Efficient Real-Time Decision Making in IoT

Subjects: Computer Science, Artificial Intelligence Contributor: Kyoung-Don Kang

Efficient Real-Time Decision Making in IoT(the Internet of Things) is about using real-time sensor data, using fresh sensor data that represent the current real-world status to minimize.

Internet of Things	real-time decision making	timing and data freshness constraints
predicate evaluation	real-time scheduling	sensor data analytics

1. Introduction

The Internet of Things (IoT) envisions to enable many innovative applications, such as smart transportation, healthcare, and emergency response ^{[1][2][3][4]}. In IoT, timely decision-making using real-time sensor data is essential. For example, drivers in New York, Chicago, and Philadelphia lost 102, 104, and 90 h on average in 2021 despite a -27% to -37% drop since 2019 due to the reduced traffic during the COVID-19 pandemic ^[5]. Real-time decision-making for efficient traffic routing based on sensor data streams from roadside sensors (if any) or dashboard-mounted smartphones can greatly alleviate traffic congestion ^{[6][7]}. Also, an agent for real-time decision-making needs to find an available route among several alternative routes to send an ambulance to a patient when some of them are unavailable because of construction, social/political event, or disaster ^[6]. As another example, patients in an emergency department or intensive care unit with abnormal shock index values have much higher mortality rates ^[9] and higher risks to suffer from hyperlactatemia ^[10] and cardiac arrest ^[11]. Thus, making real-time triage decisions based on the analysis of physiological sensor data from wearable devices within decision-making deadlines is desirable.

In the presence of alternative actions, a real-time decision-maker needs to select one of them that is currently feasible within decision-making deadlines using fresh sensor data that represent the current real-world status to minimize, for example, traffic congestion or mortality in an emergency department. Furthermore, a real-time decision-maker should require IoT devices to provide minimal sensor data necessary for decision-making only to avoid possible network congestion and significant energy consumption in IoT devices for transmitting redundant sensor data wirelessly. Logic predicates, also called Boolean queries, can effectively evaluate alternative courses of action in IoT ^{[8][12][13]}. For example, an ambulance may try to find an available route among several alternative routes to a patient where some of them are unavailable due to construction, a social/political event, or disaster. Let us suppose that there are two alternative routes, *A-B-C* and *D-E-F*, which are expressed as $(A \land B \land C) \lor (D \land E \land F)$

where Λ and ν represent the logical AND and OR operator, respectively. If road segment A of the route A-B-C is unavailable, the data indicating the status of the road segment B or C does not have to be retrieved from the sensors and analyzed for real-time decision making, but can be short-circuited to reduce the latency and resource consumption ^{[8][12][13]}. Similarly, effective treatment can be selected among alternative treatments by efficiently analyzing the logic predicate in a timely manner using fresh data that represent the current status of the patients in an emergency department or intensive care unit. Emergency vehicle routing and triage/treatment as running examples for real-time decision support were used.

2. Pull Model and Data Freshness

In ^{[8][12][13]}, the real-time decision-maker employs the pull model, in which it pulls (retrieves) data from sensors over a single wireless connection upon an event of interest to analyze, for example, the availabilities of alternative routes. To make decisions based on fresh data representing the current real-world status, the real-time decisionmaker in ^{[8][12][13]} periodically retrieves sensor data based on their validity intervals—the notion originated in realtime databases (RTDBs) ^{[14][15]}. A sensor data object is fresh within its predefined validity interval; however, the real-time decision-making system considers it stale after the validity interval expires. By doing this, the system ensures that it makes real-time decisions based on fresh data representing the current real-world status.

Although managing the data freshness (data temporal consistency) via validity intervals could be effective in RTDBs with its own sensors, it can be too strict and expensive in IoT. First, sensor data, such as indoor temperature readings, may not normally change significantly in a short time period. Thus, the data could be still valid even after its validity interval expires. Periodic updates even in the absence of any noteworthy change may incur unnecessary consumption of the precious wireless bandwidth and energy in IoT devices without enhancing real-time decision making.

