Edge-Cloud Intelligence in Self-Diagnostic of Land Mobile Radio Systems

Hung Cao
University of New Brunswick
Fredericton, Canada
hcao3@unb.ca

Monica Wachowicz
RMIT University
Melbourne, Australia
monica.wachowicz@rmit.edu.au

James Craig
Rimot.io Inc
Halifax, Canada
james.craig@rimot.io

***Abstract*—IIoT sensors are usually deployed on a massive scale with stringent scalability, modularity, and interoperability requirements. It is indisputable that they produce a large amount of high-speed and heterogeneous data streams that pose many challenges to perform management, processing, and analytical tasks. This paper proposes an integrated edge-cloud continuum platform that can harvest IIoT data streams from a variety of sensors deployed at a remote RF site; and can harmonize different machine learning models for diagnosing problems that enhance infrastructure monitoring and long-term structural resilience. A real-world experiment was carried out to evaluate the proposed platform for supporting a self-diagnostic process for intelligent maintenance of Land Mobile Radio (LMR) infrastructures.**

***Index Terms*—cloud computing, edge computing, Land Mobile Radio Systems, Industrial Internet of Things, streaming analytics, intelligent maintenance**

I. Introduction

Industrial IoT (IIoT) is a branch of IoT that integrates the domains of machine-to-machine (M2M) and automation applications with industrial communication and sensing technologies [1]. Due to the legacy of industrial systems, the IIoT applications are usually developed on top of a traditional infrastructure, targeting at analytics such as predictive maintenance and improved logistics. Some noticeable IIoT applications include smart logistics, smart factories and manufacturing, and remote maintenance [2].

Traditionally, IIoT sensor data is streamed and offloaded to a large shared pool of network, storage, and computing resources that can provide on-demand processing services that are released through a selection of cloud architectures (e.g. Private, Public, Community, and Hybrid Cloud) [3]. Knowledge and insights can be yielded through a training and inference process using current machine learning models running on the virtually unlimited resources of the cloud [4]. However, this approach is only suitable for delay-tolerance applications [5], since the long transmission distance between centralised cloud data centers and IIoT sensors can generate a delay overhead and bandwidth consumption.

Recently, the edge computing paradigm has been leveraged to alleviate the network burden of transporting IIoT data to the cloud [6]. Edge computing resources include edge servers (cloudlets), edge accelerators, and edge gateways [7], which are referred to as edge nodes in this paper. By bringing edge nodes in proximity to IIoT devices, latency and network bandwidth consumption can be considerably reduced due to the edge decentralisation nature; and IIoT data streams can be pre-processed and analyzed in part or in whole at the edge nodes, optimizing the existing legacy cloud resources.

However, edge nodes, in practice, have less storage capacity and computing power than the centralised cloud data centers. Therefore, leveraging only edge nodes may not be a suitable solution to support the diversity of computing and storage requirements of IIoT applications. The research challenge is to develop architectural frameworks capable of integrating edge and cloud resources that are necessary to fulfill the diversified and rigorous requirements of IIoT applications, aiming to unlock hidden value of IIoT data streams and deliver intelligence to innovative business models and services.

Towards this end, this paper proposes an intelligent platform based on the widely accepted three-tier architecture, i.e. sensors, edge nodes, and cloud, which are are connected by their geographical proximity and network access. The goal is to develop a new automated data workflow capable of performing remote monitoring and sensing of critical infrastructures by powering IIoT sensors with continuously integrated machine learning models, running at both edge and cloud resources.

The proposed platform is evaluated using real-world IIoT data for self-diagnostic of Land Mobile Radio (LMR) systems, which are characterised by unique functional designs and operations, having key strengths such as power, resilience, and purpose-built devices for supporting critical operations such as paging firefighters, dispatching fleets of trucks, navigating ships and aircraft, and communicating on police radios.

