International Journal of Core Engineering & Management
Volume-7, Issue-01, 2022 ISSN No: 2348-9510

EDGE AI: DEPLOYING DEEP LEARNING MODELS ON MICROCONTROLLERS FOR BIOMEDICAL APPLICATIONS: IMPLEMENTING EFFICIENT AI MODELS ON DEVICES LIKE ARDUINO FOR REAL-TIME HEALTH MONITORING

Vedant Singh

Abstract

Real time health monitoring in biomedical applications is now possible through efficient usage of deep learning models on portable microcontrollers through edge AI. Edge AI is different from most preexisting AI systems that require massive cloud support for complex computations and analytics. It is most effective in healthcare since time is vital in treatment, and data must be safeguarded securely. Arduino, ESP32, Raspberry Pi Pico, and other microcontrollers serve as the main components of the Edge AI systems used in biomedical practice. These small and, low-cost, low-power devices enable the use of complex AI methods on limited-featured hardware. Deep learning specialists also note that using techniques such as quantization, pruning, and model compression, deep learning models can be fine-tuned for microcontrollers without achieving a reduction in performance outcomes. This capability covers the ability to track and monitor an individual's health status in real-time using wearable devices and portable diagnostic equipment, including the rate of heartbeat, respiration, and blood oxygen levels. The combination of Edge AI with the Internet of Things (IoT) takes advantage of this to allow communication between devices for full health analysis. Similarly, innovations in one's microcontrollers and enhancements in model advancement procedures provide hope for extending the utilization of biomedical systems even to the neglected parts of the globe. This paper reviews the deployment workflow, assessing key issues and future trends of Edge AI while focusing on the importance of its application in the healthcare sector. Edge AI truly sets the stage for effective, efficient, and intelligent healthcare systems by successfully merging technology with reality.

Keywords: Edge AI, Microcontrollers, Biomedical Applications, Deep Learning, Healthcare, Real-Time Monitoring, Optimization Techniques, Artificial Intelligence (AI), Model Deployment, Wearable Devices.

I. INTRODUCTION

Edge AI is another trend that has emerged in the new age of AI, where instead of linking everything to the cloud, data is processed on the device itself. Edge AI is the deployment of model algorithms on edge devices, including microcontrollers, sensors, and other resource-scarce hardware. Different from other forms and types of artificial intelligence that require

International Journal of Core Engineering & Management
Volume-7, Issue-01, 2022 ISSN No: 2348-9510

computation and inference to take place in the cloud, edge AI operates at the periphery of the network. This transformation is major in settings where real-time processing is important, with low latency and high privacy. Edge AI integrates the advantages of both artificial intelligence and edge computing technologies. In essence, Edge AI enables devices to take charge of computing, analyzing data, making decisions, or performing tasks with no recourse to the cloud for frequent support. This decentralized strategy works to decrease the response time, increase stability, and strengthen the protection of users' data by avoiding a constant exchange with other servers.

Importantly for healthcare, Edge AI is changing how medical devices function. As it empowers timely decision-making at the edge, it paves the way for real-time health checks, immediate notifications, and self-operations during important contingencies. For instance, the smart wearable devices and compact form diagnostic tools that are integrated into Edge AI technology mean that patients can immediately get an idea of their health status, something that is especially beneficial in a critical healthcare situation or where the availability of donor resources is limited. Additionally, the resulting Edge AI minimizes the reliance on high internet speeds that can effectively bring the healthcare solutions to the doorsteps of patients located in even the rural regions.

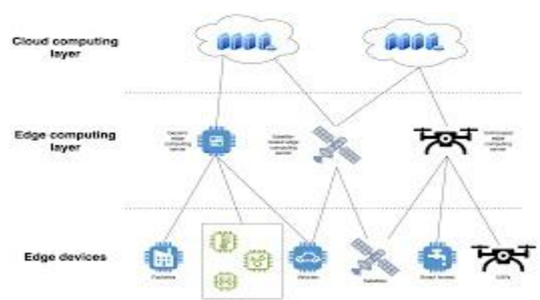


Figure 1: Combining Machine Learning and Edge Computing

The benefits of Edge AI in healthcare are numerous and deep-seated. The feature of keeping data locally processed avoids delay, which is vital for apps like heartbeat rate and identifying falls among the elderly. Second, it provides better privacy as patient data remain confined to the smartphone and are never transmitted anywhere else. Last but not least, the implementability of Edge AI solutions on a large scale and with low costs drives the solutions' applicability to multiple settings. HC Edge AI can be attributed directly to the contribution of microcontrollers. These are portable power-constrained computing devices optimized to perform limited tasks effectively. These microcontrollers include Arduino boards, ESP32, Raspberry Pi Pico, and others due to their popularity, low cost, energy saving, and versatile features of this elevator that make them more suitable for biomedical applications.

Another strong prospect for microcontrollers is their low power use; they require proportionately less electricity than other microcomputers. This means that most of the equipment, like wearable and portable diagnostic devices used in healthcare, require long-

lasting power and are rarely charged. Microcontrollers offer this potential, and it is guaranteed that they will work accurately all the time. Another important aspect is cost reconnoiter. While typical processors used in other types of AI systems are more complex and expensive, microcontrollers can be bought for a relatively low price, facilitating the creation of cost-effective healthcare solutions. This makes it possible to extend best-practiced healthcare innovations, especially to the neediest societies. This aspect is equally important in biomedical applications of nanoparticles. Microcontroller-based devices are portable and small in size and can hence be built into a watch, a human implant, or portable diagnostic equipment. For instance, a wristband can be fitted with a microcontroller system that continuously and in real-time detects a patient's heartbeat rate and instantly notifies caregivers of any likely anomalies.
Combining deep learning with Edge AI is revolutionizing the healthcare segment by presenting sophisticated solutions for analyzing the intricate existing data. Deep learning is a particular branch of AI that uses neural networks to identify the relationships between data points in large datasets. In the field of healthcare, deep learning models can diagnose diseases, examine images, and even read patients' vital signs with a high degree of precision. For instance, CNNs can be used to analyze the electrocardiogram (ECG) signals for the identification of arrhythmias, whereas RNNs can be applied to predict trends in patients' data. This means that with these models deployed on microcontrollers, real-time health monitoring becomes possible, including in situations that characterize low-resource health contexts. This advancement has enabled healthcare providers to deliver timely interventions and, in the process, improve patient outcomes enormously.
This paper discusses the real-time biomedical application of efficient deep-learning models on microcontrollers in detail. Regarding digital devices such as Arduino, it delves into issues, approaches, and real-life applications of Edge AI in the healthcare sector. The purpose of this work is to present a set of recommendations to scholars and practitioners interested in advancing the approaches based on Edge AI to improve patients' treatment. Edge AI holds the key to the implementation of intelligent healthcare systems with portability, efficiency, and cost optimization at its best.

II. UNDERSTANDING EDGE AI AND MICROCONTROLLERS

2.1 What is Edge AI?

Edge AI can be described as the practice of running AI models on edge devices, including microcontrollers and sensors, where data is analyzed on the device itself without the need for the cloud (Merenda et al., 2020). This approach is different from a previously developed AI system that needs an external server or cloud to analyze the information. Specific attributes of Edge AI include low latency times, better energy management, and improved data privacy. As such, Edge AI helps run applications and services that require immediate response time and assurance that the data being used is stored locally on the device and not synched with any cloud database regularly.

Conventional AI architectures normally rely on complex computing assets for model learning

International Journal of Core Engineering & Management
Volume-7, Issue-01, 2022 ISSN No: 2348-9510

and prediction activities. Nonetheless, Edge AI works within the confines of small hardware devices, including but not limited to memory space and the ability to process data. To accommodate such environments, specific measures such as model quantization and model compression are applied to render the models light and energy exigent (Nyati, 2018). Therefore, Edge AI can enable model deployment in scenarios where communication and computing ability are restricted, such as the continuous monitoring of patients' health or using medical wearable technologies.

