Ingeniería y Competitividad
Vol 28 (1) 2026 doi: 10.25100/iyc.v28i1.15038

Vol 28 (1)

ISSN
0123-3033
e- 2027-8284

Original Article

Ultra-Low-Cost Device for Liquor Classification using TinyML Technology

Dispositivo de ultra-bajo costo para la clasificación de licores usando tecnología TinyML

Ilber-Adonayt Ruge-Ruge[1] Ingrid-Carolina Ortiz-Álvarez[1] Fabián-R. Jiménez-López[1]

[1] Grupo de Investigación I2E, Escuela de Ingeniería Electrónica, Facultad de Ingeniería, Universidad Pedagógica y Tecnológica de Colombia. Tunja - Boyacá, Colombia.

How to cite?

Ruge-Ruge IA, Ortiz-Álvarez IC, Jiménez-Lopez FR. Ultra-Low-Cost Device for Liquor Classification using TinyML Technology. Ingeniería y Competitividad, 2025, 28(1)e-20215038

https://doi.org/10.25100/iyc.v28i1.15038

Received: 15/06/25
Reviewed: 25/09/25
Accepted: 21/01/26
Online: 10/02/26

Correspondence

ilber.ruge@uptc.edu.co

Spanish version

Abstract

Objective: To design and implement a liquor identification system using TinyML, employing an MQ-135 sensor and an ESP32 microcontroller from Espressif Systems. The objective is to optimize local processing for high precision and low latency, thereby validating its viability as an accessible solution in resource-constrained environments.
Methodology: The methodology employed an Artificial Neural Network (ANN) to classify liquors based on volatile compounds detected by the sensor. A dataset comprising 6,000 measurements was collected, and the ANN model was trained in MATLAB® R2018a. For performance evaluation, the dataset was split into 70% for training, 15% for validation, and 15% for testing, with the Mean Squared Error (MSE) used as the primary metric.
Results: The deployed ANN model converged after 375 epochs, achieving a minimal Mean Squared Error (MSE) of 1.05×10□□ and a correlation coefficient (R) of 1.0. Synthesized onto the ESP32 microcontroller, the model utilized only 24.8% of Flash memory and 6.3% of RAM. The inference time recorded for execution was 2.43 ms, yielding a classification accuracy of 84.4%. This cost-efficient solution, priced at 15 USD, outperforms existing commercial options.
Conclusions: It achieves an exceptionally fast inference time of 2.43 ms with a minimal operational energy consumption ranging from 5 mA to 10 mA. This performance, coupled with the low fabrication cost (15 USD), contrasts sharply with commercial solutions often exceeding hundreds of dollars. Consequently, the system is highly viable for implementation in embedded quality control and food industry applications.

Keywords: TinyML, Liquor classification, Artificial neural networks, MQ-135 sensor, ESP32 microcontroller.

Resumen

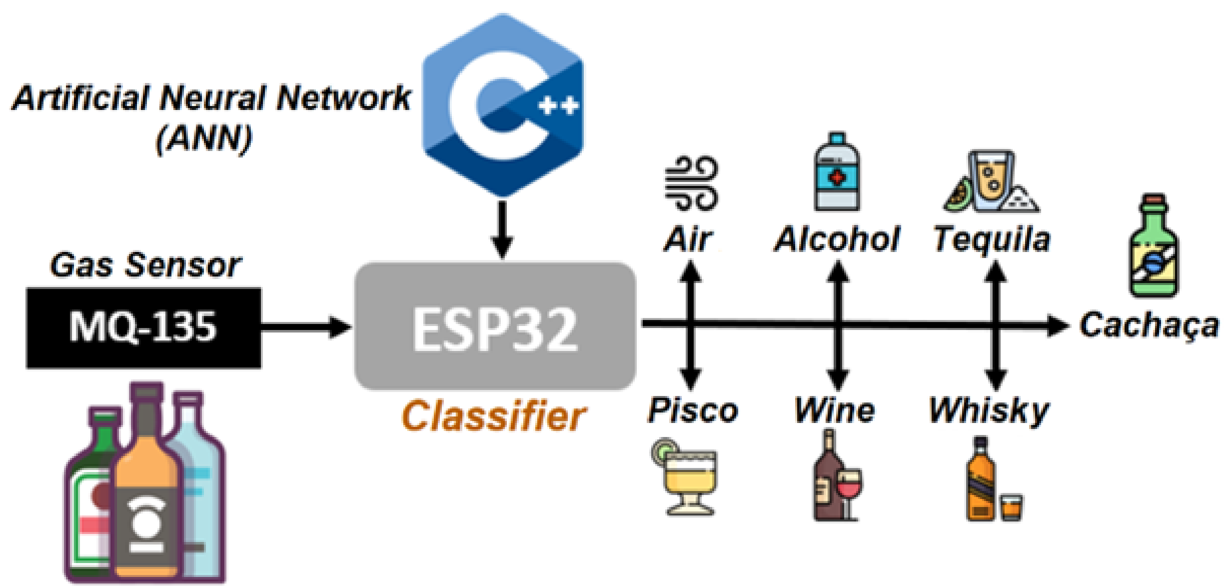
Objetivo: Diseñar e implementar un sistema de identificación de licores mediante TinyML, empleando un sensor MQ-135 y un microcontrolador ESP32 de Espressif Systems. El objetivo es optimizar el procesamiento local para lograr alta precisión y baja latencia, validando así su viabilidad como solución accesible en entornos con recursos limitados.
Metodología: La metodología empleó una Red Neuronal Artificial (RNA) para clasificar licores según los compuestos volátiles detectados por el sensor. Se recopiló un conjunto de datos con 6000 mediciones y el modelo de RNA se entrenó en MATLAB® R2018a. Para la evaluación del rendimiento, el conjunto de datos se dividió en un 70 % para entrenamiento, un 15 % para validación y un 15 % para pruebas, utilizando el Error Cuadrático Medio (EMM) como métrica principal.
Resultados: El modelo de ANN implementado convergió tras 375 épocas, alcanzando un Error Cuadrático Medio (EMM) mínimo de 1,05 × 10□□ y un coeficiente de correlación (R) de 1,0. Sintetizado en el microcontrolador ESP32, el modelo utilizó solo el 24,8 % de la memoria Flash y el 6,3 % de la RAM. El tiempo de inferencia registrado para la ejecución fue de 2,43 ms, lo que arroja una precisión de clasificación del 84,4 %. Esta solución rentable, con un precio de 15 USD, supera las opciones comerciales existentes.
Conclusiones: Alcanza un tiempo de inferencia excepcionalmente rápido de 2,43 ms con un consumo energético operativo mínimo, de entre 5 mA y 10 mA. Este rendimiento, sumado al bajo coste de fabricación (15 USD), contrasta marcadamente con las soluciones comerciales, que a menudo superan los cientos de dólares. En consecuencia, el sistema es muy viable para su implementación en aplicaciones integradas de control de calidad y de la industria alimentaria.

Palabras clave: TinyML, Clasificación de licores, Redes neuronales artificiales, sensor MQ135, Microcontrolador ESP32.

