iJournals: International Journal of Software & Hardware Research in Engineering
ISSN:2347-4890, ijournals.in/ijshre
Volume 12 Issue 7, July 2024

The Intelligent Edge: AI and Machine Learning in Edge Computing for IoT

Author: Harish Janardhanan[1]

Independent Researcher, 101 Mount Pleasant Ave, Edison, NJ, USA [1]

E-mail: harishjan@gmail.com [1]

DOI: 10.26821/IJSHRE.12.7.2024.120706

ABSTRACT

Integrating AI and ML with EC can be viewed as a paradigm shift in the IT IoT world. Thus, this article discusses the possibilities of applying AI and ML in IoT with the help of edge computing, its advantages and disadvantages, and future trends. Edge computing has the primary benefit of reducing latency and the physical amount of data transmitted at the network level due to processing data closer to the source. This paper aims to discuss the current state of edge AI technologies, approaches, and performance indicators in carrying out IoT applications.

Keywords: Edge Computing, Internet of Things (IoT), Artificial Intelligence (AI), Machine Learning (ML), Edge Devices.

1. INTRODUCTION

Smart objects or the Internet of Things (IoT) have tremendously grown within the last decade to have billions of devices. [1] [2] [3] these devices produce tremendous amounts of data, and the current cloud paradigm fails to handle the data with optimal performance. This strain results in several issues.

- Latency: The data must go to large central computing facilities for processing and back to the device and this takes time.
- Bandwidth: Pushing large amounts of data to the cloud is expensive in terms of bandwidth.
- Scalability: Several issues related to the growth of IoT devices and data collected by the devices are obstacles to the centralized structures of the architecture.

To these challenges, Edge computing brings computation and storage closer to the source of data, hence reducing a lot of latency and usage of bandwidth.

A. Importance of Edge Computing in IoT

Edge computing enhances the performance [4] of IoT systems in several critical ways:

- Reduced Latency: Edge computing performs data computation near the data-generating source and reduces the time taken to transmit data, making real-time decisions possible.
- Bandwidth Conservation: Part of the data is transmitted to the cloud when required, thus optimizing the use of the bandwidth and the costs involved.
- Improved Reliability: Smart processing allows IoT applications to execute on local constraints to enable them to continue running when they are not connected to the cloud or when the connection is weak.

B. Role of AI and ML in Edge Computing

Edge computing is dependent on two major aspects that is AI and ML to unlock its full capability. These technologies enable edge devices to [5]:

- Process and Analyze Data Locally: This is because the AI and ML algorithms can also be hosted in the edge devices so that data collected can be analyzed, predictions made, and insights generated at the edge without the need for the data to be sent to the cloud.
- Enable Real-time Decision-making: Thus, when data is processed near the processing location, the AI and the ML models provide an instant response to changes in conditions.
- Optimize Resource Utilization: In this case, the design of the ML models can be fine-tuned to work on small and low-powered devices so that the available resources for computation and power use are conserved.

C. AI and ML Techniques Used in Edge Computing

1) Supervised Learning: Methods that are applied for activities including anomaly detection and predictive maintenance are decision trees and support vector machines.

2) Unsupervised Learning: Hawkin cluster and feature reduction assist in data grouping and pattern identification.

3) Reinforcement Learning: When used in conditions that are frequently changing, such as the context of self-driving vehicles, the edge devices must work in a manner that adapts to these changing circumstances.

D. Advantages of AI & ML at the Edge

1) Enhanced Performance: AI and ML-based local processing enhances the rates of IoT methods and boosts their performance.

2) Data Privacy: Local data processing entails lesser transmission of data to the cloud; hence, it improves data privacy.

3) Scalability: AI and ML models can be deployed multiple at the edges to support many IoT devices.

E. Challenges

1) Resource Constraints: Some of these devices are resource-constrained; therefore, the AI and ML models used need to be power efficient and require minimal storage.

2) Model Compression: Their use demands efficient algorithms to reduce the size of the AI models while at the same time not having a large impact on their performance [6].

3) Security: Probably the most sensitive area when it comes to edge computing is the security of data and models in terminal equipment.

F. AI and ML in Edge Computing: Historical Analysis

1) Initial Integration: When the approach of using AI and ML in edge [8] computing was first adopted, the algorithms used were simple and created primarily to perform simple data processing tasks. Initially, applications were related to data preprocessing for filtering and outlier detection to prove the role of AI and ML in the improvement of edge computing.

