An Intelligent Microprocessor Integrating TinyML in Smart Hotels for Rapid Accident Prevention

Angelos Zacharia*, Dimitris Zacharia*, Aristeidis Karras*, Christos Karras*, Ioanna Giannoukou†, Konstantinos C. Giotopoulos†, Spyros Sioutas*

*Decentralized Systems Computing Group, Computer Engineering and Informatics Department, University of Patras, Greece
{anzacharia, akarras, c.karras, sioutas}@ceid.upatras.gr, zade@zade.com.cy

†Department of Management Science and Technology, University of Patras, Greece
{igian, kgiotop}@upatras.gr

2022 7th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM) | 979-8-3503-9858-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/SEEDA-CECNSM57760.2022.9932982

***Abstract*—In the modern era of Internet of Things (IoT) and Industry 4.0 there is a growing need for intelligent microcontrollers that can collect, sense and analyse data effectively and efficiently. Such devices can be installed in large scale IoT deployments ranging from smart homes to smart cities and smart buildings. The aim of these devices shall be not only data monitoring but at the same time energy saving and overall building management. In the context of this paper, an all-in-one microprocessor is presented, namely ZAC888DP, which can sense data from multivariate sources and perform data analytics on top of the collected data. Moreover, machine learning (ML) models are deployed in the embedded memory of the device and specifically TinyML methods using a tflite file. The aim of the developed ML model is to collect data from four heterogeneous sources (water sensor, light sensor, humidity and temperature) in order to identify and forecast possible lavatory accidents. The experimental results of this work are encouraging as the model managed to achieve 100 percentage accuracy after 256 iterations. Future directions include the integration of the device with a neural network that will be trained on top of the pre-trained model in order to increase the overall precision even further.**

***Index Terms*—IoT, TinyML, TensorFlow Lite, Smart Hotels, Smart Buildings, Energy Saving, Accident Prevention System**

I. Introduction

The Internet of Things (IoT) has revolutionised our lives since its initial emergence. Nowadays, people can automate, schedule or monitor smart home actions as well as whole facilities ranging from small buildings to large companies. The emerge of IoT applications has lead to the creation of tiny microcontrollers but conversely with high capabilities. These microcontrollers can aid in several sectors such as energy management, appliance control utilizing event detection mechanisms [1] and even more in the accident prevention sector in one or more areas within a building.

Early accident prevention using sampling approaches [2]–[5] or even more prediction based on deep learning methods as the ones presented in [6] [7] can significantly help in saving peoples lives from injuries. As contradictory as it might be, 234 thousand accidents occurred in the lavatory area within a building as the research [8] conducted by the Centers for Disease Control and Prevention (CDC) highlights. This number is 1% of all -non-fatal- injuries and especially in those over 65 years old 2.5%, causing some really serious and painful injuries. Hence, early accident prevention and information systems are of note as they can forecast and avoid user-dangerous situations by utilizing signals from multivariate sources that play a vital role analysing the users' routine.

This work aims to identify the variables that enhance the likelihood of a possible fall of a person and to develop techniques for early prediction of a potential accident, so as to promptly alert the user. This is accomplished via the use of machine learning (ML) methods and, more particularly, the TinyML technique, which is optimised for microprocessors. The ESP32 microprocessor, one of the most well known devices in the IoT sector is highly compatible for integration with TinyML and is also utilized in this work. A secondary objective of this work is the actual implementation of the aforementioned mechanism in a simulated environment, which is coupled to an existing automation system named as the ZAC888-DP device. This device is nowadays installed in several facilities and is responsible for the entire management of the features within a room.

Additionally, we also present the innovative and all-in-one microprocessor ZAC-888DP in detail, which is currently utilised for home and hotel automation incorporating TinyML techniques for accident prevention in areas of the house and specifically the lavatory area and to alert the user promptly. The pre-trained ML model is initially built and tested within the ESP32 environment whereas later on, is exported as a tflite file and transferred through the Arduino environment to the device. Ultimately, the ZAC888DP device makes use of the ML model and forecasts the likelihood of a possible lavatory accident based on variables obtained from sensor readings each of which has its own weight.

This rest of the paper is structured as follows. In Section II related work regarding smart buildings and energy management is covered along with accident prevention systems. In Section III the hardware and software aspects of the ZAC888-DP device are covered along with the features and functions of the microcontroller. Additionally, the model architecture is presented as well as the machine learning model implemented in TinyML. Section IV summarises the experimental results and their findings and finally, the conclusions and future directions of this work are presented in Section V.