The scientific contributions of this paper are as follows:

- Our research work is unique in proposing an edge-cloud intelligence approach to achieve an automated self-diagnostic process by combining edge and cloud analytics for generating early warnings based on discovered patterns about problems before they become failures. Previous research work related to remote monitoring telecommunications infrastructure has been focused on cell outage and sleeping cell phenomenon, which are the main causes for access failure in mobile networks.
- Since intelligent maintenance is revolutionizing every aspect of industrial processes, we propose a new automated data workflow capable of harmonizing different machine learning models that collaborate for an auto-

mated self-diagnostics of LMR systems. Real-world IIoT data streams of seven remote sites are used to validate the proposed workflow.

The remaining of this paper is organized as follows. In Section 2, previous research work is described. Section 3 describes the proposed self-diagnostic analytics in IIoT. Section 4 provides an overview of the implementation and the experiment used for collecting real-world IIoT data. In Section 5, we discuss the preliminary results. Finally, Section 6 concludes and indicates future research work.

II. Related work

The most basic maintenance approach is reactive maintenance, which comes into play when the equipment is broken and needs to be fixed [8]. Preventive maintenance is another approach that replaces the equipment based on their expected lifetime, regardless of their condition. Both types of maintenance deemed to be inefficient and costly since they might replace parts still in good operating conditions or with machine failures [9].

Previous research work related to reactive and preventive maintenance has been focused on the purpose of remote monitoring telecommunications infrastructures. Problems happen silently and immediately affect network performance, making failures very difficult to detect. Manzanilla-Salazar et al. [10] proposed a solution using Key Performance Indicators (KPIs) to detect failures, but they were only able to validate the proposed approach using simulated data from the location of base stations and IIoT devices.

The recent innovations in the IIoT, edge, and cloud technologies are fostering new developments towards intelligent maintenance [8]. A number of edge hardware accelerators such as Google Edge TPU, NVIDIA Jetson Nano Edge GPU, Apple's Neural Engine, Intel Vision Processing Unit (VPU) has recently emerged with the specific goal of supporting edge-based AI [11]. With an improved normalized performance, these edge resources are paving the way for a proliferation of edge intelligence in the near future [7].

Based on the path length of data offloading, Zhou et al. [12] describe the operation workflows into Edge2Edge, Edge2Cloud, Cloud2Cloud, and Cloud2Edge; where training and inference of machine learning models have been primarily focused on using cloud-based resources and gradually moving towards the edge. Moreover, Garcia et al. [13] applied an ensemble approach that combines several supervised machine learning algorithms such as SVM, ANN, Random Forest, and Naive Bayes to identify and prioritize alarms for fault management in cellular networks. This work has been done offline using an alarm data set provided by a network operator.

Therefore, intelligent maintenance plays an important role in the functionality and performance of Land Mobile Radio (LRM) systems [14]–[16]. LMR systems generate large amounts of operational data and system alarms on a daily basis. Most of these vital data go unused and require sending a technician out to investigate every alarm manually. This not only incurs the cost of additional truck rolls but also lost revenue during system downtime. Therefore, a remote monitoring solution must address the transmitter site shelter, the transmitters themselves, and the antenna system. It must provide visibility to each site and its critical components, so that management can better ensure telecommunications reliability by identifying and resolving problems in advance before they become failures [17], [18].

To best of our knowledge, our research work is a first attempt towards an intelligent management approach by devising an automated self-diagnostic process based on the ensemble of different machine learning models where automated tasks are performed on IIoT data streams available at edge and cloud resources.

III. Self-Diagnostic Analytics in IIoT

A. IIoT Stream Data Lifecycle

IIoT data streams can be categorized into two types: *accumulated data streams* and *continuous data streams* [3], [19]. Accumulated data streams are transmitted from the IIoT sensors to an edge or cloud resource, and the data tuples are accumulated using the sliding time window model until a workflow task is triggered or finalized. In contrast, continuous data streams are active incoming data tuples which require to be processed immediately when they arrive at an edge or cloud resource. Although continuous data streams do not required high cost of storing the data as accumulated data streams, they tend to use more processing power to run computations.