Why was this work conducted?
This work was conducted to demonstrate the capability of Tiny Machine Learning (TinyML) in liquor classification. TinyML enables the execution of machine learning models on resource-constrained devices, representing a novel solution compared to conventional approaches. The technology was applied to the classification of six types of alcoholic beverages: medicinal ethyl alcohol, wine, pisco, cachaça, tequila, and whiskey. The core objective was to develop a system characterized by its ultra-low cost (approximately 15 USD) and high computational efficiency (inference times of 2.48 ms, making it viable for implementation in embedded devices.

What were the most relevant results?
The results indicated a classification accuracy of 84%, which is comparable to some more expensive commercial solutions, such as the Gasboard-3210Plus (800 USD) or the Anton Paar analyzers (1,500 – 2,100 USD). With a latency of 32 ms, the system allows for fast and efficient real-time classification.

What do these results contribute?
The developed classification system, with an estimated cost of only 15 USD, represents an accessible alternative for beverage authentication. These findings suggest its strong potential for application in quality control, adulteration detection, and monitoring within the alcoholic beverage industry.



doi: 10.25100/iyc.v28i1.15038

Introduction

Currently, the development of low-cost devices for substance classification and detection has gained significant relevance, particularly in sectors such as the food and beverage industries. Tiny Machine Learning (TinyML), which enables the execution of machine learning models on resource-constrained devices, presents itself as an innovative solution for real-time classification tasks (1,2).

This technology offers considerable potential for implementing efficient and accessible systems that facilitate product identification, such as in the case of liquors, without the need for expensive or specialized equipment. The determinant factor for selecting TinyML as the core technology in this work is its capability, as demonstrated by Schizas et al. (3), to enhance performance and efficiency by processing data locally with minimal latency and significantly reduced energy consumption. By minimizing data transmission to the cloud, TinyML strengthens security and decreases operational costs, thereby consolidating its role as the foundation for intelligent, low-cost, and highly autonomous embedded systems.

A systematic literature review was conducted following the PRISMA (*Preferred Reporting Items for Systematic Reviews and Meta-Analyses*) methodology to identify TinyML applications in classification systems. In the identification phase, the keywords "*TinyML,*" "*Classification Machine Learning*," and "*Arduino*" were employed in databases available at the Universidad Pedagógica y Tecnológica de Colombia (UPTC, Tunja): ScienceDirect (25), Web of Science (18), Scopus (30), Elsevier (12), and Google Scholar (40), yielding 125 records. During the screening stage, pertinent titles and abstracts were considered, and subsequently, in the eligibility phase, only articles with full text available were analyzed. Ultimately, 12 studies met the inclusion criteria and were incorporated into the final analysis.

Diverse applications of TinyML and low-cost instrumentation have been explored in the literature. Tsoukas et al. (4) developed an autonomous TinyML–based system for ammonia leak and smoke detection, achieving an F1-Score of 0.77 for smoke and 0.70 for ammonia. Bagheri et al. (5) evaluated TinyML on microcontrollers and environmental sensors for real-time aquatic pollutant monitoring, highlighting its low cost and cloud-independent autonomy. Atanane et al. (6) implemented a water leak detection system in buildings using acoustic sensors, accelerometers, Convolutional Neural Networks (CNN), and TinyML on an Arduino Nano 33 BLE, achieving a precision of 97.45%.

Regarding liquor classification, Zhou et al. (7) reported 100% accuracy in classifying eight brands of Chinese liquor using gas sensors and Discriminant Function Analysis (DFA). Zhang et al. (8) utilized an electronic nose with doped ZnO sensors and Principal Components Analysis (PCA), Cluster Analysis (CA), and Learning Vector Quantization (LVQ) techniques, achieving up to 94.1% precision in characterizing Chinese vinegars. Scorsone et al. (9) developed an electronic nose using conductive polymer (CP) sensors for early fire detection, reducing false alarms through PCA analysis. Related research includes Zhang et al. (10) who employed zinc oxide (ZnO) gas sensors doped with MnO_2, TiO_2, and CO_2O_3, reaching a precision of 89.3% with LVQ.

doi: 10.25100/iyc.v28i1.15038

Furthermore, Wongchoosuk et al. (11) designed a portable electronic nose with hybrid carbon nanotube–SnO_2 sensors to detect methanol contamination in whisky, successfully identifying 1% methanol contamination. Other relevant applications include the wireless carbon monoxide monitoring system (MQ-9 sensors and LPWAN) by Vega-Luna et al. (12), achieving 11.8 km coverage, and the system by Nagy et al. (13) for measuring seven pollutant gases using MQ-2 and MQ-5 sensors on a Raspberry® Pi 3.

Finally, Jiang et al. (14) applied Machine Learning models, specifically Random Forest, achieving over 90% accuracy in predicting white wine quality. Specifically, within the Colombian context, Botero-Valencia et al. (15) successfully applied TinyML in an environmental monitoring station, achieving a 2.67% improvement in precision. This demonstrated a practical success case of utilizing the specified low-resource microcontrollers for high-impact environmental sensing in local settings, further justifying the architectural approach taken in the present study.

The relevant results of the literature review, highlighting the year, application, and location, are shown in Table 1.

Table 1. Literature Review Results

Year	Authors	Application	Microcontroller	Measured Variables	Sensors Used	Country
2023	Tsoukas, V. et al. (4)	Gas leak detection	Not specified	Smoke, ammonia	Gas sensors	Greece
2023	Bagheri, M. et al. (5)	Environmental monitoring in water	Not specified	Water pollutants	Environmental sensors	Iran
2023	Atanane, O. et al. (6)	Water leak detection	Arduino™ Nano 33 BLE	Vibrations, acoustics	Accelerometers, acoustic sensors	Morocco
2011	Zhou, Q. et al. (7)	Classification of Chinese liquors	Not specified	Volatile compounds	MOX sensor	China
2006	Zhang, Q. et al. (8)	Classification of Chinese vinegars	Not specified	Volatile compounds	Doped ZnO sensors	China
2006	Scorsone, E. et al. (9)	Early fire detection	Not specified	Smoke	Conducting polymer sensors	Italy
2005	Zhang, Q. et al. (10)	Classification of Chinese liquors	Not specified	Volatile compounds	Doped ZnO sensors	China
2010	Wongchoosuk, C. et al. (11)	Methanol detection in whiskey	Not specified	Methanol	CNT-SnO_2 sensors	Thailand

doi: 10.25100/iyc.v28i1.15038

Year	Authors	Application	Microcontroller	Measured Variables	Sensors Used	Country
2017	Vega-Luna, J. I. et al. (12)	Carbon monoxide monitoring	PIC18F8722	Carbon monoxide	MQ-9 sensor	Mexico
2020	Nagy, A. S. et al. (13)	Simultaneous gas measurement	Raspberry® Pi 3	Gases, temperature, humidity	MQ-2, MQ-5, DHT11	Cuba
2023	Jiang, X. et al. (14)	Wine quality prediction	Not applicable	Physicochemical properties	Not applicable	China
2023	Botero-Valencia, J. S. et al. (15)	Environmental pollution monitoring	ESP32-CAM	Air, noise, light	SPS30, SCD30, SCD40, BME680, BH1750, AS7341	Colombia

Complementary research has explored the application of TinyML across various distinct areas. Hayajneh et al. (16) proposed a framework based on TinyML and transfer learning to predict soil moisture in smart agriculture, utilizing drones and IoT sensors. Karras et al. (17) developed TinyML algorithms for data management in IoT, enhancing data cleaning, compression, and storage, with implementation on the Raspberry® Pi platforms.