979-8-3503-9858-8/22/$31.00 ©2022 IEEE

II. Previous Work

In the context of Internet of Things (IoT) applications, sensors and actuators play a vital role in the energy management of intelligent power systems [9] [10]. Energy management is crucial for the development of smart power systems. It is relevant to several areas of smart power systems, such as microgrids, smart homes, and demand side management. Due to the unpredictability of renewable energy supplies and loads, uncertainty presents a difficulty for energy management.

Energy management in smart buildings has placed a large emphasis on determining the optimal contract power value selection [11]. High-end smart meter devices are also utilized for energy desegregation where the separation of an aggregate energy signal into the consumption of individual appliances in a household is presented in [12]. Smart Home management and analytics utilizing big data are presented in [13] where the proposed system uses a system on a chip (SOC) device to collect data and transmit it to a centralized server for further processing. Advanced IoT based systems for energy management in buildings are presented in [14] where the system incorporates cross-domain data, such as the data of the building (e.g., energy management systems), energy output, energy pricing, weather data, and end-users' behaviour, to provide daily and weekly action plans for energy end-users with actionable, personalised information.

Edge computing infrastructures employed with deep reinforcement learning are of note, where intelligent systems as in [15] are used for energy management in smart cities ecosystems and p2p scenarios [16] where devices act as users while in [17], the authors propose the use of an Edge-IoT platform along with a Social Computing framework to develop a smart energy efficiency system for public buildings. Forecasting short-term future energy usage, an effective communication between energy distributors and customers is presented in [18]. Key contributions of the article include of real-time energy management on edge devices through a shared cloud-based data supervisory server, optimum normalisation method selection, and an unique sequence learning energy forecasting mechanism with decreased time complexity and error rates.

Energy demand prediction in hotels is a crucial aspect as it aids the administrators to adjust their proportion of usage to energy waste. With the use of advanced techniques for intelligent hybrid modelling as indicated in [19], it is feasible to predict the energy load in a hotel by integrating neural networks and support vector machines.

Accident prevention is also an important sector in IoT infrastructures among transportation systems [20] smart vehicles [21] [22] and modern city transportation infrastructures [23]. Similarly to accident prevention, real-time alert systems are introduced in [24]. In a broader scope of smart cities, crash avoidance systems are shown in [25].

Although each of the preceding works include a vast amount of contribution to the scientific field, they do not incorporate all the proposed models in an all-in-one scenario as the microcontroller presented in the context of this paper.

III. Methodology

In this section the custom microprocessor is presented as well as the methodology for the integration of Machine Learning techniques using TinyML.

A. ZAC888-DP Microprocessor

The ZAC888-DP is a smart microcontroller designed to provide the user with the most pleasant, friendly, and accessible experience possible in a given environment. It is used in either homes or hotels, often one per room, since it can handle up to eight lighting circuits, the opening and shutting of curtains, and numerous inputs such as motion sensors in order to automate several operations. Moreover, it interfaces with and controls HVAC systems, DALI lighting, the DMX512 protocol, and a number of other protocols. Its extensive connectivity is one of its greatest benefits, and all of its features and capabilities are outlined here.

1) Features: The ZAC888DP is an All-In-One device because it combines several roles as an intermediate or actuator. Adjustments are made to eight channels of 5 Ampere adjustable light intensity to enhance the lifespan of the luminaires. With the same channels, we may open and shut curtains, garage doors, water heaters, and adjust water pressure. It features eight inputs for receiving basic connections, as well as eight outputs for unlocking doors and locks with low-current requirements. In addition, it is possible to control up to 64 secondary devices in tandem for DALI luminaires. Using the DMX protocol, we are able to control any RGB lighting from our device, and a DMX console can control the ZAC888DP. It connects to TCP/IP on the network and communicates via the RS485 protocol through CAT6 cable. In addition, there is a USB connection through which updates may be installed. The energy metre, which computes usage and establishes maximum cost limitations, is one of its most essential components. This allows the gadget to define its own safety boundaries, preventing overload or short circuit. In addition to a built-in temperature sensor for its safety, it cuts intensity at very high measures to prevent overheating. Its aluminium structure is suitable for a panel rail. It has a colour TFT technology screen that is controlled by 15 buttons for menu and channel navigation. In addition, there are ten LEDs that alert the user about the state of the channels and whether or not communication and power are functioning properly. The wires are terminated in detachable phoenix termination blocks, making it extremely simple and quick to replace faulty devices. Lastly, the ZAC888DP is an eco-friendly gadget due to its 0.35 Watt power consumption.