The types of raw IIoT streams include time-series (i.e. recorded timestamped readings at successive and equal time intervals) and event triggered data (i.e. recorded timestamped readings when a sensors are triggered due to an activity). In our proposed automated data workflow, the results of one workflow task are also the input for the other workflow tasks. We also anticipate that raw IIoT data streams are usually small in volume at the edge, and they can be subject to LMR system requirements that prevent data from being moved to a cloud, especially when natural disasters and incidents occur.

Therefore, a stream data lifecycle is supported in the proposed platform to: (1) continuously analyze and monitor incoming data tuples aiming to detect problems and understand them; (2) understand component or system behavior under a variety of conditions to constantly enhance further the current component or system; and (3) trigger specific actions to respond to changes when certain thresholds in the system are identified.

B. Computational Resources

The platform encompasses the edge-cloud continuum as shown in Figure 1. The main components can be described as follows:

- Sensors that can be proprietary or commercial-off-the-shelf (COTS). They usually generate IIoT data streams at different data rates and accuracy levels. They monitor some phenomenon, sending their observations with uplink communications from sensors to controllers, and downlink communications from controllers to actuators.

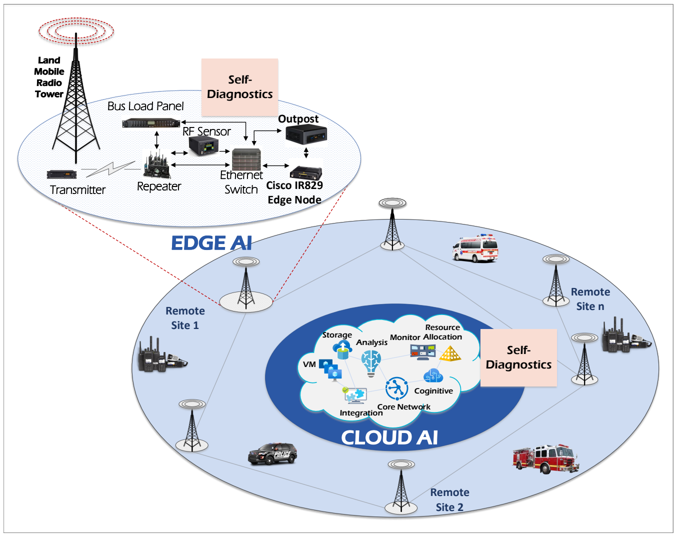


Fig. 1. Overview of the edge-cloud resources

- Edge Nodes which are continuously harvesting IIoT data streams and diagnosing what is currently happening with a remote site of an LMR infrastructure. Edge Intelligence provides information to diagnose the occurrence of a problem at the transmitter site shelter, the transmitters themselves, or the antenna system that might lead to a failure in LMR systems.
- Cloud resources are devised to support storage, compute engine, data flow management, network connection, analytics, and visualization. Cloud Intelligence can be explored to diagnose a problem at the network level of LMR systems.

We assume that communications between the resources are hierarchical: only sensors can communicate with the edge nodes, and only edge nodes can communicate with the cloud. Some edge nodes are expected to work together in parallel to make sure that the service is always on. Other edge nodes in the network play the role of both publisher/producer and subscriber/consumer. All data streams will be passed through the broker and they will be filtered and sent to the correct destination. Several protocols such as the binary protocol over TCP (Apache Kafka), MQTT (RabbitMQ) and MQTT-SN (MQTT for Sensor Networks) can be used to deploy our architecture.

C. IIoT Automated Data Workflow

Figure 2 provides an overview of the automated data workflow that is devised for supporting an automated self-diagnostic process. The sensing phase is rendered operational by a communication network that collects and exchanges useful data tuples to fully leverage the advantages of IIoT sensors. Energy-efficient and low-complexity security are required for supporting LMR IIoT sensing applications.

