UNIVERSITY of RWANDA

ACEIoT

Website: www.aceiot.ur.ac.rw
Mail: aceiot@ur.ac.rw

COLLEGE OF SCIENCE AND TECHNOLOGY

AFRICAN CENTER OF EXCELLENCE IN INTERNET OF THINGS

Offline Prediction of Cholera in Rural Communal Tap Waters Using Edge AI inference

Submitted in partial fulfilment of the requirements for the award of

MASTER OF SCIENCE IN INTERNET OF THINGS

WIRELESS INTELLIGENT SENSOR NETWORKING

(MSC in IoT-WISENET)

Submitted by

Marvin Muyonga Ogore - Reg. No. 220014154

Submission Date: December, 2021

UNIVERSITY of RWANDA

Website: www.aceiot.ur.ac.rw
Mail: aceiot@ur.ac.rw

COLLEGE OF SCIENCE AND TECHNOLOGY

AFRICAN CENTER OF EXCELLENCE IN INTERNET OF THINGS

Offline Prediction of Cholera in Rural Communal Tap Waters Using Edge AI inference

Submitted in partial fulfilment of the requirements for the award of

MASTER OF SCIENCE IN INTERNET OF THINGS

WIRELESS INTELLIGENT SENSOR NETWORKING

(MSC in IoT-WISENET)

Submitted by

Marvin Muyonga Ogore - Reg. No. 220014154

Under the supervision of

Supervisor: **Dr. NSENGA Jimmy**

Co-Supervisor: **Dr. NKURIKIYEYEZU Kizito**

December 2021

DECLARATION

I hereby declare that all information in this document is original and it has never been presented in any University or other Institutions of Higher Learning. I also declare that, as required by rules I have fully cited and referenced all material and results that are not original to this work.

Student Name and Number

Marvin Ogore

220014154

Date: 28/11/2021

BONAFIDE CERTIFICATE

This is to certify that the dissertation report work is a record of the original work done by Mr. Marvin Muyonga Ogore a postgraduate student in MSc in Internet of Things (IoT) with specialization in Wireless Intelligent Sensors Networks (WISeNet), at University of Rwanda – College of Science and Technology in African Center of Excellence in Internet of Things (UR/CEST/ACEIoT). We certify that the work reported doesn't form a part of any other research project.

Supervisor:

Dr. NSENGA Jimmy

Signature:

26/11/2021

Co-Supervisor:

Dr. NKURIKIYEYEZU Kizito

Signature:

The Head of Masters and Training ACEIoT

Dr. James RWIGEMA

Signature:

ACKNOWLEDGEMENTS

First of all, I would like to thank the Almighty God, who gave me the grace to accomplish this work. I would like to express my sincere and special thanks to my supervisory team consisting of Dr. Jimmy NSENGA and Dr. NKURIKIYEYEZU Kizito for their great guidance towards the achievement of this work from which I gained invaluable knowledge. Secondly, I wish to thank the African Center of Excellence in Internet of Things, College of Science and Technology, University of Rwanda, especially the ACEIoT authorities for allowing me to pursue this Masters and for the skills they gave to me. I also wish to thank Mr. Peter Ogore for invaluable advice and emotional support together with Umulisa. Furthermore, my mataifa family as well as my classmates have been integral in all academic activities for their direct and indirect contribution to the achievement of my objectives. My acknowledgement goes to everyone who supported me in my everyday life to reach this moment.

I wish blessings upon you all.

ABSTRACT

Africa accounts for 54% of the world disease burden due to the lack of access to safe drinking water, with the majority of rural area populations or endemic zones getting access to water through potentially unsafe communal water taps. Unfortunately, the expensive laboratory processes and resources used in water processing centers to detect water-borne diseases like cholera cannot be massively deployed on all those taps to guarantee safe water for everyone, anywhere at any time. Thanks to the integration of Internet of Things (IoT) and Artificial Intelligence (AI), the prediction of water-bone cholera can be done by monitoring water's physicochemical patterns. However, related state of the art IoT/AI solutions rely on a cloud-centric architecture with edge water parameter sensors sending collected data to the cloud for inference. Unfortunately, anytime wireless connectivity is not always guaranteed in rural areas, but also it is very consuming in terms of energy for a system expected to run several years without maintenance. Last but not least, low latency detection is mandatory to warn the tap user on time. This Master thesis research project focuses on prototyping design and development of an offline edge AI rapid water-bone cholera detector kit pluggable into existing taps to lower the cost of mass deployment. Our simulation results in an embedded context show a good accuracy of edge inference with respect to live cloud classification.
Keywords: Cholera, Smart Water, Physicochemical water parameters, IoT, Edge AI,

LIST OF ACRONYMS

ACEIoT: African Center of Excellence in Internet of Things

AI: Artificial Intelligence

APW: Alkaline Peptone Water

GSM: Global System for Mobile

HW: Hardware

IDE: Integrated Development Environment

IoT: Internet of Things

KNN: K-Nearest Neighbor

LAMP: Loop-mediated isothermal amplification

LCD: Liquid Crystal Display

MCU: Microcontroller Unit

ML: Machine Learning

TDS: Total Dissolved Solids

SVM: Support Vector Machine

SW: Software

LIST OF FIGURES

TABLE OF CONTENTS

CHAPTER I: GENERAL INTRODUCTION

1.1 INTRODUCTION

The quality of drinking water in taps goes down when the water goes into the distribution system. Tap water is used by more than 50% of the people in urban settings [1] or at standpipes. Pipes in rural areas and informal settings are centrally placed in locations where people can access in large numbers during specific times of the day. Any contamination on the water can affect the whole community. Studies show that in some African settings, waste management systems are not implemented as humans defecate in open spaces. Humans who defecate in open spaces fall within a range of 4%-16% from the study done in Cameroon and as a result rain water comes into contact with the human wastes in open spaces and can find its way into sources of water which shows why cholera outbreaks occur during the rainy season [1] . In Uganda, investigations were done to show presence of fecal matter in water sources [2] . Results show that the further away you move from water sources, that is, further along the pipeline, the water becomes more contaminated. Furthermore, some water companies that do not sufficiently chlorinate water at the source and that is a cause for cholera in household settings as depicted in India how when the water system has leaks there is a high possibility of the drinking water coming into contact with the sewerage system [3]. The contamination can be linked to stagnation of sewage water around water pipes.

1.2. BACKGROUND AND MOTIVATION

Cholera is an acute abdominal infection characterized watery diarrhea and vomiting which, if left untreated, the disease can quickly lead to severe dehydration and death [4] . The World Health Organization (WHO) estimates yearly cholera infections at 1 million to 4 million people and up to 143 000 deaths. In 2016 Africa accounted for 54% of the cases reported [5] . Death incidences decline with communities' ability to access clean drinking water and food. Currently, cholera outbreaks mostly occur in rural areas and endemic zones of developing countries due to lack of safe drinking water and poor sanitation spread through local transmission [2] [6]. In those regions, shared communal tap waters are the main access point to water for most populations. The level of preparedness for health facilities in Africa show a lot of weakness presented in the sparse number of facilities in rural settlements [7]. This shows the limited capability in sub-Saharan Africa in the surveillance of cholera which is fully dependent on lab facilities provided by health facilities.

1.3. PROBLEM STATEMENT

Tap water in households is shown to be a cause for contraction of waterborne diseases such as cholera. The existing water quality analysis methods using Internet of Things which involves the use of interconnection of devices, people or objects, focus on pollution with a provision of how it affects health in a general way as highlighted in [14] and therefore lack a way to link the quality of water to any specific disease. Improvements have been implemented on existing IoT based water quality infrastructure with the hybrid combination of IoT and ML. Despite that, the current IoT/ML solutions are centered on the cloud which reduces viability in rural areas which face limitations of wireless internet connectivity [8]. Furthermore, wireless communication is highly energy consuming for a system expected to run for several years without maintenance, either harvesting ambient energy or running on small-sized batteries. State of the art techniques to detect cholera are made for laboratory-like settings involving expensive, complex and high latency processes and resources. Those solutions are not practical for a mass deployment on all communal water taps in Africa as a way to warn users in real-time about a potential waterborne disease such as cholera. To give an example, one of the most used processes is based on a product called Alkaline Peptone Water (APW) which costs around $29/ml [9]. With 1% APW per 500ml samples, it takes up to 24-48 hours depending on the enrichment process of samples in order to increase cell count in samples that have low cells per volume [10],[11] . This process furthermore requires the presence of highly skilled laboratory staff to carefully drive the all process. This solution is certainly not scalable to the thousands of water taps already deployed in rural areas.

It is therefore necessary to detect water quality and waterborne diseases at the edge. This creates a need for a device that can perform water quality analysis for waterborne diseases at the edge using machine learning. Machine learning is important as it reduces the need for human expertise especially in regions where disease experts are scarce. Importantly, warning the water tap consumer that he/she is potentially taking a risk to use the tap should be fast/immediate, thus making low latency of inference a strong requirement as well.

1.4. STUDY AIM AND OBJECTIVES

1.4.1 AIM

The aim of this research is to enable day to day cheap, laboratory-free water-borne disease diagnosis for the Rural African population in order to enable them to have access to clean and safe tap water. This aim will be achieved through prototyping a diagnostic kit for water borne disease detection relying on open source IoT and AI technologies. This is in line with the global development goals to provide clean water and sanitation. Chosen for this scope of this study is cholera. This study aims at a quick detection of cholera in water in effort to prevent an epidemiological crisis and improve on the reporting of the occurrence of cholera in water.

1.4.2 OBJECTIVES

There are several ways to make sure the aims are achieved. For this study, the aims, the objectives and milestones to be achieved are as follows:

1. To understand how physico-chemical parameters affect existence of cholera cells in water
2. To identify IoT and embedded technologies that are used in sensing and transmit physico-chemical parameters of water.
3. To identify open source AI and Non-AI technologies that can be used in the detection of waterborne diseases and identify possibilities for embedded operationalization.
4. To design and simulate a smart embedded diagnostic kit for waterborne disease from parameters of water.

1.5 HYPOTHESES

The hypotheses of this research are as follows:

1. IoT Sensing technologies are able to efficiently collect physico-chemical water parameters of water.

2. ML can be leveraged to detect cholera from physico-chemical water parameters
3. There is available technology that can be used to produce ML models adapted for embedded device

1.6 SIGNIFICANCE OF THE STUDY

Cholera reporting is very poor especially in African countries. Having an IoT device that can report cholera causes to cloud platforms will help health officials have a clear view of which areas are affected and necessary mitigation and treatment measures will be taken to ensure the disease is contained. The numbers provided in 2017 provide an incomplete assessment due to lack of reporting by many countries [12] . African countries showed a drop in the reporting of cholera cases in 2017. The disease could be more severe than speculated due to under reporting. Therefore, a solution that can report consistently will give a more accurate epidemiological view.

Despite increased levels of chlorinated water at the source, cholera has the ability to emerge in scenarios where floods occur or shortage of water as shown in [13] . According to WHO, the death rate of cholera is about 1.8% and in 2016, 17 African countries reported cholera cases despite the low reporting trends in Africa [5]. This project can help the overall African setting to be healthier and bring the death rate and number of cases further down through early detection techniques and monitoring the water the end user intakes.

From WHO, online JMP data of 2017, 29% of the population in rural settings in Rwanda can access improved piped drinking water while a staggering 75% of the population in Rwanda's urban settings can access improved piped drinking water. A total number of 1,567,038 in urban areas and 2,978,400 in rural areas will potentially benefit from having piped water analyzed at the tap.

1.7 ORGANIZATION OF THE STUDY

This work is organized into five chapters:

- Chapter I: General Introduction, this chapter focuses on Objectives of the project, Problem statement, Hypothesis and the significance highlighting the potential impact.
- Chapter II: Literature review, this offers theoretical concepts regarding the related work done by the other researchers.

- Chapter III: Research Methodology;
- Chapter IV: focuses on the results of the projects and discussions.
- Chapter V: The last chapter is made up of the conclusion and recommendation for further improvements in the project.

1.8 CONCLUSION

In this chapter, a brief description of the project has been stated that introduces the setting of and background. Furthermore, the aim and objectives of the study are detailed. The problem statement highlights the scope of what is to be overcome and proposes techniques and technologies that will be used to prototype in accordance with the assumptions the project is planning to achieve at the end of design and implementation. The related works and literature review are detailed in the following chapter.