2) Functions: The startup screen of ZAC888DP is shown in Figure 1. The details provided in the initial screen are as follows: ID is the identification of the microcontroller unit. IP is the IP address of the network. T Indicates the temperature within the Bus device. R indicates if the device is operational. P Indicates the energy used which is calculated hourly. OUT1-8: Indicates if voltage is present at outputs 1-8. INPUT1-8: Indicates if voltage is present at inputs 1-8. Loc01: Amb00C

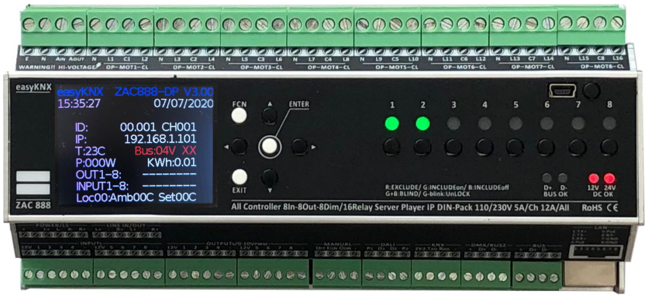

Fig. 1. ZAC888DP Module

Set00C informs the user with the current room temperature and (Set) setting.

Additionally, on the first page:

- Network: We set the ID, if our IP is static or dynamic, the address as well as the other basic elements of a network (mask, gateway, protocol, port).
- Channels/OUTs: Select the minimum and maximum limits of each channel, and whether the output is dimmable (0-100) or switch (0 or 100).
- SCENEs: We define the components that each scenario will contain, in how much time it will have reached its maximum or minimum, as well as whether we want it to turn off automatically in a specific time frame. We can also create script groups using Sequence. In the same menu we program the DALI fixtures.
- TIMERs: We set the date time and hours of sunset and sunrise (it can be done automatically from the network). The main purpose of this menu is to set scheduled fixed tasks, for example in winter to turn on the water heater for two hours in the morning before the children leave for school, and in summer to turn on 30 minutes since due to temperature the water is already a good temperature.
- INPUTs: In the Inputs menu the user can choose whether the contact will work with logic normally open, normally close, toggle etc. and what it will activate after firing, this could be done once or repeatedly as well as throughout the day or at specific times.
- HVACs: The control of all possible heating, ventilation and cooling systems is done here with possible choices from underfloor heating to air conditioners using IR among others. We also define which will be the temperature sensors and the preset values that the user can choose.
- DMX: Activate and specify as input or output of the protocol in relation to our device. COLOR MACROS: Adjusts the speeds when the automated scenarios are running with RGB lights.
- PROTECTIONs: This menu is quite important, in which we define the overload and overheating but also the limits at which the ZAC888DP stops being fully functional in order to achieve the drop of loads and local temperature.

3) Room Management System: ZAC888DP module is suitable for efficient room management as it focus on energy saving among automations. For example, when doors or windows are opened the Air-conditioning device are turned off in order to maximize energy saving. Additionally, the device can utilize maximum and minimum values for a specific device so as when these limits are met, certain actions are taken. This can lead to a 5-20% energy saving due to our findings based on the data from hotels that our device is deployed.

4) A Real Green Product: The aim of the ZAC888DP product is to not only maximize energy savings, but at the same time to pay the minimum electricity required for the appliances unitized. For example, for one room utilizing the ZAC888DP device we have:

$$\begin{aligned}&10\text{mA controller} + 5\text{mA thermostat} + (5\text{mA sensors} \times 2)\times\\&24\text{volts} = 0.5\text{Watt}\end{aligned} \tag{1}$$

while by taking the scenario (1) and applying it to 300 hotel rooms we have the scenario (2).

$$\begin{aligned}&300 \times 0.5\text{W/controller/room} =\\&150\text{W} \rightarrow 0.150\text{KW} \times 24\text{hours} \times €0.21/\text{KWh} =\\&€0.756/\text{day} \rightarrow \textbf{Total: € 276 /year}\end{aligned} \tag{2}$$

While if we use scenario (1) for any other system we have scenario (3).

$$10\text{W/Room} \tag{3}$$

As every other room utilizes its own power supply. As anticipated, a massive increase from 0.5 watts to 10 watts will also increase the power cost required.

By taking the scenario of any other system from (3) and applying it to a 300 room hotel we have scenario (4).

$$\begin{aligned}&300 \times 10\text{W} =\\&3000\text{W} \rightarrow 3\text{KW} \times 24\text{hours} \times €0.21/\text{KWh} =\\&€15.12/\text{day} \rightarrow \textbf{Total: € 5,519 /year}\end{aligned} \tag{4}$$

That is an astonishing 95% savings originating only from the system consumption. The system operates on a wired network and is powered over Bus (PoB) allowing us to make use of one central intelligent and industrial degree SMPS to limit 240VAC to 24VDC conversion losses to one only and not to 300 as with the case of system using Distributed Power Supplies.

During operation of the system, no extra energy loss is added as the system utilizes its own latching relays. These relays have zero consumption while all other type of relays/-contactors will add extra cost to the energy required. Note that, wireless systems also have an extra cost as they require battery replacements every 2 years.