Schizas et al. (3) conducted a systematic review on TinyML in low-power IoT deployments, highlighting benefits concerning transition, bandwidth, security, privacy, latency, energy efficiency, and low cost. Dutta and Bharali (18) analyzed the integration of TinyML in IoT, emphasizing advantages such as cost reduction and cloud independence, alongside associated hardware challenges. Srinivasagan et al. (19) designed a TinyML sensor to estimate the shelf life of fresh dates using VisNIR spectrometry.

Capogrosso et al. (20) classified development approaches into three categories: algorithmic, hardware-centric, and co-design. Banbury et al. (21) proposed benchmarks to evaluate TinyML platforms. Hymel et al. (22) presented Edge Impulse, a platform simplifying model development and deployment. Finally, Lê et al. (23) reviewed optimization techniques in neural networks for resource-limited microcontrollers. The relevant applications using TinyML technology are shown in Table 2.

doi: 10.25100/iyc.v28i1.15038

Table 2. Literature Review Results on TinyML Applications.

Year	Authors	Application	Country
2023	Hayajneh, A. M. et al. (16)	Soil moisture prediction in smart agriculture. UAVs, IoT devices TinyML	Jordan / United Kingdom
2024	Karras, A. et al. (17)	Big Data Management in IoT with TinyML Algorithms	Greece
2022	Schizas, N. et al. (3)	TinyML in low-power IoT deployments	Greece
2021	Dutta, L. y Bharali, S. (18)	TinyML integration in IoT	India
2023	Srinivasagan, R. et al. (19)	Shelf life estimation of fresh dates	Saudi Arabia / Egypt
2023	Capogrosso, L. et al. (20)	Review of TinyML Learning Algorithms	Italy
2020	Banbury, C. R. et al. (21)	Benchmarking TinyML platforms	Not Specified
2023	Hymel, S. et al. (22)	Edge Impulse platform for TinyML Deployment	Not Specified
2023	Lê, M. T. et al. (23)	Optimization techniques for neural networks on microcontrollers for TinyML	France

With this background, this study aims to propose an ultra-low-cost TinyML based device with the goal of classifying alcoholic beverages. Alcoholic beverages is a relevant context where speed and accuracy is critical since the importance around making the right classification within order of terms. The aim is to create an accurate, portable, and easy to follow path to classify types of alcoholic beverages based on conventional sensors and algorithms. The present work addresses the identified gap by proposing an ultra-low-cost device based on TinyML technology for liquor classification, an application where both precision and speed are critical factors.

The term ultra-low-cost characterizes embedded systems whose total cost of materials (Bill of Materials – BOM) is minimal, prioritizing economical components without compromising essential system functionality. The solution is implemented on the ESP32 microcontroller from Espressif Systems, selected specifically due to its low market cost (approximately 8 USD), wide market availability, and extensive developer documentation. This microcontroller features a dual-core Tensilica LX6 processor (up to 240 MHz), 520 KB of SRAM, 4 MB of Flash memory, and UART, SPI, I2C, and 12-bit ADC interfaces.

This architecture allows for the efficient execution of machine learning models despite its limited computational resources, integrating low-cost sensors to develop an accurate and economically accessible tool for alcoholic beverage identification. This proposed system is anticipated to contribute to both industrial operational efficiency and the democratization of advanced technologies in contexts with budgetary constraints (24,25).

doi: 10.25100/iyc.v28i1.15038

Methodology

The classification system is based on the MQ-135 gas sensor and an Artificial Neural Network (ANN). The MQ-135 sensor is a widely adopted metal oxide semiconductor (MOS) device capable of detecting various gases, including carbon monoxide (CO), alcohol, carbon dioxide (CO_2), toluene (C_7H_8), ammonia (NH_4), alcohol, carbon dioxide (26). The sensor's operating principle relies on the variation in the resistance of the sensitive material R_s in response to gas concentration.

Calibration is performed following the manufacturer's recommendations to determine the coefficient R_0 which corresponds to the sensor's baseline resistance in clean air. This R_0 value is crucial for normalizing the sensor response, as the ratio R_s / R_0 isolates the resistance change specifically due to the presence of target gas concentrations. This normalization is essential for validating the model's robustness against typical ambient fluctuations, such as changes in temperature and humidity. The relationship between the measured resistance R_s and the gas concentration C is expressed by a logarithmic model derived from the sensor's characteristic curve, as presented in Equation 1.

$$\frac{R_s}{R_o} = A * (C)^{-B} \tag{1}$$

where R_s is the sensor resistance at a gas concentration C, R_0 is the resistance in clean air, and A with B are experimentally determined constants specific to the type of gas.

The ANNs were employed to identify liquors based on their distinct volatile organic compound (VOC) signatures, resulting in robust and high-precision models suitable for quality control and beverage authentication. Specifically, the system is designed to perform quality control by verifying the VOC signature against a known standard and beverage authentication by detecting unauthorized variations, such as dilution or the presence of adulterants like methanol. In general terms, a Multilayer Perceptron (MLP) network can be represented by the following expressions shown in Equations 2 and 3.

$$z_j = \sum_{i=1}^{n} w_{ij} x_i + b_j \tag{2}$$

$$y_k = f(z_j) \tag{3}$$

where x_i represents the input variable corresponding to the voltage value obtained from the MQ–135 sensor, w_{ij} are the synaptic weights, b_j are the associated biases, and y_k is the activation function (27).

The neural network training was conducted in MATLAB® R2018a utilizing the 'nntraintool', which simplifies the configuration and training via algorithms such as backpropagation and provides graphical tools for evaluating model performance (28). The optimized model was then implemented on an ESP32 board, a microcontroller with performance adequate for IoT, TinyML, and Edge Processing applications. Edge processing is understood as the capacity to execute analysis and decision-making tasks directly on the device without reliance on a continuous cloud connection (29).

doi: 10.25100/iyc.v28i1.15038

The ESP32 development board integrates a dual-core Tensilica Xtensa LX6 processor operating at up to 240 MHz, 520 KB of SRAM, and 4 MB$ of Flash memory, along with communication interfaces such as Serial Peripheral Interface (SPI), Inter-Integrated Circuit (I2C), Universal Asynchronous Receiver-Transmitter (UART), and a 12–bit Analog-to-Digital Converter (ADC). This architecture enables efficient integration with sensors and data acquisition modules (30). Figure 1 shows the block diagram of the proposed TinyML system for liquor classification.