CHAPTER II: LITERATURE REVIEW

2.1 INTRODUCTION

This chapter captures the relevant studies, research and state of the art technologies done by other researchers, individuals and stakeholders in the field and provides an overview of existing technologies that have been used to drive our approach to a solution towards offline Water-borne detection using edge AI.

2.2 WATER POLLUTION

In [14], we see how the move to urban areas has caused humans to over utilize land especially in revolutionizing industries that drive the urban setting leads to deteriorating quality of water. The paper analyses tap water, surface water which covers rivers and lakes and also sea water as water sources. The parameters measured are pH, conductivity, oxidation reduction levels, and temperature. This is an overall system that aims to ensure the water is not only safe for human consumption but water that aquatic water can thrive on considering diseases and death of aquatic life. Any deviation from the range of the set baseline is sent to an FTP server in the local network rather than a cloud platform. Communication technologies used are cellular with the use of a GSM shield. The analysis of the tap water required manual extraction and testing by inserting sensors in water that had been fetched.

With the help of IBM cloud Watson one can be able to predict the deteriorating quality of water using statistical methods by attaching parameter values to a timestamp [15] . Parameters measured against their respective thresholds are turbidity, pH and temperature. The objective is to predict whether the quality of water is deemed to go low. No Machine learning methods are employed here. Furthermore, the system is not linking the quality of water to any specific disease. This helps to determine the trend of water quality and predict whether the water will be safe in a future setting as well as a real time monitoring of the water quality.

The paper [16] , urban areas of India lack proper waste management which causes depreciation of quality in soils and water of soil and water. To tackle this challenge, various sensors are distributed to measure parameters like pH, conductivity, and dissolved oxygen, turbidity, so as to monitor the water quality in water sources that is, an open well and a freshwater canal. Taps in the region are not studied as they are presented to be dry. The proposed IoT framework shows how soil and water pollution are interrelated. The parameters for the water resources are: Temperature, pH, Turbidity, electrical conductivity sensors, Chemical Oxygen Demand (COD), Total hardness ($CaCO_3$), Total Dissolved Solids (TDS), Magnesium (Mg), and Chloride (Cl). Data is sent to a gateway via ZigBee communication and data is sent to the cloud via Wi-Fi.

2.3 CONVENTIONAL TECHNIQUES FOR WATER-BORNE DISEASE DETECTION

2.3.1 LAB TECHNIQUES

Conventional cholera detection in many laboratory settings and cholera detection studies use enrichment steps with alkaline peptone water APW at a pH of 8.0. Stool samples are incubated for 4–6 hours at 37 °C while water samples take up to 24 hours [4],[17],[3]. Additional steps are needed for further characterization and confirmation of V. cholerae such as in TCBS agar characterized by a shiny yellow color due to sucrose fermentation [10] . Preparing the samples for testing requires a lot of preparation steps and skilled human intervention.

Other techniques for detecting cholera require DNA sequencing before detection or expensive PCR equipment. As examples, (1): the loop-mediated isothermal amplification [10], [5] , (2) Multiplex polymerase chain reaction (PCR) as presented in [18] , [19] and (3) The Molecular beacon-based real-time nucleic acid sequence-based [10] . Furthermore, Cholera Toxin Gene encoding in V. Cholerae as shown in [5] requires PCR methods to generate assays or DNA complementary base pairing knowledge. Vibrio cholerae can also be extracted from water samples but the method is inefficient due to the process requiring incubation at 35–37 °C for 18–24 h and a testing kit that costs $716.00/each as shown in [20].

2.3.2 Lab on a chip Techniques

A sensor node is developed to be used in the detection of waterborne pathogens in environmental water [21] . Researchers focus on E. coli bacteria. The color change of water is observed once water is mixed with reagents. For a real-time analysis, a live image captured by a webcam is sent to the web browser of the end user with a Wi-Fi connected sensor node. An Arduino and a connected motor control sampling rates of the pumps. The process of testing the water is repeated until the reagents are depleted. From the web browser, users can post to a python server to record for long term data storage. Human intervention is required to determine color change and send data manually to the python server. The server is internal and is running on the embedded system.

Other methods shown in [22],[23]are effective but require the assembly of materials such as microfluidic chip that is expensive to develop and the use of cholera detection methods using PCR and LAMP methods that require the purchasing and refilling of expensive fluids such as Streptavidin, fluorescent nanoparticles, LAMP master mix and a biotinylated LF primer [24], [25]. The process of assembling such materials requires experts and is complicated.

2.4 AI WATER QUALITY MEASUREMENT

An intelligent water quality system is presented in [26] . A comparison of Machine learning algorithms is done. The algorithms are random forest and K-Nearest Neighbor (KNN) and KNN in the results is more accurate. Data is collected from sensors and sent to a Django server where data preprocessing occurs and prediction is carried out. The implementation is done using a NodeMCU with attached sensors. The model makes informed decisions on how the water can be used based on drinking, industrial usage, agricultural use, home usage or if it is dirty water (unusable). There is no clear impact on water quality metrics associated with diseases.

Emerging technologies such as IoT, machine learning and cloud technologies are implemented to monitor the quality of water in rural areas [27] . A pH sensor, turbidity sensor and temperature sensor are attached to a NodeMCU which sends data to Azure hub. Azure hub storage stores structured data. The machine learning model is used to predict the weather and adjust the cooler and heater accordingly. When the turbidity threshold is surpassed, relevant authorities are alerted to take necessary action. This paper does not focus on implementing machine learning models on parameters such as pH and turbidity to predict the water quality.

In Tanzania a machine learning model known as XGBoost is deployed to monitor weather patterns in order to predict the occurrence of cholera [28] . The dataset used is irregular, meaning there is no consistency in the data. The goal is to be able to predict the occurrence of cholera based on the season as influenced by weather patterns. The project does not look into water as a cause of cholera therefore not associated with water quality measurement but rather cholera predictability based on climate.

An IoT technology is used to assess how human activities have caused water pollution in tap water [29] . The proprietary solution is able to recognize six pollutants namely Sulphur Acid, Phosphoric Acid, Acetic Acid, Formic Acid, Hydrogen Peroxide, Ammonia. Machine learning technologies are deployed where K-Nearest Neighbor (KNN) proves to be a more accurate classifier for implementation. The aim is to deploy a proprietary sensor node and an embedded processor using machine learning to be able to identify and classify the six contaminants. Furthermore, the performance of different classifiers is analyzed in order to make necessary trade because the machine learning classifiers take up a lot of computing resources. Tap water is fetched in a beaker for analysis using IoT, aspects of running water are not highlighted.

2.5 THE SUMMARY AND IDENTIFIED GAP

State of the art techniques to detect cholera are made for laboratory-like settings involving expensive, complex, high latency processes and resources and highly skilled laboratory staff to carefully drive the all process. Those solutions are not practical for a mass deployment on all communal water taps in Africa as a way to warn users in real-time about a potential waterborne disease such as cholera. Using the Internet of Things (IoT) sensors, has been proven that one can monitor water's physicochemical properties like potential hydrogen (pH), oxidation and reduction potential (ORP), conductivity, temperature, turbidity and so on. The sensed water data is transmitted to the cloud where trained machine learning (ML) models perform water quality analysis (inference) to predict water safety, either in general or specifically against a given water-borne disease like cholera. Solutions centered on the cloud for inference are not viable in rural areas which face limitations of wireless internet connectivity.

Our study’s approach is to design and develop a prototype of an offline edge AI rapid water-bone cholera detector kit, pluggable on existing taps to instantaneously infer water safety from physicochemical patterns of water

2.6 CONCLUSION

A clear gap on how Cloud centric solutions are not suitable for a rural based community mass deployment has been highlighted. Existing solutions still require laboratory-like settings, labor and resources that cannot be feasible in resource constrained countries especially in sub-Saharan Africa. Therefore, this study proposes using IoT technologies to capture physico-chemical parameters of water geared towards inference of water borne diseases using Edge AI.

CHAPTER III: RESEARCH METHODOLOGY

3.1 INTRODUCTION

This chapter describes how the research will be conducted in order to achieve the stated objectives. It demonstrates the research design and procedures, sample selection, data collection techniques and instruments. The scientific methods for conducting research have been stated in this section as well as the experimental research approach.

3.1.1 RESEARCH APPROACH AND DESIGN OF THE SYSTEM

The study approach will be in two phases:

- **HW/SW embedded simulation**

In our first step, we simulate the edge AI inference in a virtual embedded platform simulator before testing on a real development board. In the simulation context, the performance of edge AI inferencing is validated by reading test data from a file and comparing the inference results with the ideal results calculated by the cloud platform used during the training.

- **Embedded Hardware implementation**

This step involves the process of combining different physical embedded components in effort to achieve specific objectives against the study hypothesis, data collection and evaluation of the hardware system.