In the data ingestion phase, two main tasks are devised. First, the data control task aims to ensure secure remote access to the IIoT sensors and resolve troubleshoot issues. It is also designed to view and download sensor logs, and reset sensor state. Second, the data filtering task is used to filter and throttle data streams to reduce system bandwidth and minimize pre-processing and storing unneeded data tuples at the edge node.

The pre-processing phase is focused on performing pre-processing tasks on both accumulated and continuous IIoT data streams. The envisaged tasks include data cleaning and format conversion, as well as data contextualization, fusion, and partition, These tasks are performed at the edge nodes that are deployed at the remote sites.

The analytical phase combines different machine learning models that collaborate for an automated self-diagnostics of LMR systems. The edge analytics is performed using accumulated IIoT data streams. The machine learning models such as the SARIMA [20] and Prophet [21] models are selected to diagnose a problem taking place at a remote site. At the edge, a Python script containing a forever loop is also used to detect the incoming IIoT data streams which are visualized to support the monitoring of the health conditions of a remote site.

Cloud analytics is performed by applying the machine learning model using the accumulated data streams from all remote sites. The aim is to diagnose a problem at the network level of the LMR infrastructure using the Random Forest model [22]. The multi-correlation approach is also proposed to detect partial and semi-partial correlation between the sensor observations and improve the user's understanding about their relationships. Finally, based on the edge and cloud intelligence, notifications and actions can be delivered during the activation phase.

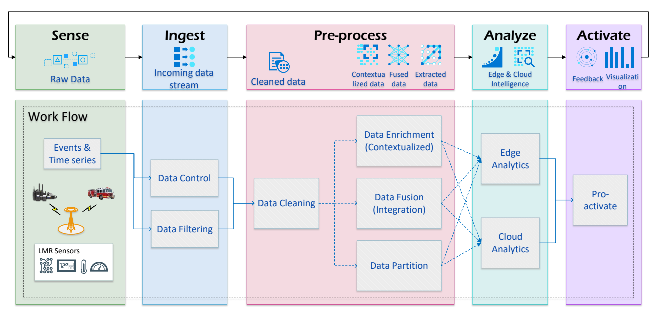


Fig. 2. Overview of the automated data workflow

IV. Experiment

A. Architecture Implementation

The sensors component was built based on general-purpose sensors (motion, temperature, humidity, wind) and commercial sensors (MOTOTRBO, RF Sensor). The edge nodes were implemented using the Cisco IR829 Industrial Integrated Services Routers (IR829) that have a compact form factor, multimode 4G LTE and 3G wireless WAN (dual-active LTE and single LTE models), IEEE 802.11a/b/g/n WLAN, Ethernet (RJ45 and SFP), serial connections, integrated storage and compute capability for edge application hosting, and integrated

9-32 VDC power input. The cloud environment was developed using AWS services.

B. IIoT Data Streams

The IIoT data streams were collected from different sensors located at seven remote sites. At each remote site, the incoming data was continuously pushed to the outposts (edge nodes) before sending it to the cloud. For this study, the IIoT data for nearly 3 months (from April 29th to July 24th 2019) was used and can be described as follows[1]:

- IIoT data from site related sensors: These sensors include environmental sensors such as motion, temperature, humidity, and wind at the remote site. Furthermore, they also include sensors that continuously read and monitor the cellular network signal strength data, detect asynchronous events, and monitor non-contact alternate current at the remote site. This IIoT data was collected every 2.5 minutes.
- IIoT data from RF sensors: These sensors collect important information related to VSWR in order to see how efficiently radio-frequency power is transmitted from a power source into a load, through a transmission line. Forward power and reflected power values are also collected to measure the power delivered to the load. They can also detect if an RF sensor is active or not. This IIoT data was pushed to the edge nodes every 15 minutes.

