centric Paradigm

Depending on the actual cases and business models, different management modes exist. IoT systems can be managed either by owners themselves locally or by specialized vendors/SPs at the edge or clouds. We consider two key factors: owners' characteristics and capabilities, and the vendors/SPs' expertise and capability. On one hand, various owners may have very different expertise and capability. The first category is that the owners run very dedicated applications such as Industry 4.0 factories, smart vehicle charging networks, and camera-based security events detection. In these cases, the owners are either very capable and can manage all the specialized devices by themselves locally or at the edge, or rely on very powerful vendors/SPs such as Google and Amazon to manage at the edge or cloud. For these ideal cases, existing management methods may suffice. The second category includes less ideal cases in that the common IoT owners are much less capable, and they barely have adequate expertise or tools to manage things all by themselves. On the other hand, various vendors/SPs on which the owners rely may also have very different capability. For example, they may range from powerful companies such as Amazon and Google to vendors of many cheap devices that barely provide any management.

Meanwhile, it has become common that IoT owners may own and run multiple systems in their networks, and they rely on various vendors/SPs to manage these systems separately. For example, a smart home owner may run NEST on Google's cloud for thermostat, Amazon's Echo on AWS cloud for voice assistant, and some cheap original equipment manufacturer (OEM) IP cameras from small vendors on rented cloud spaces. These subsystems are either managed by owners themselves or in a vendor/SP-centric way, that is, by different device vendors or SPs vertically as separate silos [4] that typically do not share interfaces, data, and insights. Within the foreseeable future, the IoT market will remain scattered with various sizes of vendors/SPs using different software and platforms. The standardization process remains relatively slow, and the interoperability between different platforms is limited.

In addition, a series of emerging trends will bring even more difficulties to management. First, there are increasing numbers of devices and growing management interests. Second, the devices

Depending on the actual cases and business models, different management modes exist. IoT systems can be managed either by owners themselves locally or by specialized vendors/SPs at the edge or clouds. We consider two key factors: owners' characteristics and capabilities, and the vendors/SPs' expertise and capability.

Devices and applications	Data type	Priority (1-4: H to L)	Computing intensity	Data intensity	Latency sensitivity
Emergency real-time response (e.g., gunshot detection)	Video/audio	1	High	High	High
VR/AR related applications	Video	2 or 3	High	High	High
Home voice assistant	Audio	2	Medium	Medium/low	High
Cognitive assistance	Video/audio	2 or 3	High/medium	High/medium	Medium
Building access face detection	Video	3	Medium	Medium/low	Medium
Personal identification	Audio/ image/text	3	Medium/low	Medium/low	Medium
Home health monitoring	Text	2	Low	Low	Low
Common smart home devices	Text/audio	4	Medium/low	Low	Low
Low-level sensors	Text	4	Low	Low	Low

TABLE 1. Example Edge-IoT systems and characteristics.

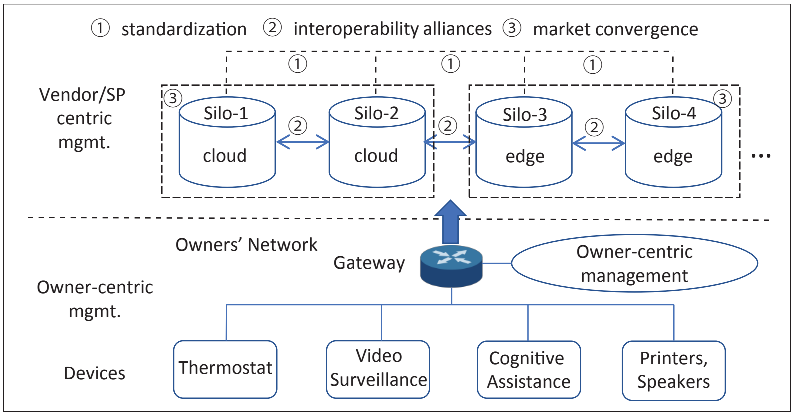


FIGURE 1. Owner-centric vs. vendor/SP-centric paradigms.

can be very diverse [9] in constraints, network types (wireline/wireless), protocols (Wi-Fi/Ethernet/5G), network patterns (point-to-point and point-to-multipoint [P2P/P2MP]/multihop), media types (text/audio/video), and characteristics (running modes, bandwidth, and response frequency). They may show a wide range of behavior patterns. Third, heterogeneous IoT systems have various quality requirements. Some typical examples are shown in Table 1, which includes resource-intensive and latency-sensitive applications (in light gray). The existing management practices of the IoT systems have fallen short, which has been partially reflected in widespread large-scale botnet attacks and significant financial losses caused by millions of poorly managed smart IoT devices in recent years [3].