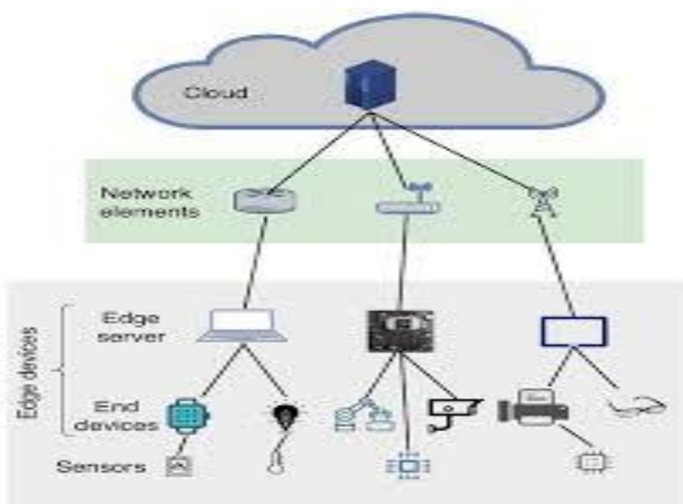


Figure 2: Edge AI Overview

2.2 Introduction to Microcontrollers

Microcontrollers are small-sized system chips developed to perform set functions effectively. They act as the core of Edge AI since they allow small resource-constrained devices to compute and interact with other components, like sensors (WoldeMichael, 2018). Subsequently, utilizing microcontrollers in the context of Edge AI results in positive trends in costs, portability, and energy efficiency. For the same reason, they are best suited for real-time biomedical applications such as heart monitoring and respiratory rate measurement. They are familiar ones like Arduino, Raspberry Pi Pico, and ESP32, among others. All of these devices provide differentiated features for the specific Edge AI applications. For instance, Arduino is relatively easy to use and offers abundant community support, something that makes it even more suitable for rookies. Raspberry Pi Pico has better features that include the availability of two processing cores. At the same time, ESP32 comes with features like integrated Wi-Fi and Bluetooth that allow for wireless transfer of data in real-time.

Microcontroller	Processor	Memory	Connectivity	Price (USD)	Best Use Case

International Journal of Core Engineering & Management
Volume-7, Issue-01, 2022 ISSN No: 2348-9510

Arduino Nano 33 BLE	ARM Cortex-M4	256 KB Flash	Bluetooth	20-25	Simple AI tasks (e.g., gesture recognition).
Raspberry Pi Pico	Dual ARM Cortex-M0+	2 MB Flash	None	04-May	Advanced processing (e.g., data preprocessing).
ESP32	Xtensa Dual-Core LX6	520 KB SRAM	Wi-Fi, Bluetooth	06-Oct	Wireless applications (e.g., IoT health monitoring).

Table 1: Comparison of Popular Microcontrollers for AI Deployment

2.3 Why Edge AI in Biomedical Applications?

The healthcare industry is now experiencing adoption, especially in real-time monitoring with the Edge AI techniques. In the biomedical field, the first benefit of using Edge AI is certain: data latency is reduced. Such systems help process data locally and thus reduce time delays, which are used to access data from cloud-based servers (Sun et al., 2020). For instance, in cases like heart rate or glucose monitoring, fast responses definitely have a positive impact on the illness-affected individual. Another important advantage is improved data privacy (Nyati, 2018). By inputting, processing, and storing electronic and patient-sensitive information at the local level, the problem of invasion and breach is controlled. This feature matches well with preventive measures in healthcare compliances like the Health Insurance Portability and Accountability Act (HIPAA), which requires particular measures to safeguard medical information.

Further, Edge AI functioning can be offline, which is valuable for use in settings with poor connectivity or limited internet connectivity. Just like any other cloud-based system, cloud-dependent systems are only efficient when the internet connection is reliable throughout the region. Edge devices, on the other hand, can work autonomously and continuously, so the monitoring and analysis of data do not stop. For instance, smartwatches that use microcontrollers and Edge AI can monitor patients' vital signs on a real-time basis, including in zones where there is no network connectivity at all (Amin, & Hossain, 2020). Such capabilities make Edge AI a necessity for current and future biomedical applications, including wearable fitness trackers and more sophisticated diagnostic devices. This blend of low power consumption, real-time processing, and data privacy opens the door to mass deployment and efficient healthcare solutions.

III. DEEP LEARNING MODELS FOR EDGE AI

The incorporation of deep learning in edge devices, including microcontrollers, has now precipitated a new wave of solutions for scientific biomedical real-time applications. In the past, deep learning was often related to computationally intensive systems; however, with the recent achievements in model optimization and hardware design, deep learning can now be executed on resource-constrained devices (Gill, A. (2018). This section covers the native architectures of deep learning, the concerns and approaches for running such models on microcontrollers, the optimization methods available to them, and a field example of their use.

3.1 Overview of Deep Learning Models

International Journal of Core Engineering & Management
Volume-7, Issue-01, 2022 ISSN No: 2348-9510

Specifically, deep learning is a branch of machine learning that uses artificial neural networks to process the data patterns existing in the information flow. Several architectures play a pivotal role in the deployment of deep learning models, particularly for Edge AI applications:

3.1.1 Convolutional Neural Networks (CNNs): CNNs are particularly drawn to tackle spatial data and, therefore, are useful when dealing with image or video information (Zha et al., 2015). These networks use convolutional layers to identify features such as edge, shape, and texture in data, making them perform well in areas such as diagnosis by inference and gesture recognition. Due to their layered design, feature extraction can be accomplished with far fewer parameters than can fully connected networks, and thus, they are feasible in restricted platforms.

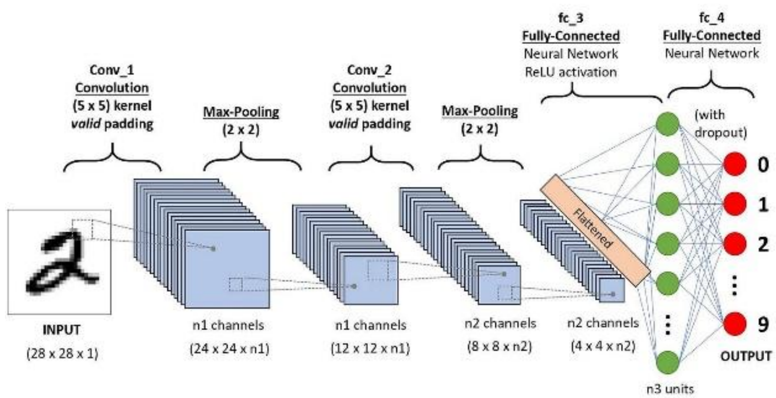


Figure 3: An Example Convolutional Neural Networks

3.1.2 Recurrent Neural Networks (RNNs): RNNs work well with sequential data input, either a time series or a text. Thus, by introducing loops into their construction, RNNs are capable of remembering the earlier inputs, which is particularly useful in interpreting tendencies in the variability of heartbeat rates or identifying pathologic features in ECG signals (Xue & Yu, 2021). Still, conventional RNNs may be computationally intensive and may be improved for integration with microcontroller systems.

3.1.3 Long Short-Term Memory (LSTM) Networks: LSTM, especially, is a less general type of RNN that can solve the problem of long-term dependencies and gradients vanishing (Sherstinsky, 2020). Owing to their rather excellent performance in tasks such as speech recognition or physiological signals processing, they are ideal for real-time health monitoring procedures.

Although these architectures look promising when deployed on microcontrollers, they are not without challenges.

3.2 Limitations of Using Deep Learning on Microcontrollers

Using deep learning models in microcontrollers is not a small task since they have limited resources. The main challenges include:

International Journal of Core Engineering & Management
Volume-7, Issue-01, 2022 ISSN No: 2348-9510

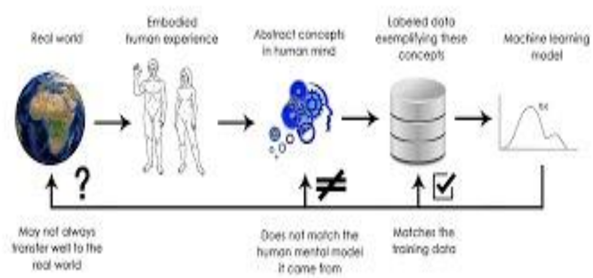


Figure 4: The limitations of deep learning

3.2.1 Limited Memory: Typically used microcontrollers work with tens to thousands of times less memory, from several KiB of RAM to a few MiB, which is insufficient for feeding and working with large multidimensional neural networks. For instance, a simple CNN with a reasonable number of filters may need a memory size that could not fit in devices such as Arduino Uno or ESP32.