Both types of IIoT data were pre-processed at the edge nodes for data cleaning and data format conversion from JSON to CSV. However, as the VSWR data streams were being processed at the edge nodes, they were being sent to the cloud. In contrast, the site sensor data streams were accumulated for further performing edge analytics tasks.

V. Discussion of Results

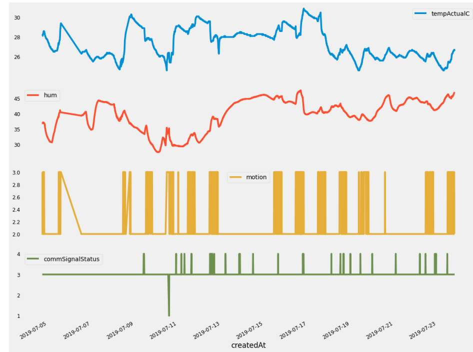
Two scenarios have been selected to illustrate the results of the self-diagnostic process based on the proposed automated data workflow. They are described in detail in the next sections.

A. Scenario 1: Self-diagnostics at the remote site

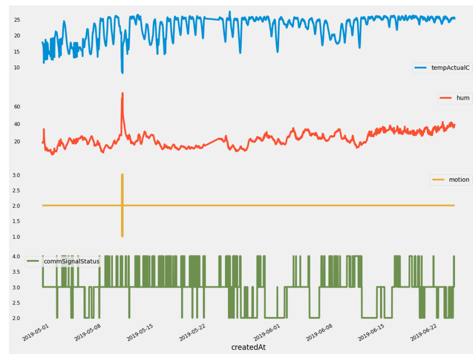
We have selected two remote sites to illustrate the self-diagnostic process at the edge using the outcomes from our automated data workflow. In this scenario, sensors such as humidity, temperature, motion, signal status, and AC Power are used to support intelligent maintenance. The goal is to achieve a self-diagnostic outcome that can reduce outages by 50% and provide pro-active services. Figure 3 illustrates two samples of the IIoT data streams being generated at two remote sites. In this case, the data streams show a higher disturbance occurrence at the site A in comparison to the site B.

For the site A, the Prophet model was applied to the time-series data generated by the temperature sensor, where linear trends were fit with daily and weekly seasonality as shown in Figure 4.

[1]See the link https://github.com/hung-cao/LMR_paper for further details of these data streams



(a) Site A



(b) Site B

Fig. 3. Real-time incoming data streams

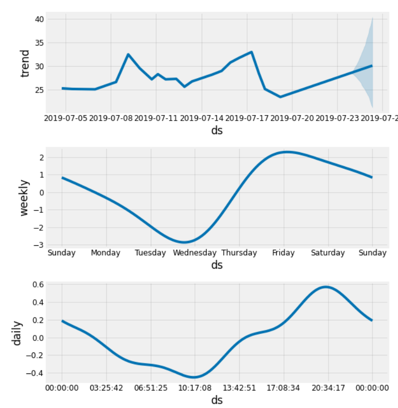


Fig. 4. Observed trends at Site A

The forecast results are shown in Figure 5, where the black line represents the observed temperature values, meanwhile the blue line represents the forecast values from the Prophet model. During the training using only 80% of the historical data, it was clear that the Prophet model had a robust fitting to high and medium observed temperature values rather than lower values. The uncertainty for the forecast trend was based on the assumption that the future will see the same average frequency and magnitude of rate changes that were seen in the historical temperature data.

In contrast, the SARIMA model was selected for the site B due to the seasonal patterns exhibited by temperature trends

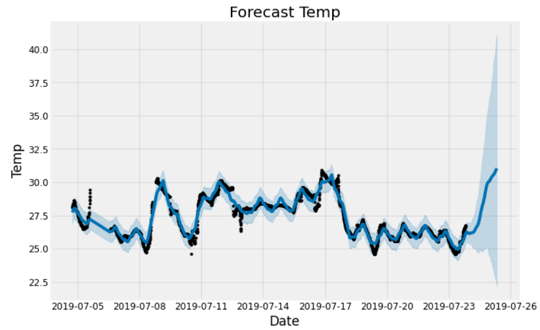


Fig. 5. Predicted temperature at the site A for a 3-day horizon forecast

observed at this site. This model is based on the assumption that a future value of a variable is assumed to be a linear function of several past observations and random error. Figure 6 shows how this linear regression model has used its own lags as predictors when the predictors were not correlated and were independent of each other.