B. Simulation interface

In this section we outline the interconnection among ZAC888DP and ESP32, a lightweight microcontroller for TinyML integration in the proposed system so as to create an environment close to real-world for accident prediction. In our experiments, sensors where utilized that measured light intensity, water content, humidity, and temperature. Using these four factors, we could recreate the lavatory environment as closely as possible to real-world scenarios. Our objective is to trigger the user's accident-prevention systems when certain situations coincide. For almost 35 days, tens of thousands of values were sampled using the Arduino IDE and the Ardu-spreadsheet extension in order to get a reasonably accurate data set and improve the precision of our model. The samples of the measurements are within a 5-minute interval rate.

$$\frac{60}{5} = 12\backslash\text{hour},\ 12\times 24 = 288 \text{ per day},\ \frac{10000}{288} = 34,72 \text{ days} \tag{5}$$

Initially, the light factor is one of the most significant determinants of potential accidents, as the chance of occurrence in any part of the home quickly rises in the absence of illumination. As sliding in the restroom is a regular hazard, we were then given water pricing. Through our DHT-11 sensor, we obtained data for humidity and temperature that are significant, particularly when they are extremely high. Therefore, we assign weights of 50% to the brightness, 30% to the volume of water, 10% to the humidity, and 10% to the temperature for our four variables. The sensors for water and light are analogue, whereas the sensors for temperature and humidity are digital.

This indicates that the first two numbers range from 0 to 4095, which corresponds to the analogue subdivision after its digital conversion. To apply the proposed model, we therefore normalise the numbers so they are all in between the $[0, 1]$ range. To perform so, we use the formula as in (6).

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{6}$$

where x is the value we get each time and the min/max the maximum and minimum that can be sampled by our sensor. Then, we select the threshold at which, after applying the normalised data and considering the weights, we have a considerable risk of dropping, at 0.75. The formula is as in (7). The variables below water, light, humidity and temp, refer to the normalized values

$$0.3 \times \text{Water} + 0.5 \times \text{Light} + 0.1 \times \text{Humidity} + 0.1 \times \text{Temp} \tag{7}$$

By utilizing supervised learning, the output is 1 for every result over 0.75 possibility and 0 otherwise. The following table provides a summary of the preceding variables.

The accurate and efficient implementation of TinyML, TensorFlow, Arduino Ide, and ESP32 needs a sequence of particular processes. In the computer language Python, where our model is trained, we input our data through a csv file. Then, we present the TensorFlow libraries and develop the calculation procedure for the issue we want to solve by using them. At the conclusion of the course, we will convert our outcomes into a hex file and register this table in the Arduino IDE environment. Finally, we will upload it to our microprocessor and provide different control values through our sensors to see whether our model has been successfully implemented. Algorithm 1 presents the model execution as well as the inner workings of the proposed system.

TABLE I
DATA DISTRIBUTION AND NORMALIZATION

	Water Level	Light Value	Humidity	Temperature
Weight	0,3	0,5	0,1	0,1
Minimum	0	4095	20	0
Maximum	4095	0	90	50
Normalize	$\frac{x}{4095}$	$\frac{x-4095}{4095}$	$\frac{x-20}{70}$	$\frac{x}{50}$
Possibility (P)	Formula (7)	Formula (7)	Formula (7)	Formula (7)

Algorithm 1 Accident Prevention Alert System (APAS)

Require: Water, Light, Humidity and Temperature Values
Ensure: Accident Prevention Alert System
1: **for** $0.3 \times \text{Water} + 0.5 \times \text{Light} + 0.1 \times \text{Humidity} + 0.1 \times \text{Temp}$ **do**
2: Set threshold value of alert $\longrightarrow 0.23$
3: Calculate likelihood of alert
4: **if** Formula value of step $1 > 0.23 \rightarrow$ formula = 1 **then**
5: Return result as ALERT
6: Trigger the alert system
7: **else if** Formula value of step $1 < 0.23 \rightarrow$ formula = 0 **then**
8: Return result as FINE
9: **end if**
10: **end for**

For the model creation, the *pandas* and *numpy* libraries are imported. The *pandas* library is used to read the data from the csv file, while the *numpy* library is used to modify it. Then, we provide the train test split technique, which is used to divide our data set into training set and evaluation set. In addition to the confusion matrix, we import from *sklearn* the evaluation metrics of the model that we will show at the conclusion of our work. For the data representation we also utilize the *matplotlib* as the plotting library, which enables the creation of graphs. Next, we import the applicable model and field from the *TensorFlow* library, to perform Machine Learning techniques.