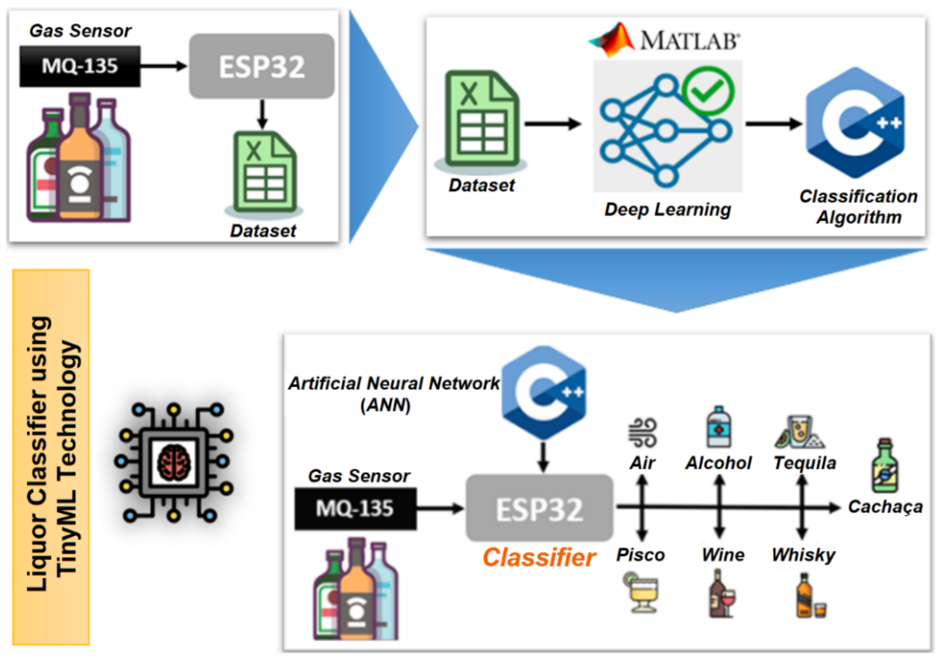


Figure 1. The development workflow of the TinyML liquor classifier.

The connection with the MQ-135 sensor was managed through the 'MQUnifiedsensor.h' library in the standard Arduino™ IDE, allowing for value reading and parameter configuration for each measured gas.

The dataset for training the neural network was constructed from MQ-135 sensor measurements, capturing the concentration of volatile compounds such as alcohol, hexane, carbon monoxide (CO), benzene, and Liquefied Petroleum Gas (LPG). The acquisition process involved samples of ambient air, medicinal ethyl alcohol, wine, pisco, cachaça, tequila, and whisky. Reference voltage values and the R_0 coefficient were recorded to monitor sensor stability.

The final classification system was designed with seven output categories: Alcohol, Wine, Pisco, Cachaça, Tequila, Whisky, and Ambient Air (used as the reference). To ensure robust data representation, 100 samples were collected for each of the seven categories, with 10 repetitions for each sample to account for signal capture variation. This generated an initial total of 7 categories × 100 samples × 10 repetitions, resulting in 7,000 records. As noted in preliminary analysis, the first 50 initial readings were discarded due to the settling time and inherent instability of the MQ-135 sensor during warm-up. Consequently, the final usable dataset consisted of 6,950 records. The input

doi: 10.25100/iyc.v28i1.15038

features captured the concentration of the various volatile compounds, and the output was labeled using a binary scheme corresponding to the presence of each liquor type or ambient air.

Figure 2 gives an example of a portion of the dataset structure within Excel™, with each capture and the corresponding classification in rows.

Alcohol	Hezane	CO	Benzene	LPG	Volt	R0	Alcohol	Wine	Pisco	Cachaca	Tequila	Whisky	Air
0.13	878.1	28949.11	0.64	3840.68	4.35	0	1	0	0	0	0	0	0
0.13	888.46	28498.36	0.64	3840.68	4.35	0	1	0	0	0	0	0	0
0.13	888.46	28949.11	0.64	3840.68	4.35	0	1	0	0	0	0	0	0
0.07	324.93	7630.96	0.25	1250.89	4.11	0.01	0	0	1	0	0	0	0
0.07	327.95	7726.26	0.25	1237.77	4.11	0.01	0	0	1	0	0	0	0
0.08	331.01	7822.94	0.25	1264.17	4.12	0.01	0	0	1	0	0	0	0
0.05	175.79	3347.78	0.14	609.18	3.94	0.01	0	0	0	0	1	0	0
0.05	175.79	3347.78	0.14	614.84	3.94	0.01	0	0	0	0	1	0	0
0.05	174.37	3311.51	0.14	609.18	3.94	0.01	0	0	0	0	1	0	0
0	0.07	0.09	0	0.08	0.92	0.15	0	0	0	0	0	0	1
0	0.07	0.09	0	0.08	0.92	0.15	0	0	0	0	0	0	1
0	0.07	0.09	0	0.08	0.92	0.15	0	0	0	0	0	0	1

Figure 2. Structure of dataset for training the TinyML liquor classification system.

The neural network training was conducted in MATLAB® R2018a using the 'nntraintool', facilitating configuration and training via the Levenberg–Marquardt algorithm (31). This algorithm leverages a combination of the steepest descent and Gauss–Newton methods to efficiently minimize the Mean Squared Error (MSE). Mathematically, the weight update is defined as shown in Equation 4.

$$\Delta w = -|\mathbf{J}^T\mathbf{J} + \mu\mathbf{I}|^{-1}\mathbf{J}^T\mathbf{e} \tag{4}$$

where **J** is the Jacobian matrix of the error partial derivatives with respect to the weights, **e** is the error vector, $\mathbf{I}$ is the identity matrix, and μ is an adjustment factor that regulates the transition between the steepest descent behavior (when μ is large) and the Gauss–Newton method (when μ is small).

The network architecture was configured with 5 input neurons, one hidden layer, and 6 output neurons, corresponding to the final classification classes. Tests were performed by varying the number of neurons in the hidden layer to analyze its effect on convergence. The primary loss function utilized was the Mean Squared Error (MSE), and the data were partitioned into 70\% for training, 15% for validation, and 15% for testing.

doi: 10.25100/iyc.v28i1.15038

Results and discussion

Sensor Data Preprocessing and Stability

The data used for training exclude the first 50 readings of the sampling process for each input class, as it was observed that the sensor values remained unstable during that time interval, as shown in Figure 3. This initial variability, which stabilizes after the warm-up period, necessitates the exclusion of early samples to ensure that the model is trained only on reliable data, thereby maximizing the model's predictive precision.

