Gas and Smell Detection System using ML Algorithm

Dhir Gandhi [21BEI14],
Het Patel [21BEI034]
Dept. of Electronics & Instrumentation Engineering
Nirma University
Under the guidance of Dr. Vijay Savani

Abstract - To differentiate and recognize beverages like Tea, Coffee, we offer an electronic nose application case study in this paper that is based on an inexpensive Bosch BME688 AI (artificial intelligence) capable sensor module development kit. The module works with the Bosch BME AI Studio software and has eight BME688 sensors. We developed and tested a variety of machine learning (ML) models to differentiate between Coffee, Tea, and Air. The purpose of the case study is to ascertain whether the module can differentiate between beverages. It was found that the differentiation between air and different samples is possible.

Keywords: Machine learning, freshness detection, gas detection, smell detection, machine learning for scent, electronic nose, odour identification

1. Introduction

The use cases and research opportunities made possible by modern sensor technology were before unattainable and just partially applicable. The Internet of Things (IoT) and Edge Computing trends of today make it possible to find novel solutions to a wide range of issues in both science and industry. This holds true for human perception-based sensing, healthcare, and human-machine interactions.
The trends align nicely with the use of AI, which can identify and forecast factors based on data that is supplied to it through sensors. Accessible, low-power embedded systems with AI-based platforms are made possible by the TinyML application paradigm. In the past few decades, there has been extensive study on digital, electronic, or artificial noses, with compactness frequently playing a significant role.
The constraints of the sensor can provide a variety of hurdles, including issues with sensitivity, selectivity, stability, repeatability (calibration), and noise in gas detection applications. Delivery of gas can also be crucial. Depending on how it is approached, it can be either passive or proactive. As demonstrated by Brandt et al. and Bosch in their demonstrations, the passive delivery can take the shape of a sample holder built out of a glass chamber. The active method, sometimes referred to as "sniffing," involves exhale-inhale sequencing. Sniffing can be managed using varied tactics (deep/short sniffs) to adjust to diverse circumstances. Either way, both qualities can be relevant in specific use scenarios.
Electronic nose technology powered by machine learning (ML) could advance robotics, food engineering, environmental monitoring, and medical diagnostics. Electronic olfaction can be usefully facilitated by ML-based AI applications for a variety of illness diagnosis scenarios. They typically provide a more significant edge in intricate scent discrimination. An AI-powered e-nose system can be used to emphasize the fragrant

differences between various coffee beans and tea scents. Additionally, the idea can be expanded to explore the freshness of food (such as meat) using colorimetric barcode combinatorics.
To present a simple application that expands on previously published results, we present in this paper a compact, accessible BME688 devkit-based e-nose application where the approach of detecting and identifying beverages is in focus.

1.1) Background:
With the use of sophisticated technology, machine olfaction detects and classifies odorous things by discerning distinctions based on descriptors. Electronic nostrils that replicate biological noses in their extraction of descriptors from objects use arrays of gas sensors and odour identification algorithms. While understanding the psychological aspects of how people perceive scent has long been a crucial issue in olfactory research, researchers have dedicated their efforts to developing some thorough guidelines for the characteristics of how odour quality is measured and predicted. Additionally, odour categorization has already been included in a lot of industrial jobs. For example, meat quality can be checked by quick detection methods, which are necessary for proper product management. There are two basic categories into which odour categorization methods can be divided: linear and non-linear. Principal component analysis (PCA), linear discriminant analysis (LDA), and other linear techniques were widely utilized for categorization or identification of odours. Still, these approaches typically fall short of satisfactory results because of the model's strong non-linearity. Techniques that are not linear, like artificial neural networks (ANN) and Convolutional neural networks (CNNs) have advanced significantly in the last several years. However, the Neural network applications or demonstrations of odour classification were insufficient to establish a comprehensive assessment of the work. In contrast to linear techniques, which restrict the quantity of training data, non-linear techniques often need a sizable amount of data for both training and validation, which significantly raises the challenge when experimenting.

1.2) Related Work:
In recent years, odour identification and description have gained a lot of respect. Nevertheless, rather than categorizing materials made up of several odorous molecule kinds, most studies focus on modelling quantitative structure onto a particular odour. Relevant research mainly examined machine olfaction, the quantitative structure-odour link, and simple labelling for smells. A distinct approach is used in sensor-based machine olfaction with a Neurodynamics model of the olfactory bulb. Once signal processing is complete, odour patterns are produced with recognition in mind. The patterns might show how concentrated the smells are. A miniature electronic nose system was created by Wang et al. to evaluate the freshness of food in refrigerators. A MOS gas sensor array and a gas sampling module make up the system the model was constructed based on the sensor array findings and a comparison with the outcomes of a human sensory evaluation. An Odor Labelling Convolutional Encoder-Decoder (OLCE) for odour has been proposed recently by Wen et al. Olfaction in Machine Sensing. The study's scents from seven Chinese herbal medications that were not crushed were gathered with the electronic PEN-3 nose. The OLCE model was developed using self-collecting gas response data for training and testing. However, it wasn't broadly applicable because of the limitations on sample sizes and types.