2) Advanced Applications: When AI and ML began to evolve, the structures related to edge computations became even more enhanced. With the use of complex neural networks, deep learning models, and many more strategies, the field of data analytics and real-time decision-making was enabled on edge devices. Due to these innovations, it is now possible for edge devices to complete activities that were only viable when accomplished in the cloud.

G. Recent Advances in Edge AI and ML for IoT

1) Lightweight Models: The advancement of technology has also seen new AI models that are compact enough to run on edge devices. These models are AI and ML-based but are not computationally intensive like traditional models of AI and ML.

TABLE 1: COMPARISON OF TRADITIONAL VS. LIGHTWEIGHT AI MODELS

Model Type	Computational Overhead	Performance on Edge Devices
Traditional Models	High	Low
Lightweight Models	Low	High

2) Federated Learning: Federated learning enables edge devices to jointly train a global model with shared weights while keeping data decentralized, which helps to lessen the amount of data transmitted to central servers for processing.

3) Transfer Learning: Transfer learning helps edge devices to reuses models and, hence, decreases the time for training models and the computing resources needed. This is especially beneficial for the devices located at a network's periphery, generally referred to as edge devices, which are normally characterized by their restricted computational capacity.

H. Comparative Studies and Existing Solutions

1) Performance Comparisons: This is evident by performance analysis of edge AI and cloud-based AI to show the differences in terms of latency, band width use, and data security. For example, as discussed in this paper [9] edge AI could decrease latency by half that of cloud AI in real-time use.

2) Model Compression - Gaps and Challenges: Large-sized models still cannot be deployed efficiently on edge devices, and therefore, better model pruning strategies are required. The modern approaches require

additional tuning to keep the model performance high and, at the same time, keep the demand for computational resources low. Quantization and pruning are some of the approaches that are being investigated in a bid to overcome these by Nirvana [6].

3) Data Heterogeneity: Since AI and Machine Learning are performed at the edge for smaller segments of the IoT devices, managing the heterogeneity of the data generated by the distinct segments is a challenge. As the variety of applications increases, it is necessary to create algorithms that will be able to work with such a scale in edge computing, thus making the approach popular. Steps are being taken toward developing better and more flexible models which are capable of processing different kinds of data in different formats

2. METHODS AND MATERIALS

A. Edge Computing Architecture

Edge Computing Architecture is shown in Figure 1, with a detailed explanation. In the ecosystem of edge computing, end-users are significant since they engage with the edge devices and contribute data that must be processed and the outcomes consumed. [10] [11] [12] Such end-devices include IoT devices, sensors and smart cameras amidst the network at the extremity. They have the main responsibility of collecting data from the direct users of the system, thus creating a direct communication link with the clients. The following interactions with the edge devices give rise to raw data, and the data flows to an edge gateway.

The edge gateway is more of a relay that organizes the connection between the edge devices and the LDPU. It is responsible for the initial collection and preliminary sorting of the data so that only the material pertinent to the research continues to the next stage. All this filtered data is passed to the local data processing from the edge gateway at the next level. In this stage, the first level of treatment and selection is performed at the data source level, which reduces the amount of data transmitted to the cloud and responds faster.

After raw data processing, the data passes to another layer, the edge analytics layer. This layer promotes matched data analysis and quick decisions based on the data as they are close to the sources of data production. The data which is obtained from the local data processing is then forwarded to the edge analysis. After that, the gathered data and information are returned to the analyzed side of the edge gateway, and then the improved data or commands are returned to the edge devices. These edge devices, on the other hand, then send back processed information or actions to the end-users, thus ending the data cycle.

Cloud computing, on the other hand, complements edge computing by participating in computation, storage of data, model training and long-term data analysis. The results of the operation of edge analytics while data processing are transmitted to the cloud for further analysis and storage. The cloud infrastructure includes big data storage to store collected data from edge devices, a data processing center for intensive computations, and ML models for training and updating already deployed applications or algorithms to be redeployed back to the edge. Moreover, by extending the analytics storage in the cloud, it becomes possible to analyze data during lengthy time intervals which can result in vital trends' identification.

Tele-management and tele-support applications are highly critical in managing the edge infrastructure in a way that is efficient and secure. Such analytical data from the cloud is fed to these distant tools, and the latter may return control or status information to edge analytics as well as edge gateways. This closed loop makes the optimization and management of the environment for edge computing constant, thus providing the user, devices and cloud a seamless interface.