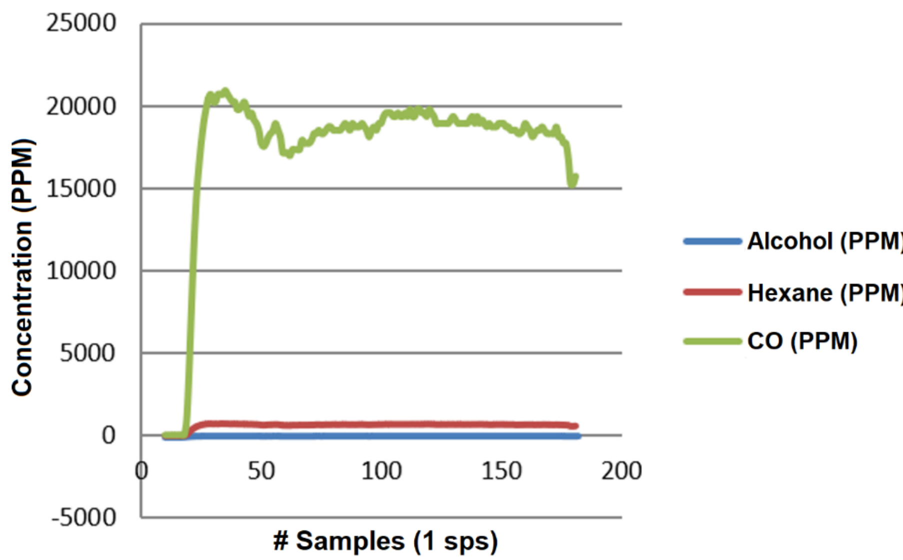


Figure 3. Variability of the MQ135 sensor during a new measurement.

With respect to preprocessing, this step was integral in ensuring that the training data was representative of a sensor behaving at steady state, and did not allow the model to learn the surrounding patterns of instability, but rather, the composite characteristics of the target substances. Implementing this form of data conditioning is a common practice in sensor-based machine learning applications to aid in performance improvements and reliability.

Neural Network Performance and Convergence

The neural network training process demonstrated adequate convergence under conditions of gradient stability and minimal variation between the training, validation, and testing sets. This stability is evidenced by the progressive reduction of the Mean Squared Error (MSE) over 375 epochs, reaching a minimum value of 2.8083×10^{-9} in the validation phase (See Figure 4). Furthermore, the final gradient (9.91×10^{-8}) and the error (1.05×10^{-9}) remained within optimal convergence ranges, showing no discernible signs of overfitting during the training process.

doi: 10.25100/iyc.v28i1.15038

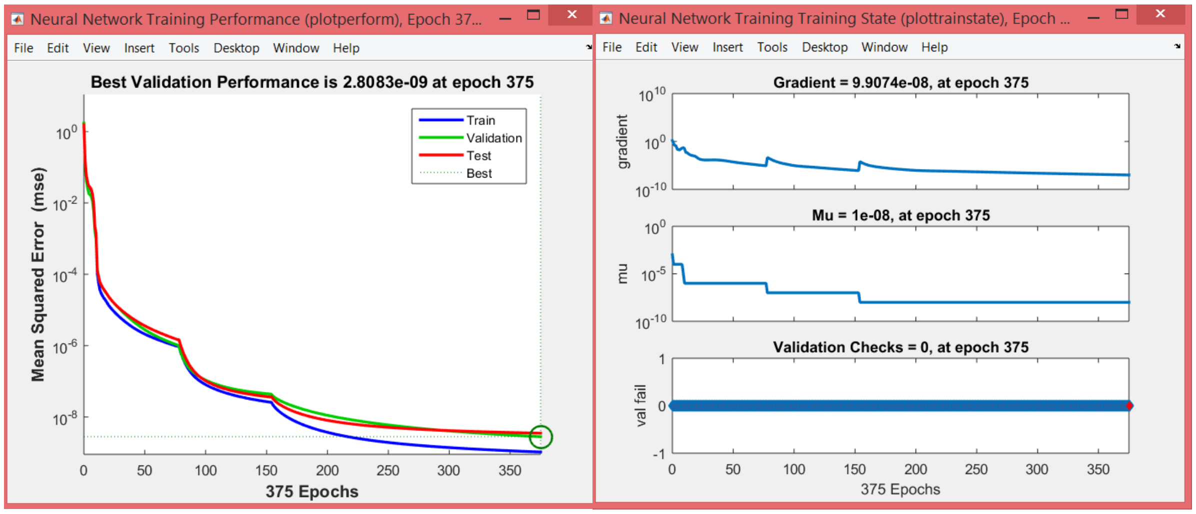


Figure 4. Training performance of the classification model.

Model Regression Analysis

In addition to the reduction of the MSE during training, the regression evaluation between the predicted outputs and the target values showed an extremely high model fit. As illustrated in Figure 5, the correlation coefficients (R) for the training, validation, testing, and total data sets all achieved values equal to 1.0. This result suggests a near-perfect linear relationship between the network outputs and the expected target values.

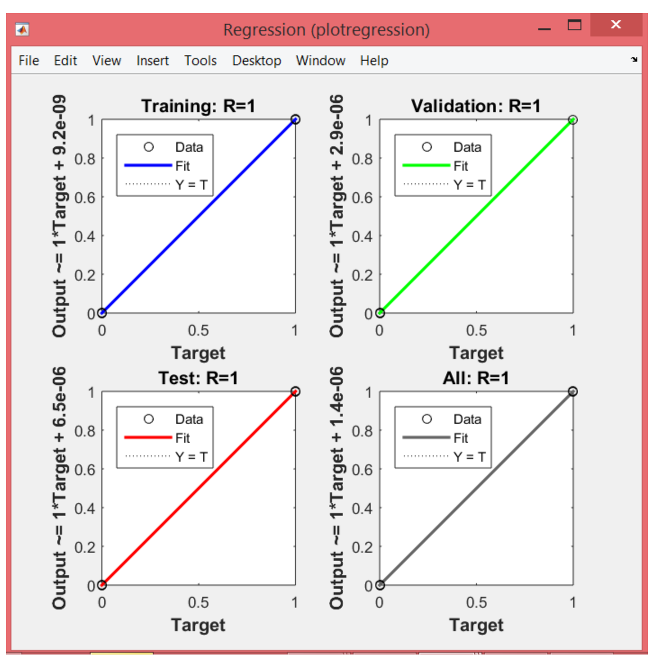


Figure 5. Regression evaluation of the classification model.

The observation of MSE values approaching zero and R = 1 across all data partitions, despite the relatively limited size of the dataset, is primarily attributed to two factors: (1) the inherent separability of the classes and (2) the efficiency of the Levenberg–Marquardt optimization algorithm. Given that the volatile organic compound (VOC) profiles of the seven categories (six liquors

doi: 10.25100/iyc.v28i1.15038

and ambient air) are chemically distinct, the neural network was able to learn highly separable boundaries with minimal error.

The non-linear nature of the sensor response, when mapped through the MLP's hidden layer, likely results in a clear projection space, leading to a near-perfect fit on the training and validation sets. This behavior has also been reported in similar studies, such as that by Aljohani et al. (32), who obtained R = 1 when applying neural networks trained with the Levenberg–Marquardt algorithm for analyzing wire coatings in Sisko fluids, using comparable dataset sizes and epochs. Nonetheless, considering the constrained size of the present dataset, this result is interpreted as a high capacity for pattern fitting under controlled conditions, rather than a definitive validation of the model's full generalization capability to unknown or adulterated samples.