1.3) Problem Description:
The identification of smelly food using a classification algorithm, whose inputs are measured by the Bosch BME688 electronic nose, serves as the basis for our study. In order to optimize the utility, we tried to

identify the type of food as well as its freshness. If effectively finished, this novel accomplishment could be used in practical situations in the workplace, such the construction of an intelligent refrigerator. Customers would benefit from the state-of-the-art features when sorting through bad food or assessing the rancidity of the meal.

2. Materials and Experiment Setup

2.1) The sensor setup (hardware + software) used for measurements:

For the experiment, we used the official Bosch BME688 devkit, which consists of 8 BME688 sensors and an Adafruit HUZZAH32 ESP32 Feather microcontroller.

BME688 is a gas sensor with AI capabilities (Figure 1 presents the sensor). It has a compact package of 3x3x0.9mm size, developed for applications requiring compact size. The sensor is a 4-in-1 package, where volatile organic compounds (VOCs), volatile sulfur compounds (VSCs), carbon monoxide and hydrogen in PPB (parts per billion) range can be measured.

Figure 1. BME688 sensor

The HUZZAH32 is an ESP32-based board to enable wireless and microprocessing capabilities. The chip has a dual-core 32-bit Tensilica LX6 processor, 520 KB SRAM and 4 MB flash memory. This is the recommended microcontroller for the BME688 development kit. The development kit itself is a pre-programmed system for gas sensor development. The statistical analysis capabilities are improved with the eight-sensor configuration. The shuttle board has two further buttons and a real-time clock (RTC) battery. Figure 2 presents the application package: devkit shuttle board + HUZZAH32 board and the assembled sensor matrix on top.

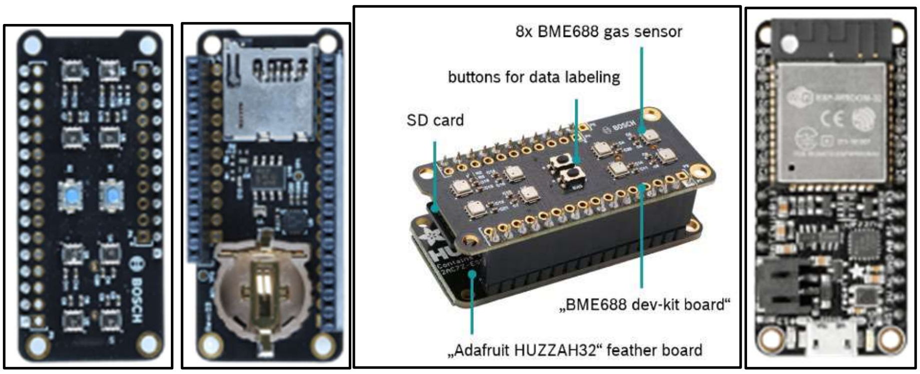


Figure 2. BME688 dev kit

From the software side, Bosch BME AI Studio was used. We wanted to use the gas sensor's capabilities to distinguish between different beverages based on their gas profiles. By using ML methods with the BME688 sensor, it is possible to create and recognize unique gas profiles, so its devkit should be able to perform such a task. The first step was to run the devkit for a full day to stabilize the 8 BME688 sensors. It is essential to create a configuration file in the software to be uploaded to the dev kit device. In doing so, three sets of settings can be changed. During the configuration file creation, you can set the method and how the sensors collect data during the learning phase. The three types of settings are the different heating profiles (HP, heater profile), the active cycle time or recording duty cycle (RDC) settings, and the possibility of dividing these between the eight sensors (board layout). According to the standard configuration file recommended by Bosch Sensortec at the start of the project, all eight sensors are simultaneously configured to collect data based on the HP-354 heating profile and the RDC-1 cycle time. After selecting the appropriate settings, the software generates a file, which must be copied to the dev kit memory card before starting data collection.

Once the data has been appropriately labelled and sorted, ML algorithms can be created using the "New Algorithm" menu. Here, classes can be created from the labelled collected data. You can select which data channels (humidity, gas resistance, temperature, barometric pressure) the algorithm should consider and how many measurement points the software should handle simultaneously. This can be set in the "batch size", between 4 and 64. This affects the speed of learning. It can be said that the larger the batch size, the slower the learning process. The only optimization method available so far is the "ADAM optimizer" type. ADAM was introduced as a method in 2014 at the ICLR 2015 conference. Its name comes from the English term "adaptive moment estimation". The optimization algorithm is an alternative to the SGD, or "stochastic gradient descent" method. ADAM optimization works by adapting the first and second-moment estimates of the gradient. It is also possible to set the total percentage of the acquired data spent on training and validation, and the number of training rounds. The standard settings recommended by Bosch Sensortec are a "batch size" of 32, 2048 training rounds and a data split of 70 % training and 30 % validation data. The software will indicate that there is a balance in the amount of data recorded for the classes required for training, which will be measured on time. Then the "Train Neural Net" option starts the training, during which a graph helps to keep track of its status, i.e. how long the training is taking, how much more time it may take, how many heating cycles have been trained out of the maximum of 4 and how many training rounds out of 2048 have been completed. The accuracy and validation accuracy values are expected to be successful if they approach a value of 1 during the training (starting from 0). In contrast, the values of loss and validation loss are expected to approach 0. The software exports two files from the results obtained, which are copied to the memory card of the dev kit to test the algorithm operation. The so-called validation matrix shown in the results shows how often the algorithm has failed out of all the trials. The higher the number of hits at a given position, the darker the colour of the square in the matrix. This example algorithm distinguishes between air and white wine, with 100% accuracy according to the software, since it did not make any errors according to the "Confusion Matrix" shown in Figure 3. The Confusion Matrix is a detailed performance analysis for a classification algorithm.