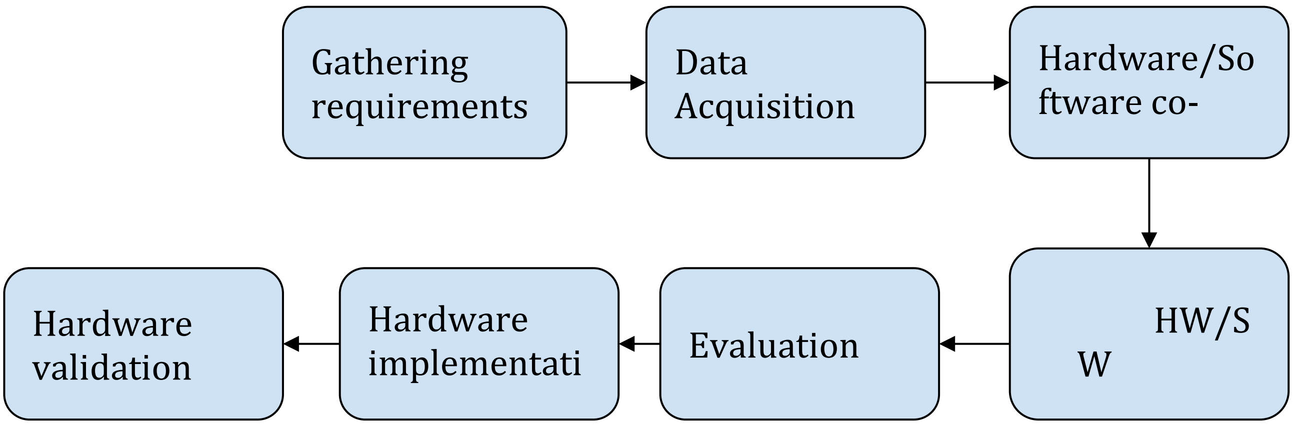

FIGURE 1: RESEARCH METHODOLOGY FLOW PROCESS

3.2 HARDWARE/SOFTWARE CO-DESIGN

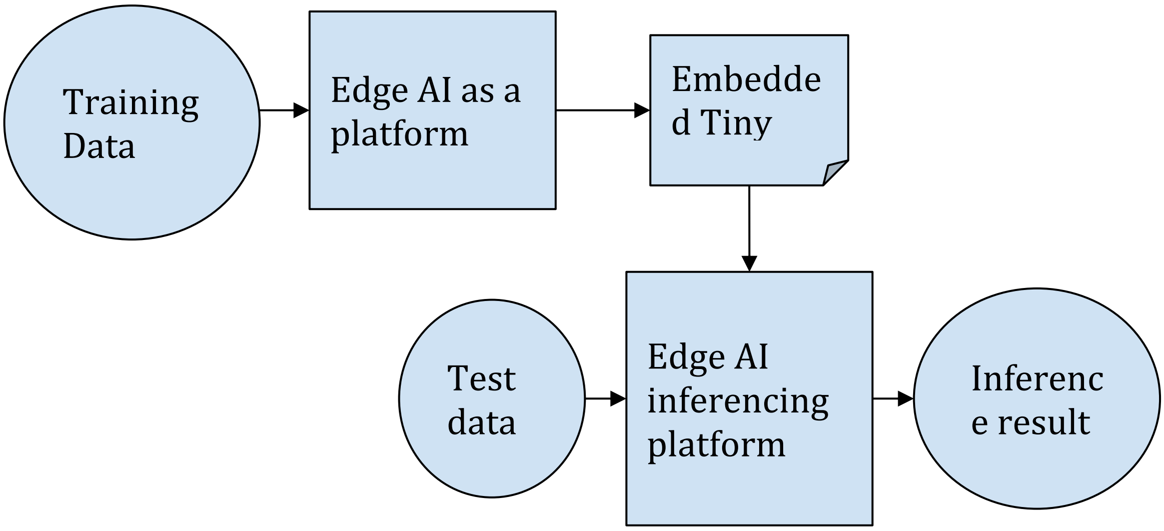

FIGURE 2: HARDWARE/ SOFTWARE CO-DESIGN

3.3 EDGE AI PROTOTYPING TOOL STACK

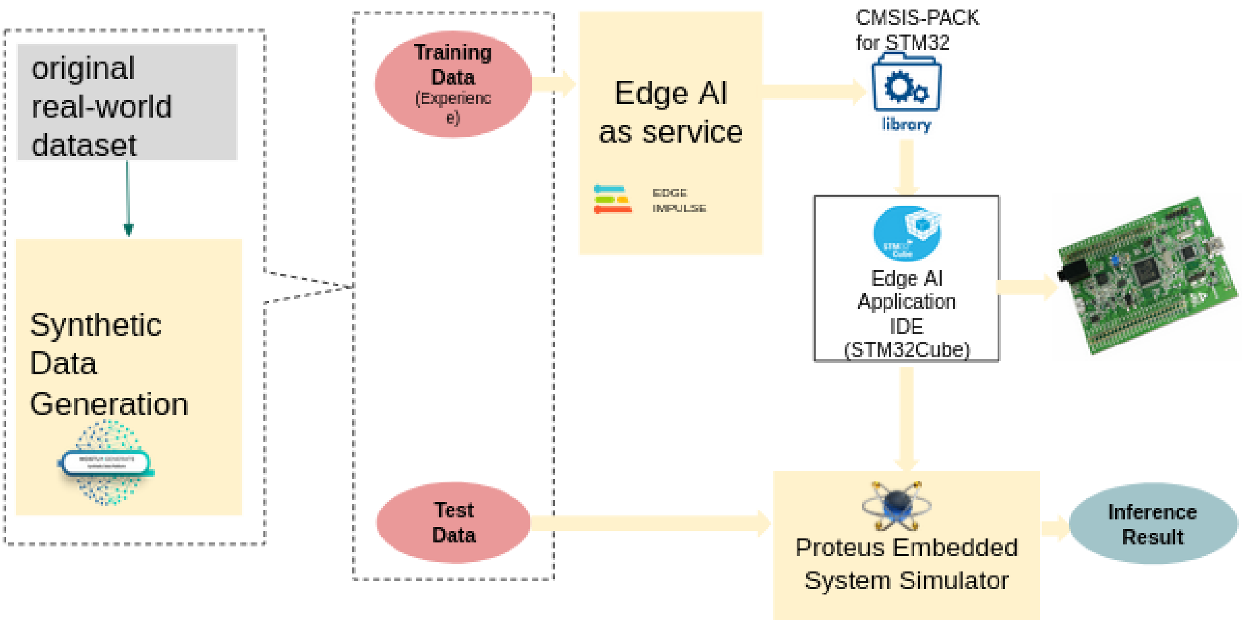

FIGURE 3: PROTOTYPING TOOL STACK

3.4 SOFTWARE REQUIREMENTS

3.4.1 SOFTWARE TOOL STACK

The software tools used as per figure 3 are as follows:

1. MOSTLY GENERATE-Mostly GENERATE is a free synthetic data generator platform which allows the simulation of realistic & representative synthetic data. By automatically learning patterns, structure and variation from existing original real-world dataset. It leverages generative deep neural networks. This way a dataset that is realistic, can be acquired from the already existing dataset, freely processed and analyzed. Generally, the data does not replicate already existing values in the file. Performance during training shows that the data generated performs exceptionally well when tested on real world data. This in effect validates the use of synthetic data.
2. Edge Impulse is a platform that enables IoT developers to embed ML and deep learning techniques onto resource constrained embedded devices. Edge Impulse as a development platform for embedded machine learning, enables development for machine learning on embedded devices for sensors, audio, and computer vision, at scale in order to solve real problems using machine learning in embedded solutions, speeding up development time from years to weeks.
3. STM32 Cube IDE. The STM32CubeIDE is chosen to integrate STM32 configuration and project creation functionalities from STM32CubeMX to offer all-in-one tool experience for simulation purposes. The IDE is an all-in-one multi-Operating System development tool, which is part of the STM32Cube software ecosystem. STM32CubeIDE is an advanced C/C++ development platform with peripheral configuration, code generation, code compilation, and debug features for STM32 microcontrollers and microprocessors. Arduino IDE
4. Proteus: For embedded engineers, Proteus VSM bridges the gap in the design life cycle between schematic capture and PCB layout. It enables you to write and apply your firmware

to a supported microcontroller on the schematic and then co-simulate the program Proteus allows you to interact with the design using on screen indicators such as LED and LCD displays.

3.5 DATA ACQUISITION: DATA ACQUISITION AND SYNTHETIC DATA GENERATION

3.5.1 THE ORIGINAL REAL-WORLD DATASET

The dataset used for this study is acquired from a study of observed cholera cases in the Katana health zone shown in figure 4. Physico-chemical characteristics of Lake Kivu, temperature and salinity, were collected against the concentration of V. cholera found in lake water and inside the gills and abdomen of fish [30]. The dataset was collected twice a month over a period of 48 sampled weeks over the 2016–2017. The dataset consists of 289 observations. As the first step to understand the data and make sense of it, feature engineering is performed to help understand the relationship between the features and the labels. Despite having the two features being correlated, it is established that temperature and salinity are important in the occurrence of vibrio cholerae in water. Furthermore, you cannot derive temperature directly from salinity which supports the argument of having the two features to be used in the ML process. The resulting dataset is small due to the use of conventional methods to detect cholera count in water used to acquire the dataset. This challenge can be addressed by using appropriate ML algorithms for low datasets and exploration of generation of synthetic datasets. As shown in Fig. 5, there is a clear separation of features for each class. This shows that the acquired model should be able to perform very well upon completion of training.

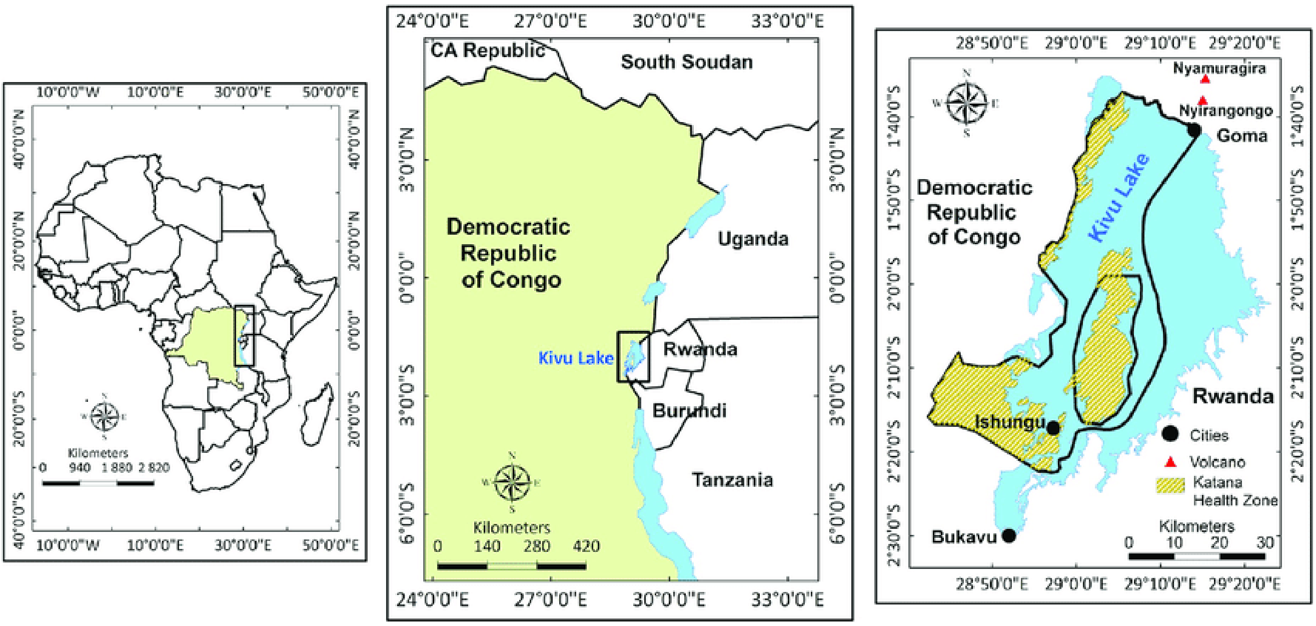

FIGURE 4: MAP OF THE KATANA HEALTH ZONE

3.5.2 SYNTHETIC DATA GENERATION.

The volume of data is important when you are performing ML. The more data you have, the higher the chances that you will be able to acquire a model that has high accuracy when tested on real world data. Synthetic data is chosen to address the question of volume, cost and time efficiency. For us to be able to perform machine learning, we require data that has been acquired through the traditional cholera testing techniques which have been proven to be time consuming and costly. As an alternative to conventional cholera detection data collection techniques, we leverage technologies at our disposal and perform synthetic data generation from already existing datasets.

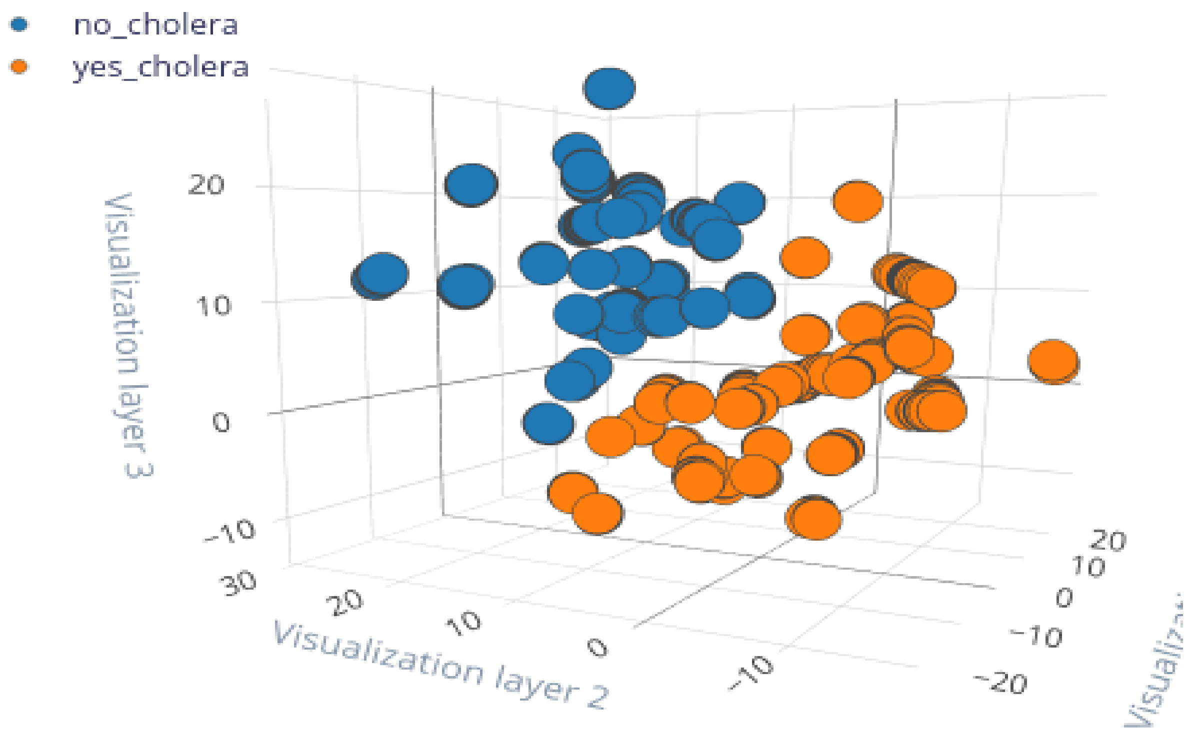

FIGURE 5: VISUALIZATION OF DATA FROM DSP BLOCK

3.6 DIGITAL SIGNAL PROCESSING

We observed the collected data on a digital signal processing block to enable us to have a clear view of the features that we want to feed into the neural network. The flattened modules that looked at averages, standard deviation, minimum, maximum, skewness and kurtosis are suitable to the kind of data that is being used. Upon review, feeding raw data of the available dataset without further processing showed a clear separation of features for both classes as shown in figure 5.