3.2.2 Processing Power: Microcontrollers are different from GPUs or other intelligent AI accelerators in that the latter involve low-power CPUs designed to handle routine operations (Nyländen, 2018). Arithmetic operations like matrix multiplication or convolutions can be computationally expensive, and adding them anywhere in the process of model inference dilutes real-time responsiveness.

3.2.3 Energy Efficiency: Energy use is always one of the main concerns, especially in biomedical applications, especially for portable or implantable devices. Analyzing massive data volumes using complex models on computational units with battery-rechargeable energy sources substantially drains such batteries, reducing the systems' availability (Larminie & Lowry, 2012).

Solving these problems necessitates advanced strategies to enhance and fine-tune deep learning for effective implementation.

3.3 Model Optimization Techniques

Several optimization techniques have been developed to tailor deep learning models for microcontroller compatibility without compromising performance:

3.3.1 Quantization: Quantization minimizes the number of significant bits by limiting the weights and activations of the model to low-bit values, for example, to 8-bit integers (Vandersteegen et al., 2021). This approach reduces memory consumption and calculation time to the minimum optimal for the microcontroller to handle, which makes it perfect for microcontrollers. For instance, TensorFlow Lite Micro provides post-training quantization to reduce model size and boost velocity.

3.3.2 ***Pruning:*** Reducing removes small-size weights within a neural network, making it smaller in size. Pruning reduces the size of a neural network. In this sense, while the model is lighter as weights with insignificant participation are eliminated, accuracy is maintained. This is helpful, especially in CNNs with some of the filters that may not play much role in feature extraction.

3.3.3 Model Compression: Methods such as knowledge distillation merge the large model into a smaller model by transferring knowledge. The smaller model is called the "student" and replicates the outputs of the larger "teacher" model. This allows the installation of lightweight networks with acceptable degradation levels of performance parameters.

When used correctly, these strategies facilitate the utilization of deep learning on microcontrollers. The following case study illustrates this in reference to the use of a monitor to record heart rate.

3.4 Case Study: Improving a CNN for Tracking Heart Rates

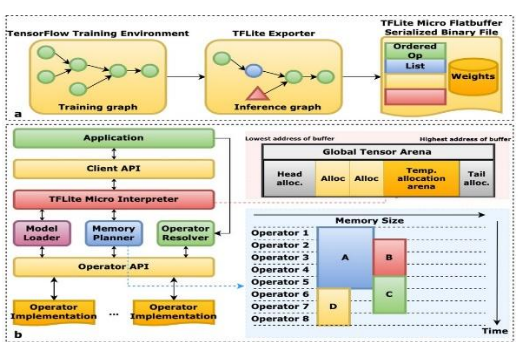
One important application of current RHMS is continuous heart rate tracking, which may sometimes entail the use of proper algorithms for processing physiological signals. This work presents an overview of the measures implemented to prepare a CNN for real-time implementation on an Arduino Nano 33 BLE Sense.

3.4.1 ***Model Selection:*** A lightweight CNN was chosen as PPG signals are captured by optical sensors. The initial model had three layers of convolution and two layers of full connection and contained 420k parameters (Ibtehaz, 2020).

3.4.2 Quantization: To keep the floating point numbers to a minimum, post-training quantization was employed to convert the weights and the activations to 8-bit integers. This reduced the model size by about 75%, making it easy for it to fit in the Arduino Nano's 256 KB SRAM.

3.4.3 ***Pruning:*** Moreover, it utilized the pruning technique to eliminate 40% of the network's weights, thereby decreasing the computational intensity while tolerating a small level of inaccuracy. The process of pruning required repeated fine-tuning to maintain the model's accuracy.

3.4.4 Testing and Deployment: The optimized model was deployed using TensorFlow Lite Micro (Warden & Situnayake, 2019). In testing, the model was said to offer labor of 150 milliseconds per inference, which makes it ideal for real-time monitoring.



International Journal of Core Engineering & Management
Volume-7, Issue-01, 2022 ISSN No: 2348-9510

Figure 5: Operation of TensorFlow

3.5 Performance Evaluation:

The final implementation also shows energy-efficient running through the system, which only uses 10 mW while constantly monitoring (Qiu et al., 2011). This also proved the proof of concept of the presented approach of developing deep learning models on microcontrollers for biomedically important tasks.

Consequently, the choice of combining model optimization techniques with the tools for microcontroller deployment is discussed as a practical approach in Edge AI in this case study. When it comes to microcontroller deployment, though, which necessarily entails model optimization, deep learning models are still feasible. On resource scarcity issues, quantization, pruning, and model compression play a pivotal role in sorting out efficiency needs and performance (Ganesh et al., 2021). The above-mentioned heuristic strategies can prove to be valuable in the development of low-cost and real-time health monitors that are clearly scalable. The enhancement of the CNN-based HRM system serves as a perfect illustration of how Edge AI can revolutionize the biomedical industry and result in the development of more efficient and affordable approaches to increase the quality of people's lives.

IV. DEPLOYMENT WORKFLOW

Deep learning models for biomedical applications on microcontrollers consist of different steps, from choosing the correct hardware to using the model and debugging it in the field.

4.1 Pre-deployment Preparation

The preparation process is the most crucial as it involves assessing the environment in which the program is to be deployed, ensuring a smooth deployment process for the planning team.

4.1.1 Choosing the Right Microcontroller: Microcontrollers are the central component of any Edge AI architecture because it is they that perform computations associated with AI. Choosing the right microcontroller is very important, and it involves several important facets, such as superior processing ability, massive storage, power consumption, and AI toolkit compatibility. These, among others, include Arduino Nano 33 BLE Sense, Raspberry Pi Pico, and ESP32. For biomedical applications, microcontrollers should also have the support of ADC to process the signal from biomedical sensors. For example, the Arduino Nano 33 BLE Sense has an NPU for AI processing, which makes it perfect for low-power health monitoring applications (Taivalsaari et al., 2021). On the other hand, the ESP32 is well renowned for its wireless communication functionality, which relays information to cloud storage for additional processing.

Microcontroller	Processor	Memory (RAM)	AI Framework Support	Power Consumption
Arduino Nano 33 BLE Sense	ARM Cortex-M4	256 KB	TensorFlow Lite Micro	Low

International Journal of Core Engineering & Management
Volume-7, Issue-01, 2022 ISSN No: 2348-9510

Raspberry Pi Pico	ARM Cortex-M0+	264 KB	MicroPython, TinyML	Medium
ESP32	Xtensa Dual-Core	520 KB	TensorFlow Lite Micro	Medium

Table 2: Comparison of Popular Microcontrollers for AI Deployment

4.2 Training and Optimizing the Model

Most of the deep learning models that are trained to run on microcontrollers must be small and compact. High-performance platforms in the training typically performed include, for example, GPUs, which can handle large input data sets (Wang et al., 2019). Once trained, the model is optimized using techniques such as:

4.2.1 **Quantization:** Scales down model parameters from the generally used 32-bit floating point to 8-bit integers in order to reduce both memory and computation.

4.2.2 **Pruning:** It reduces the number of units present in a neural network by removing parameters to decrease model complexity without a substantial impact on performance.

4.2.3 **Model Compression:** Cuts down on computational needs by adding pruning and quantization all in one.

4.3 Tools for Deployment

Several tools are available to assist in deploying AI models on microcontrollers:

4.3.1 **TensorFlow Lite Micro:** TensorFlow Lite Micro has been optimized to run on resource-scarce IoT devices.

4.3.2 **Edge Impulse:** An intelligent development platform for building, training, and deploying AI models directly onto microcontrollers.

4.3.3 **TinyML Frameworks:** A set of MicroPython tools and libraries and TinyML for Arduino to ease the integration process.

These tools offer inbuilt libraries and automated processes to help developers integrate Edge AI into their devices easier.