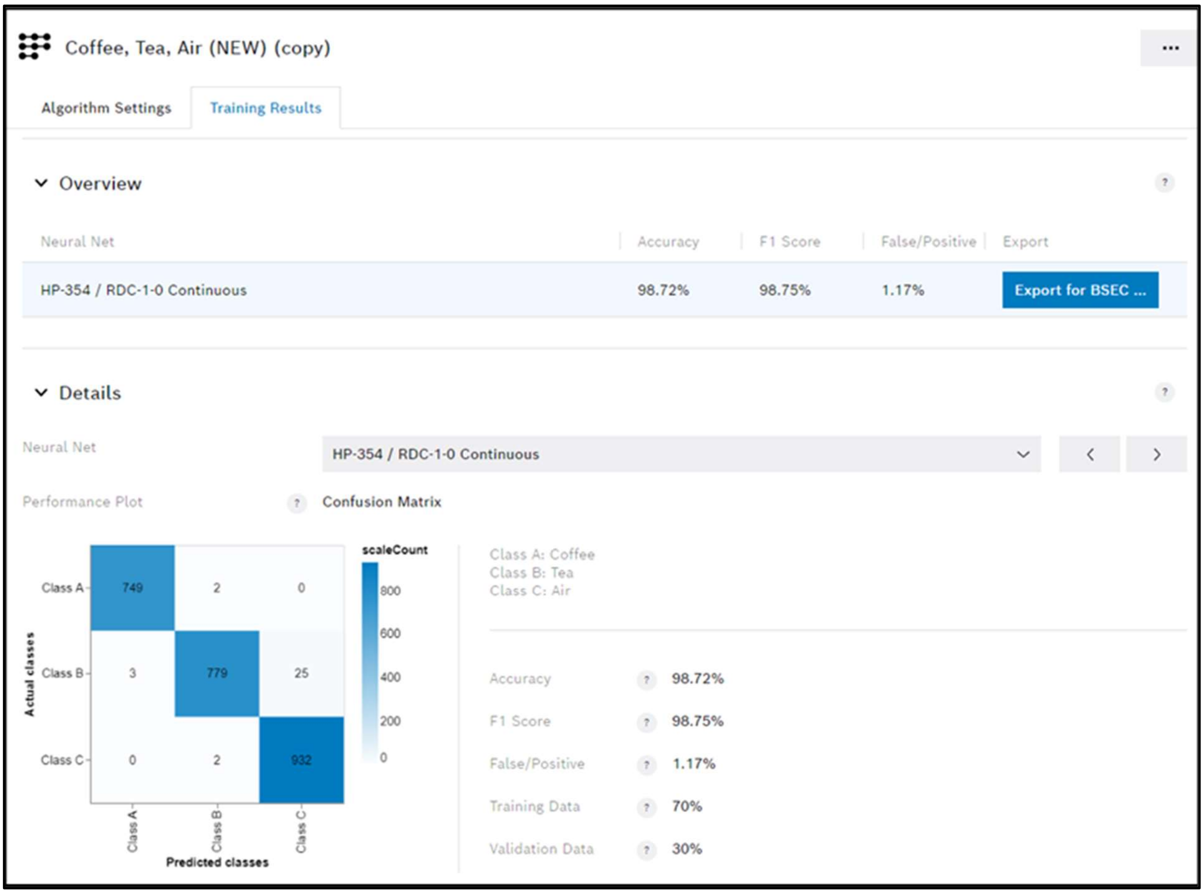


Fig. 3. Accuracy, F1 Score, False/Positive, Training/Validation Data according to the software

The results shown in the figure can be read from the confusion matrix, "Accuracy", "F1 Score" and "False/Positive". Here, the "Training Data" and "Validation Data" values are also present, i.e. what percentage of the total data set is used for training and validation. Accuracy is a measure of the performance of the classification algorithm. It expresses the probability that the algorithm gives a correct definition by giving the ratio between correct and incorrect Definitions. The F1 value also indicates how well the correct classification or definition of the validation dataset works. A false positive value indicates how many incorrect predictions the algorithm makes. The lower its value, the better the algorithm's performance.

2.2) Flow chart:

The process for detection should be a confluence of hardware for data collection and software for classification (Figure. 4), which consists of two parts. The first part requires you to place the gas sensor adjacent to the food or drink for detection, and the SD card embedded in the sensor will extract the corresponding features as input. In the second step, recorded data are imported from the SD card to the BME AI-Studio Desktop application, which will automatically execute the pre-trained model and then display the outcomes of predicted labels and freshness. This elaborately end-to-end designed system would be efficient, especially for freshness tests in routine cases.