Model Deployment and Resource Utilization

The 'nntraintool' generated a mathematical representation of the trained model (via the 'deploy solution – > Matlab Function' option, which was subsequently adjusted to C/C++ language for use in the Arduino™ IDE and final implementation on the ESP32 microcontroller. This adjustment process included the syntactic definition of the weight vectors and the input/output normalization functions. Figures 6 and 7 present the most relevant excerpts of the code developed in Arduino™ IDE version 1.8.19, illustrating these adjustments.

```
/*************************Globals E-Nose*******************************************/
//double x_Input[7]={0.05,180.14,3459.37,0.14,632.2,3.95,0.01};
double x_Input[7];
double x_OffSet[]= {0,0.06,0.08,0,0.07,0.9,0};
double x_Ganancia[]={15.3846153846154,0.00225123818099955,0.000680081732222579,3.125,0.0005138
double x_Y_min= -1;
double a1[20];
double a2[7];
double y_OffSet[]= {0,0,0,0,0,0,0};
double y_Ganancia[]={2,2,2,2,2,2,2};
double y_Y_min= -1;

// Layer 1
double b1[]= {-1.6523867461356561,6.1138055577454509,2.5944933158635131,0.82639102310003743,2.4
double IW1_1[20][7]= {{0.75440637200062111,0.66256452398938537,1.5879691272379033,0.9512061550

//Layer 2
double b2[] = {0.79132620384296259,-0.97149375244564684,-0.098346155977758351,-0.94149342719753
double LW2_1[7][20] = {{1.3472180990584768,0.024523386294063212,-0.50440969930700508,0.03169865
```

Figure 6. Syntactic adjustment of weight vectors from Matlab Function to C/C++.

doi: 10.25100/iyc.v28i1.15038

```c
//****************************************************
//*********** TinyML_Model Liquor identifier ********
//****************************************************
void evaluate_TinyML__Model()
{
  x_Input[0]= coPPM;
  x_Input[1]= alcoholPPM;
  x_Input[2]= co2PPM;
  x_Input[3]= toluenPPM;
  x_Input[4]= nh4PPM;
  x_Input[5]= sensor_volt;
  x_Input[6]= R0;


//************** Data Input Normalized [-1,1] *******
  for(int i=0;i<7;i++)
  {
  x_Input[i]= x_Input[i]-x_OffSet[i];
  x_Input[i]= x_Input[i]*x_Ganancia[i];
  x_Input[i]= x_Input[i]+x_Y_min;
  }

//******** Layer 1 a1 = (b1+(IW1_1*Data_Input)) *********
  for(int i=0;i<20;i++)
  {
    double Acum=0.0;
    for(int j=0;j<7;j++)
    {
      Acum=IW1_1[i][j]*x_Input[j]+Acum;
    }
    a1[i]=Acum+b1[i];
    a1[i] = 2/(1+exp(-2*a1[i]))-1;
  }

//******** Layer 2 a2 = (b2+(LW2_1*a1)) *********
  for(int i=0;i<7;i++)
  {
    double Acum=0.0;
    for(int j=0;j<20;j++)
    {
      Acum=LW2_1[i][j]*a1[j]+Acum;
    }
    a2[i]= Acum+b2[i];
    a2[i]= a2[i]-y_Y_min;
    a2[i]= a2[i]/y_Ganancia[i];
    a2[i]= a2[i]+y_OffSet[i];     //**** Data Output Reverse Normalized [-1,1]
  }
```

Figure 7. Syntactic adjustment of the classification model from Matlab Function to C/C++.

The process of loading the model onto the ESP32 demonstrates efficient use of the embedded system's resources. The utilized Flash memory measured 297,525 representing 24.8% of the available Flash memory (1.2 MB for the application partition APP), which leaves sufficient margin for future expansions. Regarding RAM usage, it totaled 20,432 bytes, equivalent to 6.3% of the 327,680 bytes of available RAM, indicating a lightweight implementation with low impact on the microcontroller's resources. The programming latency was low, with 2.3 seconds required for writing, indicating a rapid and stable firmware upload, optimizing the model's execution in an embedded environment.

The estimated execution time of the algorithm on the ESP32 is approximately 2.48 ms. This calculation considers that the loaded code occupies 297,525 bytes in Flash memory and that each instruction, with an average size of 2 bytes, requires 2 clock cycles for execution. Operating at a frequency of 240 MHz, the microcontroller processes 595,050 cycles, which reflects an extremely low latency and efficient real-time execution suitable for embedded classification and pattern recognition applications.

The resources utilized in the deployment of the TinyML model on the ESP32 are summarized in Table 3. The achievable execution speed and the low resource usage indicate that TinyML does in fact provide a capability for advanced machine learning on inexpensive, resource-limited hardware; a major benefit for involving the real-time decisions that are often called for in food and beverage production.

doi: 10.25100/iyc.v28i1.15038

Table 3. Summary of resources utilized in the deployment of the TinyML model on the ESP32.

Resource Type	Used Value	Description
Flash Memory	297525 bytes (**24.8%**)	Space occupied by the code in Flash memory.
RAM Memory	20432 bytes (**6.3%**)	RAM used to execute the model.
Programming Latency	**2.3 seconds**	Time it takes to load the code into the ESP32.
Execution Time	**2.48 ms**	Latency (Processing time).

System Validation and Performance

The liquor classification system validation was conducted in an environment ensuring stable conditions of temperature (22°C) and relative humidity (75%) to minimize external variations. Six types of liquors were selected for testing: Medicinal Ethyl Alcohol, Wine, Pisco, Cachaça, Tequila, and Whisky. During the testing phase, the embedded system captured real-time data, executed the classification model, and displayed the prediction on the Virtual Terminal of the Arduino™ IDE configured at a baud rate of 9600.

To evaluate system stability, 30 measurements were performed per sample, accumulating a total of 180 tests, allowing for the analysis of the repeatability and consistency of the predictions. The predictions were compared with the expected values, yielding an accuracy percentage that reflects the system's capacity to correctly identify each liquor type. The results indicate an average accuracy of 84.4%, with individual values ranging between 80.0% and 86.7%, depending on the specific liquor analyzed. In total, out of the 180 samples evaluated, the system correctly classified 152, demonstrating solid performance in identifying the different beverages. The validation results of the TinyML classification device are shown in Table 4.

Table 4. Validation Results of the TinyML Classification Device

Liquor Type	Total Samples	Correctly Classified	Accuracy (%)	Score
Ethyl Alcohol	30	26	86.7	0.87
Wine	30	25	83.3	0.83
Pisco	30	26	86.7	0.86
Cachaça	30	25	83.3	0.82
Tequila	30	24	80.0	0.78
Whisky	30	26	86.7	0.88
Overall Average	**180**	**152**	**84.4**	**0.84**

doi: 10.25100/iyc.v28i1.15038

Cost-Effectiveness and Comparative Analysis

Finally, a system is considered ultra-low-cost when it prioritizes cost reduction without compromising functionality or the required precision. Key criteria include the total cost of hardware, commercial availability of components, low energy consumption, ease of implementation, and scalability. According to Ciuffoletti (33), low-cost IoT systems achieve a balance between performance, simplicity, and economic sustainability, making them appropriate for educational, commercial, or resource-limited environments.