Moreover, if a decision-making task uses several sensor data with different validity intervals, the real-time decisionmaker may have to retrieve the data repeatedly to ensure that all of them are fresh until the decision task completes. The system also should undo and redo any analysis performed using stale data. Hu et al. ^[8] investigate this problem for a single decision task that uses sensor data pulled over a wireless connection. Their algorithm, called the LVF (Least Volatile First), pulls the data with the longest validity interval first. By doing this, LVF minimizes repeated data retrievals for one decision task that pulls sensor data with different validity intervals.

Kim et al. ^{[16][17]} extend LVF to schedule multiple real-time decision-making tasks with potentially different deadlines using fresh data. Their algorithm, called EDEF-LVF (Earliest Deadline or Expiration First-Least Volatile First), schedules the real-time task with the earliest deadline or the shortest time to the expiration of the validity interval first. Within each task, the least volatile sensor data is retrieved first, similar to ^[8]. They assume that there is a single bottleneck resource, such as a wireless connection, and real-time tasks do not share any data. Under the assumptions, EDEF-LVF is optimal in the sense that it can schedule real-time decision-making tasks to meet their deadlines and data validity constraints if such a schedule exists. In addition, Kim et al. ^[18] devise several

suboptimal heuristics to efficiently schedule real-time decision-making tasks that share sensor data with each other.

However, none of these approaches ^{[8][12][13][16][17][18]} is free of repeated sensor data retrievals and re-executions of data analytics upon expiration of any validity interval. As a result, the precious wireless bandwidth and energy of IoT devices can be wasted and many deadlines for real-time decision making can be missed. In an extreme case, it may become impossible to run a task using fresh data as per the strict notion of validity intervals. For the sake of simplicity, let us suppose that there is only one real-time task that needs to pull data A and B from sensors deployed in a wide area over a wireless connection with relatively low bandwidth. Using LVF, the task pulls A with the longer validity interval first. When it tries to pull B, however, the wireless connection may become unstable. As a result, the sensor should retransmit B several times. Meanwhile, the validity interval of A expires. By the time a new version of A arrives, the validity interval of B may expire, and the whole process may repeat indefinitely. Finally, the system misses the deadline of the real-time decision-making task, wasting the bandwidth and energy. If there are multiple real-time decision-making tasks in the system, the problem may become worse. In addition to the situations described above, a real-time task can be preempted by a higher priority task, such as a task with an earlier deadline under the EDF (Earliest Deadline First) scheduling algorithm. When all higher priority tasks are completed, the preempted task may have to pull certain sensor data again, if their validity intervals have expired already.

The root cause of the problem is using the rigid freshness requirements based on data validity intervals. Surprisingly little work has been done to address this critical issue for cost-efficient real-time decision-making in IoT. A viable way to address the problem is the adaptive updated policy based on flexible validity intervals [19][20][21]. Instead of using fixed validity intervals, the validity intervals of sensor data are dynamically adapted based on their access to update ratio in RTDBs such that the validity intervals of the data updated frequently but accessed infrequently are extended, if necessary, to reduce update workloads under overload [19][20][21]. The notion of flexible validity intervals can be extended to efficiently manage the data freshness for real-time decision-making in IoT. Instead of requiring the real-time decision-maker to pull data from IoT devices, sensors start to push data into the decision-maker when they detect an event of interest, e.g., a moving object in surveillance or traffic congestion in transportation management. After sending the first sensor readings to the decision-maker upon an event, the sensors only send new data if they differ from the previous version by more than the specified threshold. They periodically send a heartbeat message to the real-time decision-maker to indicate that they are still alive and monitoring the event of interest, even though they have not transferred new data to the decision-maker due to little changes. When the decision-maker receives a heartbeat message from a device, it extends the flexible validity interval to the next heartbeat period. On the other hand, when the sensor data changes by more than the threshold, the device sends new data to the decision-maker. By doing this, the decision-maker can avoid significantly wasting the network bandwidth, computational resources, and energy to repeatedly pull sensor data from IoT devices due to the expirations of strict validity intervals even when the actual data values hardly change.