3.7 ML TRAINING

The next step is to perform training with the extracted features from the dataset with an embedded-aware ML framework. Edge impulse provides a native design of a classification neural network shown in figure 6. Raw features are extracted from the digital signal processing block of edge impulse before they are fed as input for the ML classifier algorithm. Due to the volume of the acquired dataset, it is recommended to have a simple neural network that can be used for low datasets. The default layered structure of edge impulse fits the requirements. The neural network consists of an input layer, two dense fully connected hidden layers, one having 20 neurons and another 10 neurons and an output layer for optimal performance of the derived model as shown in Fig. 4. A predictive model is generated with varying accuracies that are hinged on the number of training cycles applied for the training process. To achieve accuracies of up to 80%, a number of up to 500 training cycles is used. When inference is done on test data, the accuracies achieved are 60% which shows a discrepancy when compared to the validation accuracies. This discrepancy therefore leads to the conclusion that the model is overfitting. We are therefore able to show that the experimental use of Deep Neural Networks using the tool stack is not ideal for low datasets. To tackle this, we explore the use of techniques suitable for low datasets.

In the implementation of ML algorithms suitable for low datasets, Support Vector Machine (Quasi-SVM in Keras) is explored for classification methods. SVM has several unique benefits in solving small samples, and nonlinear and high-dimensional pattern recognition which can be extended to function in the simulation of other machine learning problems. It uses the hyperplane to separate the points of the input vectors and finds the needed coefficients. After training the SVM model using Edge impulse, the estimated real-time resources on an ARM-cortex microcontroller are: 1.5K of RAM memory for processed data, a latency of 1ms and 14.9K ROM memory to save the tinyML model. The model evaluation in cloud settings achieves an accuracy of 94% for the targeted model optimizer EON compiler using the SVM ML algorithm.

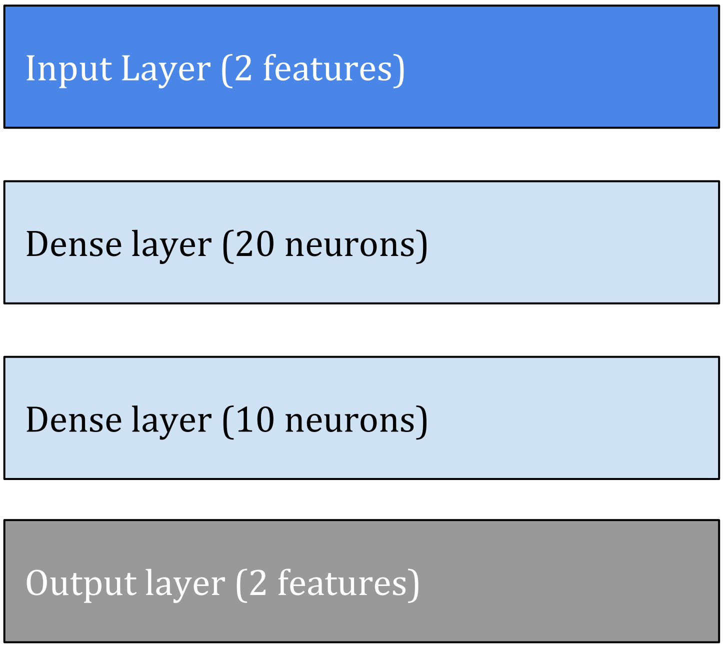

FIGURE 6: NATIVE NEURAL NETWORK ARCHITECTURE

3.8 GENERATION OF A COMPILED TINYML SW LIBRARY

The resulting AI model packaged as a software library is then compiled to the targeted processor architecture using an integrated development environment (IDE).

3.9 HW/SW EMBEDDED SIMULATION

In our first step, we simulate the edge AI inference in a virtual embedded platform simulator before testing on a real development board. In the simulation context, the performance of edge AI inferencing is validated by reading test data from a file and comparing the inference results with the ideal results calculated by the cloud platform used during the training.

We use Proteus as our embedded modelling and simulation tool. Figure 7 shows the layout of our design in Proteus.

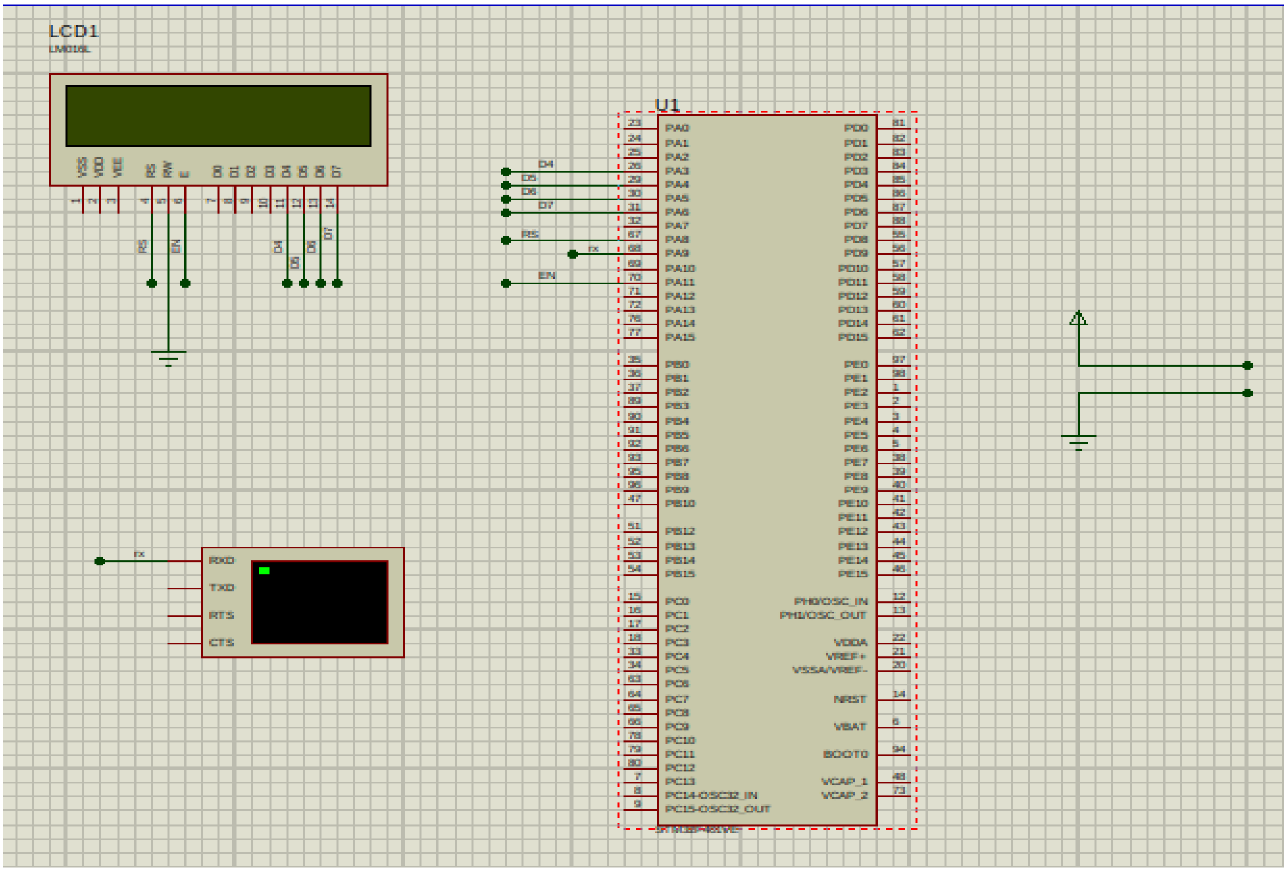

FIGURE 7: PROTEUS LAYOUT

3.5 HARDWARE REQUIREMENTS AND SPECIFICATIONS

3.5.1 SYSTEM LEVEL DESIGN

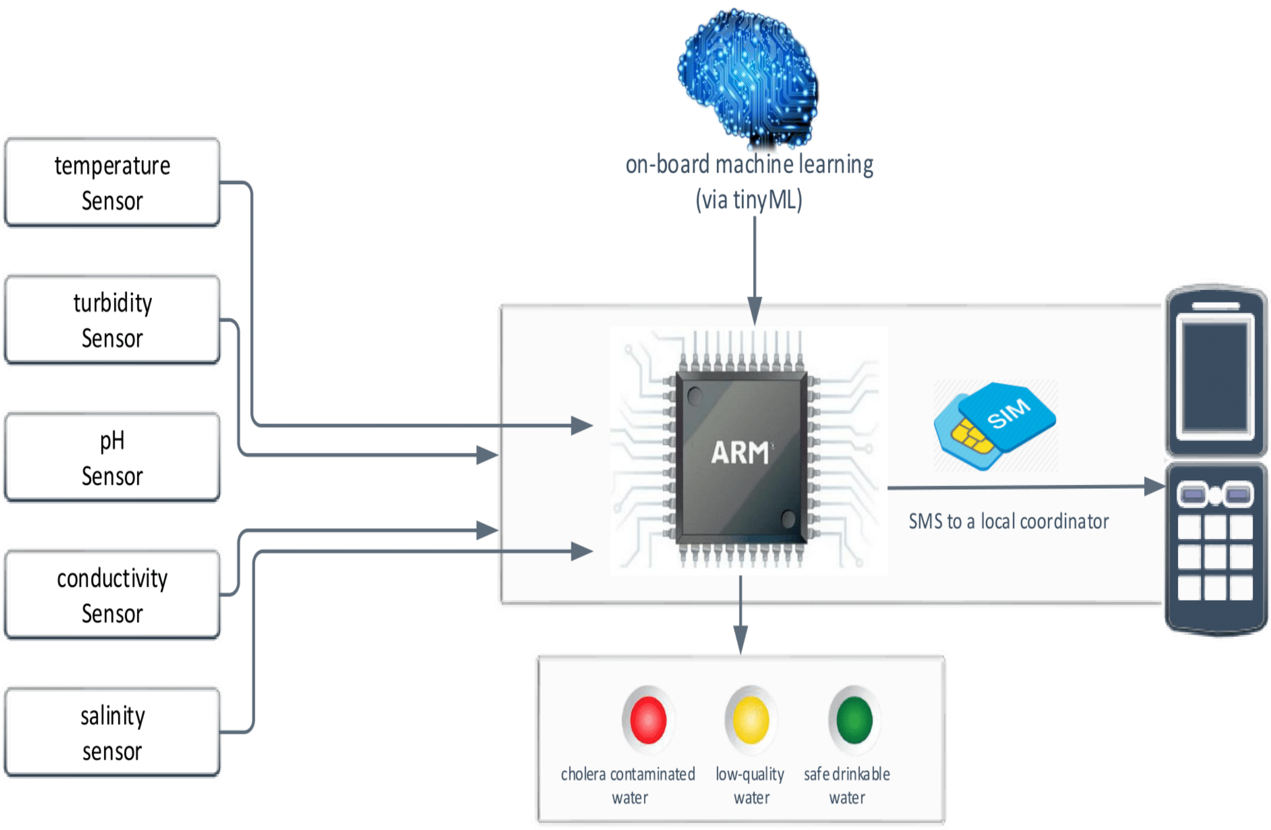

FIGURE 8: HW SYSTEM-LEVEL DESIGN MODELLING

Figure 8 presents the HW system-level design modelling of the proposed edge AI kit which is made up of

- Sensors: 2 vital water physicochemical sensors, temperature and conductivity, that are essential and impacts life for both flora and/or fauna within the aquatic systems as per [24]
- A microprocessor: Based on ARM Cortex M4 instruction set architecture (ISA)
- Actuators: A set of two LEDs, with green indicating cholera-free water and red warning the cholera infected water.
- We consider outdoor communal taps to be powered either by harvesting ambient solar energy or by non-rechargeable small-sized batteries for several years of operations without maintenance.
- Communication module: A wireless GPRS/GSM interface enables water notifications via SMS.

3.5.2 Sensors Concept

Sensors are relevant as they take a form of physical stimuli and convert them into a digital signal that can be processed.

The following are sensor specifications for the prediction of waterborne disease using Edge AI:

TEMPERATURE SENSOR

FIGURE 9: GRAVITY WATERPROOF DS18B20 SENSOR KIT.