In this context, the developed system meets these criteria by employing accessible hardware—ESP32 (\$5–10 USD) and MQ-135 sensor (≈ \$5 USD)—reaching a total cost of approximately \$15 USD. This cost is significantly lower than that of commercial equipment such as the Gasboard-3210Plus from Cubic Instruments (34) or the analysis system from Anton Paar (35), which often exceed \$800 USD.

Figure 8 demonstrates the physical application of the developed TinyML based liquor classification device as both compact and practical. Table 5 outlines a direct comparison of estimated costs considering the proposed TinyML system and the commercial systems discussed, demonstrating the vast cost effectiveness of the developed prototype, but still showcasing a reasonable degree of accuracy for its intended use.

The TinyML device, with this cost-effectiveness and established accuracy of classification, can be thought of as a realistic, democratizing technology for beverage authentication and quality control in markets or applications focused on cost restriction. The performance/price balance is appealing for further adoption in industries that may not have access to elite laboratory equipment.

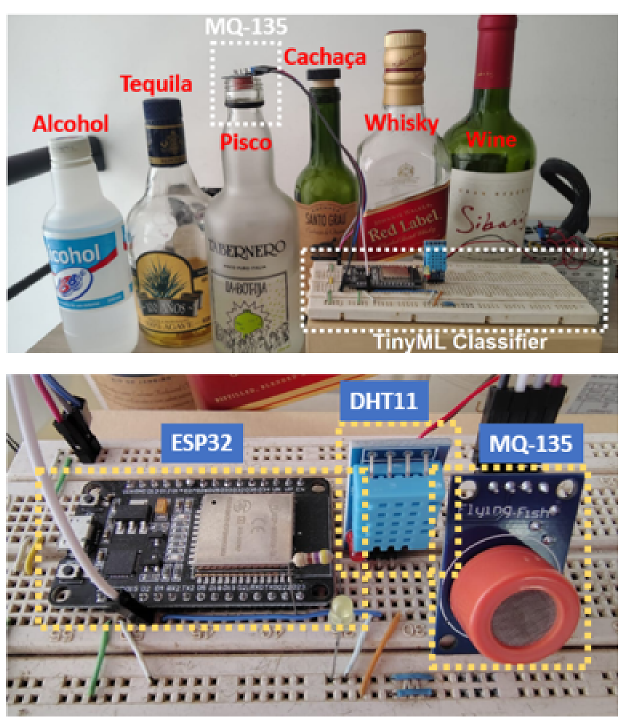


Figure 8. TinyML-based liquor classification device.

doi: 10.25100/iyc.v28i1.15038

Table 5. Cost Comparison Between the TinyML-Based Classification System and Some Commercial Solutions.

Characteristic	TinyML Classifier	Gasboard-3210Plus	Anton Paar Vine Meter	Anton Paar Alcohol Meter
Estimated Cost	15 USD	800 USD	2100 USD	1500 USD
Main Hardware	ESP32 Microcontroller + MQ135 Sensor	Gas Spectrometry Sensors	Electrochemical Sensors	Electrochemical Sensors
Required Software	Code in C/C++ (Arduino™ IDE)	Proprietary Software	Proprietary Software	Proprietary Software

Conclusions

The TinyML-based liquor classification system developed in this study is demonstrated to be an efficient and cost-effective solution when benchmarked against commercial devices. With an estimated hardware cost of approximately $15 USD, the developed prototype significantly exceeds the accessibility of costly alternatives, such as the Gasboard-3210Plus ($800 USD) or Anton Paar analyzers ($1500–$2100 USD). Furthermore, the rapid inference time of 2.48 ms and minimal memory footprint (only 24.8% of available Flash memory utilized) substantiate the system's viability for deployment in resource-constrained embedded systems without compromising performance.

In terms of predictive accuracy, the classifier achieved an average classification rate of 84.4%. The accuracy varied depending on the specific liquor type, reaching up to 86.7% for the identification of medicinal ethyl alcohol, pisco, and whisky. Although these results are promising for a system relying on simple hardware, there remains scope for performance enhancement through the employment of more robust machine learning models or the integration of additional sensors capable of capturing a wider range of chemical characteristics. Nevertheless, the performance level attained is competitive considering the simplicity and low cost of the utilized hardware platform.

This investigation establishes a foundation for future research concerning the application of TinyML for substance detection and classification. Model optimization, the use of advanced preprocessing techniques, and the adaptation of the system to other types of liquids or gases could substantially broaden its applicability in industries such as food processing and quality control. The synergistic combination of low cost, ease of implementation, and fast inference times effectively positions this technology as a viable and democratizing alternative in environments with strict budgetary or resource limitations.

Credit authorship contribution statement

Conceptualization – Ideas: Ilber Ruge, Ingrid Ortiz, Fabián Jiménez; Data Curation: Ingrid Ortiz; Formal Analysis: Ilber Ruge, Fabián Jiménez; Research: Ilber Ruge, Ingrid Ortiz, Fabián Jiménez;

doi: 10.25100/iyc.v28i1.15038

Methodology: Ilber Ruge; Supervision: Ingrid Ortiz; Validation: Ilber Ruge, Ingrid Ortiz; Writing – Preparation: Fabián Jiménez; Writing - Revision and Editing: Ilber Ruge; Preparation: Ilber Ruge, Ingrid Ortiz, Fabián Jiménez.

Ethical implications: The authors have no ethical implications that should be declared in the writing and publication of this article.

Conflicts of Interest: The authors do not declare.

References

1. Saraan A, Hussain N, Zahara SM. Tiny Machine Learning (TinyML) Systems [Internet]. 2024 [cited in mar 2025]. Available in: https://www.researchgate.net/publication/386579238_Tiny_Machine_Learning_TinyML_Systems

2. Murshed MGS, Murphy C, Hou D, Khan N, Ananthanarayanan G, Hussain F. Machine learning at the network edge: A survey. ACM Comput Surv [Internet]. 2021 [cited in mar 2025]; 54(8):Article 170, 1-37. https://doi.org/10.1145/3469029

3. Schizas N, et al. TinyML for ultra-low power AI and large scale IoT deployments: A systematic review. 2022. https://doi.org/10.3390/fi14120363

4. Tsoukas V, Gkogkidis A, Boumpa E, Papafotikas S, Kakarountas A. A gas leakage detection device based on the technology of TinyML. En: 2023 12th International Conference on Modern Circuits and Systems Technologies (MOCAST); 2023. p. 1-4. IEEE. https://doi.org/10.1109/MOCAST54814.2022.9837510

5. Bagheri M, Farshforoush N, Bagheri K, Shemirani AI. Application of TinyML in water environments. Environ Monit Assess. 2023;195(5):1-14.