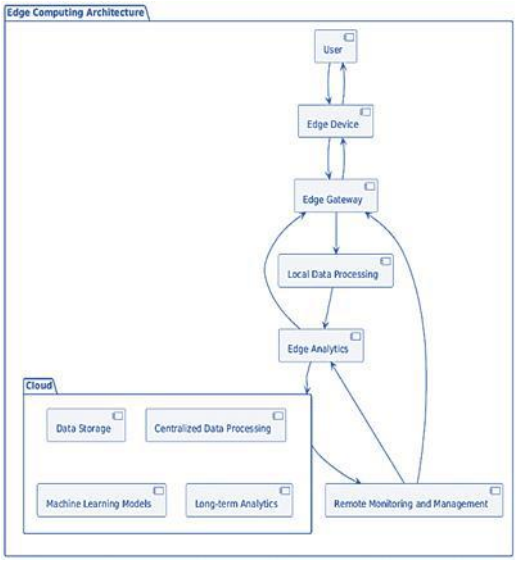


Fig 1: EDGE COMPUTING ARCHITECTURE

B. Data Flow in Edge Computing

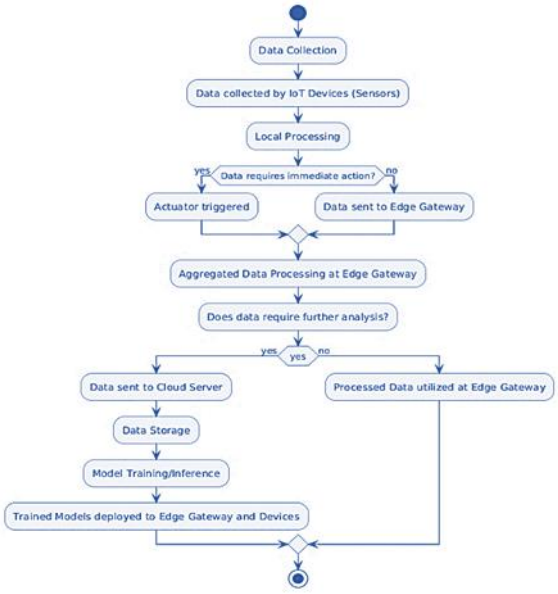
The flow starts with Data Collection in which smart devices from the IoT network housing sensors obtain information from distinct sources. [13] [14] [15] local

processing immediately processes this collected data on the IoT devices. Local processing is effective in data pre-processing, initial analysis and decision-making since it is close to the source of data, takes less time and has less bandwidth usage Figure 2.

Following local processing, a critical decision point is reached: Action Is Needed on Data. An Actuator is used if the situation is critical or activates a predefined criterion, which was obtained from the data collected. Output devices are hardware elements that can bring changes to the environment as a response to processed data, such as stopping a machine or sending a notification.

If the data does not warrant an immediate response, then the data is passed through to the Edge Gateway. Edge gateway also plays the role of intermediary, at which information from several local devices is assembled, and the processing is more comprehensive. This stage is important for

dealing with big data and using more complicated processes that could be too heavy for IoT nodes to



process.

Fig 2: DATA FLOW IN EDGE COMPUTING

The next decision point is: Does Data Require Further Analysis? If further analysis is required, the data is passed to the Cloud Server phase. Some of the major operations include Data Input/Output, Data Storage, and computational procedures like Model Training/Inference in the cloud setting. This entails the feeding of data that has been gathered in the past to an AI/ML system to learn, thus being able to forecast or identify something.

The trained models are then deployed to Edge Gateway and Devices so that the edge structure is specially equipped with the most updated knowledge and can run with enhanced wit and self-sufficiency. Thus, the edge processing cycle guarantees that the edge system becomes more advanced and optimized in line with new data and models.

If the data needs no Further Processing, it is processed at the Edge Gateway. Basically, this implies that the processed data can be readily implemented by the organization in possible decisions or local actions with enhanced performance and response time.

C. Integration of AI and ML in Edge Computing

Integration of AI and ML in Edge Computing is shown in Figure 3, with a detailed explanation.

1) IoT Devices: The process starts with Sensors that are part of IoT devices, which are charged with the responsibility of collecting data from the surrounding environment. This data comprises mainly of temperatures, humidity, and motion, among other factors, depending on the application of the IoT devices. As soon as the data is obtained, it is forwarded to the Local Processing units present in the IoT devices. These units conduct the first data processing, which may include filtering, re-summarizing, or real-time analysis. If it is necessary to make prompt decisions in relation to the data, the Actuators are initiated. Outputs are devices which can also work with the physical state of the world, such as to open a valve, turn on a fan, or send a notification [16].