4.4 Integrating AI models on microcontrollers

After the model is calibrated and fine-tuned, the final model is then deployed into the microcontroller domain.

In this case, TensorFlow Lite Micro for Model Integration is employed.

International Journal of Core Engineering & Management
Volume-7, Issue-01, 2022 ISSN No: 2348-9510

Figure 6: Microcontrollers for Machine Learning and AI

4.4.1 TensorFlow Lite Micro is a flexible library that enables the functioning of artificial intelligence on devices with low processing power. The process typically involves:

4.4.2 **Converting the Model:** The trained deep learning model is then exported to TensorFlow Lite format with the extension '.tflite' (David et al., 2020).

4.4.3 **Integrating with Firmware:** The converted model is then compiled into the microcontroller's firmware using the TensorFlow Lite Micro library.

4.4.4 **Loading the Model:** The firmware is then loaded into the microcontroller for processing and execution.

- **Code Example: Loading a Model onto Arduino**

```
#include "TensorFlowLite.h"

// Include the model
extern const unsigned char model[];
extern const int model_len;

// TensorFlow Lite library objects
tflite::MicroErrorReporter micro_error_reporter;
tflite::ErrorReporter* error_reporter = &micro_error_reporter;

// Memory allocation for the TensorFlow Lite model
constexpr int kTensorArenaSize = 1024;
uint8_t tensor_arena[kTensorArenaSize];

// Setup TensorFlow Lite interpreter
void setup() {
 Serial.begin(115200);

 // Initialize TensorFlow Lite components
 tflite::MicroInterpreter interpreter(model, tensor_arena, kTensorArenaSize,
```

International Journal of Core Engineering & Management
Volume-7, Issue-01, 2022 ISSN No: 2348-9510

```
error_reporter);
  Serial.println("Model loaded successfully");
}

void loop() {
  // Your inference logic here
}
```

This code provides a fundamental structure for uploading a TensorFlow Lite model on Arduino. Developers can extend it to perform real-time health monitoring using biomedical sensors

4.5 Testing and Debugging

It is also advisable to deploy the developed AI model into a real-world environment and troubleshoot when there are errors or bugs.

4.5.1 Evaluating Performance on Real-World Data

Online and offline performance fully entails feeding the microcontroller with appropriate data derived from the biomedical sensors. Key metrics include:

- **Inference Time:** The mapping is the delay between the input received and the subsequent generation of a prediction.
- **Accuracy:** Refers to the ability of the model to perform well when estimated with unseen data samples.
- **Power Consumption:** Performance of the system with respect to energy efficiency when the system is in operation for an extended period (Dincer & Rosen, 1999).

For example, researchers assume the heart rate monitoring system is tested using data from a photoplethysmograph (PPG) sensor. The results can be analyzed to determine whether the model allows accurate predictions in real-time.

4.5.2 Common Troubleshooting Techniques

Developers may encounter issues such as:

- **Memory Overflow:** This can be fixed either by optimizing the model even further or by allocating more memory for the task in the firmware.
- **Latency Issues:** The problem might not occur if the model is smaller or if some calculations are made easier.
- **Sensor Noise:** Signal preprocessing techniques, namely filtering, can help to avoid this problem.

4.6 Workflow for Deploying AI Models on Microcontrollers

Employing the chart of the stages of the model's deployment on microcontrollers provides a clear and logical display of the process, with a focus on the sequence of the actions required for successful deployment. The solution refines each step so that the process results are efficient and optimized for application, especially in biomedical monitoring systems.

International Journal of Core Engineering & Management
Volume-7, Issue-01, 2022 ISSN No: 2348-9510

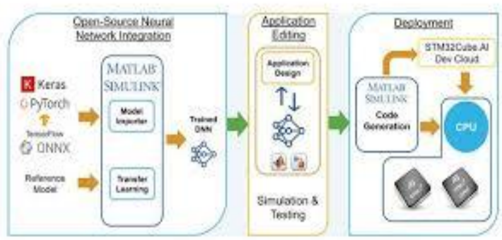


Figure 7: Workflow for Deploying a Neural Network

Training and optimizing the model is the first process in the workflow and is done in a high-performance system. This step includes feeding the model to larger datasets using graphical processors or other powerful hardware. To make identifiers learn correctly, biomedical data, for instance, the heart signal or respiratory patterns, are passed through the layers (Faust et al., 2018). Optimization comes next where ways such as quantization, pruning, and model compression are used to minimize the size and the computational resources of the model for microcontroller use.

Subsequently, the model is converted to TensorFlow Lite, which is more compatible with Microcontroller environments. The architecture of the model is reduced, and most computations are simplified in TensorFlow Lite to make it work on devices' limited hardware. This step involves converting the trained model into a form consumable by libraries such as TensorFlow Lite Micro.

The third step is the development of microcontroller firmware based on the presented model. During this phase, the TensorFlow Lite model of the neural network is compiled and transferred to the microcontroller using libraries such as TensorFlow Lite Micro or Edge Impulse. The integration also helps to enable the microcontroller to communicate with the model and perform inferences in real time (Novac et al., 2021). Programmers create firmware to link the AI model with the input sensors and the output devices to gain real-world applications including, yet not limited to, health monitoring.

The last process is to cross-check and validate the system's functionalities by using actual or real data. The predictions are made using an artificial intelligence model in collaboration with the live biomedical signals from sensors that are processed by the microcontroller. Developers quantify the system to assess how efficiently it supports the required functions by fixing the accuracy, time to produce an inference, and energy utilization required (Lee & Brooks, 2006). Ad hoc testing checks for latent bugs to eliminate them through trial and error testing encounters such as memory overflow, latency, and sensor noise. Some of the methods used are signal preprocessing and firmware optimization to improve the dependability and efficacy of the system. This workflow ensures that the deployment of models in microcontrollers is structured, thereby making it easier to employ AI models that may be pivotal in high applications such as real-time health monitoring. This chart helps to explain complicated procedures and is best used as a guide in developing and researching situations.

International Journal of Core Engineering & Management
Volume-7, Issue-01, 2022 ISSN No: 2348-9510

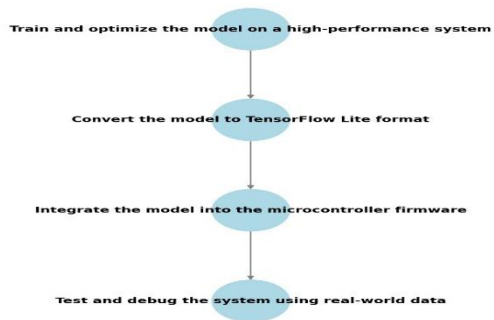


Figure 8: An Overview of Workflow for Deploying AI Models on Microcontrollers

V. REAL-TIME HEALTH MONITORING APPLICATIONS

5.1 Overview of Health Monitoring Systems

Over the last couple of years, the healthcare domain has shifted from a reactive monitoring system to a more proactive health monitoring system. These systems are put in place to monitor physiological parameters on a consistent basis and provide instantaneous information to the patient and caregivers. Real-time monitoring also increases the chances of identifying chronic diseases, cutting on hospitalization, and offering holistic solutions to people's health needs (Valero-Ramon et al., 2020). Health has to be checked frequently in humans with chronic diseases, including cardiovascular diseases, diabetes, and respiratory issues. Wearable devices and microcontroller-based systems, in particular, can help gather enough data on a patient's state so that it can be analyzed in real time and useful actions to be taken at the earliest opportunity, leading to the saving of many lives. For instance, a constant heart rate check can analyze arrhythmias, while a constant oxygen level analysis can alert patients of hypoxemia. All these developments are driven by Edge AI, where most of the computations are processed within the device rather than through clouds, thus making proceedings faster, private, and efficient in terms of power usage.

5.2 Biomedical Applications

5.2.1 Heart Rate Monitoring: Real-time health systems are widely used in utilizing information from Physiological sensors, and one of the most commonly used applications is in using the data collected by Heart rate sensors. Some of the conventional methods of monitoring include Holter monitors, which prove to be awkward and, therefore, can only be used for a short time. These limitations are, however, overcome by Edge-AI wearables that incorporate PPG and ECG sensors with microcontrollers. In Edge AI, deep learning models are embedded in microcontrollers to process raw data gathered from PPG sensors in real-time (Alessandrini et al., 2021). They are capable of identifying arrhythmias and counting things like HRV, which is used to determine stress and other cardiovascular factors. For example, when using lightweight neural networks pruned and quantized for operation in microcontrollers, such as Arduino Nano 33 BLE Sense, it becomes possible to process ECG signals in real time.