6. Atanane O, Mourhir A, Benamar N, Zennaro M. Smart buildings water leakage detection using TinyML. Sensors. 2023;23(9):4473. https://doi.org/10.3390/s23229210

7. Zhou Q, Zhang S, Li Y, Xie C, Li H, Ding X. A Chinese liquor classification method based on liquid evaporation and a metal oxide gas sensor. Sens Actuators B Chem. 2011;158(1):241-6. https://doi.org/10.1016/j.snb.2011.08.015

8. Zhang Q, Xie C, Zhang S, Wang A, Zhu B, Wang L, et al. Identification and pattern recognition analysis of Chinese liquors by doped nano ZnO gas sensor array. Sens Actuators B Chem. 2005;110(2):370-6. https://doi.org/10.1016/j.snb.2005.02.017

9. Scorsone E, Pisanelli AM, Persaud KC. Development of an electronic nose for fire detection. Sens Actuators B Chem. 2006;116(1-2):55-61. https://doi.org/10.1016/j.snb.2005.12.059

10. Zhang Q, Zhang S, Xie C, Zeng D, Fan C, Li D. Characterization of Chinese vinegars by electronic nose. Sens Actuators B Chem. 2006;120(2):694-9. https://doi.org/10.1016/j.snb.2006.01.007

11. Wongchoosuk C, Wisitsoraat A, Tuantranont A, Kerdcharoen T. Portable electronic nose based on carbon nanotube-SnO_2 gas sensors and its application for detection of methanol contamination in whiskeys. Sens Actuators B Chem. 2010;147(2):392-9. https://doi.org/10.1016/j.snb.2010.03.072

12. Vega-Luna JI, Lagos-Acosta MA, Salgado-Guzmán G. Monitoreo de concentración de monóxido de carbono usando tecnología long-range. In: Congreso Internacional sobre Innovación y Desarrollo Tecnológico (CIINDET); 2017. p. 1-6. IEEE. https://doi.org/10.17163/ings.n18.2017.09

doi: 10.25100/iyc.v28i1.15038

13. Nagy AS, Polanco Risquet A, Martínez de la Cotera OL, Carralero Ibargollen O. Medición simultánea de gases con sensores MQ. RIELAC Revista de Ingeniería Electrónica, Automática y Comunicaciones. 2020;41(1):34-43. Disponible en: http://scielo.sld.cu/pdf/eac/v41n1/1815-5928-eac-41-01-34.pdf

14. Jiang X, Liu X, Wu Y, Yang D. White wine quality prediction and analysis with machine learning techniques. Highlights in Science, Engineering and Technology. 2023;39(1):321-326. doi: https://doi.org/10.54097/hset.v39i.6548

15. Botero-Valencia JS, Barrantes-Toro C, Márquez-Viloria D, Pearce JM. Low-cost air, noise, and light pollution measuring station with wireless communication and TinyML. HardwareX. 2023;14:e00422. https://doi.org/10.1016/j.ohx.2023.e00477

16. Hayajneh AM, et al. Tiny Machine Learning on the edge: A framework for transfer learning empowered UAV-assisted smart farming. 2023. https://doi.org/10.1049/smc2.12072

17. Karras A, et al. TinyML algorithms for big data management in large-scale IoT systems. 2024. https://doi.org/10.3390/fi16020042

18. Dutta L, Bharali S. TinyML meets IoT: A comprehensive survey. 2021. https://doi.org/10.1016/j.iot.2021.100461

19. Srinivasagan R, Mohammed M, Alzahrani A. TinyML-Sensor for shelf life estimation of fresh date fruits. Sensors. 2023;23(16):7081 https://doi.org/10.3390/s23167081

20. Capogrosso L, Cunico F, Cheng DS, Fummi F, Cristani M. A machine learning-oriented survey on Tiny Machine Learning. IEEE Access. 2024;12:3365349. https://doi.org/10.1109/ACCESS.2024.3365349

21. Banbury CR, Zhou W, Fedorov I, Matas R, Thakker U, Gope D, et al. Benchmarking TinyML systems: Challenges and direction. arXiv preprint arXiv:2003.04821. 2020. Available in: https://arxiv.org/abs/2003.04821

22. Hymel S, Banbury CR, Montemerlo R, Reddi VJ. Edge Impulse: An MLOps platform for Tiny Machine Learning. arXiv preprint arXiv:2212.03332. 2023. Available in: https://arxiv.org/abs/2212.03332

23. Lê MT, Bouzid Y, Miloud B, Hajjaji MA, Bensrhair A. Efficient neural networks for Tiny Machine Learning: A comprehensive review. arXiv preprint arXiv:2311.11883. 2023. Available in: https://arxiv.org/abs/2311.11883

24. Ray PP. A review on TinyML: State-of-the-art and prospects. J King Saud Univ Comput Inf Sci. 2022;34(4):1595-623. https://doi.org/10.1016/j.jksuci.2021.11.019

25. Sze V, Chen Y, Yang T. Efficient processing of deep neural networks: A tutorial and survey. Proc IEEE. 2017;105(12):2295-329. https://doi.org/10.1109/JPROC.2017.2761740

26. Henan Hanwei Electronics Co. Ltd. Technical Data MQ-135 Gas Sensor. Hanwei Electronics. 2013. Available in: https://www.electronicoscaldas.com/datasheet/MQ-135_Hanwei.pdf

27. Biglari A, Tang W. A review of embedded machine learning based on hardware, application, and sensing scheme. Sensors. 2023;23(4):2131. https://doi.org/10.3390/s23042131

28. MathWorks. nntraintool Neural network training tool - MATLAB. 2025 Feb 19 [cited in Mar 2025]. Available in: https://www.mathworks.com/help/deeplearning/ref/nntraintool.html

29. Bagheri M, Farshforoush N, Bagheri K, Shemirani AI. Applications of artificial intelligence technologies in water environments: From basic techniques to novel tiny machine learning systems. Process safety and environmental Protection. 2023;180(1):10-22. doi: https://doi.org/10.1016/j.psep.2023.09.072

doi: 10.25100/iyc.v28i1.15038

30. Espressif Systems. ESP32 series datasheet. Espressif Systems. 2025 Feb 19 [cited in Mar 2025]. Available in: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf

31. Lourakis MIA. A Brief Description of the Levenberg-Marquardt Algorithm Implemented by levmar. Institute of Computer Science, Foundation for Research and Technology - Hellas (FORTH); Heraklion, Greece; 2005. Disponible en: https://users.ics.forth.gr/lourakis/levmar/levmar.pdf

32. Aljohani JL, Alaidaorous ES, Raja MAZ, Alhothuali MS, Shoaib M. Backpropagation of Levenberg-Marquardt artificial neural networks for wire coating analysis in the bath of Sisko fluid. Ain Shams Engineering Journal. 2021; 12(4): 4133-4143. doi: https://doi.org/10.1016/j.asej.2021.03.007

33. Ciuffoletti A. Low-Cost IoT: A Holistic Approach. Journal of Sensor and Actuator Networks. 2018; 7: 19. https://doi.org/10.3390/jsan7020019

34. Cubic Instruments. Portable Headspace Analyzer (MAP) Gasboard-3210Plus. 2025 Mar 5 [citado 2025 Mar 5]. Available in: https://www.gasanalyzers.com/ProcessGasAnalyzer/info190

35. Anton Paar. Alcohol and Extract Meter for the Beverage Industry. 2025 Mar 5 [cited in Mar 2025]. Available in: https://www.anton-paar.com

doi: 10.25100/iyc.v28i1.15038