3. Sensor Data Analytics via Machine Learning for Real-Time Decision Making

Machine learning is effective to analyze sensor data. For example, the availability of a bridge or a road segment can be analyzed by a CNN (Convolutional Neural Network) ^[22], which is very effective for image processing and computer vision. Thus, machine learning is useful to evaluate the literals of a DNF predicate for real-time decision support. Sequence models are also useful for real-time decision support in IoT. For example, Markov decision processes ^[23] and partially observable Markov decision processes ^[23] are leveraged for near real-time health monitoring, treatments, and interventions in various medical applications ^[24]. More recently, long-short term memory (LSTM), which is an artificial recurrent neural network (RNN) architecture effective for sequence modeling, has been applied to detect emotion ^[25], to predict cardiovascular disease risk factors ^[26], and to predict healthcare trajectories ^[27]. Machine learning is applied to smart homes ^{[28][29][30]}. Guo et al. have designed a graph CNN optimized for traffic predictions ^[31]. In ^{[32][33][34][35]}, GRNN (General Regression Neural Network) and GRNN-SGTM (GRNN-Successive Geometric Transformation Model) are used to recover missing IoT data, respectively. Wang et al. ^[36] devise a GRNN and a multivariate polynomial regression model to estimate unmeasurable water quality parameters from measurable parameters. In addition, Tien ^[37] gives a high-level view of IoT, (near) real-time decision making, and artificial intelligence instead of focusing on technical approaches for real-time decision support in IoT.

Although it is effective for data analytics, machine learning is resource hungry. A complex machine learning model often consumes a significant amount of memory and computational resources, such as CPU cycles and GPU (Graphics Processing Unit) thread blocks, that may not be available in IoT devices with relatively little resources. Thus, in IoT devices, it is hard to run sophisticated prediction models in a timely manner to meet stringent timing constraints. A naive approach to address this challenge is transferring all sensor data from IoT devices to the cloud with virtually infinite resources. However, this approach is unsustainable, as described before. Therefore, the question of "where to analyze sensor data?" is as important as the question of "how to analyze them efficiently?". Ultimately, it is desirable to optimize the tradeoff between the timeliness and bandwidth conservation of real-time data analytics near IoT devices vs. the scalability of data analytics in the cloud. In this regard, the relative advantages and disadvantages of sensor data analytics in IoT devices were summarized, at the network edge, and the cloud in **Table 1**, and discuss them in the following.

Table 1.	Comparisons of	real-time	decision-making	at different	places.
----------	----------------	-----------	-----------------	--------------	---------

	Cloud	Edge	IoT End-Devices
Resources	High	Medium	Low
Latency	High	Medium	Low
Bandwidth consumption	High	Medium	Low
Energy consumption	High	Medium	Low

	Cloud	Edge	IoT End-Devices
Geographic coverage	High	Medium	Low

The first category is centralized analytics of sensor data in the cloud. A cloud has abundant computational resources and provides rich functionalities, such as very deep learning with many layers and training complex machine learning models using big datasets. Another advantage of real-time analytics in the cloud is that it can support real-time data analytics in a more global geographic area. However, centralized data analytics for real-time decision making in the cloud has several serious drawbacks:

- It requires IoT devices to transmit all sensor data to the cloud for analytics, incurring long, unpredictable latency, and many deadlines miss in real-time decision making. (The Internet backbone latency is relatively long and varies significantly from tens to hundreds of milliseconds ^[38].) Tardy decisions may lead to undesirable results, such as severe traffic congestion or chaos in an emergency department.
- Such a naive approach may saturate the core network with the limited bandwidth as the number of sensors and IoT devices is increasing rapidly ^{[39][40]}. It may substantially impair the performance, scalability, and availability of the Internet. Thus, centralized analytics of sensor data in IoT is unsustainable.
- In addition, IoT devices may consume a lot of precious energy and wireless bandwidth to transfer all their sensor data to the cloud for centralized data analytics in the cloud. Typically, IoT devices communicate wirelessly for the ease of deployment in a distributed area. Wireless networking consumes a significant fraction of the energy in an IoT device ^{[41][42]}. Wireless IoT networks, such as LPWAN (Low-Power Wide-Area Network) ^{[43][44]}, often have stringent bandwidth constraints.