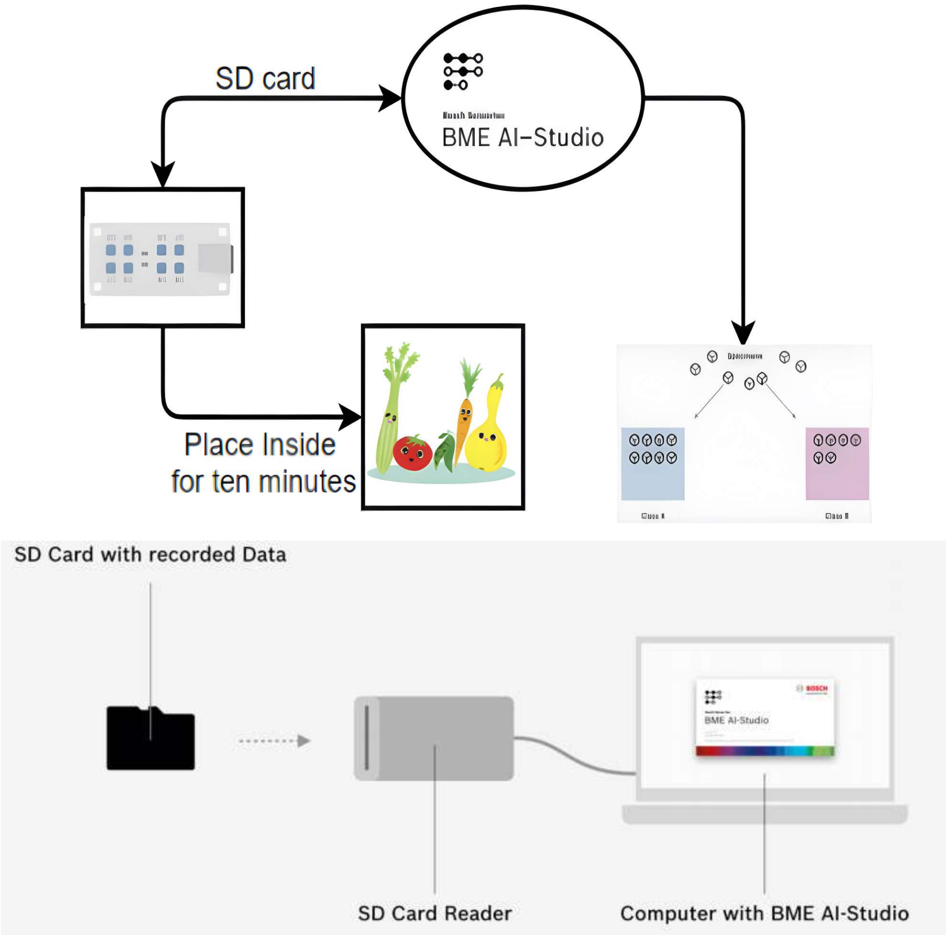


Figure 4. Process for specimen collection

2.3) Materials:

The tools used in the experiment include a processor (computer), a power bank, a USB cable, and a development kit that incorporated electronic noses and an SD card. The computer was installed with BME studio to extract raw data. As the selected electronic nose, the BME688 gas sensor can distinguish different gas compositions by measuring unique electric fingerprints and features high sensitivity and selectivity. This works on the principle that molecules going into the sensor area are charged either negatively or positively, which imposes on the electric field inside the sensor directly. The sensor provides straightforward customization for specific cases, such as detecting spoiled food and air quality. The sensor board had a heater profile, which will be run through by the sensor during a scanning cycle. A brief setting and the configuration of the sensor board are illustrated in the figure below.