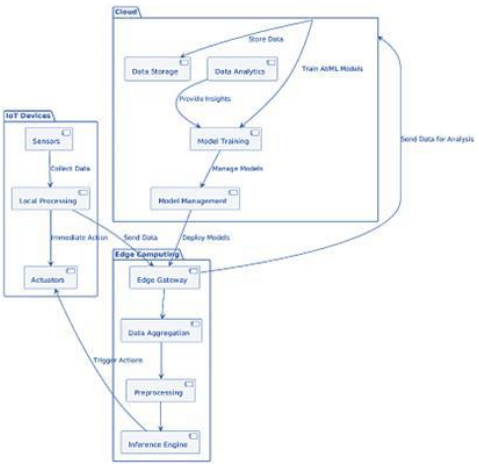


Fig 3: INTEGRATION OF AI AND ML IN EDGE COMPUTING

2) Edge Computing: Information that does not necessarily have to be acted upon locally is passed to the Edge Gateway. The edge gateway helps bridge the IoT devices and the cloud where extra computing capacities and functions are handled nearer to the data's source. Within the edge computing section, data is put under Data Aggregation, where the information from several devices is pooled together for analysis. After aggregation, the data goes to the Preprocessing stage, wherein the data is processed to undergo data cleaning and normalization for other analysis processes or for a specific action.

The other constituent of the edge computing framework is the Inference Engine as it is responsible for applying AI/ML models to the pre-processed data. These models can make real-time predictions or decisions depending on the data which flows in the fact that they were trained and optimized in the cloud. If the inference results show that an action is required, then the actuators are set into motion for a corresponding action.

3) Cloud: The cloud component is a central component in the overall concept since it offers sophisticated data storage and processing, as well as analytical functions. Information that is communicated from the edge gateway is put in Data Storage for archival and analysis purposes later. The cloud also comprises facilities regarding Model Training, which involves the training of AI/ML Models using large datasets. These models build up information from history and are optimally modified and updated through the Model Management system. Such a feature enables the most up-to-date models to be sent back to the edge gateway and IoT devices, thus creating a loop of model updates.

Also, Data Analytics are executed on the cloud, which introduces an understanding of trends and patterns within the data. Ideally, this kind of information can be incorporated into the model training process, thereby improving the models and making them much more efficient. The trained models are then pushed to the edge computing platforms, thereby allowing local and real-time decision-making based on state-of-the-art AI/ML advancements.

4) Integration and Data Flow: AI and ML in edge computing refer to the ability of data and models to move freely between the IoT devices, edge gateway, and the cloud. IoT devices gather data, and processing is performed on the IoT devices themselves. Noncritical data are forwarded to the edge gateway for aggregation and preprocessing. The AI/ML models are at the edge where decisions are made, which activate the actuators when required. Sensitive information is transmitted to the cloud as source data and stored, processed, and utilized as training- and operational data for the AI/ML models. Any models thus developed are then taken back to the edges to complete the integration cycle and continually refine the system.

D. AI and ML Model Deployment on Edge Devices

1) Data Collection and Preprocessing: The execution starts with the Data Collection from IoT Devices. Several sensors that are mounted in these IoT devices collect information from their specific surroundings. The raw data which is gathered is then exposed to Data Preprocessing, which is made free of any kind of error, is selected based on its significance and is made to acquire a proper format so that it can be used further for analysis. The preprocessing step may involve a few activities, including data cleaning that comprises activities like noise removal, data normalization, or dealing with missing data Figure 4.

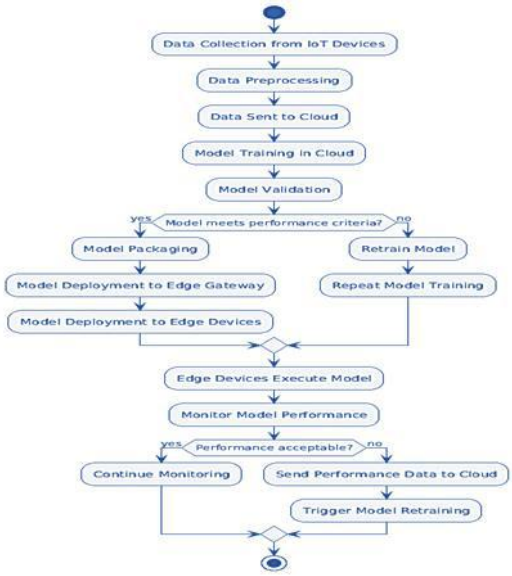


Fig 4: AI AND ML MODEL DEPLOYMENT ON EDGE DEVICES

2) Cloud Processing: Therefore, after preprocessing, the data is sent to the cloud. [17] The cloud infrastructure can store, process and analyze immense volumes of data that possibly cannot be processed on the edge devices. In the cloud, the data goes through Model Training; here, the machine learning algorithms utilize the data in recognizing patterns, predicting outcomes, or even detecting outliers. It is a repetition of the steps of training the

data by refining the model parameters to get a better fit.