Owner-Centric Paradigm and Rationale

In this section, we discuss the new paradigm and the designing rationale.

Owner-Centric vs. Vendor/SP-Centric Paradigms

For common owners, overly relying on the vendor/SP-centric paradigm may incur significant limitations. Thus, we propose a new owner-centric paradigm named ORCA to fill the existing gaps left by the vendor/SP-centric paradigm (not to replace the application logic of individual silos), and provide much-needed advanced designs to empower those vulnerable and incapable IoT owners to better manage various heterogeneous IoT systems. Specifically, ORCA addresses the challenges as follows. First, various vendors/SPs may have very uneven technical capabilities, and the powerful vendors/SPs cannot manage devices across silos. This may result in poor management of some devices (e.g., some OEM IP cameras) by less capable vendors/SPs. They can become weak links, and be compromised and used as springboards by hackers to launch internal attacks. ORCA provides owners with basic management for these vulnerable devices so that they will not be easily exploited by hackers. Second, the management functions of various vendors/SPs are typically separate and not transparent, and the owners generally have no way to gain cross-subsystem insights. For example, the owners may want to know whether the shared resource pool is being responsibly used by different subsystems, and whether devices abnormally interact with others managed by different vendors/SPs. ORCA will allow the owners to manage beyond silos and extract useful cross-system or group insights by performing better data analysis defined by the owners and serving their own objectives. The owners are in a better position to judge how these groups of devices from different vendors can be managed to serve the owners' purposes and fit with its current facilities. Third, when specific vendors/SPs experience temporary/permanent situations, stop support, or go out of business, the owners are at risk of losing basic management capabilities over their own assets. ORCA will continue providing basic management support even if the above situations occur, and it will reduce management lapses that hackers can exploit. Fourth, instead of managing from the cloud, ORCA manages at the owners' network premises, which does not incur large volumes of data transmission and long delay, and does not require the owner network to be always online.

We compare the two paradigms in Fig. 1 via a simple smart home example with four silos including subsystems using the cloud or edge. It also shows the three types of horizontal efforts including standardization, interoperability alliances, and market convergence, and illustrates their relationship. Horizontal progress can potentially alleviate some of the challenges and help management with better data quality and availability, and hence benefit ORCA's performance. However, horizon-

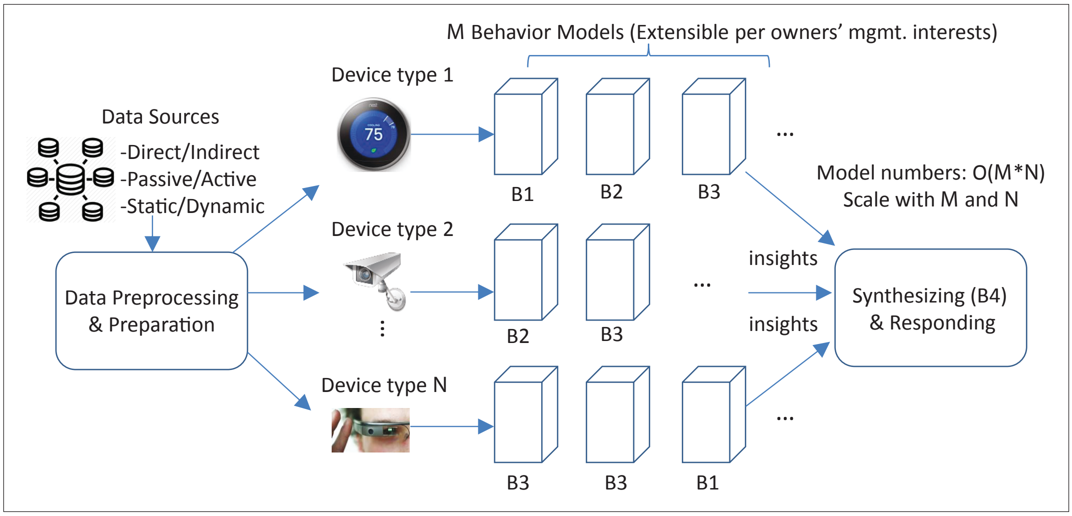


FIGURE 2. Scalable and extensible behavior models.