5.2.2 Respiratory Rate Monitoring: Respiratory rates can help diagnose conditions such as asthma, COPD, or pneumonia and assess them during management. Current methods, such as spirometry and manual counts, are usually performed in clinical settings and involve human labor. In other environments, Edge AI enables constant respiratory monitoring. Another physiological parameter is the respiratory rate (Greco et al., 2021). It can be recorded by automatic wearable devices having an accelerometer, gyroscope, RFID sensors, or respiratory inductance plethysmography. These models are implemented on dialects used in edge AI that work on these microcontrollers to analyze sensor data to detect breathing patterns, including tachypnea or apnea. For instance, by incorporating an Arduino microelectronic with a chest-worn sensor, data can be processed at the edge in order to detect abnormalities and hence notify users or medical practitioners promptly in case of any.

5.2.3 Blood Oxygen Level Detection: Continuous SpO2 (blood oxygen saturation) monitoring has become increasingly important lately, especially after the outbreak of the COVID-19 virus (Shah et al., 2020). Hypoxaemia can be an early sign of severe respiratory compromise, and thus, early identification is important. The possibility of utilizing portable devices with Edge AI means that SpO2 can be persistently monitored, in contrast to pulse oximeters, where it is only occasionally checked. This technology also enables portable devices such as microcontrollers and reflectance-based PPG sensors to monitor blood oxygen levels by analyzing the absorbance level of the corresponding tissue. In these devices, Edge AI algorithms local to these gadgets conduct computations to determine SpO2 and enable users to monitor their oxygen levels while exercising, during illness, or while sleeping. The present application is especially useful for people with diseases such as sleep apnea or chronic lung disorders.

5.3 Case Study: Deploying an AI-Based Heart Rate Monitor

5.3.1 Implementation Details: Real-time heart rate monitoring through Edge AI includes sensors, microcontrollers, and efficient AI algorithms. In this case study, an Arduino 33 BLE Nano sensor is connected to a pulse oximetry sensor that records the number of heartbeats per minute. The steps for implementation are as follows:

- **Sensor Integration**: An optical PPG sensor is interfaced with the microcontroller to acquire raw photo-plethysmographic signals, which measure blood volume variations in the microvascular tissue bed (Moreno Sánchez, 2015).
- **Data Preprocessing:** It then filters the raw signals and enhanced signals by reducing noise during motion or effects of light variation.
- **Model Deployment:** Empirically, a lightweight convolutional neural network (CNN) model is proposed to perform heart rate prediction, utilizing TensorFlow

International Journal of Core Engineering & Management
Volume-7, Issue-01, 2022 ISSN No: 2348-9510

Lite for optimization with heart rate datasets (He et al., 2018). This optimized model is then implemented in the Arduino microcontroller.

- **Real-Time Analysis:** The microcontroller analyzes sensor data using the deployed model. It computes heart rate in beats per minute (BPMs) and identifies other abnormalities, such as arrhythmia.
- **Output and Alerts:** The measurements are shown in numerations on an LCD screen, and alarms are given if the pattern changes.

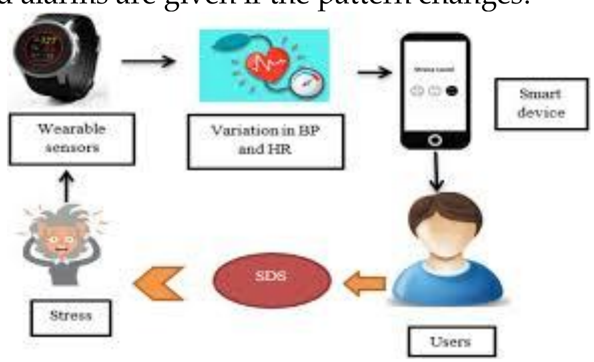


Figure 9: Advanced artificial intelligence in heart rate and blood pressure monitoring for stress management

5.3.2 Code Snippet: Processing Real-Time Sensor Data

Below is an example of Python-based TensorFlow Lite Micro code used to deploy a heart rate monitoring model:

```
python
#include <Arduino.h>
#include <TensorFlowLite.h>
#include "model.h"  // Include the trained model header
// Initialize the PPG sensor and TensorFlow Lite model
void setup() {
  Serial.begin(9600);
  setupPPGSensor();  // Function to initialize PPG sensor
  tflite::InitializeTensorFlowLite(); }
// Real-time data processing loop
void loop() {
  float sensorData = readPPGData();  // Function to read PPG sensor data
  float result = predictHeartRate(sensorData);  // TensorFlow Lite prediction
  Serial.print("Heart Rate (BPM): ");
  Serial.println(result);
  delay(1000);}
```

This implementation demonstrates how Edge AI models can process real-time physiological data on resource-constrained devices

5.4 Future Trends in Real-Time Monitoring

5.4.1 **Wearable Devices Powered by Edge AI:** Smartwatches are also changing with the implementation of Edge AI to create new forms of real-time health monitoring. The next generation of wearables will be compact, highly intelligent, and power-efficient devices capable of measuring a number of physiological parameters concurrently (Seneviratne et al., 2017). For example, next-generation smartwatches and fitness trackers could have multiple machine-learning algorithms for multimodal analysis, fusing heart rate, SpO2, and respiratory rate.

5.4.2 **Integration with IoT and Telemedicine:** Combined with Edge AI and the Internet of Things, it is possible to create progressively more sophisticated health monitoring systems. The IoT-endpoint edge AI devices can relay the health data to the cloud-level systems for further processing and sharing with physicians or surgeons. This integration improves telemedicine and offers precise distant consults and individualized treatment based on true real-time data.

5.4.3 **Advancements in Sensor Technologies:** Continuous advancements towards better sensor technologies are pushing the development of better wearable devices. The comfort and durability of the sensors used are a matter of ongoing research and development, and there is ongoing work to develop flexible and biocompatible sensors for long-term applications (Liu et al., 2018). Further, the association of heterogenous multiple sensors in smart environments yields rich datasets for the training and execution of Edge AI models, enhancing the accuracy of health monitoring applications.

5.4.4 **AI for Predictive Analytics:** The applications of Future Edge AI in health care may include preventive health care. This means identifying a health condition such that it can be treated even before symptoms begin to show. This can potentially prevent outcomes like Cardiac Arrhythmias or Asthma attacks through insights generated by Edge AI models on real-time data patterns.

VI. ADVANTAGES AND CHALLENGES OF USING EDGE AI IN HEALTHCARE

Decision-edge AI has become a revolutionary solution in healthcare, with substantial improvements in monitoring, diagnostics, and patients. Nevertheless, it also raises certain issues when implemented. Based on the above findings, this section seeks to discuss the major opportunities and main issues regarding Edge AI in biomedical.

6.1 Advantages of Edge AI in Healthcare

6.1.1 **Scalability:** Another advantage of Edge AI is that it is designed to be scalable. The application of edge computing devices such as microcontrollers and IoT sensors is mobile. It can be placed in many different locations, thereby allowing comprehensive health monitoring across the population without the need for extensive physical infrastructure. This factor also comes in handy when it comes to large areas where special well-served facilities are not well served. For instance, as portable devices, intelligent gadgets can continuously assess the clients' status and lessen the load on

International Journal of Core Engineering & Management
Volume-7, Issue-01, 2022 ISSN No: 2348-9510

health care facilities as well as broaden access to medical care. Furthermore, the concept of Edge AI also has a perspective in relation to the growth of the individualized approach to treatment (Patel et al., 2009). With the support of devices capable of local AI computations, healthcare providers can accumulate and analyze patient-specific data permanently, increasing the efficacy of the interventions performed.