We begin by reading the data and converting it to the desired format. Then, we divide our data into input and output components. For the output, we set the value 1 when "Alarm" is present, i.e., when the specified threshold is met. After establishing the input and output, we split the data into training and evaluation data, with training accounting for 67% and evaluation comprising the remaining 33%. Using the methods of the relevant library, we transform the data sets to a *numpy* array format before concluding. We use the Sequential model to which we have added a Dense layer containing a neuron that predicts the output. The sigmoid function is utilized, which applies logistic regression, as the activation function. When compiling, the loss counter is defined as 'binary crossentropy'

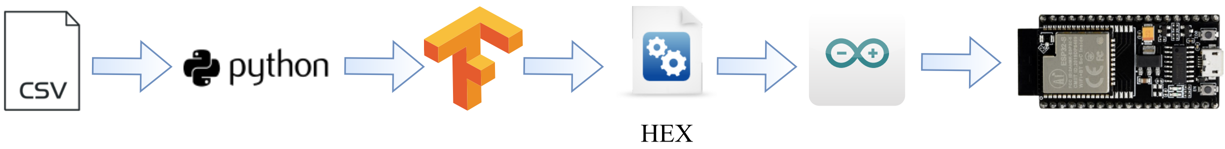

Fig. 2. Steps of model training and deployment

and 'rmsprop' optimizers, as well as the measure 'accuracy'. The model is trained for 256 episodes, and based on the assessment of the input dataset, we create a forecast and set the warning threshold to 0.23. Having trained the model, the presentation and analysis follows using the described metrics precision, recall, accuracy, and F1 Score. We generate the Confusion matrix and assign the real values from the Dataset to the y test and temp input variables, respectively. Next, we display the plots of accuracy and loss for both training and validation of our output data in proportion to the 256 epoches that represent the number of model learning attempts specified. Using the converter, we do the required conversion from keras to tflite and export our model to the converted model tflite file.

To start the integration process of the pre-trained machine learning model, we imported the $EloquentTinyML$ TensorFlow libraries and the digit model files that our model resides in, as well as DHT in order to use sensor-specific components. Next, we specify the distance from our CPU at which our sensors will be placed. At 22 the DHT11 was inserted, at 36 the water sensor (active when it receives values through 13), and at 39 the light sensor. For our model, we define four inputs, one output, and its function.

Finally, we regulate our threshold to trigger the Alarm and the input of ZAC888DP by setting Y PIN to low, therefore engaging the preventative measures and producing a possible difference in the input of our device. With the delay instruction, the whole procedure within the loop function is repeated every 2 minutes.

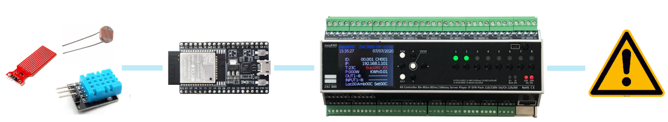

Fig. 3. Model

IV. Experimental Results

In this section the assessment of our model takes place by metric evaluation and using the approach of confusion matrix measuring how many of our binary outputs have been classified properly. In addition, instances of usage and their consequences are described. Having split our data set into 67% training and 33% validation, we have determined that our total number of instances is 3299 (9996 total examples). The following metrics are calculated:

$$\text{Accuracy:} = \frac{TP + TN}{TP + TN + FN + FP} \tag{8}$$

$$\text{Precision:}\ \frac{TP}{TP + FP} = \frac{241}{241 + 138} \tag{9}$$

$$\text{Recall:}\ \frac{TP}{TP + FN} = \frac{241}{241 + 0} \tag{10}$$

$$\text{F1-Score:}\ \frac{2 \times (\text{ Recall } \times \text{ Precision })}{\text{Recall } + \text{ Precision}} \tag{11}$$

Results for Eq.(8),(9),(10),(11) are summarized in Table II. The accuracy is sufficiently high at 94.11% while the recall score is 100%. The precision metric appears low at 63% but this is due to the algorithm's aggressive threshold at 0.23 value.

TABLE II
Evaluation Metrics

Metric	Score
Accuracy	94.11%
Precision	63.58%
Recall	100.00%
F1-Score	77.30%

The next two Figures 4 and 5 examine the accuracy and loss of the 256 training attempts of the model in detail. The blue line indicates training, whereas the orange line shows assessment/validation of findings. The model accuracy is sufficiently high and the model loss is acceptably low.