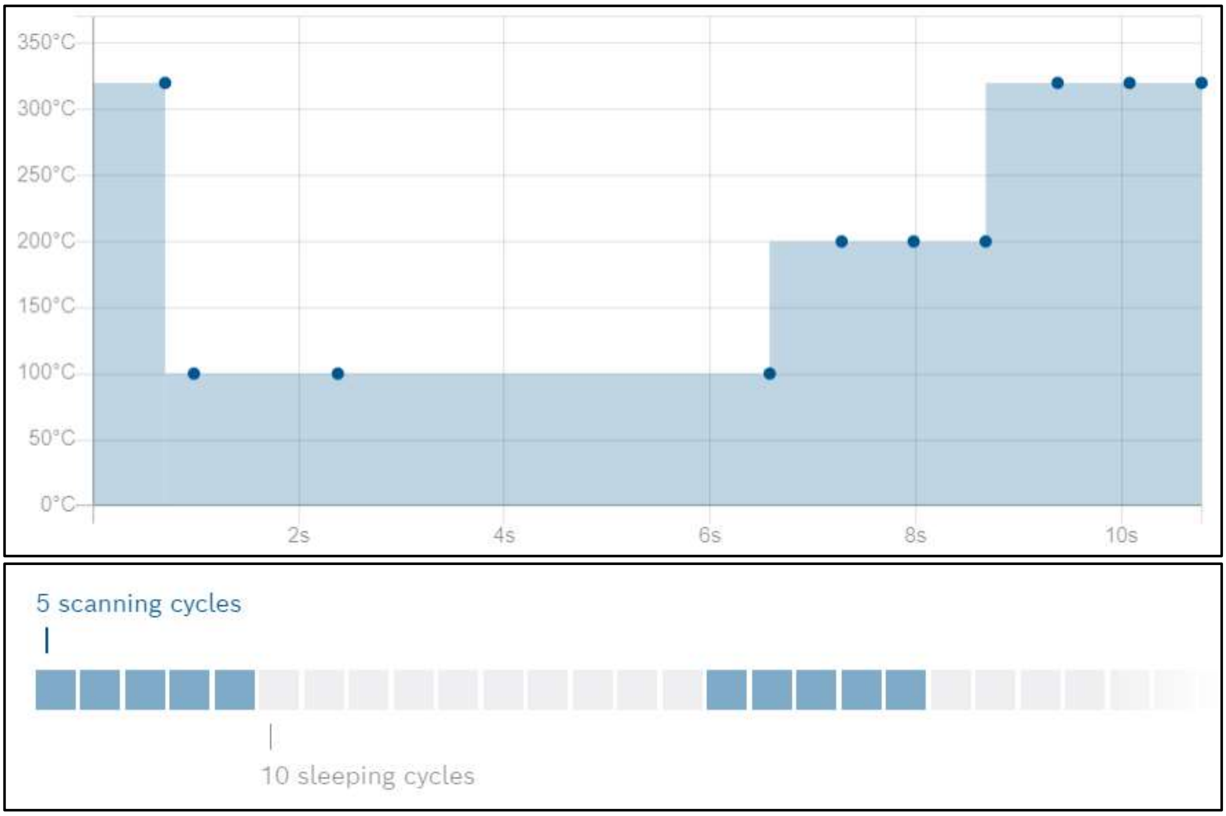


Figure 5. BME Board configuration (Up: Settings of the heater profile, down: Information of a duty cycle)

The SD card inside the dev kit starts working as soon as connected to the power. The scanning will be executed corresponding to the given heater profile and duty cycles that have been set up previously. The sensor will be heated according to the given heater profile during each scanning cycle. When the sensor is in sleep mode, it will not perform the measurement. In the experiment, we used the default setting of the board configuration, which sufficed for data training. In Figure 5, the default heater profile consists of ten measurement points, which will correspondingly generate ten sample points within a scanning cycle.

2.3) Environment:

We were devoted to simulating a scenario for practical use, so coffee was selected as our detection environment. Each time, we put one piece of the specimen in front of the sensor and waited for minutes to ensure that the odour volatilizes from the specimen. The sensor was then connected to a power bank and placed near the specimen. The time length was about 60 minutes for each detection.

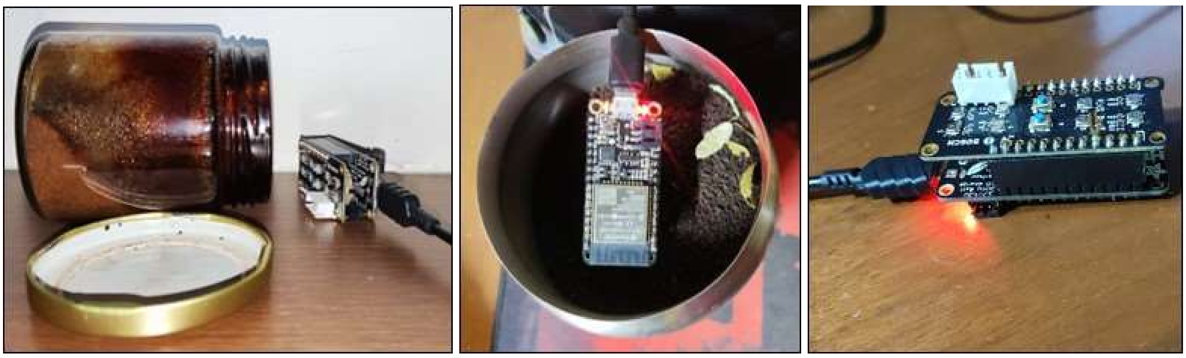
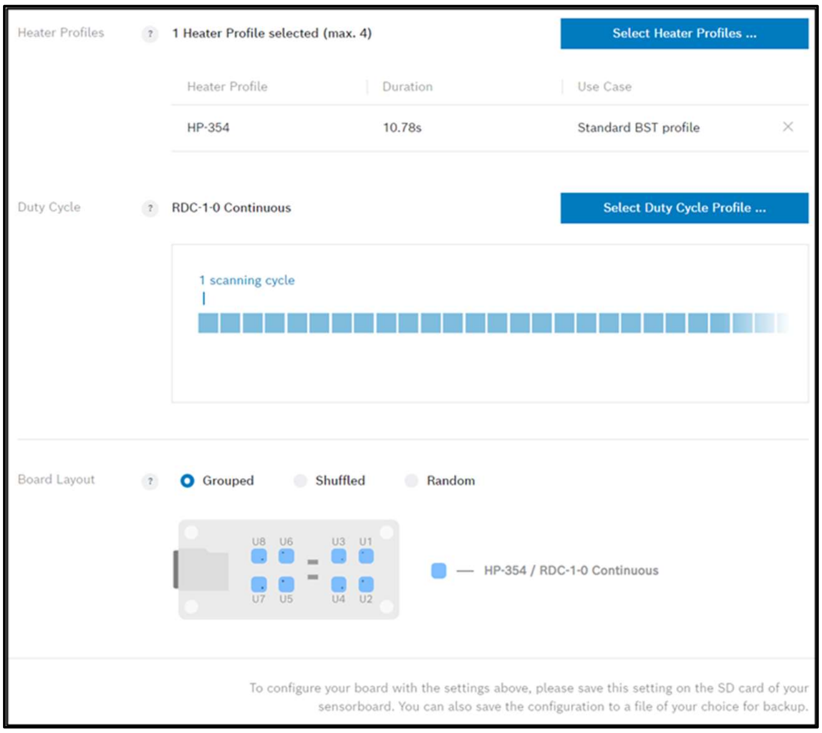


Figure 6. Environment for specimen detection (From left to right: Coffee, Tea, Air)

2.4) Board configuration:

The BME688 devkit, equipped with Bosch's environmental sensors, is first set up and configured. This involves flashing firmware onto the devkit's ESP32 microcontroller board and configuring the heater profile, duty cycle and board layout using the BME AI-Studio Desktop application. We have used the default heater profile HP-354 and RDC 1-0 continuous duty cycle. After selecting the parameters, the config file is saved on a microSD card.