6.1.2 Cost-Efficiency: Edge AI deploys intelligence at the perimeters away from centralized servers and the cloud to cut data transportation and internet bandwidth costs. This approach is important for the provision of health monitoring solutions in low-resource environments due to its affordability. Boards such as Arduino and ESP32, with the inclusion of intelligent features, are relatively cheap to allow for intelligent health monitoring functionalities (Aghenta, & Iqbal, 2019). Furthermore, Edge AI minimizes healthcare facilities' expenses since they can run the network independently of extensive IT help. Through timely diagnosis of illnesses and reduction in hospitalization due to complications of chronic diseases, there are reasonable costs that healthcare organizations can lower.

6.1.3 Enhanced Patient Outcomes: The present use of Edge AI ensures that the devices are capable of handling data locally, which is essential in healthcare since response has to be quick. For instance, the Smartwatch with Edge AI can identify a change in the client's heart rate, such as an abnormal heart rhythm or a low oxygen saturation rate, and notify caregivers in real-time. Such close contact can avert dangerous consequences to one's health and, in some cases, death. In addition, Edge AI protects the patient's data privacy by reducing the flow of the patient's personal and health reports to other servers. By processing data locally, it aligns itself with provisions of strict laws such as HIPAA and GDPR, and staff and patients trust the facilities (Determann, 2019).

Figure 10: Benefits of Artificial Intelligence in Healthcare

6.2 Challenges of Using Edge AI in Healthcare

6.2.1 Limited Hardware Resources: Smart sensors, microcontrollers, and other edge devices are limited in terms of computational power, memory, and energy resources.

International Journal of Core Engineering & Management
Volume-7, Issue-01, 2022 ISSN No: 2348-9510

These limitations make it rather difficult to traverse such deep-learning models directly on these devices. For instance, recent models for biomedical imaging or signal processing may have highly demanding computations that cannot be executed in microcontrollers (Alcaín et al., 2021). To minimize this, developers rely on a variety of optimization strategies, such as quantizing the model, pruning, and model compression. Although these approaches can optimize the model and the computation needed, they may do so at the expense of model efficiency.

6.2.2 **Maintaining Model Accuracy:** One major challenge presented by Edge AI systems is that the field exposes how deep learning models lose their accuracy when implemented in devices with low computational power. In the process of quantization or compression, optimized models may lose precision even though they are lightweight. For instance, decision-making could mean a precision reduction of a model from the high precision level of a 32-bit floating point to an 8-bit integer, which will see the level of accuracy in predictions drop (Brewer 1995). Further, due to variability in the biomedical data and diversification of patient demographic range, environment conditions, and sensor standards, the model fails to generalize. The reliability of an AI system across multiple contexts will only be as good as the testing done before its rollout and subsequent model updates.

Figure 11: AI implementation challenges in healthcare

6.2.3 **Maintaining Model Accuracy:** The main concerns of the healthcare ecosystem are the various types of devices and data formats, which make it complicated to achieve the necessary level of interoperability. The goals of the clinically deployed Edge AI systems include that these systems should work with currently existing medical devices, EHRs, and IoT systems (Rahmani et al., 2018). Nevertheless, the absence of a set of guidelines makes this integration challenging in most cases, thereby leading to problems such as the formation of isolated data structures. Based on the current nature of the wetware, it has become crucial for developers to pay attention to building open and friendly interfaces that enable systems to interact with many devices and platforms. This entails using common protocols such as HL7 FHIR for the exchange of health data and supporting the use of more than one type of sensor.

6.2.4 **Energy Constraints:** Healthcare applications require data to be processed at edge devices, which are predominantly battery-operated; hence ene, energy consumption is usually a paramount factor. Uninterrupted data processing and ongoing AI calculations will consume the power in the device, leading to shorter time between

International Journal of Core Engineering & Management
Volume-7, Issue-01, 2022 ISSN No: 2348-9510

charges, which makes it unsuitable for some continuous monitoring use cases (Pramanik et al., 2019). For instance, products such as wearable devices for measuring heart rate or blood oxygen levels require a balance between computational requirements with energy consumption to meet the need for continuous operation. New technologies, including deep low-power AI algorithms and energy catching and storage, are solving the problem of reducing power consumption. As pointed out earlier, these developments are crucial in supporting the deployment of sustainable Edge AI systems in healthcare.

Aspect	Advantages	Challenges
Scalability	Enables widespread deployment and personalized care.	Requires significant initial investment for device setup.
Cost-Efficiency	Reduces reliance on expensive cloud services.	Limited by the affordability of advanced microcontrollers.
Enhanced Outcomes	Provides rapid response times and real-time monitoring.	May face accuracy issues in optimized models.
Privacy and Security	Processes data locally, ensuring compliance with regulations.	Risk of local device breaches if not adequately secured.
Energy Efficiency	Can operate in low-power settings.	Energy constraints limit long-term usability.

Table 3: Pros and Cons of Edge AI in Biomedical Applications

6.3 Future Directions to Overcome Challenges

The present study has shown that Edge AI is integral to healthcare, and the potential of overcoming the challenges identified is feasible through a collaborative approach with various fields. Lack of resources can be tackled through advances in hardware design, for example through the incorporation of AI-specific microcontrollers on the chips. At the same time, improvements in software optimization such as NAS, as well as new trends like federated learning, help to make models more efficient, yet not less accurate (Park et al., 2015). Cooperation among all parties in relation to research, clinical practice, and policy development is also important. Aligning the practices for data exchange and device communication will help integrate Edge AI systems in healthcare settings. In addition, appropriate investments in developer education and training can help introduce innovative solutions.

VII. FUTURE OF EDGE AI IN BIOMEDICAL APPLICATIONS

Biomedical applications with Edge AI integration also suggest how biomedicine in general and healthcare in particular may be transformed into intelligent, personalized, real-time, and efficient approaches. This shift is informed by new trends and possible developments that hold kinetic possibilities in the innovation of health technology.

7.1 Emerging Trends

7.1.1 AI-Enabled Personalized Healthcare: Another area that has the most potential for Edge AI is the healthcare sector. Leveraging AI algorithms on the edge devices helps the medical systems to tailor treatment to the patient. Devices using microcontrollers for various wearable applications can also analyze data from body signals instantaneously, including heart rate, blood pressure, and glucose levels (Dias & Cunha, (2018). It also avoids overly generalized interpretations; instead, the findings are prompt to inform context-specific action without requiring cloud support. For instance, a patient with diabetes can wear an Edge AI wearable glucose monitor that, in addition to monitoring the patient's glucose level in real-time, offers the patient a nutrition and dosing plan. Incorporations such as these make method delivery more attentive and enhance treatment efficacy, thereby increasing patient compliance.

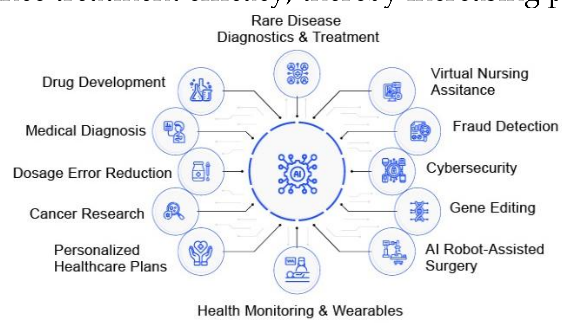


Figure 12: Artificial Intelligence (AI) in Healthcare

7.1.2 Integration with IoT for Smarter Health Solutions: Edge AI is fast transforming into an essential part of IoT elements, thus producing a well-connected environment. This integration makes it possible for the MD to work in cooperation with other devices so that a patient's health can be viewed holistically. For example, an IoT-based health monitoring system may combine the data from a Watch, Sleep tracker, and Heart rate monitor to give an all-encompassing picture of health (Farahani et al., 2018). Edge AI also reduces the time required for the data to be transmitted to centralized servers, which is an essential feature given that it reduces latency while maintaining the data's security. It is forecasted that with the advancement of IoT technology, smart healthcare systems will be developed where devices diagnose abnormalities and report to healthcare providers.