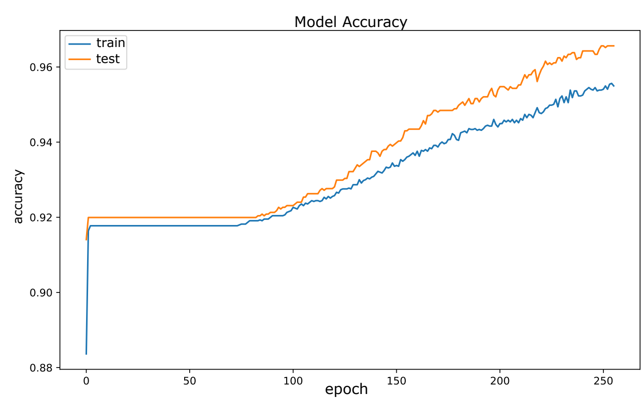

Fig. 4. Model Accuracy

Below, Figure 6 shows the confusion matrix and the distribution percentages of the results.

A. Use cases

Four instances of particular scenarios that we would want to explore are provided below. The conditions are the following:

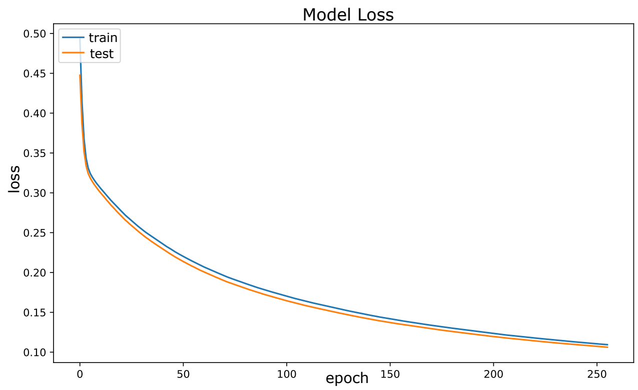

Fig. 5. Model Loss

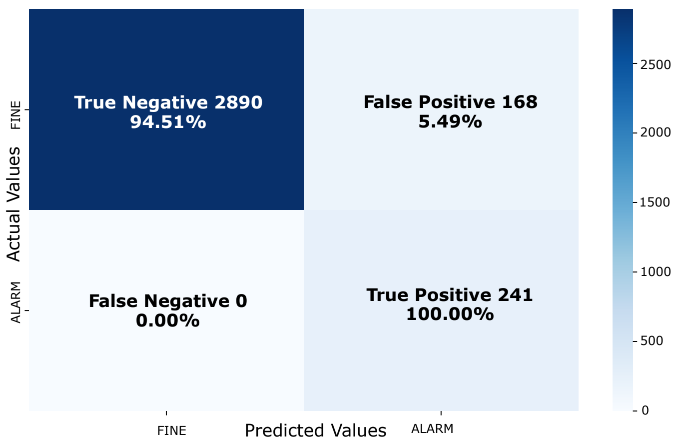

Fig. 6. Confusion Matrix based on the ML model

- During the day, either natural light or the lights of the room are on.
- Water is detected only when one is careless and creates water sources in specific locations.
- Despite low gravity and influence on our output, temperature and humidity remain at normal levels because we are discussing indoor space and the user can control these values.

Consequently, we have two instances of normal measurements: one that eventually has a Fine impact but is near the limit, and one that is at the extremes and immediately generates an Alarm signal.

Use Case I: Normal values - Fine Result. Zero quantity of water, high light intensity, low humidity, and normal room temperature result in very low normalised values and the output of 0.00 with the indicator that everything is OK. The results for this use case are shown in Table III. In this use case, the output of formula (7) is 0.00 as each of the normalized values is zero.

Use Case II: Normal values - Fine result. Partial quantity of water, medium-intensity illumination, low humidity, and normal to low room temperature all contribute to low normalised values and, therefore, the output of 0.00 with the indicator that everything is well. The results for this use case are shown in Table IV. In this use case, the output of formula (7) is 0.00 as most of the normalized values are zero except light value.

TABLE III
SIMULATION USE CASE I

Values	Water	Light	Humidity	Temperature
True Variables	0.00	3930	41.00%	24.80°C
Norm. Variables	0.00	0.04	0.30	0.50
Output	0.00			
Result	Fine			

TABLE IV
SIMULATION USE CASE II

Values	Water	Light	Humidity	Temperature
True Variables	1194.00	1852.00	29.00%	21.00°C
Norm. Variables	0.29	0.54	0.12	0.38
Output	0.00			
Result	Fine			

Use Case III: Borderline values - Fine Result. Partial quantity of water, very low light intensity, high humidity, and normal room temperature result in pretty high normalised values and an output of 0.22, which is marginal since 0.23 and above triggers an alert, therefore in this case we get a Fine signal. The results for this use case are shown in Table V. In this use case, the output of formula (7) is 0.00 as most of the normalized values are zero except light value. The overall output of the formula is 0.22 which is borderline value as from 0.23 and so on the alarm is triggered.