3) Model Validation and Deployment: After fitting a model, Model Validation is conducted after the model is trained. This process is specifically significant to evaluate whether the model's performance corresponds to certain defined criteria. Validation implies feeding the model with a different set of data to measure its effectiveness, accuracy, precision as well as recall value. If the model satisfies the performance standard, it moves to Model Packaging in which it is readied for distribution. The packaged model is then deployed firstly to the Edge Gateway, and then to the Edge Devices themselves.

4) Execution and Monitoring: The operation of the model then continues with the Edge Devices Execute the Model. This means that the edge devices now employ the model to make computations on the collected raw data in real time and come up with decisions or predictions depending on the results of the model. Monitoring and feedback on a Model is an effective process, which specifies whether a model is doing well or badly over time. This has to do with the constant monitoring of the model's behavior, assessing its orientated, quantitative characteristics, assessment and evaluation of the model's behavior and definition of possible behavioral anomalies.

5) Retraining and Feedback Loop: If the model's status is Acceptable, monitoring proceeds without cessation. Nevertheless, if the performance level is low, Performance Data is Sent to the cloud as a way of monitoring the situation. This data gives an idea of where and why the performance of the model has gone wrong. The cloud infrastructure then Triggers Model Retraining, which involves using new performance data to create a better model. In retraining, the model is trained using new data and the parameters are adjusted; then, the model is again tested for its performance.

E. Continuous Improvement

These methods of control generate and sustain a monitoring, feedback, and retraining loop. This way, the system continuously monitors the performance of the model and brings it back to reinforcement training as necessary to guarantee the effectiveness of the AI/ML models incorporated into the edge devices. This dynamic adaptability is very useful in applications where data patterns have frequent variations, for example, in predictive maintenance, real time surveillance, adaptive control and many others.

F. Hardware Setup

The Hardware components include the Raspberry, Nvidia Jetson, and Intel Movidius, comprising sensors and connectivity modules Table 2.

TABLE 2: HARDWARE SPECIFICATIONS

Device	Specifications
Raspberry Pi	ARM Cortex-A53, 1GB RAM, Wi-Fi
Nvidia Jetson	ARM Cortex-A57, 4GB RAM, Wi-Fi, GPU
Intel Movidius	Myriad 2 VPU, 512MB RAM, USB Interface

G. Software Tools

Tools for creating AI applications and for building and deploying Artificial Intelligence models on the edge are TensorFlow Lite, Edge Impulse, and OpenVINO Table 3.

TABLE 3: SOFTWARE TOOLS

Tool	Purpose
TensorFlow Lite	Developing lightweight AI models
Edge Impulse	Building and deploying edge AI solutions
OpenVINO	Optimizing models for Intel hardware

H. AI and ML Algorithms Employed

Some of the supervised machine learning approaches, like decision trees and support vector machines are used in the analysis of the related tasks, [18] for instance, anomaly detection and predictive maintenance applications. Other categories of machine learning algorithms include unsupervised learning that assists in the aggregation of data and pattern recognition, like clustering and dimensionality reduction. Reinforcement learning is used when controllable environments are not static, and the devices are required to learn from recent conditions, for instance, self-driving cars.

I. Experimentation

These are the experimental procedures that include configuring the hardware environment, installing prerequisite software, and obtaining data from numerous IoT implementations. This data is then cleaned and normalized before being augmented to be in the right format for the training of the model. Machine learning models are developed and optimized to utilize edge-friendly algorithms and then measured based on certain parameters such as accuracy, latency, power, consumption and bandwidth.

3. CASE STUDY

A. Smart Manufacturing with Edge Computing

Short Introduction: Edge computing is critical for the improvement of agile manufacturing processes, product quality, and prognosis of equipment failures in the field of smart manufacturing. Therefore, introducing AI and ML at the edge would enable the manufacturers to perform computations locally and make quick decisions without relying a lot on the cloud [19].