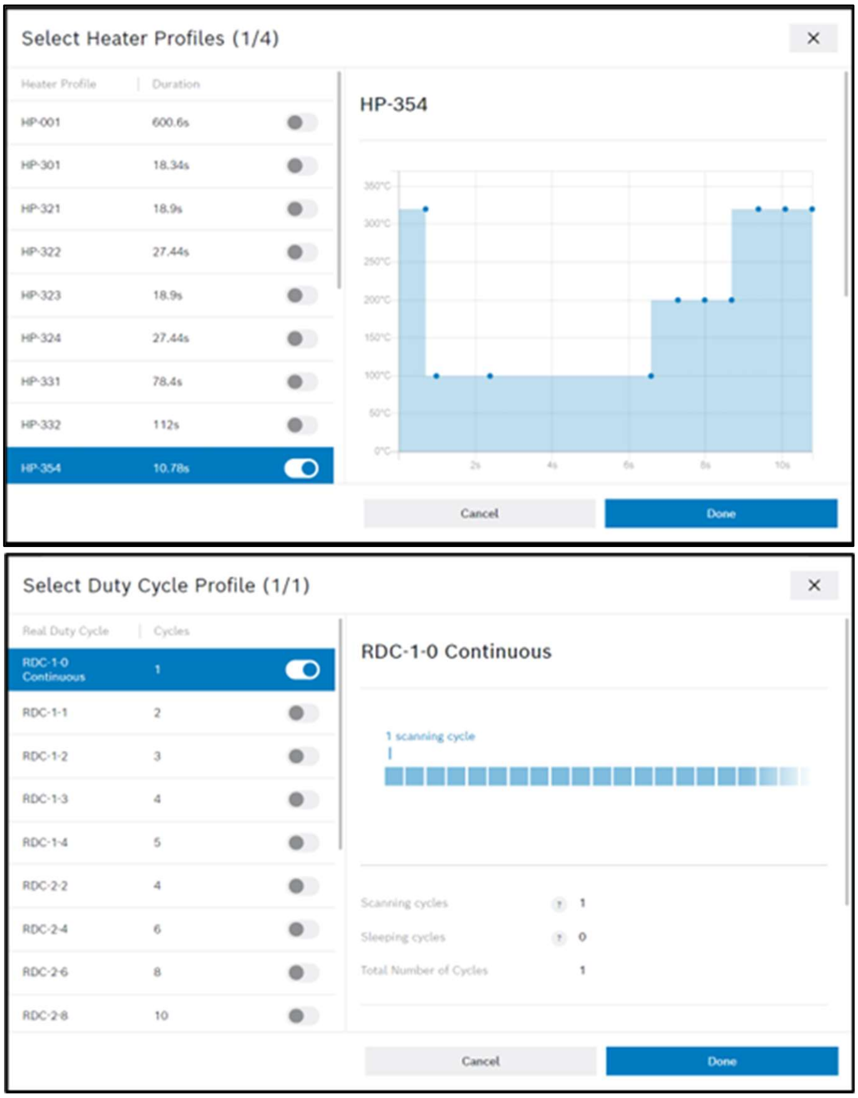


Figure 7. Board Configuration

3. Model Training

In the BME AI-Studio Desktop application, the environmental data from the sample collection phase is used to train machine learning models. The AI-Studio leverages this data to create predictive models or identify patterns, such as classifying air quality, detecting specific gas compositions, or learning environmental behaviour. Users can define what scenarios they want the model to focus on, adjusting the training process according to their use cases. AI-Studio's training algorithms apply deep learning techniques to make the model more accurate and adaptable to various environmental conditions.