The selected component for this scope is the Gravity Waterproof DS18B20 Sensor Kit shown in figure 9. The sensing module is capable of taking readings of temperatures ranging from -55~125^{O}C. Given the acquired dataset temperature values, the sensing module fits the range of temperatures and given the application, a waterproof sensor kit is best suited.

SALINITY SENSOR

Salinity can be described as TDS (Total Dissolved Solids). A TDS sensor as shown in figure 10 indicates how many milligrams of soluble solids dissolved in one liter of water using a probe that measures the conductance of the ions in the water between the probe tips. In general, the higher the TDS value, the more soluble solids dissolved in water, and the less clean the water is.

The TDS Measurement Range is: 0~1000ppm with a measurement Accuracy: ± 10%.

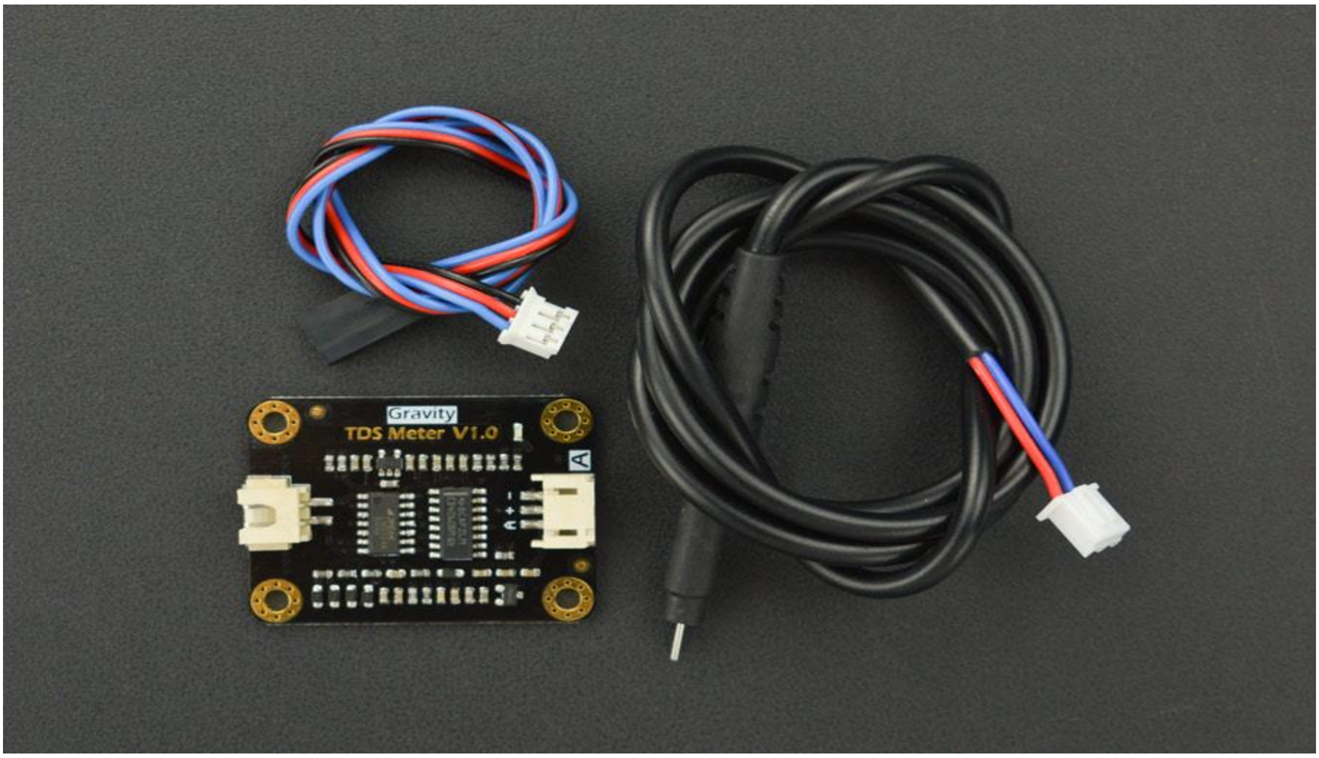

FIGURE 10: GRAVITY: ANALOG TDS SENSOR/METER FOR ARDUINO

3.6 HARDWARE SET-UP

The Hardware experiment is composed of a breadboard, an Arduino Nano BLE Sense 33, and an LCD connected through jumper wires. The aim of this set up is to show that we can on-board a tinyML on a constrained microcontroller and inference the same results as acquired from the cloud platform.

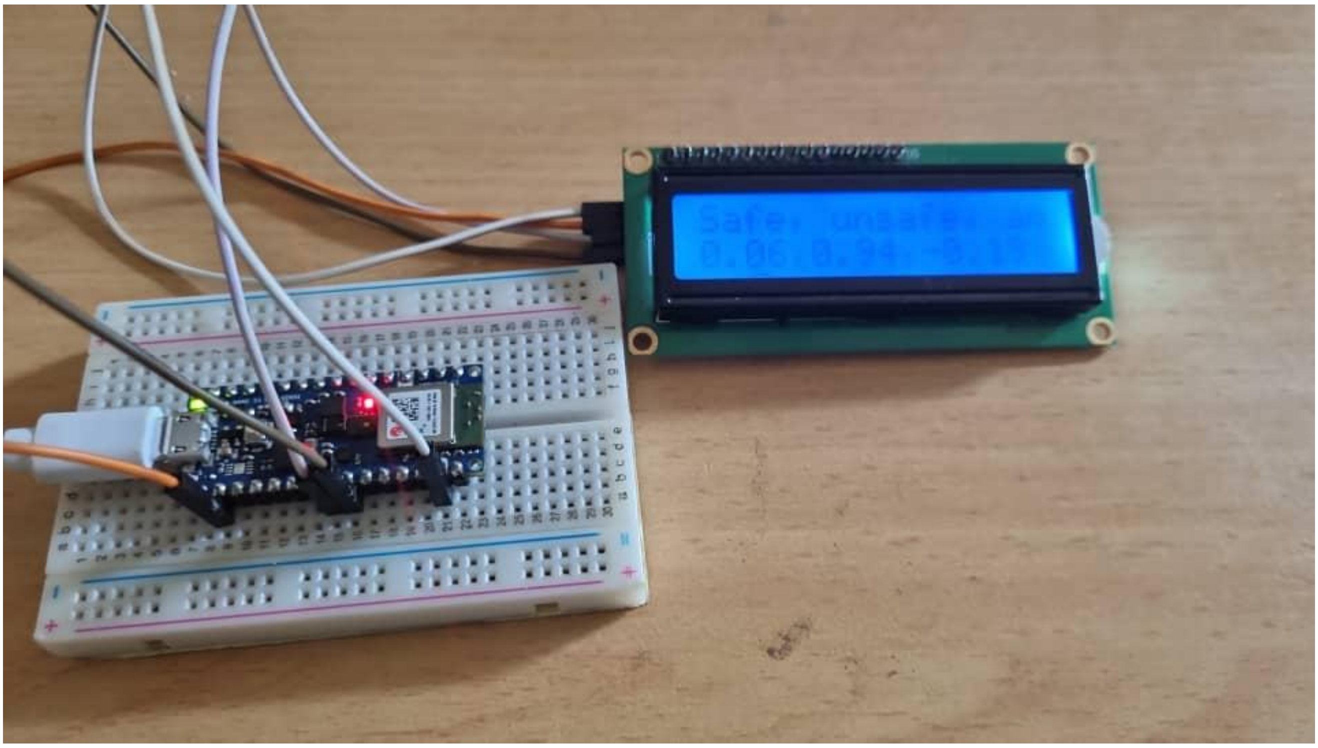

FIGURE 18: HARDWARE SET-UP

3.7 COMMUNICATION TECHNOLOGY REQUIREMENTS

FIGURE 11: SIM 800 GSM MODULE

Considering the background on this study, the chosen means of transmission will be SMS messages to health officials. A suitable module selected to achieve this is a Mini GSM / GPRS breakout board that is based on SIM800L module, which supports quad-band GSM/GPRS network, available for GPRS and SMS message data remote transmission shown in figure 11. The board's fitting features compact size and low current consumption with power saving technique, the current consumption is as low as 1mA in sleep mode. It communicates with the microcontroller via UART port. The working Voltage: 3.5~4.2V. Ultimately considering the connectivity issues in African rural areas, it is necessary to have a module that can connect onto any global GSM network with any 2G SIM

MICROCONTROLLER REQUIREMENTS AND SPECIFICATIONS

Selected for this study is the Arduino Nano 33 BLE Sense which features a powerful processor, the nRF52840 from Nordic Semiconductors, a 32-bit ARM® Cortex™-M4 CPU running at 64 MHz This allows support for its main feature which is running Edge Computing applications (AI) on it using TinyML. It allows one to create machine learning models using TensorFlow™ Lite and upload them to the board using the Arduino IDE. The main processor includes other features like ultra-low power consumption modes shown in figure 12.

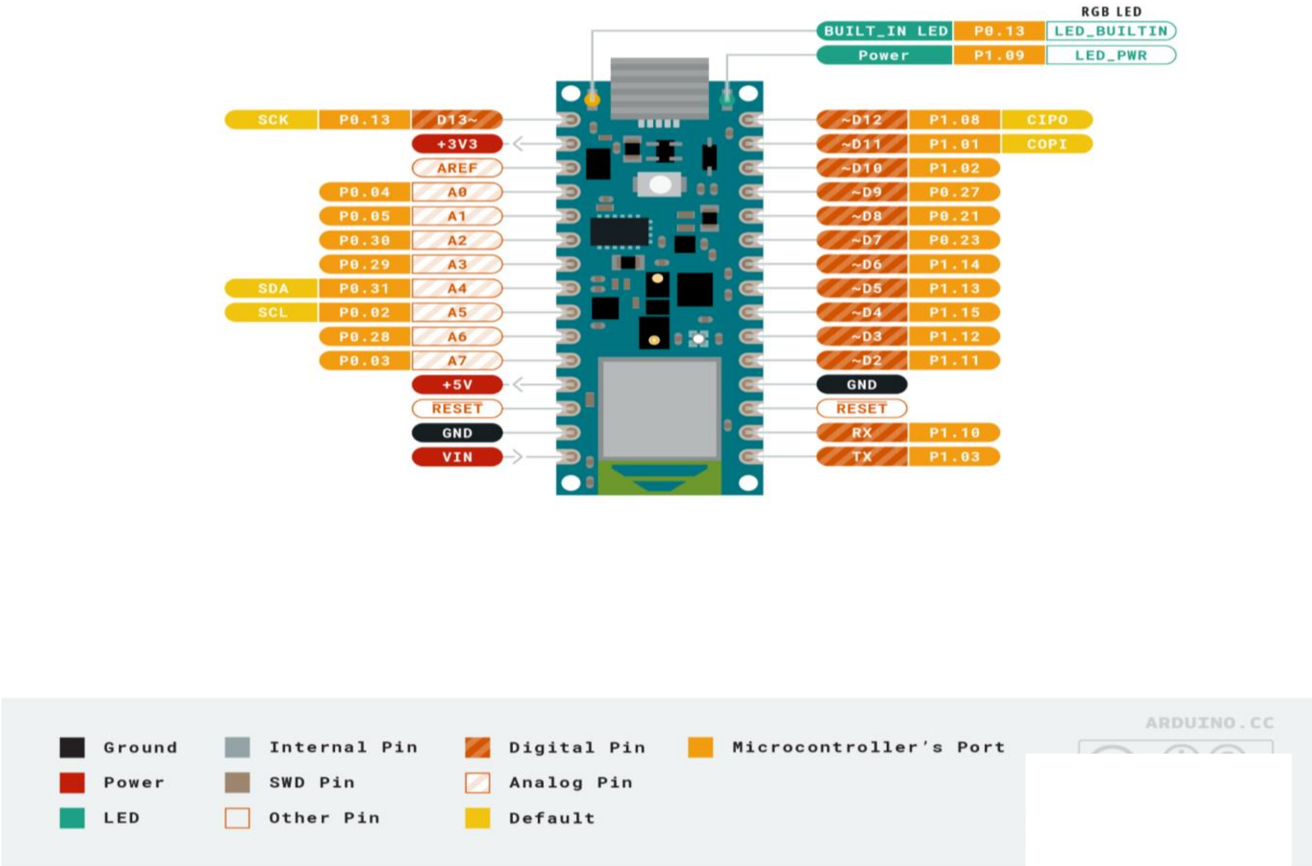

FIGURE 12: ARDUINO NANO 33 BLE SENSE WITH HEADERS

3.8 CONCLUSION

This chapter of methodology highlights how the research will be handled. The research is divided into two major phases that consist of simulation and hardware development. Relevant Edge AI tool stack is pointed out with description and relevance to each segment of the research procedure. Hardware components are also showcased with descriptions and an overall circuitry of the final system is portrayed.