The resource allocation module aims to build a QoE model, takes various behavior insights as parameters, and allocates edge resources using a two-stage deep online learning method, with the goals of maximizing users' satisfaction and edge resource utilization, and encouraging good behavior.

tal progress does not automatically lead to owner-centric management. Even in a "perfect" world with full standardization and interoperability, and with fewer silos, management is still done in a vendor/SP-centric way. The horizontal vision has a relatively long way to go.

Behavior-Oriented and Data-Driven Approaches

We define "behavior" as the patterns that the IoT owners want to observe based on their extensible management interests of different levels [10]:

1. B1: device-level interests such as hardware failure, software malfunction, and remaining lifetime
2. B2: network-level interests such as traffic patterns, unsafe connections, and botnet activities.
3. B3: cloud/edge interests such as requested resource, offloaded tasks, and response time
4. B4: group and subsystem-level interests such as how groups behave in B1 to B3

In ORCA, to generate device behavior insights in a timely manner and at suitable cost, specialized machine learning models are needed that are suitable to the behavior complexity and can appropriately balance between performance and cost for both model training and operation. The overall idea is illustrated in Fig. 2. The owner will be able to deploy behavior models flexibly and extensibly based on device types and their management interests (B1–B4). Moreover, ORCA is data- and insight-driven. The device-level behavior modeling results are used to synthesize group and system-level behavior, and they are further used to assist owners to make intelligent management responses.

ORCA Architecture and Data-Driven Three-Step Management

In this section, we focus on the ORCA architecture and the proposed data-driven three-step IoT management.

ORCA Architecture Overview

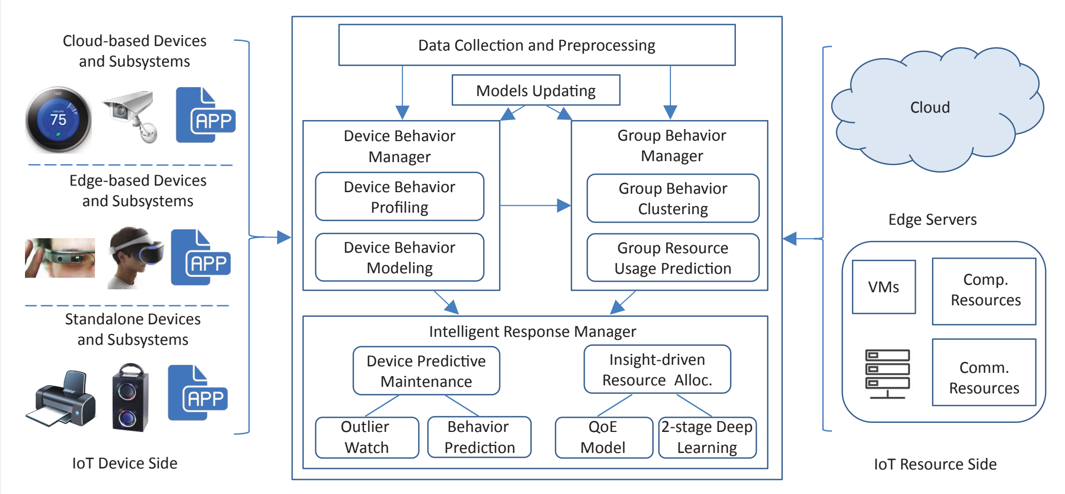
The ORCA architecture is presented in Fig. 3. It works at the network edge and integrates the IoT device side and the resource side. The device side includes multiple types of devices and subsystems. The resource side consists of multiple edge servers comprising virtual machines, and communication and computation resources. The acquired data from both sides go through quality improvement, and the resulted data samples are used to train or retrain the models. The device behavior manager will profile the device behavior based on managers' interests, and decide the appropriate candidate models based on the data time dependency and the behavior complexity. The group behavior manager synthesizes device-level behavior into group-level insights by clustering, and group resource usage trends by long short-term memory (LSTM)-based prediction [13]. The intelligent response manager will utilize the insights from the device manager and group manager to make intelligent decisions in device-level predictive maintenance, and quality of experience (QoE)-based intelligent edge resource allocation. Specifically, the device predictive maintenance module will further inspect the group outliers identified by the clustering and make behavior predictions using an online and lightweight OL-ARIMA [12] approach. It will then generate a device list for further maintenance. The resource allocation module aims to build a QoE model, takes various behavior insights as parameters, and allocates edge resources using a two-stage deep online learning method, with the goals of maximizing users' satisfaction and edge resource utilization, and encouraging good behavior.