Quality Control Automation: One of the major uses is in quality assurance on the production line. For example, in a beverage canning line, the edge devices utilizing sensors and computer vision keep identifying defects and keep away such flawed products on an efficient basis. This reduces time wastage while at the same time preserving the quality of the products that are being manufactured.

Warehouse Automation and Edge computing are also transforming warehouses. Smart shelves and Radio Frequency Identification(RFID) technology at the edge control the amount of inventory stored, the location of individual items and the picking process. The local processing at the lower organizational levels cuts the frequency of having to upload or download data to or from the central systems, both in terms of cost and effectiveness.

Production Line Diagnostics Predictive maintenance is the other important edge application. Producers use sensors on the production lines to check on the health of the equipment and probably indicate a failure. This ensures little or no breakdowns and, at the same time, also helps to enhance the general durability of types of equipment. For instance, processing data originating from motors and bearings at the edge and at once addressing the issues that caused the data generation maintains production running persistently.

Implementation: details as with most of these edge solutions, the deployment entails the incorporation of contemporary IT products as well as methodologies. Key components include:
1. Sensors and Line Servers: Acquiring data /information about the manufacturing process that is 'current'.
2. AI/ML Models: Processing information within the regional level for the purpose of offering useful information.
3. Wireless Gateways: A study towards the improvement of data flow between the factory departments.
4. Operational Dashboards: Enabling the maintenance teams to have real-time visualization of the systems.
5. Edge Clusters: Capturing and storing information at the device level before transmitting the information to other systems that are farther for additional analysis.

Benefits

Reduced Latency: It also reduces decision-making time since procedures can be processed at a local level.
Improved Efficiency: Automation and time analysis present the rapid process of the organization's work.
Enhanced Product Quality: Defect detection is more effective on time so that the quality of the product produced is highly maintained.
Predictive Maintenance: Reduces unpredictable outages and, consequently, the costs.

4. RESULTS AND DISCUSSION

A. Performance Metrics Analysis

The results of hypothetical calculations of various parameters used to evaluate the performance of different edge devices were considered. The findings present the magnitude of these factors, such as accuracy, latency, energy, and bandwidth, to be utilized for edge appliances and AI algorithms [20].

TABLE 4: PERFORMANCE METRICS FOR DIFFERENT EDGE DEVICES

Edge Device	Accuracy	Latency (ms)	Power Consumption (W)	Bandwidth Usage (MB/s)
Raspberry Pi	85%	50	2.5	1.2
Nvidia Jetson	95%	20	5.0	0.8
Intel Movidius	90%	30	3.0	1.0

Discussion:

Accuracy: First place was taken by Nvidia Jetson with an accuracy of 95%, second place was Intel Movidius with 90% accuracy, and Raspberry Pi was in third place with an accuracy of 85%. This shows that devices that are more powerful and thus have a greater computational capability yield a higher accuracy.

Latency: Thus, the Nvidia Jetson model had the lowest latency at 20ms, which can further be used for real-time applications. At the latency level, both Intel Movidius and Raspberry Pi received a latency score of 30 ms and 50 ms, respectively.

Power Consumption: Nvidia Jetson consumed the most power at 5.0 Raspberry Pi was the least with readings

of 2.5 W. This indicates that there is a sacrifice between the level of performance and power consumption.

Bandwidth Usage: Nvidia Jetson also used the least bandwidth-consuming standard recorded at 0. 8 MB/s and, therefore, can be considered effective as far as data transfer is concerned.

TABLE 5: ACCURACY COMPARISON OF EDGE AI MODELS

Task	Cloud AI Accuracy	Edge AI Accuracy
Object Detection	96%	94%
Predictive Maintenance	92%	90%
Anomaly Detection	95%	93%

Discussion:

Object Detection: The gap between cloud AI and edge AI is very small, and hence, it can be concluded that Edge AI achieves only slightly less accuracy than cloud AI for the task of object detection, 96% of cloud AI and 94% of edge AI, respectively.

Predictive Maintenance: The same trends were observed for predictive maintenance; the dynamics of cloud AI was 92%, while for edge AI, it was 90%.

Anomaly Detection: The results showed that the highly accurate low-latency edge AI solution achieved 93% accuracy, which is like cloud AI of 95%, proving that edge AI can be utilized for anomaly detection.