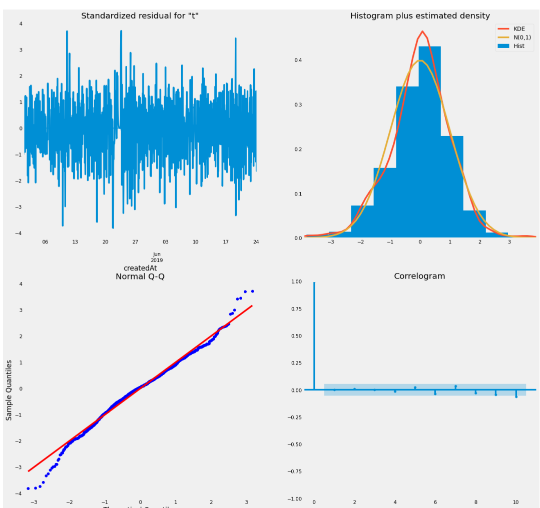


Fig. 6. Plot diagnostics results of the SARIMA model at Site B

The top left plot shows the residuals found over time were random, assuring that the model has found the trend and seasonality in the data by removing the noise during the training. In the top-right plot, the red KDE line follows closely with the N(0,1) line, indicating that the residuals were normally distributed. This line is the standard notation for a normal distribution with a mean of 0 and a standard deviation of 1. In the bottom left qq-plot, the ordered distribution of residuals (blue dots) followed the linear trend (red line) of the samples taken from a standard normal distribution with N(0, 1). Finally, the correlogram on the bottom right shows that the time series residuals had a low correlation with the lagged versions of itself.

Figure 7 shows the predicted results for the remote site B. The blue line represents the historical data used for the training of the model, the orange line represents the predicted values, and finally, the grey line represents the residuals. This machine learning model has shown flexibility to describe the behavior of actual non-stationary and seasonal time series, which makes it ideal for forecasting temperature at the site B.

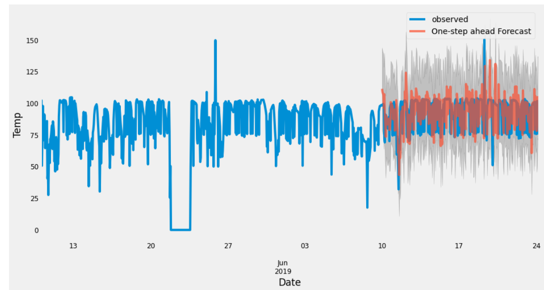


Fig. 7. Future forecast results at Site B

B. Scenario 2: Self-diagnostics of the LMR network level

In this scenario, the aim is to integrate the continuous VSWR monitoring at the LMR network level. First, the multi-correlation matrices were computed to help users to identify the main relationships between observations over time. Figure 8 shows an example of the coefficients of a multiple correlation matrix, showing a high correlation between the VSWR and FwdPwr signals. In contrast, the RefPwr signal shows a low correlation with the other signals.

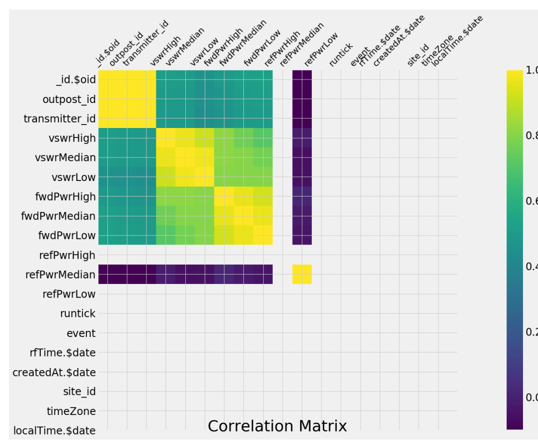


Fig. 8. Multiple correlation matrix results.