TABLE V
SIMULATION USE CASE III

Values	Water	Light	Humidity	Temperature
True Variables	1626.00	90.00	81.00%	21.40°C
Norm. Variables	0.39	0.94	0.87	0.42
Output	0.22			
Result	Fine			

Use Case IV: Extreme Values - Alarm Result.

Extremely large amounts of water, low light intensity, high relative humidity, and normal room temperature result in extremely high normalised values and the output of 0.74 with an alarm indicator. The results for this use case are shown in Table VI. The overall output of the formula is 0.74 which is high value compared to 0.23 where the alarm threshold exists.

TABLE VI
SIMULATION USE CASE IV

Values	Water	Light	Humidity	Temperature
True Variables	3580.00	400.00	80.00%	21.80°C
Norm. Variables	0.87	0.90	0.85	0.43
Output	0.74			
Result	Alarm			

V. Conclusions and Future Work

In the context of this work, a method for preventing and informing users of potential lavatory accidents was developed. To perform so, recent and suitable solutions as well as technologies that have emerged in recent years and seem to have much to offer the IT sector and human existence are utilized. We use the ZAC888DP device, which is a sturdy and favoured option for smart building installations on the market along with ESP32, a suitable alternative for a low-cost device, which was used to build our data collection in conjunction with sensors measuring water, temperature, humidity, light, and motion. Following a brief examination of related works, we concentrated on supervised learning specifically linear regression that yields a binary outcome based on the sigmoid function for model creation. TensorFlow was also utilized for TinyML integration with outstanding efficiency and dependability for the development of a forecasting and information system for user-dangerous scenarios. The model was built in Python and converted using TensorFlow lite before applying it to ESP32 and transferring the result to ZAC888DP, which activates the warning systems. Finally, evaluation metrics are used for the pre-trained machine learning model.

Future directions of this work include the assessment of additional values and parameters for more precise findings, the establishment of a connection with a mobile device of a user and the delivery of push notifications. The notifications issued by ZAC888DP, to alert the reception in the case of a hotel, will also forward the notice to the user. Moreover, another future direction is the use of wireless sensors instead of wired sensors for the implementation of the system in preexisting structures along with using IoT devices on-site to get additional data for model training. The ML model developed may be utilized to avoid accidents in various environments, such as nursing homes, by modifying the input parameters of the algorithm. In order to compare with actual numbers, the data collecting mechanism can be deployed in a real world deployment, and in the event of an accident, there must be an entry file and the value must be set at any given moment enabling us to determine if our settings and findings are precise.

References

[1] C. Karras, A. Karras, and S. Sioutas, "Pattern recognition and event detection on iot data-streams," *arXiv preprint arXiv:2203.01114*, 2022.

[2] C. Karras and A. Karras, "Dbsop: An efficient heuristic for speedy mcmc sampling on polytopes," *arXiv preprint arXiv:2203.10916*, 2022.

[3] C. Karras, A. Karras, M. Avlonitis, and S. Sioutas, "An overview of mcmc methods: From theory to applications," in *Artificial Intelligence Applications and Innovations. AIAI 2022 IFIP WG 12.5 International Workshops* (I. Maglogiannis, L. Iliadis, J. Macintyre, and P. Cortez, eds.), (Cham), pp. 319–332, Springer International Publishing, 2022.

[4] C. Karras, A. Karras, M. Avlonitis, I. Giannoukou, and S. Sioutas, "Maximum likelihood estimators on mcmc sampling algorithms for decision making," in *Artificial Intelligence Applications and Innovations. AIAI 2022 IFIP WG 12.5 International Workshops* (I. Maglogiannis, L. Iliadis, J. Macintyre, and P. Cortez, eds.), (Cham), pp. 345–356, Springer International Publishing, 2022.

[5] C. Karras, A. Karras, G. Drakopoulos, K. Tsakalidis, P. Mylonas, and S. Sioutas, "Weighted reservoir sampling on evolving streams: A sampling algorithmic framework for stream event identification," in *Proceedings of the 12th Hellenic Conference on Artificial Intelligence*, SETN '22, (New York, NY, USA), Association for Computing Machinery, 2022.

[6] A. Karras and C. Karras, "Integrating User and Item Reviews in Deep Cooperative Neural Networks for Movie Recommendation," *arXiv preprint arXiv:2205.06296*, 2022.

[7] C. Karras, A. Karras, L. Theodorakopoulos, I. Giannoukou, and S. Sioutas, "Expanding queries with maximum likelihood estimators and language models," in *Proceedings of the ICR'22 International Conference on Innovations in Computing Research* (K. Daimi and A. Al Sadoon, eds.), (Cham), pp. 201–213, Springer International Publishing, 2022.