Data-Driven Three-Step Management

"Observing": Owner-Centric Device-level Behavior Modeling: This step focuses on IoT owners' device manager role and aims to profile and model various behaviors of diverse devices for heterogeneous IoT systems from an owner-centric perspective with growing management interests.

Behavior Profiling: We first profile multi-level new behavior targets, exploit various data sources, and map the behavior targets to suitable models. The device behavior profiling workflow is shown in Fig. 4. The workflow inputs are the target behavior that the owners choose to model for a specific type of device, and the outputs are

For simpler behavior with low dimensionality of the non-time series data, we develop a simpler method based on One-class Support Vector Machine (OC-SVM) for fast and less-costly modeling, and an agile and lightweight prediction method based on Multivariate Autoregressive Integrated Moving Average (MARIMA) for time series data.



Cloud-based Devices and Subsystems
75
APP
Edge-based Devices and Subsystems
APP
Standalone Devices and Subsystems
APP
IoT Device Side
Data Collection and Preprocessing
Models Updating
Device Behavior Manager
Device Behavior Profiling
Device Behavior Modeling
Group Behavior Manager
Group Behavior Clustering
Group Resource Usage Prediction
Intelligent Response Manager
Device Predictive Maintenance
Insight-driven Resource Alloc.
Outlier Watch
Behavior Prediction
QoE Model
2-stage Deep Learning
Cloud
Edge Servers
VMs
Comp. Resources
Comm. Resources
IoT Resource Side

FIGURE 3. ORCA architecture.

the trained models that have learned the patterns of normal device behavior. To jointly consider performance, cost, and scalability, we formulate behavior modeling as a one-class classification problem [11] in order to avoid difficult and expensive labeling for anomalies of diverse devices and relatively low frequency for specific faults, and leverage the relatively ample "normal" behavior data to train the one-class classifiers. The behavior models will output the devices' current abnormal degree, comparing it to the normal or healthy condition. For example, if an IoT owner wants to define a behavior model for an IP camera in the B2 category focusing on its traffic pattern, the model output will be a score measuring its abnormal degree. If the camera is having a software malfunction or a botnet attack, the model will output a score in an alarming range. Depending on behavior types (B1 to B3), various data sources related to multi-level features across the IoT reference model are used to prepare training samples and dataset.

Behavior Modeling: We then map the behavior target to the suitable modeling methods based on two data attributes, time dependency and feature dimensionality, in order to balance between performance (speed/accuracy) and cost (training/operation). First, key behavior data may be either time series or non-time series data. Time-series data are generated periodically at fixed intervals such as sensors' data or network packets with natural timestamps, and the patterns may exist in the time dependency. Non-time-series data have little or no time dependency between instances. Second, complex behavior's high feature dimensionality may cause high data sparsity, and require advanced modeling approaches and more training samples. Thus, to effectively learn high-dimensional feature distributions in the one-class classfiers, we integrate the generative adversarial network (GAN) [14] with encoder-decoder networks (GAN-ED) for non-time-series behavior modeling, and integrate LSTM [13] with ED networks (LSTM-ED) for time-series behavior modeling. In the GAN-ED model, we incorporate an encoder network into the original GAN framework. The encoder advances its learning ability, which compresses the inputs into low-dimensional feature vectors through the mutual training process with the generative and discriminative networks in GAN. In the LSTM-ED model, we employ LSTM neurons in the hidden layers for both the encoder and the decoder networks to learn the time dependencies between the input features. For simpler behavior with low dimensionality of the non-time-series data, we develop a simpler method based on one-class support vector machine (OC-SVM) for fast and less costly modeling, and an agile and lightweight prediction method based on multivariate autoregressive integrated moving average (MARIMA) for time series data.

"Synthesizing": Owner-Centric Group and Subsystem Behavior Modeling: This step focuses on the IoT owners' application and resource manager roles, and aims to synthesize group and subsystem level behavior and resource usage insights for the following management decision making.