To address these problems, a system designer can consider another extreme—on-device analytics, where all data analytics occur in IoT end-devices. By supporting distributed analytics of sensor data, this approach can significantly reduce the latency and bandwidth consumption compared to the centralized analytics in the cloud. However, this approach also has several challenges:

- It is challenging to meet stringent timing constraints for real-time data analytics and decision support due to the stringent resource and energy constraints of IoT devices.
- IoT devices with limited resources may not be able to support sophisticated machine learning models or extensive model training. Instead, they typically use simplified models trained in the cloud to analyze local sensor data in a timely fashion ^{[45][46]}; however, the stripped-down model may suffer from lower predictive performance.
- Each IoT device is likely to have a relatively myopic view of the specific area it is monitoring only without a global view necessary to optimize, for example, the overall traffic flow in a city.

By analyzing sensor data at the network edge near IoT devices and sensors, edge analytics ^{[47][48][49][50]} aims to integrate the advantages of cloud and on-device analytics, while mitigating their shortcomings. Edge computing brings more computational resources at the network edge near data sources. It can be supported at different places. First, IoT end devices can preprocess sensor data and perform lightweight analytics ^{[51][52][53]}. Second, an edge node, such as an IoT gateway, access point, cellular base station, or software-defined routers/switches, can collect and analyze data from IoT devices ^{[52][54][55][56]}. Edge servers deployed at the network edge can be leveraged for more sophisticated data analytics ^[49].

Thus, edge analytics for real-time decision support can be performed in a hierarchical and event-driven manner. An IoT device preprocesses sensor data and performs a lightweight analysis of them to detect any event of interest while filtering irrelevant data out. An IoT gateway, if any, further analyzes data received from the devices connected to the gateway. It forwards important information, if any, to one or more relevant edge servers. For example, traffic cameras can send images to the edge server in charge of monitoring traffic flows in a specific area of a big city. Li et al. [57], on-camera filtering is performed for efficient real-time video analytics. In [58], an IoT camera analyzes the traffic flow using a low-resolution image and the edge server also analyzes the image, identifies an important part of the image (if any) in terms of data analytics, and requests an important part in high resolution from the device. IoT devices in a smart building can transfer their sensor readings to the IoT gateway on the same floor for efficient HVAC (Heating, Ventilation, and Air Conditioning). In these examples, IoT devices can do relatively simple data analytics to drop redundant or low-quality data, such as blurry images ^[57]. Edge servers analyze real-time sensor data from multiple IoT devices/gateways to derive a more comprehensive view of the real-world status essential for real-time decision making. They can also communicate with each other to exchange information for a global view of real-world situations, such as the overall traffic flow in a city or hurricane paths in a nation. Edge computing and analytics are a booming area of research and industrial adoption due to their significant potential. Leveraging emerging edge computing for cost-efficient real-time decision support is in an early stage of research with ample room to grow.

Overall, efficient evaluations of predicates are important across IoT devices, gateways, and edge/cloud servers to significantly reduce latency as well as energy and bandwidth consumption. The efficiency of real-time decision-making can also be further enhanced by effectively exploiting cloud, on-device, and edge analytics frameworks and synthesizing them to optimize timing, predictive performance, bandwidth, and other resource consumption. Relatively little work, however, has been done for real-time decision-making in IoT from this holistic, overarching perspective.

Another promising direction for real-time analytics of sensor data on IoT devices is model compression ^{[59][60][61][62]}. The key idea of model compression is to compact a machine learning model to minimize the resource requirements without significantly reducing the predictive performance of the compressed model. Especially, deep learning has been very successful and outperformed other machine learning techniques in killer applications, such as computer vision and natural language processing. DNNs (Deep Neural Networks) with many hidden layers and parameters, however, consume a lot of memory, computation time, and energy. They are too big and too expensive to learn on low-end IoT devices. The motivation for model compression is to significantly reduce the memory

consumption and computational complexity of DNNs without significantly comprising their accuracy. Effective approaches for model compression include (1) compact models, (2) tensor decomposition, (3) data quantization, and (4) network sparsification ^[59]:

- Compact CNNs (Convolutional Neural Networks) are created by leveraging the spatial correlation within a convolutional layer to convolve feature maps with multiple weight kernels (Compact RNNs (Recurrent Neural Networks) for sequence data analysis has also received significant attention from researchers ^[59].). They also leverage the intra-layer and inter-layer channel correlations to aggregate feature maps with different topologies. In addition, network architecture search (NAS) aims to automatically optimize the DNN architecture.
- Tensor/matrix operations are the basic computation in neural networks. Thus, compressing tensors, typically via matrix decomposition, is an effective way to shrink and accelerate DNNs.
- Data quantization decreases the bit width of the data that flow through a DNN model to reduce the model size and save memory while simplifying the operations for computational acceleration.
- Network sparsification attempts to make neural networks sparse, via weight pruning and neuron pruning, instead of simplifying the arithmetic via data quantization.