The accumulated IIoT data from all remote sites were also used as the input to the Random Forest model. The labels FwdPwr (critical event) and No Event were used in the prediction. The aim is to alert technicians of critical events to help mitigate future damage and maintain the performance of the system as a whole. The accuracy of the prediction is shown in Table I.

It was also interesting from the self-diagnostic perspective to identify the sensor observations that were significant in predicting the occurrence of critical events happening at the network level. Figure 9 shows the importance feature scores obtained from the RF sensor data from all remote sites that

TABLE I
ACCURACY OF PREDICTED RESULTS

	Precision	Recall	F1 Score
FwdPwr	0.72	0.93	0.81
No event	0.94	0.77	0.85
Accuracy			**0.83**
Macro Average	0.83	0.85	**0.83**
Weighted Average	0.86	0.83	**0.84**

had an effect on the diagnostics process. The Reflected Power, Forward Power, and and the location of a site were the most important observations to diagnose critical events at the remote sites.

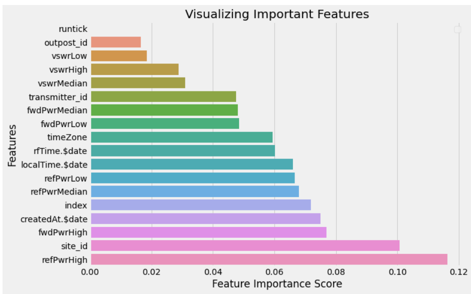


Fig. 9. Feature Importance Scores found in the RF sensor data

VI. Conclusions and Future Research Work

We have described the preliminary results obtained from exploring the integration of IIoT sensors with edge and cloud resources, having as the main goal to develop self-diagnostics of Land Mobile Sytems. The platform was implemented to generate an intelligent management based solution for monitoring RF sites.

The implemented automated data workflow was capable of harmonizing three machine learning models that collaborate for an automated self-diagnostics of LMR systems. The SARIMA, Prohet and Random Forest models have produced accurate predictions using real-world IIoT data streams generated from seven remote sites.

The results are promising, paving the way for further investigating other machine learning models to diagnose problems that can lead to failures in critical communication and operation, indicating that a component is damaged or a weather-related incident has occurred. We will also work towards a centralized scheduler which will dynamically manage the load balance during the entire IIoT stream data lifecycle of the automated data workflow in order to avoid running out of storage and processing capacity at the edge nodes.

Acknowledgments

This project was funded by the MITACS Accelerate Funding Program under the Investigations and Analysis of Industrial Internet of Things Ecosystems Project. We would like to thank Rimot.io for providing the real-world data used in our research.

References

[1] T. Qiu, J. Chi, X. Zhou, Z. Ning, M. Atiquzzaman, and D. O. Wu, "Edge Computing in Industrial Internet of Things: Architecture, Advances and Challenges," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 4, pp. 2462–2488, 2020.

[2] P. Bellavista, R. D. Penna, L. Foschini, and D. Scotece, "Machine Learning for Predictive Diagnostics at the Edge: an IIoT Practical Example," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, pp. 1–7, IEEE, jun 2020.

[3] H. Cao, *Developing an analytics everywhere framework for the Internet of Things in smart city applications*. PhD thesis, University of New Brunswick., 2019.

[4] R. Bianchini, M. Fontoura, E. Cortez, A. Bonde, A. Muzio, A.-M. Constantin, T. Moscibroda, G. Magalhaes, G. Bablani, and M. Russinovich, "Toward ml-centric cloud platforms," *Communications of the ACM*, vol. 63, no. 2, pp. 50–59, 2020.