Group/Subsystem Behavior Synthesizing: This stage aggregates individual devices' behavior scores, synthesizes group behavior, and identifies outliers. We allow the IoT owners to flexibly define "groups" as IoT subsystems (e.g., all the IP cameras for a video surveillance subsystem), locations (e.g., all devices on a specific floor of a building), or device batches (e.g., all devices procured in the same batch with similar software and hardware configurations). With such grouping, we can run clustering to synthesize group insights and identify potential "outliers," and help owners find out the device numbers with lowest or highest scores, and the current and historical score distributions for IoT subsystems. These insights can help understand system dynamics and identify problematic devices. They can help owners find out whether the devices have lower scores than other areas for a location/network. These insights can help identify large-scale botnet activities or malfunction caused by faulty IoT gateways.They can also help decide whether they are showing similar low scores for a device batch. These insights can help identify batches that need updates or attention.

Group/Subsystem Resource Usage Prediction: This stage synthesizes how the shared edge resources are currently being or will be used by groups or subsystems. For example, if a video cognitive subsystem uses 90 percent of the total edge resource while contributing only 10 percent of the financial revenue, IoT owners may be motivated to rebalance the resource budgets among subsystems. Similarly, for abnormal cases when the group resource usage with certain locations/networks or device batches reaches alarming ranges, it may justify certain management actions. In addition, if the group resource usage insights are combined with accurate and prompt prediction, they can help IoT owners to efficiently allocate the shared edge resources to the different subsystems, better prepare for large-scale abnormal incidents, and protect the overall welfare of these systems. A typical candidate tool is LSTM.

"Responding": Owner-Centric and Data-Driven Management Responses: This step takes all IoT owners' roles, as device, application, and resource manager, and aims to enable the IoT owners to make well-informed and intelligent management decisions at the edge to manage individual devices and the shared edge resources among all subsystems.

Device-Level Predictive Maintenance: The IoT devices' behavior can be very dynamic. To identify devices that need future maintenance with confidence, we conduct behavior prediction over the group outliers identified in the above second step. Behavior prediction in heterogeneous Edge-IoT needs to handle multiple behavior models, large amounts of historical records, random behavior pattern changes with stationary and non-stationary distributions, varied prediction window lengths for different reaction delays, and fast and cost-efficient prediction.

Intelligent Edge Resource Allocation: With the obtained insights, we then aim to intelligently allocate edge resource to jointly optimize user experience and resource utilization, while containing bad behavior. We design a novel resource allocation scheme that does two things. First, we build a new QoE model to quantify the devices' satisfaction, which comprises heterogeneous subsystems' QoS requirements and priorities. The model accounts for a device's current and predicted behavior, group behavior, and predicted edge resource usage. Second, we build a novel two-stage deep online learning [15] scheme to jointly optimize user experience and edge resource usage across subsystems.

Evaluation and Discussions

In this section, we conduct preliminary evaluations on scalability both qualitatively and quantitatively.

Qualitative Evaluation and Discussions

First, for the vendor/SP-centric paradigm, a vendor like Google or Amazon may have to manage in central locations (clouds) for large numbers of devices residing in many owners' domains. In comparison, for ORCA, in each owner's domain, the number of devices they own and manage is much less. Second, ORCA incorporates scalability and extensibility supports. As illustrated in Fig. 2, in ORCA, the required number of trained and deployed behavior models is approximately $O(M*N)$. It is decided by the number of device types N (e.g., cameras and drones) and the number of behavior models M (B1 to B4) for each device type. With such design, the overall model cost of ORCA will increase polynomially, regardless of the possible exponential increase of the number of devices. Thus, qualitatively speaking, when a new device type comes to the market, ORCA will be able to scale up efficiently with the IoT device variety N. When the owner wants to extend his/her management interests, ORCA will be able to scale up by increasing new management interests M. In addition, to balance the cost (training/operation) and performance (speed/accuracy) for the IoT behavior models, ORCA provides four candidate modeling approaches for the owners to choose based on the feature dimensionality and data time dependency.