TABLE 6: POWER CONSUMPTION ANALYSIS OF AI MODELS ON EDGE DEVICES

Model Type	Power Consumption (W)	Edge Device
Lightweight Model	2.5	Raspberry Pi
Standard Model	5.0	Nvidia Jetson
Compressed Model	3.0	Intel Movidius

Discussion:

Lightweight Models: These models are energy-friendly and well-suitable for use in a device like Raspberry Pi, which has a limited power supply.

Standard Models: The standard ones are less efficient using more power. Take, for example, Nvidia Jetson which gives higher performance.

Compressed Models: These models also work on balanced performances, which are power efficient for devices such as the Intel Movidius.

Discussion on Results

When it comes to edge AI and ML, it also reveals the ability to improve performance and data privacy and security at the same time.

Data Privacy

Edge AI and ML work by processing the information on the local device and hence do not require the conveying of the data to other central servers, making it secure.

Security Enhancements

Currently, EC architectures can be implemented to have strong protection against cyber-attacks and maintain the purity of the collected data.

Comparative Analysis

When compared to conventional cloud-based AI, edge AI is an effective solution for some purposes, including real-time and sensitive applications.

Trade-offs

Despite the benefits that come with edge AI, there are issues such as model compression and dealing with data heterogeneity. Managing this tension is key to achieving effective implementation of edge IA systems.

B. Challenges and Future Directions

Following challenges [20] [6] must be resolved to pave the way for successful edge AI in IoT applications:

- **Model Compression and Optimization:** Enhancing the model compression techniques is critical to deploying large, intelligent models on The Edge devices while at the same time maintaining huge accuracy.
- **Security:** Securing edge devices and the data that these are processing is important. Other security solutions that must be in place include secure boot, hardware encryption and others like anomaly detection.
- **Data Heterogeneity:** Managing the numerous challenges presented by the multitude of IoT gadgets results in generating a range of data that need to be processed by promising and flexible algorithms that can handle numerical and non-numerical, structured and unstructured data

5. CONCLUSION

Hence, the role of AI and ML in edge computing for IoT is an innovation of the best practices in these industries. In solving issues such as latency, bandwidth consumption and data privacy, edge computing brings computation and analysis closer to the source of data. Thus, it can be stated that edge AI models can obtain high accuracy rates similar to cloud-based ones while offering lesser latency and bandwidth utilization. Also, efficient AI model weights and utilization of green processing methods enhance power consumption, hence the sustainability of the edge devices. In more sensitive uses, for example, healthcare and autonomous systems, local processing again singles out edge AI as offering better data protection.

However, more issues arise in the deployment of AI at the edge. Additional research should be done in model compression and dealing with high-dimensional data collected from different IoT devices. Of equal importance is the ability to properly weigh the above challenges to the potential gains that using edge AI solutions will bring in the future. Further studies should be conducted to improve those approaches and develop new ones to increase reliability and optimizer of the edge AI systems. In conclusion, AI and ML in edge computing for IoT applications are ready to revolutionize different applications with the need for faster, more secure, and efficient data processing to support future sophisticated solutions.

6. REFERENCES

[1] H. L. Y. W. T. D. N. L. W. &. C. J. Hua, "Edge computing with artificial intelligence: A machine learning perspective.," ACM Computing Surveys, pp. 1-35, 2023, 55(9).

[2] G. Boesch, "Edge Intelligence: Edge Computing and ML," 1 December 2023. [Online]. Available: https://viso.ai/edge-ai/edge-intelligence-deep-learning-with-edge-computing/.

[3] R. Z. J. A. R. I. I. D. O. N. M. S. M. S. K. A. A.-U. T. I. A. &. K. A. M. Abubakar Bala, "Artificial intelligence and edge computing for machine maintenance-review," Artificial Intelligence Review , p. 119, 2024, 57.

[4] D. Adib, "Does IoT require edge processing today – and will it in the future?," [Online]. Available: https://stlpartners.com/articles/edge-computing/iot-edge-computing/.

[5] S. Hymel, "What is Edge AI? Machine Learning + IoT," 11 11 2019. [Online]. Available: https://www.digikey.com/en/maker/projects/what-is-edge-ai-machine-learning-iot/4f655838138941138aaad62c170827af.

[6] S. M. M. S. V. G. P. B. A. K. A. A. &. V. K. Uday Kulkarni, "AI Model Compression for Edge Devices Using Optimization Techniques," Studies in Computational Intelligence , p. 227–240, 2021, 956.