Model compression in hardware, as well as hardware and algorithm co-design, is also effective. Good surveys are given in [59][60].

References

- 1. Lee, I.; Lee, K. The Internet of Things (IoT): Applications, investments, and challenges for enterprises. Bus. Horiz. 2015, 58, 431–440.
- 2. Sisinni, E.; Saifullah, A.; Han, S.; Jennehag, U.; Gidlund, M. Industrial internet of things: Challenges, opportunities, and directions. IEEE Trans. Ind. Inform. 2018, 14, 4724–4734.
- Stoyanova, M.; Nikoloudakis, Y.; Panagiotakis, S.; Pallis, E.; Markakis, E.K. A survey on the internet of things (IoT) forensics: Challenges, approaches, and open issues. IEEE Commun. Surv. Tutor. 2020, 22, 1191–1221.
- Fortino, G.; Savaglio, C.; Spezzano, G.; Zhou, M. Internet of Things as System of Systems: A Review of Methodologies, Frameworks, Platforms, and Tools. IEEE Trans. Syst. Man Cybern. Syst. 2020, 51, 223–236.
- 5. INRIX 2021 Global Traffic Scorecard. Available online: https://inrix.com/scorecard/ (accessed on 10 January 2022).
- 6. Ji, B.; Zhang, X.; Mumtaz, S.; Han, C.; Li, C.; Wen, H.; Wang, D. Survey on the internet of vehicles: Network architectures and applications. IEEE Commun. Stand. Mag. 2020, 4, 34–41.

- 7. Yang, F.; Wang, S.; Li, J.; Liu, Z.; Sun, Q. An overview of Internet of vehicles. China Commun. 2014, 11, 1–15.
- Hu, S.; Yao, S.; Jin, H.; Zhao, Y.; Hu, Y.; Liu, X.; Naghibolhosseini, N.; Li, S.; Kapoor, A.; Dron, W.; et al. Data Acquisition for Real-Time Decision-Making under Freshness Constraints. In Proceedings of the IEEE Real-Time Systems Symposium, San Antonio, TX, USA, 1–4 December 2015.
- Kim, S.Y.; Hong, K.J.; Shin, S.D.; Ro, Y.S.; Ahn, K.O.; Kim, Y.J.; Lee, E.J. Validation of the Shock Index, Modified Shock Index, and Age Shock Index for Predicting Mortality of Geriatric Trauma Patients in Emergency Departments. J. Korean Med. Sci. 2016, 31, 2026–2032.
- Berger, T.; Green, J.; Horeczko, T.; Hagar, Y.; Garg, N.; Suarez, A.; Panacek, E.; Shapiro, N. Shock Index and Early Recognition of Sepsis in the Emergency Department: Pilot Study. West. J. Emerg. Med. 2013, XIV, 168–174.
- 11. Kennedy, C.E.; Turley, J.P. Time series analysis as input for clinical predictive modeling: Modeling cardiac arrest in a pediatric ICU. Theor. Biol. Med. Model. 2011, 8.
- 12. Abdelzaher, T.F.; Amin, M.T.A.; Bar-Noy, A.; Dron, W.; Govindan, R.; Hobbs, R.L.; Hu, S.; Kim, J.; Lee, J.; Marcus, K.; et al. Decision-Driven Execution: A Distributed Resource Management Paradigm for the Age of IoT. In Proceedings of the IEEE International Conference on Distributed Computing Systems, Atlanta, GA, USA, 5–8 June 2017.
- Lee, J.; Marcus, K.; Abdelzaher, T.; Amin, M.T.A.; Bar-Noy, A.; Dron, W.; Govindan, R.; Hobbs, R.; Hu, S.; Kim, J.-E.; et al. Athena: Towards Decision-centric Anticipatory Sensor Information Delivery. J. Sens. Actuator Netw. 2018, 7, 5.
- 14. Ramamritham, K.; Son, S.H.; DiPippo, L. Real-Time Databases and Data Services. Real-Time Syst. 2004, 28, 179–215.
- 15. Ramamritham, K. Real-Time Databases. Int. J. Distrib. Parallel Databases 1993, 1. Available online: https://link.springer.com/article/10.1007/BF01264051 (accessed on 10 January 2022).
- 16. Kim, J.E.; Abdelzaher, T.F.; Sha, L.; Bar-Noy, A.; Hobbs, R.L.; Dron, W. On Maximizing Quality of Information for the Internet of Things: A Real-Time Scheduling Perspective (Invited Paper). In Proceedings of the IEEE International Conference on Embedded and Real-Time Computing Systems and Applications, Daegu, Korea, 17–19 August 2016; pp. 202–211.
- 17. Kim, J.E.; Abdelzaher, T.F.; Sha, L.; Bar-Noy, A.; Hobbs, R. Sporadic Decision-centric Data Scheduling with Normally-off Sensors. In Proceedings of the IEEE Real-Time Systems Symposium, Porto, Portugal, 29 November–2 December 2016.
- 18. Kim, J.; Abdelzaher, T.F.; Sha, L.; Bar-Noy, A.; Hobbs, R.L.; Dron, W. Decision-driven scheduling. Real-Time Syst. 2019, 55, 514–551.