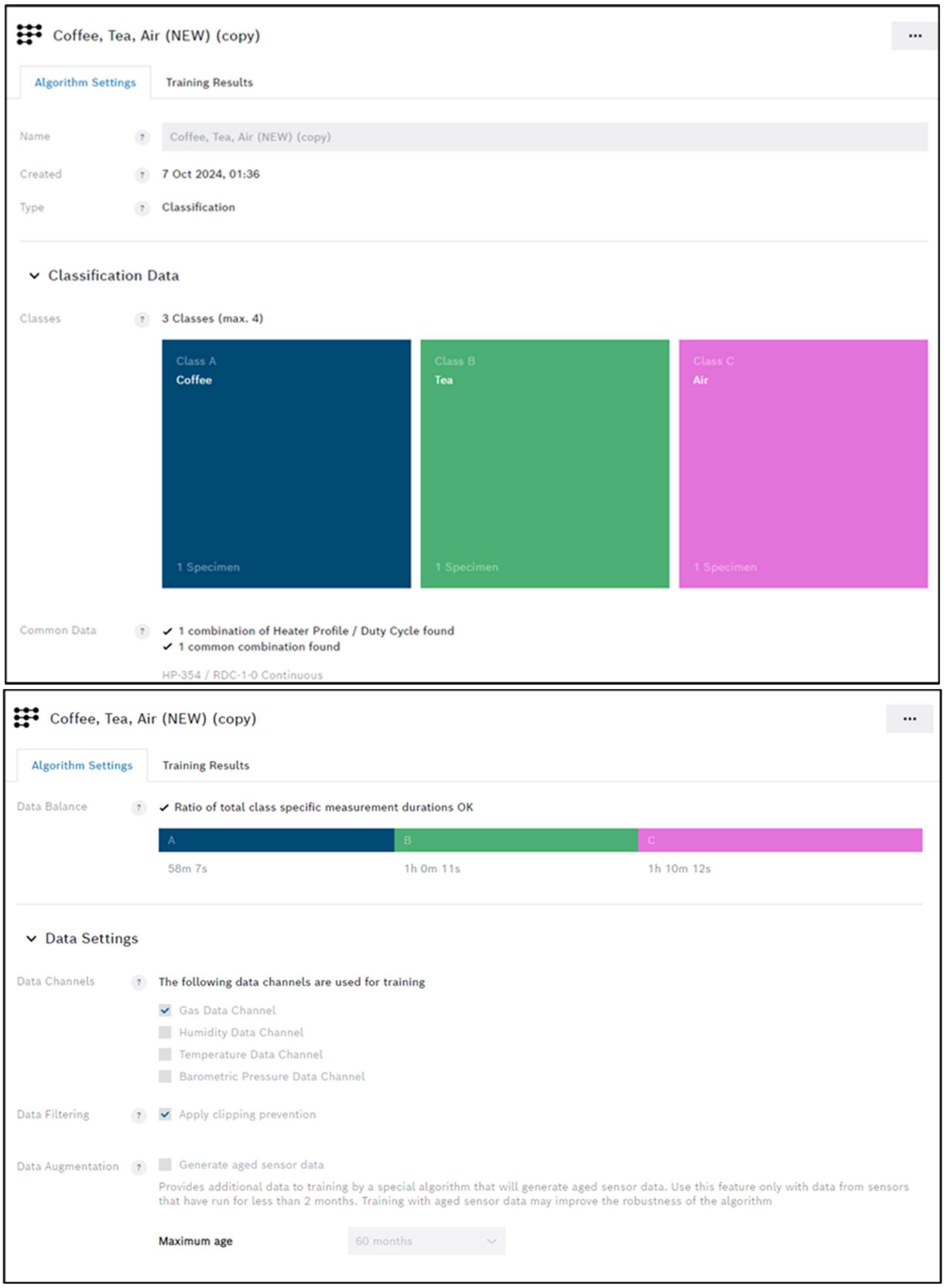


Figure 8. Data Classes, Algorithm settings, data settings

The application offers various options for selecting data and multiple parameters for model training. It also has features for clipping prevention and data augmentation. Classification and regression, both options are available. We can also select the batch size, epoch and data splitting. Once data training is completed, a confusion matrix, accuracy, F1 score and false/positive are generated. The trained model can now be exported and saved on the microSD card for use with the BME688 devkit. We achieved 86.66% accuracy, 86.73% F1 score and 13.22% false/positive. The confusion matrix can be seen in the figure below.