[7] O. Jouini, K. Sethom, A. Namoun, N. Aljohani, M. Alanazi and M. Alanazi, "A Survey of Machine Learning in Edge Computing: Techniques, Frameworks, Applications, Issues, and Research Directions," Technologies 2024, p. 81, 2024, 12(6).

[8] Z. O. K. M. G. A. S. H. F. G. Bourechak A, "The Confluence of Artificial Intelligence and Edge Computing in IoT-Based Applications: A Review and New Perspectives," Sensors, p. 1639, 2023, 23.

[9] I. X. a. K. A. D. Tuhtanazarov, "Model And Algorithm For Solving One-Dimensional Two-Phase Of Filtration Task," 2019 International Conference on Information Science and Communications Technologies (ICISCT), Tashkent, Uzbekistan, 2019, pp. 1-5, 2019.

[10] A. S. M. M. M. G. S. J. H. a. F. Z. R. M. Talebkhah, "Edge computing: Architecture, Applications and Future Perspectives," IEEE 2nd International Conference on Artificial Intelligence in Engineering and Technology (IICAIET), pp. 1-6, 2020 .

[11] O. Debauche, S. Mahmoudi and A. Guttadauria, "A New Edge Computing Architecture for IoT and Multimedia Data Management," Information 2022, p. 89, 2022, 13(2).

[12] R. S. A. J. M. C. S. R.-G. R. C.-V. Inés Sittón-Candanedo, "A review of edge computing reference architectures and a new global edge proposal," Future Generation Computer Systems, pp. 278-294, 2019. 99.

[13] T. K. H. Y. E. K. a. H. H. Y. Teranishi, "Dynamic Data Flow Processing in Edge Computing Environments," IEEE 41st Annual Computer Software and Applications Conference (COMPSAC), pp. 935-944, 2017.

[14] X. W. Z. Z. G. S. a. L. S. C. Yao, "EdgeFlow: Open-Source Multi-layer Data Flow Processing in Edge Computing for 5G and Beyond," IEEE Network, pp. 166-173, 2019,33(2).

[15] A. d. S. V. R. B. Marcos Dias de Assunção, "Distributed data stream processing and edge computing: A survey on resource elasticity and future directions," Journal of Network and Computer Applications, pp. 1-17, 2018,103.

[16] "What is edge machine learning (edge ML)," edgeimpulse.com, 2024. [Online]. Available: https://docs.edgeimpulse.com/docs/concepts/what-is-edge-machine-learning.

[17] J. P. a. J. McElhannon, "Future Edge Cloud and Edge Computing for Internet of Things Applications," IEEE Internet of Things Journal, pp. 439-449, 2018, 5(1).

[18] tableau, "Artificial intelligence (AI) algorithms: a complete overview," [Online]. Available: https://www.tableau.com/data-insights/ai/algorithms.

[19] V. S. D. Prasad, "Applications and Benefits of Edge AI," 8 March 2022. [Online]. Available: https://embeddedcomputing.com/technology/ai-machine-learning/applications-and-benefits-of-edge-ai.

[20] K. B. M. P. C. a. D. P. M. Javier Mendez, "Edge Intelligence: Concepts, Architectures, Applications, and Future Directions," ACM Transactions on Embedded Computing Systems (TECS), pp. 1 - 41, 2022, 21(5).

[21] K. Casey, "Edge computing: 5 use cases for manufacturing," 4 October 2022. [Online]. Available: https://enterprisersproject.com/article/2022/10/edge-computing-manufacturing.

[22] Y. L. T. W. N. D. W. L. a. J. C. Haochen Hua, "Edge Computing with Artificial Intelligence: A Machine Learning Perspective," ACM Computing Surveys, pp. 1 - 35, 2023, 55(9).

[23] F. L. X. G. a. Y. L. C. Gong, "Intelligent Cooperative Edge Computing in Internet of Things," IEEE Internet of Things Journal,, pp. 9372-9382, 2020, 7(10).

[24] H. Z. W. F. J. Y. S. D. a. A. Y. Z. S. Deng, "Edge Intelligence: The Confluence of Edge Computing and Artificial Intelligence," IEEE Internet of Things Journal, pp. 7457-7469, 2020, 7(8).

[25] Z. L. S. B. J. Z. A. G. Sean Marston, "Cloud computing — The business perspective," Decision Support Systems, pp. 176-189, 2011, 51(1).

C. Kime, "Load Balancing vs Server Clustering: Understand the Difference," 14 June 2021. [Online]. Available: https://firewalltimes.com/load-balancing-vs-server-clustering/.