- 19. Kang, K.D.; Son, S.H.; Stankovic, J.A.; Abdelzaher, T.F. A QoS-Sensitive Approach for Timeliness and Freshness Guarantees in Real-Time Databases. In Proceedings of the Euromicro Conference on Real-Time Systems, Vienna, Austria, 19–21 June 2002.
- 20. Kang, K.D.; Son, S.; Stankovic, J.A. Managing Deadline Miss Ratio and Sensor Data Freshness in Real-Time Databases. IEEE Trans. Knowl. Data Eng. 2004, 16, 1200–1216.
- Kang, K.D.; Oh, J.; Son, S.H. Chronos: Feedback Control of a Real Database System Performance. In Proceedings of the IEEE Real-Time Systems Symposium, Tucson, AZ, USA, 3–6 December 2007.
- 22. Goodfellow, I.; Bengio, Y.; Courville, A. Deep Learning; MIT Press: Cambridge, MA, USA, 2016; Available online: http://www.deeplearningbook.org (accessed on 10 January 2022).
- 23. Bertsekas, D.P. Dynamic Programming and Optimal Control, 3rd ed.; Athena Scientific: Belmont, MA, USA, 2005; Volume I.
- Zois, D.S. Sequential decision-making in healthcare IoT: Real-time health monitoring, treatments and interventions. In Proceedings of the 2016 IEEE 3rd World Forum on Internet of Things (WF-IoT), Reston, VA, USA, 12–14 December 2016; pp. 24–29.
- 25. Awais, M.; Raza, M.; Singh, N.; Bashir, K.; Manzoor, U.; ul Islam, S.; Rodrigues, J.J. LSTM based Emotion Detection using Physiological Signals: IoT framework for Healthcare and Distance Learning in COVID-19. IEEE Internet Things J. 2020, 8, 16863–16871.
- 26. Islam, M.S.; Umran, H.M.; Umran, S.M.; Karim, M. Intelligent Healthcare Platform: Cardiovascular Disease Risk Factors Prediction Using Attention Module Based LSTM. In Proceedings of the International Conference on Artificial Intelligence and Big Data (ICAIBD), Chengdu, China, 25–28 May 2019; pp. 167–175.
- 27. Pham, T.; Tran, T.; Phung, D.; Venkatesh, S. Predicting healthcare trajectories from medical records: A deep learning approach. J. Biomed. Inform. 2017, 69, 218–229.
- 28. Khan, N.S.; Ghani, S.; Haider, S. Real-Time Analysis of a Sensor's Data for Automated Decision Making in an IoT-Based Smart Home. Sensors 2018, 18, 1711.
- 29. Machorro-Cano, I.; Alor-Hernández, G.; Paredes-Valverde, M.A.; Rodríguez-Mazahua, L.; Sánchez-Cervantes, J.L.; Olmedo-Aguirre, J.O. HEMS-IoT: A big data and machine learningbased smart home system for energy saving. Energies 2020, 13, 1097.
- 30. Rashidi, P.; Cook, D.J. Keeping the resident in the loop: Adapting the smart home to the user. IEEE Trans. Syst. Man Cybern.-Part Syst. Hum. 2009, 39, 949–959.
- Guo, K.; Hu, Y.; Qian, Z.; Liu, H.; Zhang, K.; Sun, Y.; Gao, J.; Yin, B. Optimized graph convolution recurrent neural network for traffic prediction. IEEE Trans. Intell. Transp. Syst. 2020, 22, 1138– 1149.