CHAPTER IV: INFERENCE ACCURACY AND VALIDATION OF EDGE AI FOR PREDICTING CHOLERA

4.1 INTRODUCTION

The previous section presented our workflow and setup for co-designing the HW and SW parts of our edge AI water tap device. This section presents the validation results of edge AI inference accuracy using a real-world dataset of physico-chemical characteristics of Lake Kivu. The small size of this dataset has driven the exploration of different ML configurations presented below in the context of strict real-time embedded resources.

4.2 TRAINING RESULTS

Edge impulse metrics of how the model will perform on the actual hardware are as follows: 1.6K RAM usage, a latency of 1ms, and 15.1K ROM usage for an accuracy of 94.03% for the targeted model optimizer EON compiler using the SVM ML algorithm. Table 3 shows comparisons.

TABLE 1: SYNTHETIC DATA VS PRE SYNTHESIZED DATA

Synthetic Data		
ML Algorithm	**Loss**	**Testing accuracy**
Neural Network	0.09	90%
SVM	0.02	94%
Pre Synthesized data		
ML Algorithm	**Loss**	**Testing accuracy**
SVM	0.49	80%
Neural Network	0.52	85

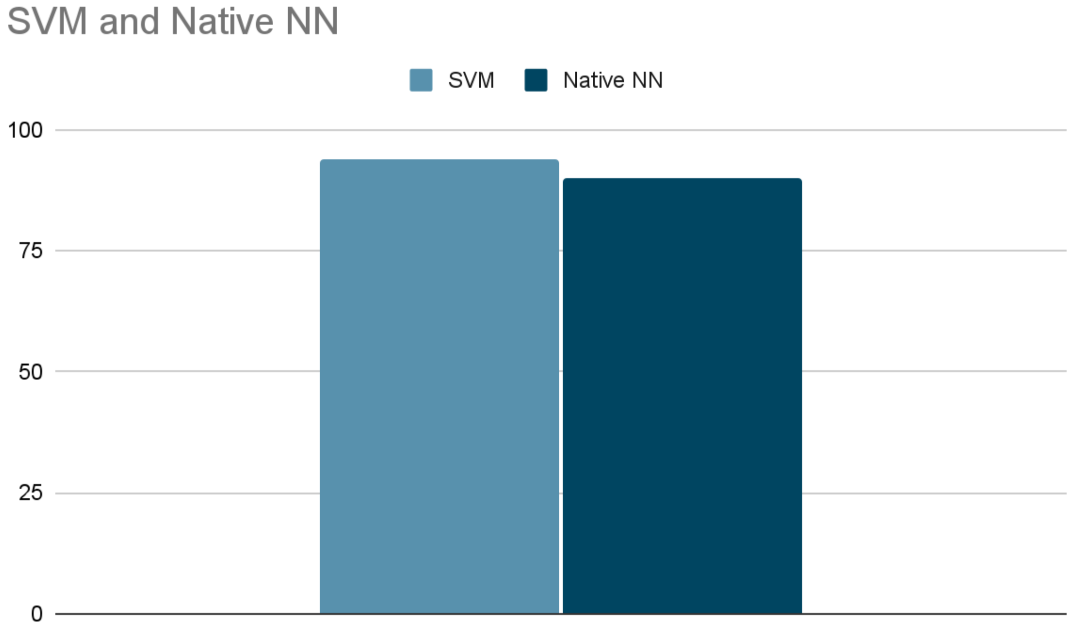

FIGURE 13: GRAPHICAL COMPARISON OF SVM AND NATIVE NN (SYNTHETIC DATASET)

4.3 SIMULATED RESULTS

Simulation results differ slightly in performance with 2ms inference time, using 57% of the CPU, 3.05Kb RAM and 54.96Kb flash on Proteus.

When simulation results are compared to the ones achieved during live classification we see some similarity. Model optimizations involve making sure the model takes up least memory and is efficient. During deployment model optimization aims to have optimal on-device performance but may reduce accuracy. In this case embedded device accuracies reflect the same accuracies achieved from Edge impulse platform as shown in figure 14 and 15. A real hardware device would perform and give us an insight on the feasibility of deploying the actual hardware device.

Upon conclusion of training, accuracies of up to 94.03% were achieved using the Quasi-SVM in Keras while with the native Neural network had accuracies of up to 90% with a higher loss value as shown in figure 13.

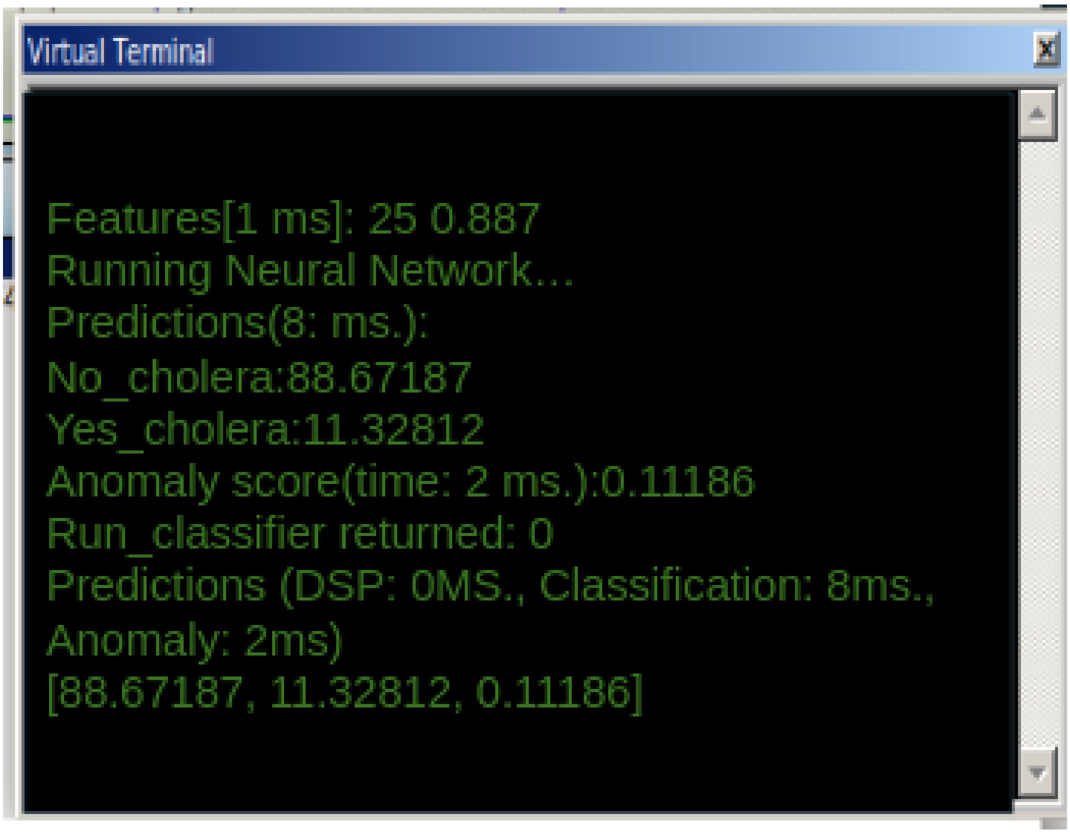

FIGURE 14: PROTEUS OUTPUT

NO_CHOLERA	YES_CHOLERA	ANOMALY0
0.89	0.11	-0.15

FIGURE 15: LIVE CLOUD CLASSIFICATION

4.4 IMPACT OF SYNTHETIC DATA

Learning rate schedules can help to converge the optimization process.

A learning rate that is too large can cause the model to converge too quickly to a suboptimal solution, whereas a learning rate that is too small can cause the process to get stuck. The graph shows that with a larger dataset we are able to increase the learning rate. This is not the same case as when we have a smaller dataset. This information is useful for tweaking the Machine learning parameters in order to achieve optimum accuracies. We are able to pinpoint learning rates for various amounts of data fed into the Machine learning algorithm as shown in figure 16.

The larger the dataset the less the risk of a suboptimal Model.