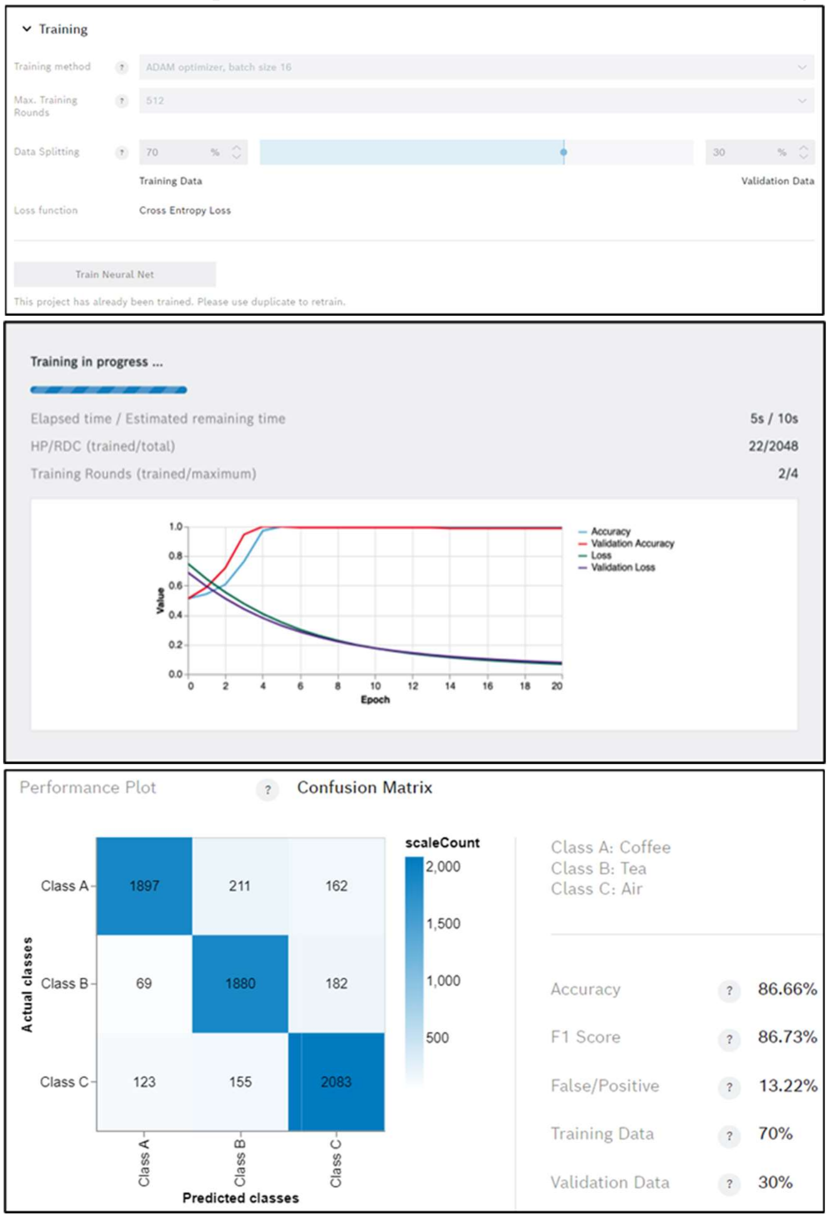


Fig. 9 Model training and results

4. Testing

After training, the model is tested in the real world. This testing phase ensures that the machine learning model is functioning as expected and can predict or classify environmental conditions based on sensor readings. The BME mobile app is used to test the model predictions. Testing helps fine-tune the system and check its accuracy in recognizing environmental patterns such as gas mixtures, pollution levels, or weather conditions.

If the model is not performing as expected, then more data can be collected, and the model can be trained again. It is also important to make sure that multiple data set is collected for the same specimen under different environmental conditions to ensure that the trained model is robust.
Below figure indicates the classification results. The current and predicted classes are displayed using the BME AI Studio mobile application. From the results, it can be concluded that the sensor is performing very well and can be used reliably for smell detection.

Figure 10. Classification results in BME AI-Studio mobile app

5. Conclusion

This study explored the application of the Bosch BME688 sensor module, leveraging AI and ML for odour classification and beverage differentiation. By employing machine learning techniques, we successfully created a low-cost, compact electronic nose system capable of distinguishing between various beverages like tea, coffee, and fresh air. The sensor's ability to detect volatile organic compounds (VOCs) was central to the success of this study, and the integration of Bosch BME AI Studio allowed for efficient model training and accurate prediction. Our work demonstrated the potential for AI-powered gas detection systems in real-world applications such as food freshness testing, air quality monitoring, and automated quality control in industries like food and beverage. While the results indicate high accuracy in beverage differentiation, challenges such as data collection in varied environmental conditions and sensor calibration were observed. Despite these, the system achieved its primary objective, proving the viability of using compact sensor arrays for intelligent olfactory systems.
This work highlights the importance of advancements in machine olfaction for both consumer and industrial applications. Future work could focus on improving the robustness of the model by training it on more

diverse datasets, exploring proactive delivery methods, and expanding the system to classify more complex gas mixtures, thereby enhancing its potential for broader use cases such as healthcare and environmental monitoring.

6. References

1. Brandt, R. et al. "An electronic nose for odour classification and diagnosis." Journal of Sensors and Actuators B: Chemical, vol. 240, pp. 213–219, 2017.

2. Bosch Sensortec. "BME688 Gas Sensor with AI Features." Bosch Sensortec Datasheet, 2022. Available: https://www.bosch-sensortec.com

3. Wang, Y. et al. "A Miniature Electronic Nose System for Real-Time Food Freshness Evaluation." Sensors and Actuators B: Chemical, vol. 308, pp. 127623, 2020.

4. Wen, H. et al. "Odor Labeling Convolutional Encoder-Decoder (OLCE) Model for Machine Olfaction in Scent Analysis." IEEE Sensors Journal, vol. 19, no. 13, pp. 5467-5478, 2019.

5. Kingma, D. P., and Ba, J. "Adam: A Method for Stochastic Optimization." Proceedings of the International Conference on Learning Representations (ICLR), 2015.

6. Bosch Sensortec. "BME AI-Studio Documentation." Bosch Sensortec, 2023. Available: https://www.bosch-sensortec.com

7. Gardner, J. W., and Bartlett, P. N. "Electronic Noses: Principles and Applications." Oxford University Press, 1999.

8. Tian, F. et al. "Machine Learning Approaches for Electronic Noses: A Review." IEEE Sensors Journal, vol. 21, no. 3, pp. 2471-2486, 2021.

9. Pawar, S. J., and Prabhu, K. "Food Quality Detection using Artificial Neural Networks and Machine Learning Techniques." International Journal of Advanced Research in Computer Science and Electronics Engineering, vol. 7, no. 6, pp. 45-50, 2022.

10. Bosch Sensortec. "BME688 Sensor: Application Programming Guide." Bosch Sensortec, 2023. Available: https://www.bosch-sensortec.com