- Izonin, I.; Kryvinska, N.; Vitynskyi, P.; Tkachenko, R.; Zub, K. GRNN approach towards missing data recovery between IoT systems. In Proceedings of the International Conference on Intelligent Networking and Collaborative Systems, Oita, Japan, 5–7 September 2019; Springer: Berlin/Heidelberg, Germany, 2019; pp. 445–453.
- Tkachenko, R.; Izonin, I.; Dronyuk, I.; Logoyda, M.; Tkachenko, P. Recovery of missing sensor data with GRNN-based cascade scheme. Int. J. Sens. Wirel. Commun. Control 2021, 11, 531– 541.
- 34. Tkachenko, R.; Izonin, I.; Kryvinska, N.; Dronyuk, I.; Zub, K. An approach towards increasing prediction accuracy for the recovery of missing IoT data based on the GRNN-SGTM ensemble. Sensors 2020, 20, 2625.
- 35. Izonin, I.; Tkachenko, R.; Verhun, V.; Zub, K. An approach towards missing data management using improved GRNN-SGTM ensemble method. Eng. Sci. Technol. Int. J. 2021, 24, 749–759.
- 36. Wang, Y.; Ho, I.W.H.; Chen, Y.; Wang, Y.; Lin, Y. Real-time Water Quality Monitoring and Estimation in AloT for Freshwater Biodiversity Conservation. IEEE Internet Things J. 2021.
- 37. Tien, J.M. Internet of Things, Real-Time Decision Making, and Artificial Intelligence. Ann. Data Sci. 2017, 4, 149–178.
- 38. Latency Across Cloud Backbones Varies Significantly. Available online: https://www.sd-wanexperts.com/blog/latency-across-cloud-backbones-varies-significantly/ (accessed on 10 January 2022).
- 39. State of IoT 2021: Number of Connected IoT Devices Growing 9% to 12.3 Billion Globally, Cellular IoT Now Surpassing 2 Billion. Available online: https://iot-analytics.com/number-connected-iot-devices/ (accessed on 10 January 2022).
- Al-Garadi, M.A.; Mohamed, A.; Al-Ali, A.K.; Du, X.; Ali, I.; Guizani, M. A Survey of Machine and Deep Learning Methods for Internet of Things (IoT) Security. IEEE Commun. Surv. Tutor. 2020, 22, 1646–1685.
- 41. Martinez, B.; Monton, M.; Vilajosana, I.; Prades, J.D. The power of models: Modeling power consumption for IoT devices. IEEE Sens. J. 2015, 15, 5777–5789.
- 42. Min, M.; Xiao, L.; Chen, Y.; Cheng, P.; Wu, D.; Zhuang, W. Learning-based computation offloading for IoT devices with energy harvesting. IEEE Trans. Veh. Technol. 2019, 68, 1930–1941.
- 43. Mekki, K.; Bajic, E.; Chaxel, F.; Meyer, F. A comparative study of LPWAN technologies for largescale IoT deployment. ICT Express 2019, 5, 1–7.
- 44. Lavric, A.; Popa, V. Internet of things and LoRa[™] low-power wide-area networks: A survey. In Proceedings of the 2017 International Symposium on Signals, Circuits and Systems (ISSCS), Iasi, Romania, 13–14 July 2017; pp. 1–5.