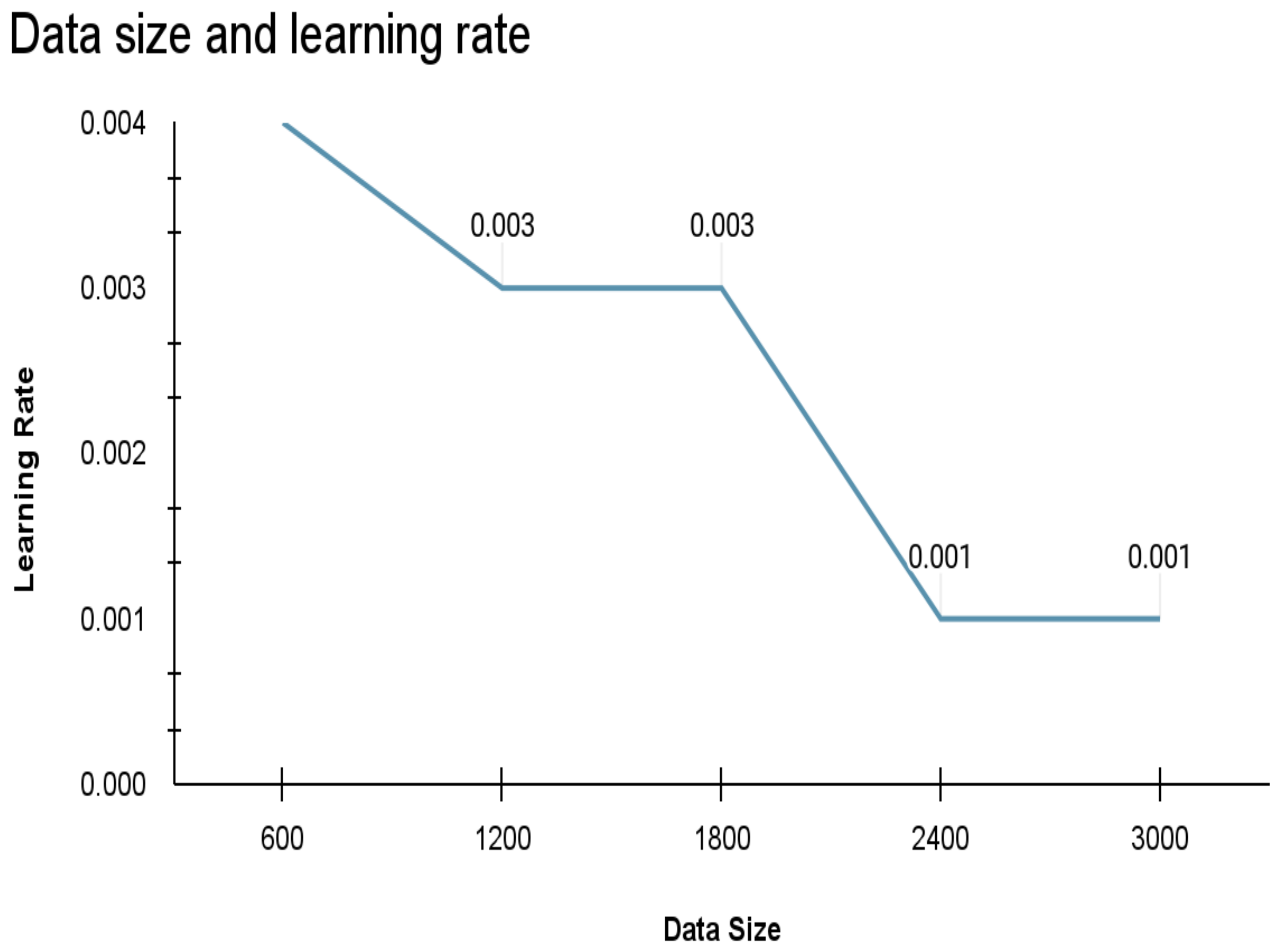

FIGURE 16: IMPACT OF SYNTHETIC DATA

4.7 EVALUATION ON DEVELOPMENT BOARD

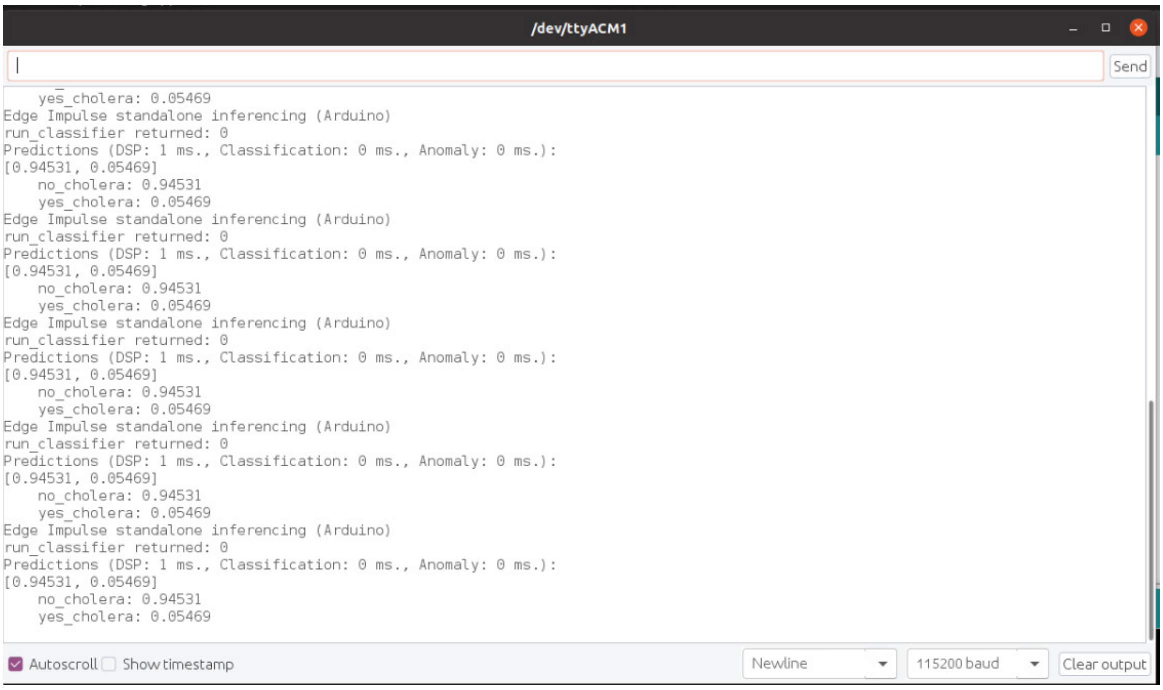

FIGURE 17: ARDUINO NANO 33 BLE SENSE OUTPUT

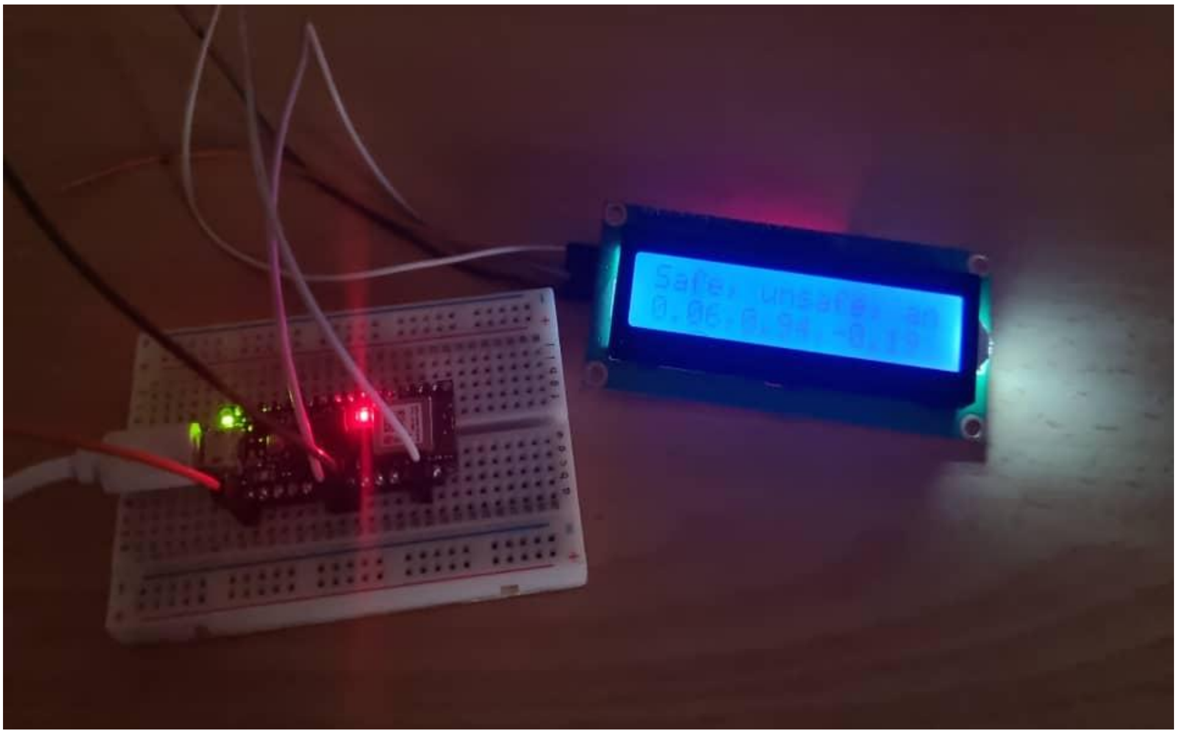

FIGURE 18: INFERENCE FROM EXPERIMENTAL SET-UP

FIGURE 19: LCD PRINT OUTPUT

Figure 17 shows performance on the Arduino Nano BLE sense's MCU nRF52840-QIAA which shows a 94% accuracy on the class no cholera for the selected features which match for temperature and salinity respectively. Cloud inference for the same features match the results output from our development board.

Further assessment was done with selected features from live classification and the results of the output compared with the LCD output of figure 18 and 19.

CHAPTER V: CONCLUSION AND RECOMMENDATION

5.1 CONCLUSION

Predicting waterborne diseases such as cholera on communal water taps will contribute in decreasing the high burden rate especially in rural areas and endemic zones. This Master Thesis work leverages on the one hand IoT technology to sense physicochemical parameters of tap water and on the other hand ML to learn waterborne disease patterns from those parameters. From this state of the art basis, we exploit the emerging edge AI technology to generate tiny ML models capable of running on resources-constrained embedded devices, thus in order to infer the risk of water to be contaminated directly on the water tap, removing the dependency to the cloud during real-time tap usage. To validate the inference accuracy in an embedded context, we have set up a prototype tool stack integrating Edge Impulse, STM32Cube and Proteus. The results confirm that inference accuracy on the used virtual embedded platform is similar to the one obtained when validating in the cloud. Furthermore, given the small size of the dataset in our disposal, we experimented tiny ML training on the one hand using a shallow ML learning suitable for small dataset namely SVM and on the other hand synthetic data generation to artificially increase the original dataset size. As a result, the accuracy has been 94% by SVM and 90% by Neural Network.

5.2 RECOMMENDATIONS AND FUTURE WORKS

As of today, acquisition of rich open datasets is a big challenge in Africa. More data based on physicochemical parameters of water could be collected using an embedded device such as the one designed in this Master Thesis and measured against cholera cell count. This project could be extended by incorporating ML algorithms for more physicochemical parameters of water. Moreover, the extended system could be more informative especially with the ability to capture turbidity of water. Furthermore, the device could be implemented to more sub-Saharan

African rural regions.

REFERENCES

[1] M. Djaouda, Z. Wadoubé, S. Liang, and M. Nola, "Scarcity of potable water and sanitation facilities in the endemic cholera region of north Cameroon," *Int. J. Infect. Dis.*, vol. 101, no. 1, p. 310, 2020, doi: 10.1016/j.ijid.2020.09.809.

[2] B. Kwesiga, G. Pande, A. R. Ario, N. M. Tumwesigye, J. K. B. Matovu, and B. P. Zhu, "A prolonged, community-wide cholera outbreak associated with drinking water contaminated by sewage in Kasese District, western Uganda," *BMC Public Health*, vol. 18, no. 1, pp. 1–8, 2017, doi: 10.1186/s12889-017-4589-9.

[3] C. K. Uthappa *et al.*, "An outbreak of cholera in Medipally village, Andhra Pradesh, India, 2013," *J. Heal. Popul. Nutr.*, vol. 33, no. 1, pp. 1–5, 2015, doi: 10.1186/s41043-015-0021-1.

[4] Center for disease Control and Prevention, "Diagnosis and Detection | Cholera | CDC," *CDC*, 2018. https://www.cdc.gov/cholera/diagnosis.html (accessed Feb. 07, 2021).

[5] "WHO | Number of reported cholera cases," *WHO*, 2017, Accessed: Mar. 08, 2021. [Online]. Available: http://www.who.int/gho/epidemic_diseases/cholera/cases_text/en/.

[6] "Cholera | WHO | Regional Office for Africa." https://www.afro.who.int/health-topics/cholera (accessed Sep. 17, 2021).

[7] J. Ateudjieu *et al.*, "Health facility preparedness for cholera outbreak response in four cholera-prone districts in Cameroon: A cross sectional study," *BMC Health Serv. Res.*, vol. 19, no. 1, Jul. 2019, doi: 10.1186/S12913-019-4315-7/TABLES/2.

[8] International Telecoms Union, "Internet Connectivity in Sub-Saharan Africa," p. 58, 2013.

[9] "Thermo Scientific™ Alkaline Peptone Water (Bottled) Products | Fisher Scientific." https://www.fishersci.co.uk/shop/products/alkaline-peptone-water-bottled/10699944 (accessed Apr. 12, 2021).

[10] T. Ramamurthy, B. Das, S. Chakraborty, A. K. Mukhopadhyay, and D. A. Sack, "Diagnostic techniques for rapid detection of Vibrio cholerae O1/O139," *Vaccine*, vol. 38, no. xxxx, pp. A73–A82, 2020, doi: 10.1016/j.vaccine.2019.07.099.

[11] G. Bwire *et al.*, “Alkaline peptone water enrichment with a dipstick test to quickly detect and monitor cholera outbreaks,” *BMC Infect. Dis.*, vol. 17, no. 1, pp. 1–8, 2017, doi: 10.1186/s12879-017-2824-8.

[12] A. S. Azman, S. M. Moore, and J. Lessler, “Surveillance and the global fight against cholera: Setting priorities and tracking progress,” *Vaccine*, vol. 38, no. xxxx, pp. A28–A30, 2020, doi: 10.1016/j.vaccine.2019.06.037.

[13] N. Sinyange *et al.*, “Cholera Epidemic — Lusaka, Zambia, October 2017–May 2018,” 2017. Accessed: Mar. 08, 2021. [Online]. Available: http://www.who.int/cholera/task_force/.

[14] A. N. Prasad, K. A. Mamun, F. R. Islam, and H. Haqva, “Smart water quality monitoring system,” *2015 2nd Asia-Pacific World Congr. Comput. Sci. Eng. APWC CSE 2015*, 2016, doi: 10.1109/APWCCSE.2015.7476234.

[15] S. Abhinav, S. Srinivasan, A. Ganesan, M. R. Anala, and T. Mamatha, “Wireless water quality monitoring and quality deterioration prediction system,” *Proc. - 26th IEEE Int. Conf. High Perform. Comput. Work. HiPCW 2019*, pp. 23–28, 2019, doi: 10.1109/HiPCW.2019.00013.

[16] M. V. Ramesh *et al.*, “Water quality monitoring and waste management using IoT,” *GHTC 2017 - IEEE Glob. Humanit. Technol. Conf. Proc.*, vol. 2017-Janua, pp. 1–7, 2017, doi: 10.1109/GHTC.2017.8239311.

[17] S. Chakraborty, M. Alam, H. M. Scobie, D. A. Sack, D. Ceccarelli, and S. R. Campagna, “Adaptation of a simple dipstick test for detection of Vibrio cholerae O1 and O139 in environmental water,” *Front. Microbiol.*, vol. 4, no. October 2013, pp. 2–5, 2013, doi: 10.3389/fmicb.2013.00320.