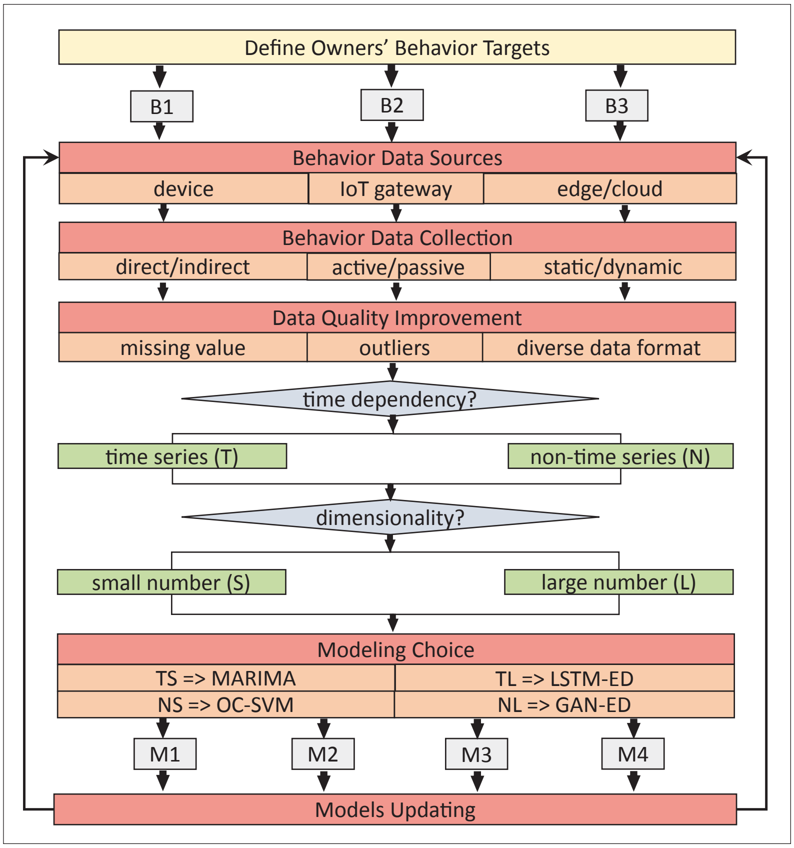


FIGURE 4. Device behavior profiling and modeling workflow.

Quantitative Evaluation and Discussions

We also present a quantitative evaluation on scalability. We build a testbed with several typical IoT devices such as IP cameras and temperature sensors, and an ORCA manager using a Raspberry Pi 3 with 1.4 GHz CPU, 1 GB RAM, and 16 GB storage. The behavior dataset consists of time-series (TS) and non-time-series (NTS) samples. TS samples are collected every 30 minutes where each sample is a single-variate sequence with 90 data points. NTS are sampled once per minute where each sample has 80 features from B1 to B4. We evaluate behavior modeling cost by implementing the four models proposed earlier, where GAN-ED and LSTM-ED have two neural layers with 64 and 32 neurons in both the encoder and decoder networks. For NTS, behavior measuring tasks are uniformly distributed in 1 minute. With these configurations, we observe the following results. First, the size of one TS sample and one NTS sample are about 1 kB and 0.5 kB, respectively. Suppose that the IoT devices number is

Model	Model size (kB)	Model running time (ms)	Model running memory (MB)
OC-SVM	103	6	97
MARIMA	36	12	84
GAN-ED	345	62	163
LSTM-ED	630	233	184

TABLE 2. Costs of the four models.

120 in the service coverage of one manager: the total behavior data size for all devices are 60 kB per minute for TS and 120 kB every 30 minutes for NTS. Second, we observe the key scalability parameters including models' sizes, running time, and running memory in Table 2. For model size, LSTM-ED is the biggest among them, and each one takes only 630 kB. For running time, the one-time behavior evaluation takes at most 233 ms, which meets the responding latency requirement of ORCA. For the running memory, we observe that the runtime memory consumption of any model does not exceed 184 MB. The evaluation results demonstrate that all the models can run on the ORCA server with very limited resource cost. In our experiment settings, a resource-constrained Raspberry Pi-based ORCA manager can manage a fair number of devices while causing limited network traffic and storage overhead.

The evaluation results demonstrate that all the models can run on the ORCA server with very limited resource cost. In our experiment settings, a resource-constrained Raspberry Pi-based ORCA manager can manage a fair number of devices while causing limited network traffic and storage overhead.