- 45. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. ImageNet Large Scale Visual Recognition Challenge. Int. J. Comput. Vis. 2015, 115, 211–252.
- 46. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Trans. Pattern Anal. Mach. Intell. 2017, 39, 1137–1149.
- 47. Satyanarayanan, M.; Simoens, P.; Xiao, Y.; Pillai, P.; Chen, Z.; Ha, K.; Hu, W.; Amos, B. Edge Analytics in the Internet of Things. IEEE Pervasive Comput. 2015, 14, 24–31.
- 48. Jedari, B.; Premsankar, G.; Illahi, G.; Di Francesco, M.; Mehrabi, A.; Ylä-Jääski, A. Video Caching, Analytics, and Delivery at the Wireless Edge: A Survey and Future Directions. IEEE Commun. Surv. Tutor. 2020, 23, 431–471.
- 49. Liu, F.; Tang, G.; Li, Y.; Cai, Z.; Zhang, X.; Zhou, T. A survey on edge computing systems and tools. Proc. IEEE 2019, 107, 1537–1562.
- 50. Liu, D.; Yan, Z.; Ding, W.; Atiquzzaman, M. A survey on secure data analytics in edge computing. IEEE Internet Things J. 2019, 6, 4946–4967.
- Mazumder, A.N.; Meng, J.; Rashid, H.A.; Kallakuri, U.; Zhang, X.; Seo, J.S.; Mohsenin, T. A Survey on the Optimization of Neural Network Accelerators for Micro-AI On-Device Inference. IEEE J. Emerg. Sel. Top. Circuits Syst. 2021, 11, 532–547.
- Sanabria-Russo, L.; Pubill, D.; Serra, J.; Verikoukis, C. IoT data analytics as a network edge service. In Proceedings of the IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Paris, France, 29 April–2 May 2019; pp. 969–970.
- 53. Hanyao, M.; Jin, Y.; Qian, Z.; Zhang, S.; Lu, S. Edge-assisted online on-device object detection for real-time video analytics. In Proceedings of the IEEE INFOCOM 2021-IEEE Conference on Computer Communications, Vancouver, BC, Canada, 10–13 May 2021; pp. 1–10.
- 54. Marjani, M.; Nasaruddin, F.; Gani, A.; Karim, A.; Hashem, I.A.T.; Siddiqa, A.; Yaqoob, I. Big IoT data analytics: Architecture, opportunities, and open research challenges. IEEE Access 2017, 5, 5247–5261.
- 55. Sharma, S.K.; Wang, X. Live data analytics with collaborative edge and cloud processing in wireless IoT networks. IEEE Access 2017, 5, 4621–4635.
- Dayalan, U.K.; Fezeu, R.A.; Varyani, N.; Salo, T.J.; Zhang, Z.L. VeerEdge: Towards an Edge-Centric IoT Gateway. In Proceedings of the 2021 IEEE/ACM 21st International Symposium on Cluster, Cloud and Internet Computing (CCGrid), Melbourne, Australia, 10–13 May 2021; pp. 690–695.

- 57. Li, Y.; Padmanabhan, A.; Zhao, P.; Wang, Y.; Xu, G.H.; Netravali, R. Reducto: On-camera filtering for resource-efficient real-time video analytics. In Proceedings of the Annual Conference of the ACM Special Interest Group on Data Communication on the Applications, Technologies, Architectures, and Protocols for Computer Communication, (Virtual Conference), New York, NY, USA, 10–14 August 2020; pp. 359–376.
- 58. Du, K.; Pervaiz, A.; Yuan, X.; Chowdhery, A.; Zhang, Q.; Hoffmann, H.; Jiang, J. Server-driven video streaming for deep learning inference. In Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the Applications, Technologies, Architectures, and Protocols for Computer Communication, (Virtual Conference), New York, NY, USA, 10–14 August 2020; pp. 557–570.
- 59. Deng, L.; Li, G.; Han, S.; Shi, L.; Xie, Y. Model compression and hardware acceleration for neural networks: A comprehensive survey. Proc. IEEE 2020, 108, 485–532.
- 60. Cheng, Y.; Wang, D.; Zhou, P.; Zhang, T. A survey of model compression and acceleration for deep neural networks. arXiv 2017, arXiv:1710.09282.
- He, Y.; Lin, J.; Liu, Z.; Wang, H.; Li, L.J.; Han, S. AMC: AutoML for model compression and acceleration on mobile devices. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 784–800.
- 62. Polino, A.; Pascanu, R.; Alistarh, D. Model compression via distillation and quantization. arXiv 2018, arXiv:1802.05668.

Retrieved from https://encyclopedia.pub/entry/history/show/45612