[18] J. F. Mehrabadi, P. Morsali, H. Rohani nejad, A. A. Imani Fooladi, and S. Najafy, “Evaluation of new primers for detecting toxigenic vibrio cholerae by multiplex PCR,” *Microbiol. Res. (Pavia).*, vol. 2, no. 1, p. 1, 2011, doi: 10.4081/mr.2011.e1.

[19] J. F. Mehrabadi, P. Morsali, H. R. Nejad, and A. A. Imani Fooladi, “Detection of toxigenic Vibrio cholerae with new multiplex PCR,” *J. Infect. Public Health*, vol. 5, no. 3, pp. 263–267, 2012, doi: 10.1016/j.jiph.2012.02.004.

[20] “Invitrogen™ LIVE/DEAD™ *Bac*Light™ Bacterial Viability Kit, for microscopy | Fisher Scientific.” https://www.fishersci.ca/shop/products/molecular-probes-live-dead-baclight-

bacterial-viability-kit-microscopy/l7007 (accessed Apr. 12, 2021).

[21] K. Kim and H. Myung, "Sensor node for remote monitoring of waterborne disease-causing bacteria," *Sensors (Switzerland)*, vol. 15, no. 5, pp. 10569–10579, 2015, doi: 10.3390/s150510569.

[22] K. N. Clayton, T. J. Moehling, D. H. Lee, S. T. Wereley, J. C. Linnes, and T. L. Kinzer-Ursem, "Particle Diffusometry: An Optical Detection Method for Vibrio cholerae Presence in Environmental Water Samples," *Sci. Rep.*, vol. 9, no. 1, pp. 1–12, 2019, doi: 10.1038/s41598-018-38056-7.

[23] T. J. Moehling *et al.*, "A smartphone-based particle diffusometry platform for sub-attomolar detection of Vibrio cholerae in environmental water," *Biosens. Bioelectron.*, vol. 167, no. July, p. 112497, 2020, doi: 10.1016/j.bios.2020.112497.

[24] A. H. Velders, C. Schoen, and V. Saggiomo, "Loop-mediated isothermal amplification (LAMP) shield for Arduino DNA detection," *BMC Res. Notes*, vol. 11, no. 1, p. 93, Feb. 2018, doi: 10.1186/s13104-018-3197-9.

[25] S. Sharma, M. Alamgir Kabir, and W. Asghar, "Lab-on-a-chip zika detection with reverse transcription loop-mediated isothermal amplification-based assay for point-of-care settings," *Arch. Pathol. Lab. Med.*, vol. 144, no. 11, pp. 1335–1343, Nov. 2020, doi: 10.5858/arpa.2019-0667-OA.

[26] K. Ashwini, J. J. Vedha, and M. D. Priya, "Intelligent model for predicting water quality," *Int. J. Adv. Res. Ideas Innov. Technol. ISSN*, vol. 5, no. 2, pp. 70–75, 2019.

[27] N. Kumar Koditala and P. Shekar Pandey, "Water Quality Monitoring System Using IoT and Machine Learning," *Proc. 2018 3rd IEEE Int. Conf. Res. Intell. Comput. Eng. RICE 2018*, 2018, doi: 10.1109/RICE.2018.8509050.

[28] J. Leo, E. Luhanga, and K. Michael, "Machine Learning Model for Imbalanced Cholera Dataset in Tanzania," 2019, doi: 10.1155/2019/9397578.

[29] A. Bria, G. Cerro, M. Ferdinandi, C. Marrocco, and M. Molinara, "An IoT-ready solution for automated recognition of water contaminants," *Pattern Recognit. Lett.*, vol. 135, pp. 188–195, 2020, doi: 10.1016/j.patrec.2020.04.019.

[30] D. B. Boloweti *et al.*, "Volcanic activity controls cholera outbreaks in the East African Rift," 2020, doi: 10.1371/journal.pntd.0008406.

APPENDICES

APPENDIX 1: GC 2021 WORKSHOP ON SUSTAINABLE ENVIRONMENTAL SENSING SYSTEMS(SESSY) ACCEPTANCE

[GC 2021 Workshop - SESSy] Your paper #1570742767 ("Offline Prediction of Cholera in Rural Communal Tap Waters Using Edge AI inference") Inbox x

GC 2021 Workshop - SESSy <gc2021workshop-sessy-chairs@edas.info> Thu, Sep 16, 8:29 AM (11 days ago)
to me, Jimmy, Kizito, Ruidong, Pietro, Nathalie, Catia, Marco

Dear Mr. Marvin Ogore:

On behalf of the Workshop Program Committee, we are pleased to inform you that your paper #1570742767 entitled "Offline Prediction of Cholera in Rural Communal Tap Waters Using Edge AI inference" has been accepted for presentation at the IEEE GLOBECOM 2021, December 7 - 11, in Madrid, Spain, and virtual, for publication in its proceedings. Please revise your paper carefully to address the reviewers' comments and suggestions, and to ensure that your final paper fits the camera-ready format. The reviews are available at https://edas.info/showPaper.php?m=1570742767.

FORMAT AND SESSION:
Please note that by submitting manuscripts, we assume that the authors have agreed to present their accepted papers at sessions organized by the Workshop Chairs.

APPENDIX 2: BEST ABSTRACT TINYML EMEA 2021

bette@tinyml.org Thu, May 13, 12:44 PM
to me

Hello Marvin,

Thanks for getting back to me. I did use this email address but it bounced, hopefully this gets to you.

I wanted to let you know that you have been accepted to speak at EMEA. Attached is speaker information, including deadlines and guidelines for materials.

Your abstract was the highest ranked out of dozens that we received. Congratulations!

Please let me know if you have any questions.

Regards,

Bette Cooper
tinyML Event Organizer
650-714-1570
http://www.tinyml.org

APPENDIX3: CODE ON ARDUINO BLE 33 SENSE

```


#include <water_quality_synthetic_data__inferencing.h>
#include "GravityTDS.h"
#include <MaximWire.h>
#include <Arduino.h>
#include <LiquidCrystal_I2C.h>

#define TdsSensorPin A7

GravityTDS gravityTds;

float temperature = 25,tdsValue = 0;

LiquidCrystal_I2C lcd(0x3F,20,4);  // set the LCD address to 0x27 for a 16 chars and 2 line display
/*
#define PIN_BUS 9

MaximWire::Bus bus(PIN_BUS);
MaximWire::DS18B20 sensors;
*/

#define THRESHOLD 0.7
#define RED 22
#define BLUE 24
#define GREEN 23
#define TdsSensorPin A1 // Where Analog pin of TDS sensor is connected to arduino
//////////////////////////////////////////////////////////////////
#define FREQUENCY_HZ        50
```

```
//////////////////////////////////////////////////////////////////
#define FREQUENCY_HZ        50
#define INTERVAL_MS         (100000 / (FREQUENCY_HZ + 1))
//////////////////////////////////////////////////////////////////


static unsigned long last_interval_ms = 0;
// to classify 1 frame of data you need EI_CLASSIFIER_DSP_INPUT_FRAME_SIZE values
float features[EI_CLASSIFIER_DSP_INPUT_FRAME_SIZE];
// keep track of where we are in the feature array
size_t feature_ix = 0;

float temp;


void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  Serial.println("Started");

  //tds sensor code
    gravityTds.setPin(TdsSensorPin);
    gravityTds.setAref(5.0);  //reference voltage on ADC, default 5.0V on Arduino UNO
    gravityTds.setAdcRange(1024);  //1024 for 10bit ADC;4096 for 12bit ADC
    gravityTds.begin();  //initialization

  // receive from temp data from arduino uno
  Wire.begin(A1);//9 here is the address(Mentioned even in the master board code)
  Wire.onReceive(receiveEvent);
```

```
  Serial.begin(9600);

  pinMode(RED, OUTPUT);
  pinMode(BLUE, OUTPUT);
  pinMode(GREEN, OUTPUT);
  lcd.init();
    // Print a message to the LCD.
  lcd.backlight();
  lcd.setCursor(1,0);
  lcd.print("Begin inferencing");

}

void receiveEvent(int bytes) {
  temp = Wire.read();//Receive value from master board
  Serial.print(temp);
}
void loop() {
    static unsigned long last_interval_ms = 0;

    if (millis() > last_interval_ms + INTERVAL_MS) {
        last_interval_ms = millis();

    gravityTds.setTemperature(temperature);  // set the temperature and execute temperature compensation
    gravityTds.update();  //sample and calculate
    tdsValue = gravityTds.getTdsValue();  // then get the value
    Serial.print(tdsValue,0);
    Serial.println("ppm");
    delay(1000);
```

```

        // keep filling the features array until it's full
        features[feature_ix++] = temp;//26.0000;
        features[feature_ix++] = tdsValue;//1.0520;
    ei_printf("Edge Impulse standalone inferencing (Arduino)\n");


        // features buffer full? then classify!
        if (feature_ix == EI_CLASSIFIER_DSP_INPUT_FRAME_SIZE) {
            ei_impulse_result_t result;

            // create signal from features frame
            signal_t signal;
            numpy::signal_from_buffer(features, EI_CLASSIFIER_DSP_INPUT_FRAME_SIZE, &signal);

            // run classifier
            EI_IMPULSE_ERROR res = run_classifier(&signal, &result, false);
            ei_printf("run_classifier returned: %d\n", res);
            if (res != 0) return;

            // print predictions
            ei_printf("Predictions (DSP: %d ms., Classification: %d ms., Anomaly: %d ms.): \n",
                result.timing.dsp, result.timing.classification, result.timing.anomaly);
              lcd.clear();
              lcd.setCursor(0,0);
              lcd.print("Safe, unsafe, anomaly");
              lcd.setCursor(0,1);

            // print the predictions
```

```

            // print the predictions
            for (size_t ix = 0; ix < EI_CLASSIFIER_LABEL_COUNT; ix++) {
                ei_printf("%s:\t%.5f\n", result.classification[ix].label, result.classification[ix].value);
                  lcd.print(result.classification[ix].value);
                  lcd.print(",");


               if (result.classification[ix].label == "yes_cholera" && result.classification[ix].value > THRESHOLD)
        {
             digitalWrite(BLUE, HIGH);

        }
        else
        {

           digitalWrite(GREEN, HIGH);
        }
            }
        #if EI_CLASSIFIER_HAS_ANOMALY == 1
            ei_printf("anomaly:\t%.3f\n", result.anomaly);
            lcd.print(result.anomaly);

        #endif

            // reset features frame
            feature_ix = 0;
        }
    }
```

```
}
void ei_printf(const char *format, ...) {
    static char print_buf[1024] = { 0 };

    va_list args;
    va_start(args, format);
    int r = vsnprintf(print_buf, sizeof(print_buf), format, args);
    va_end(args);

    if (r > 0) {
        Serial.write(print_buf);
    }
}
```

APPENDIX 4: CODE ON ARDUINO UNO-TEMP SENSOR DATA TRANSMITTED VIA I2C

```
#include <OneWire.h>
#include<Wire.h>//This library is used for I2C communication
#include <DallasTemperature.h>

// Data wire is conntec to the Arduino digital pin 4
#define ONE_WIRE_BUS A1

// Setup a oneWire instance to communicate with any OneWire devices
OneWire oneWire(ONE_WIRE_BUS);

// Pass our oneWire reference to Dallas Temperature sensor
DallasTemperature sensors(&oneWire);

void setup(void)
{
  // Start serial communication for debugging purposes
  Serial.begin(9600);
  // Start up the library
  sensors.begin();
  Wire.begin();
}

void loop(void){
  // Call sensors.requestTemperatures() to issue a global temperature and Requests to all devices on the bus
  sensors.requestTemperatures();

  Serial.print("Celsius temperature: ");
  // Why "byIndex"? You can have more than one IC on the same bus. 0 refers to the first IC on the wire
  Serial.print(sensors.getTempCByIndex(0));
```

```

  Serial.print("Celsius temperature: ");
  // Why "byIndex"? You can have more than one IC on the same bus. 0 refers to the first IC on the wire
  Serial.print(sensors.getTempCByIndex(0));
  Serial.print(" - Fahrenheit temperature: ");
  Serial.println(sensors.getTempFByIndex(0));
  delay(1000);
  int  temp=sensors.getTempCByIndex(0);
  Serial.println(temp); //check if conversion is ok
  Wire.beginTransmission(A1);//9 here is the address of the slave board
  Wire.write(temp);//Transfers the value of potentiometer to the slave board
  Wire.endTransmission();
  delay(1000);
}
```