[8] J. A. Stevens, E. N. Haas, and T. Haileyesus, "Nonfatal bathroom injuries among persons aged $\geq$15years—united states, 2008," *Journal of Safety Research*, vol. 42, no. 4, pp. 311–315, 2011.

[9] A. Delle Femine, D. Gallo, C. Landi, A. Lo Schiavo, and M. Luiso, "Low power contactless voltage sensor for low voltage power systems," *Sensors*, vol. 19, no. 16, p. 3513, 2019.

[10] M. Alonso, H. Amaris, D. Alcala, D. M. Florez R, *et al.*, "Smart sensors for smart grid reliability," *Sensors*, vol. 20, no. 8, p. 2187, 2020.

[11] Z. Foroozandeh, S. Ramos, J. Soares, Z. Vale, and M. Dias, "Single contract power optimization: A novel business model for smart buildings using intelligent energy management," *International Journal of Electrical Power & Energy Systems*, vol. 135, p. 107534, 2022.

[12] C. Koutroumpina, S. Sioutas, S. Koutroubinas, and K. Tsichlas, "Evaluation of features generated by a high-end low-cost electrical smart meter," *Algorithms*, vol. 14, no. 11, p. 311, 2021.

[13] A.-R. Al-Ali, I. A. Zualkernan, M. Rashid, R. Gupta, and M. AliKarar, "A smart home energy management system using iot and big data analytics approach," *IEEE Transactions on Consumer Electronics*, vol. 63, no. 4, pp. 426–434, 2017.

[14] V. Marinakis and H. Doukas, "An advanced iot-based system for intelligent energy management in buildings," *Sensors*, vol. 18, no. 2, p. 610, 2018.

[15] Y. Liu, C. Yang, L. Jiang, S. Xie, and Y. Zhang, "Intelligent edge computing for iot-based energy management in smart cities," *IEEE network*, vol. 33, no. 2, pp. 111–117, 2019.

[16] A. Karras, C. Karras, K. C. Giotopoulos, I. Giannoukou, D. Tsolis, and S. Sioutas, "Download speed optimization in p2p networks using decision making and adaptive learning," in *Proceedings of the ICR'22 International Conference on Innovations in Computing Research* (K. Daimi and A. Al Sadoon, eds.), (Cham), pp. 225–238, Springer International Publishing, 2022.

[17] I. Sittón-Candanedo, R. S. Alonso, Ó. García, L. Muñoz, and S. Rodríguez-González, "Edge computing, iot and social computing in smart energy scenarios," *Sensors*, vol. 19, no. 15, p. 3353, 2019.

[18] T. Han, K. Muhammad, T. Hussain, J. Lloret, and S. W. Baik, "An efficient deep learning framework for intelligent energy management in iot networks," *IEEE Internet of Things Journal*, vol. 8, no. 5, pp. 3170–3179, 2020.

[19] J.-L. Casteleiro-Roca, J. F. Gómez-González, J. L. Calvo-Rolle, E. Jove, H. Quintián, B. Gonzalez Diaz, and J. A. Mendez Perez, "Short-term energy demand forecast in hotels using hybrid intelligent modeling," *Sensors*, vol. 19, no. 11, p. 2485, 2019.

[20] B. K. Mohanta, D. Jena, N. Mohapatra, S. Ramasubbareddy, and B. S. Rawal, "Machine learning based accident prediction in secure iot enable transportation system," *Journal of Intelligent & Fuzzy Systems*, vol. 42, no. 2, pp. 713–725, 2022.

[21] U. Alvi, M. A. K. Khattak, B. Shabir, A. W. Malik, and S. R. Muhammad, "A comprehensive study on iot based accident detection systems for smart vehicles," *IEEE Access*, vol. 8, pp. 122480–122497, 2020.

[22] N. Kumar, D. Acharya, and D. Lohani, "An iot-based vehicle accident detection and classification system using sensor fusion," *IEEE Internet of Things Journal*, vol. 8, no. 2, pp. 869–880, 2020.

[23] F. Zantalis, G. Koulouras, S. Karabetsos, and D. Kandris, "A review of machine learning and iot in smart transportation," *Future Internet*, vol. 11, no. 4, p. 94, 2019.

[24] S. Goyal, P. Bedi, J. Kumar, *et al.*, "Realtime accident detection and alarm generation system over iot," in *Multimedia Technologies in the Internet of Things Environment, Volume 2*, pp. 105–126, Springer, 2022.

[25] M. Abdou, R. Mohammed, Z. Hosny, M. Essam, M. Zaki, M. Hassan, M. Eid, and H. Mostafa, "End-to-end crash avoidance deep iot-based solution," in *2019 31st International Conference on Microelectronics (ICM)*, pp. 103–107, IEEE, 2019.