7.2 Potential Advancements

7.2.1 Improved Model Optimization Techniques: Nevertheless, using deep learning models on microcontrollers represents a major challenge because they need to be optimized for hardware limitations. In the last couple of years, if it is possible to quantify the model, prune the model, or distill the model, the size and computational power can be reduced significantly without greatly affecting the model's accuracy (Kumar, 2019). Future advancements in these fields are anticipated to open the outlook for implementing more elaborate models for further distributed devices. Other students are working on more extended forms of intelligent adaptive learning that can be adapted based on usage patterns. These improvements will enable the utilization of highly developed AI algorithms in complex and low-power ML microcontrollers for biomedical applications.

7.2.2 Enhanced Microcontroller Capabilities: Over recent years, microcontrollers have become more advanced in terms of computational power and energy. Some current examples of specific microcontrollers for AI are Arduino Portenta H7 and Raspberry Pi Pico W. These devices are empowered with a complex set of processors and specific accelerators for ML operations. Subsequent generations of microcontrollers include, for instance, on-chip AI processing, longer battery duration, and additional connectivity interfaces (Benini et al., 2006). These improvements will make it possible to implement global healthcare solutions with the help of AI technologies in problem areas, including remote or poorly developed regions of the world, providing great demand for quality medical care for the population.

VIII. PREDICTED GROWTH OF EDGE AI IN HEALTHCARE

Over the next decade, the usage of Edge AI in healthcare will likely grow exponentially. Stakeholders in different markets have estimated that the global Edge AI market will reach $1.3 billion in the current year and $8.3 billion in the year 2032, and healthcare is the key sector driving demand (Padilla et al., 2019). The need for real-time health monitoring, the rise in cases of chronic illnesses, and the enhancements in AI hardware and software drive the growth of AI technologies in healthcare. Their reference to the 'future 'has a very bright future for Edge AI applied to biomedical purposes supported by substantial technological development and the emergence of various sorts of healthcare breakthroughs. This paper enlarged how, through customization of care delivery, improving the IoT connection, and overcoming the challenges of hardware restraint, Edge AI is bound to revolutionize the healthcare sector. These advancements are beneficial to patients and, indeed, provide the foundation for making the existing healthcare system more effective, available, and fair.

International Journal of Core Engineering & Management
Volume-7, Issue-01, 2022 ISSN No: 2348-9510

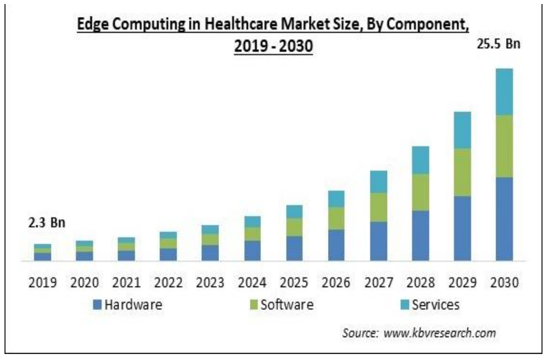


Figure 13: Prediction of Market Size & Trends of Global Edge Computing in Healthcare

IX. CONCLUSION

The adoption of Edge AI in biomedical applications is the turning point of the healthcare industry in delivering and perceiving healthcare. This technology allows data to be processed on edge devices such as microcontrollers at the same time so that it reduces latency while ensuring data privacy at the same time, not forgetting the aspect of cost. In contrast, Edge AI allows health monitoring devices to perform the analysis locally, promising less latency and faster decision-making a crucial aspect in emergency and constant health surveillance. Edge AI is most remembered for improving individual focus in the treatment course. By implementing efficient deep learning architectures within microcontrollers, wearable devices, and diagnosing tools, healthcare givers will be in a position to provide intervention. It serves to create personalized and timely delivery of health care needs coupled with the right data to enhance patient success. For instance, real-time measurement of the rate of heartbeat and glucose level enables people to respond to chronic diseases despite being stationed in rural or regions with low-internet infrastructure. Furthermore, the mobile and energy characteristics of microcontrollers enhance the extent to which they can be utilized in biomedical applications. For instance, the Arduino Nano 33 BLE and ESP32 microcontrollers are particularly effective in healthcare applications because of their low power consumption and relatively low cost. Such devices can ensure that portable, wearable, and sizeable diagnostics equipment is made available to populations. It is widely used for tracking respiratory rates and blood oxygen level analysis, and Edge AI has time and again shown the world how it can revolutionize healthcare.

Several difficulties are still present in the use of Edge AI in biomedical contexts. These factors of computational hardware impose model optimization requirements being restricted by memory size and computational and energy capacities. Quantization, pruning, and other related techniques are some of the most crucial steps needed to ensure AI models remain compact enough for the microcontroller. Moreover, the integration of the Edge AI systems with the current infrastructure of the healthcare system has not been an easy task yet. These issues thus need to be solved to further advance hardware design solutions, software solutions, and standardization endeavors in the future. In the future, better chips for Edge AI and better methods of optimization will open more opportunities in the health sector. Developing more advanced microcontrollers and better algorithms to improve the current AI will make it possible to train and deploy accurate models in resource-limited gadgets. Moreover, Edge AI, when combined with IoT, will definitely create optimized healthcare systems to deliver better and proactive techniques for consistent monitoring and care from remote locations. The main strengthening is that to implement the best-edge AI in biomedical applications, researchers, developers, and interested stakeholders have to work hand in hand. Education, research, and infrastructure expenditure will propel this innovative technology to lay foundations proactively to reach everyone. Edge AI is the key to the further development of the healthcare system, where the focus is on constant movement towards enhanced quality of services provided to patients and the overall effectiveness of the process. It will further evolve surely to complete the favorable change in the environment of health care.

REFERENCES

1. Aghenta, L. O., & Iqbal, T. (2019). Design and implementation of a low-cost, open source IoT-based SCADA system using ESP32 with OLED, ThingsBoard and MQTT protocol. *AIMS Electronics and Electrical Engineering*, *4*(1), 57-86.
2. Alcaín, E., Fernandez, P. R., Nieto, R., Montemayor, A. S., Vilas, J., Galiana-Bordera, A., & Torrado-Carvajal, A. (2021). Hardware architectures for real-time medical imaging. *Electronics*, *10*(24), 3118.
3. Alessandrini, M., Biagetti, G., Crippa, P., Falaschetti, L., & Turchetti, C. (2021). Recurrent neural network for human activity recognition in embedded systems using ppg and accelerometer data. *Electronics*, *10*(14), 1715.
4. Amin, S. U., & Hossain, M. S. (2020). Edge intelligence and Internet of Things in healthcare: A survey. *Ieee Access*, *9*, 45-59.
5. Banbury, C., Fedorov, I., & Reddi, V. J. (2020). Micronets: Neural network architectures for deploying on microcontrollers. *Proceedings of Machine Learning and Systems*.
6. Benini, L., Farella, E., & Guiducci, C. (2006). Wireless sensor networks: Enabling technology for ambient intelligence. *Microelectronics journal*, *37*(12), 1639-1649.
7. Brewer, E. A. (1995). High-level optimization via automated statistical modeling. *ACM SIGPLAN Notices*, *30*(8), 80-91.
8. David, R., Duke, J., Jain, A., Reddi, V. J., Jeffries, N., Li, J., ... & Warden, P. (2020). Tensorflow lite micro: Embedded machine learning on tinyml systems. *arXiv preprint arXiv:2010.08678*.
9. Determann, L. (2019). Healthy data protection. *Mich. Tech. L. Rev.*, *26*, 229.
10. Dias, D., & Paulo Silva Cunha, J. (2018). Wearable health devices—vital sign monitoring, systems and technologies. *Sensors*, *18*(8), 2414.
11. Dincer, I., & Rosen, M. A. (1999). Energy, environment and sustainable development. *Applied energy*, *64*(1-4), 427-440.