[5] Y. Wu, "Cloud-Edge Orchestration for the Internet-of-Things: Architecture and AI-Powered Data Processing," *IEEE Internet of Things Journal*, vol. 4662, no. c, pp. 1–1, 2021.

[6] F. Foukalas and A. Tziouvaras, "Edge AI for Industrial IoT Applications," *IEEE Industrial Electronics Magazine*, pp. 0–0, 2021.

[7] Q. Liang, P. Shenoy, and D. Irwin, "AI on the Edge: Characterizing AI-based IoT Applications Using Specialized Edge Architectures," in *2020 IEEE International Symposium on Workload Characterization (IISWC)*, pp. 145–156, IEEE, oct 2020.

[8] H. Zheng, A. R. Paiva, and C. S. Gurciullo, "Advancing from predictive maintenance to intelligent maintenance with ai and iiot," *arXiv preprint arXiv:2009.00351*, 2020.

[9] J.-R. Ruiz-Sarmiento, J. Monroy, F.-A. Moreno, C. Galindo, J.-M. Bonelo, and J. Gonzalez-Jimenez, "A predictive model for the maintenance of industrial machinery in the context of industry 4.0," *Engineering Applications of Artificial Intelligence*, vol. 87, p. 103289, 2020.

[10] O. G. Manzanilla-Salazar, F. Malandra, H. Mellah, C. Wetté, and B. Sansò, "A machine learning framework for sleeping cell detection in a smart-city iot telecommunications infrastructure," *IEEE Access*, vol. 8, pp. 61213–61225, 2020.

[11] T. Rausch and S. Dustdar, "Edge Intelligence: The Convergence of Humans, Things, and AI," in *2019 IEEE International Conference on Cloud Engineering (IC2E)*, pp. 86–96, IEEE, jun 2019.

[12] Z. Zhou, X. Chen, E. Li, L. Zeng, K. Luo, and J. Zhang, "Edge Intelligence: Paving the Last Mile of Artificial Intelligence With Edge Computing," *Proceedings of the IEEE*, vol. 107, pp. 1738–1762, aug 2019.

[13] A. J. García, M. Toril, P. Oliver, S. Luna-Ramírez, and M. Ortiz, "Automatic alarm prioritization by data mining for fault management in cellular networks," *Expert Systems with Applications*, p. 113526, 2020.

[14] A. Kumbhar, F. Koohifar, I. Güvenç, and B. Mueller, "A survey on legacy and emerging technologies for public safety communications," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 1, pp. 97–124, 2016.

[15] L. Feltrin, G. Tsoukaneri, M. Condoluci, C. Buratti, T. Mahmoodi, M. Dohler, and R. Verdone, "Narrowband iot: A survey on downlink and uplink perspectives," *IEEE Wireless Communications*, vol. 26, no. 1, pp. 78–86, 2019.

[16] S. Parida and S. K. Das, "G2a communication systems: A survey on evolving enabling technologies, technical challenges and research directions," in *2020 International Conference on Emerging Trends in Information Technology and Engineering*, pp. 1–6, IEEE, 2020.

[17] J. Luomala and I. Hakala, "Effects of temperature and humidity on radio signal strength in outdoor wireless sensor networks," in *2015 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pp. 1247–1255, IEEE, 2015.

[18] A. Wileman and S. Perinpanayagam, "Failure mechanisms of radar and rf systems," *Procedia CIRP*, vol. 11, pp. 56–61, 2013.

[19] H. Cao and M. Wachowicz, "An edge-fog-cloud architecture of streaming analytics for internet of things applications," *Sensors*, vol. 19, no. 16, p. 3594, 2019.

[20] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[21] S. J. Taylor and B. Letham, "Forecasting at scale," *The American Statistician*, vol. 72, no. 1, pp. 37–45, 2018.

[22] L. Breiman, "Random forest," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.