Conclusions

It is a significant challenge to manage massive numbers of diverse devices and heterogeneous Edge-IoT applications. The current methods mostly manage these systems as separate vertical "silos," or in a vendor/SP-centric way, which suffers from a series of limitations. To address the challenges, in this article, we propose a new owner-centric paradigm named ORCA empowered by data-driven approaches and machine learning techniques. ORCA aims to fill the gap and complement the missing pieces of the existing management approaches. It provides a scalable and extensible framework for IoT asset owners to perform data-driven three-step management to complete the "observing, synthesizing, and responding" management cycle. The preliminary evaluation demonstrates the effectiveness of the proposed ideas. Our future work includes further validation and integration of the building pieces.

Acknowledgment

The work is supported by NSF research grant No. 1909520.

References

[1] NAE Panel, "NAE Grand Challenges for Engineering," *Nat'l. Academy of Engineering*, released 2008, updated 2017.

[2] J. Pan and J. McElhannon, "Future Edge Cloud and Edge Computing for Internet of Things Applications," *IEEE Internet of Things J.*, Special Issue on Fog Computing in IoT, vol. 5, no. 1, Feb. 2018, pp. 439–49.

[3] Threatpost, "Mozi Botnet Accounts for Majority of IoT Traffic," 2020.

[4] O. Novo and M. Di Francesco, "Semantic Interoperability in the IoT: Extending the Web of Things Architecture," *ACM Trans. Internet of Things*, vol. 1, no. 1, 2020, pp. 1–25.

[5] C. L. Gan, "Prognostics and Health Management of Electronics: Fundamentals, Machine Learning, and the Internet of Things," *Life Cycle Reliability and Safety Engineering*, vol. 9, 2020, pp. 225–26.

[6] D. Smith, "Mobile Device Management," Apple macOS and iOS System Admin., Apress, Berkeley, CA, 2020, pp. 311–38.

[7] A. Al-Fuqaha *et al.*, "Toward Better Horizontal Integration among IoT Services," *IEEE Commun. Mag.*, vol. 53, no. 9, Sept. 2015, pp. 72–79.

[8] Smart Appliances REFerence (SAREF) ontology; https://ontology.tno.nl/saref/, accessed June 2020.

[9] M. Ersue *et al.*, "Management of Networks with Constrained Devices: Problem Statement and Requirements," IETF RFC 7547, May 2015.

[10] M. A. A. da Cruz *et al.*, "A Reference Model for Internet of Things Middleware," *IEEE Internet of Things J.*, vol. 5, no. 2, 2018, pp. 871–83.

[11] P. Perera and V. M. Patel, "Learning Deep Features for One-Class Classification," *IEEE Trans. Image Processing*, 2019.

[12] Pesaran H., "Time Series and Panel Data Econometrics," Oxford Univ. Press. 2015.

[13] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, Nov. 1997, pp. 1735–80.

[14] I. Goodfellow *et al.*, "Generative Adversarial Nets," *Advances in Neural Information Processing Systems*, 2014.

[15] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, MIT Press, 1998.

Biographies

Jianli Pan (pan@umsl.edu) is currently an associate professor in the Department of Computer Science at the University of Missouri, St. Louis. He obtained his Ph.D. degrees from the Department of Computer Science and Engineering of Washington University in St. Louis. He is an Associate Editor for both *IEEE Communications Magazine* and *IEEE Access*. His current research interests include the Internet of Things, edge computing, machine learning, and cybersecurity.

Jianyu Wang (jwgxc@umsl.edu) is currently a Ph.D. student with the Department of Computer Science at the University of Missouri, St. Louis. He received an M.S. in electrical and computer engineering from Rutgers University, New Brunswick, New Jersey. His current research interests include edge cloud and mobile cloud computing.

Ismail AlQerm (alqermi@umsl.edu) is a postdoctoral research associate in the Department of Computer Science at the University of Missouri, St. Louis. He received his Ph.D. in computer science from King Abdullah University of Science and Technology (KAUST) in 2017 and was among the recipients of the KAUST Provost Award. His research interests include edge computing, IoT resource allocation, machine learning in wireless networks, and software defined radio.

Yuanni Liu is an associate professor at the Institute of Future Network Technologies, ChongQing University of Posts and Telecommunications, China. She received her Ph.D. from the Department of Network Technology Institute, Beijing University of Posts and Telecommunications, China, in 2011. Her research interests include mobile crowd sensing, IoT security, and data virtualization.

Zhicheng Yang received his Ph.D. degree in computer science from the University of California, Davis in 2019. He is currently a senior research scientist at PAII Inc. His current research interests include millimeter-wave sensing, 60 GHz communications and networking, and mobile computing.