12. Farahani, B., Firouzi, F., Chang, V., Badaroglu, M., Constant, N., & Mankodiya, K. (2018). Towards fog-driven IoT eHealth: Promises and challenges of IoT in medicine and healthcare. *Future generation computer systems*, *78*, 659-676.
13. Faust, O., Hagiwara, Y., Hong, T. J., Lih, O. S., & Acharya, U. R. (2018). Deep learning for healthcare applications based on physiological signals: A review. *Computer methods and programs in biomedicine*, *161*, 1-13.
14. Ganesh, P., Chen, Y., Lou, X., Khan, M. A., Yang, Y., Sajjad, H., ... & Winslett, M. (2021). Compressing large-scale transformer-based models: A case study on bert. *Transactions of the Association for Computational Linguistics*, *9*, 1061-1080.
15. Gill, A. (2018). Developing a real-time electronic funds transfer system for credit unions. *International Journal of Advanced Research in Engineering and Technology (IJARET)*, *9*(1), 162–184. Retrieved from https://iaeme.com/Home/issue/IJARET?Volume=9&Issue=1
16. Greco, L., Percannella, G., Ritrovato, P., Tortorella, F., & Vento, M. (2020). Trends in IoT based solutions for health care: Moving AI to the edge. *Pattern recognition letters*, *135*, 346-353.
17. He, Z., Zhang, X., Cao, Y., Liu, Z., Zhang, B., & Wang, X. (2018). LiteNet: Lightweight neural network for detecting arrhythmias at resource-constrained mobile devices. *Sensors*, *18*(4), 1229.
18. Ibtehaz, N. (2020). Inferring health conditions through applying a fusion of machine learning and biomedical signal processing.
19. Kumar, A. (2019). The convergence of predictive analytics in driving business intelligence and enhancing DevOps efficiency. International Journal of Computational Engineering and Management, 6(6), 118-142. Retrieved https://ijcem.in/wp-content/uploads/THE-CONVERGENCE-OF-PREDICTIVE-ANALYTICS-IN-DRIVING-BUSINESS-INTELLIGENCE-AND-ENHANCING-DEVOPS-EFFICIENCY.pdf
20. Larminie, J., & Lowry, J. (2012). *Electric vehicle technology explained*. John Wiley & Sons.
21. Lee, B. C., & Brooks, D. M. (2006). Accurate and efficient regression modeling for microarchitectural performance and power prediction. *ACM SIGOPS operating systems review*, *40*(5), 185-194.
22. Liu, Y., Wang, H., Zhao, W., Zhang, M., Qin, H., & Xie, Y. (2018). Flexible, stretchable sensors for wearable health monitoring: sensing mechanisms, materials, fabrication strategies and features. *Sensors*, *18*(2), 645.
23. Merenda, M., Porcaro, C., & Iero, D. (2020). Edge machine learning for ai-enabled iot devices: A review. *Sensors*, *20*(9), 2533.
24. Moreno Sánchez, J. (2015). *Esport, salut i HRV: monitorització psicofisiològica de l'estrès i la recuperació amb dispositius mòbils*. Universitat Autònoma de Barcelona,.
25. Novac, P. E., Boukli Hacene, G., Pegatoquet, A., Miramond, B., & Gripon, V. (2021). Quantization and deployment of deep neural networks on microcontrollers. *Sensors*, *21*(9), 2984.
26. Nyati, S. (2018). Revolutionizing LTL Carrier Operations: A Comprehensive Analysis of an Algorithm-Driven Pickup and Delivery Dispatching Solution. *International Journal of*

Science and Research (IJSR), 7(2), 1659–1666. Retrieved from https://www.ijsr.net/getabstract.php?paperid=SR24203183637

27. Nyati, S. (2018). Transforming telematics in fleet management: Innovations in asset tracking, efficiency, and communication. *International Journal of Science and Research (IJSR)*, 7(10), 1804–1810. Retrieved from https://www.ijsr.net/getabstract.php?paperid=SR24203184230.
28. Nyländen, T. (2018). Application specific programmable processors for reconfigurable self-powered devices.
29. Padilla, P., Vonortas, N. S., Dranev, Y., Belousova, V., & Boudard, E. (2019). Analysing the Deployment of Blockchain and Distributed Ledger Technologies in the Financial Sector. *Unpublished manuscript, Background paper for Europe*, *4*.
30. Park, D. S., Chao, H. C., Jeong, Y. S., & Park, J. J. (2015). Advances in computer science and ubiquitous computing. *Lecture Notes in Electrical Engineering*, *474*.
31. Patel, V. L., Shortliffe, E. H., Stefanelli, M., Szolovits, P., Berthold, M. R., Bellazzi, R., & Abu-Hanna, A. (2009). The coming of age of artificial intelligence in medicine. *Artificial intelligence in medicine*, *46*(1), 5-17.
32. Pramanik, P. K. D., Sinhababu, N., Mukherjee, B., Padmanaban, S., Maity, A., Upadhyaya, B. K., ... & Choudhury, P. (2019). Power consumption analysis, measurement, management, and issues: A state-of-the-art review of smartphone battery and energy usage. *ieee Access*, *7*, 182113-182172.
33. Qiu, M., Gao, W., Chen, M., Niu, J. W., & Zhang, L. (2011). Energy efficient security algorithm for power grid wide area monitoring system. *IEEE Transactions on Smart Grid*, *2*(4), 715-723.
34. Rahmani, A. M., Gia, T. N., Negash, B., Anzanpour, A., Azimi, I., Jiang, M., & Liljeberg, P. (2018). Exploiting smart e-Health gateways at the edge of healthcare Internet-of-Things: A fog computing approach. *Future Generation Computer Systems*, *78*, 641-658.
35. Raza, M., & Aslam, M. W. (2021). Edge artificial intelligence: Challenges and opportunities. *Journal of Internet of Things and Machine Learning, 2*(1), 20-30.
36. Seneviratne, S., Hu, Y., Nguyen, T., Lan, G., Khalifa, S., Thilakarathna, K., ... & Seneviratne, A. (2017). A survey of wearable devices and challenges. *IEEE Communications Surveys & Tutorials*, *19*(4), 2573-2620.
37. Shah, S., Majmudar, K., Stein, A., Gupta, N., Suppes, S., Karamanis, M., ... & Patte, C. (2020). Novel use of home pulse oximetry monitoring in COVID-19 patients discharged from the emergency department identifies need for hospitalization. *Academic Emergency Medicine*, *27*(8), 681-692.
38. Sherstinsky, A. (2020). Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Physica D: Nonlinear Phenomena*, *404*, 132306.
39. Sun, L., Jiang, X., Ren, H., & Guo, Y. (2020). Edge-cloud computing and artificial intelligence in internet of medical things: architecture, technology and application. *IEEE access*, *8*, 101079-101092.

International Journal of Core Engineering & Management
Volume-7, Issue-01, 2022 ISSN No: 2348-9510

40. Taivalsaari, A., Mikkonen, T., & Pautasso, C. (2021, May). Towards seamless IoT device-edge-cloud continuum: Software architecture options of IoT devices revisited. In *International Conference on Web Engineering* (pp. 82-98). Cham: Springer International Publishing.
41. Valero-Ramon, Z., Fernandez-Llatas, C., Valdivieso, B., & Traver, V. (2020). Dynamic models supporting personalised chronic disease management through healthcare sensors with interactive process mining. *Sensors*, *20*(18), 5330.
42. Vandersteegen, M., Van Beeck, K., & Goedemé, T. (2021). Integer-only cnns with 4 bit weights and bit-shift quantization scales at full-precision accuracy. *Electronics*, *10*(22), 2823.
43. Wang, Y. E., Wei, G. Y., & Brooks, D. (2019). Benchmarking TPU, GPU, and CPU platforms for deep learning. *arXiv preprint arXiv:1907.10701*.
44. Warden, P., & Situnayake, D. (2019). TinyML: Machine learning on microcontrollers. Sebastopol, CA: O'Reilly Media.
45. Warden, P., & Situnayake, D. (2019). *Tinyml: Machine learning with tensorflow lite on arduino and ultra-low-power microcontrollers*. O'Reilly Media.
46. WoldeMichael, H. G. (2018). Deployment of AI Model inside Docker on ARM-Cortex-based Single-Board Computer: Technologies, Capabilities, and Performance.
47. Xue, J., & Yu, L. (2021). Applications of machine learning in ambulatory ECG. *Hearts*, *2*(4), 472-494.
48. Zha, S., Luisier, F., Andrews, W., Srivastava, N., & Salakhutdinov, R. (2015). Exploiting image-trained CNN architectures for unconstrained video classification. *arXiv preprint arXiv:1503.04144*