

RespiraTech: Intelligent IoT and AI-Based Breath Analyzer for Early Disease Prediction in Healthcare

N. Mahamad Gouse, Kezia Mary, Rakshitha Kanchan, Vidyashree, Fayaz Ahamed Shaikh

Department of Computer Science and Engineering (IoT, Cybersecurity Including Blockchain), Alva's Institute of Engineering and Technology, Moodbidri, Karnataka, India

Published October 2026

Abstract

This paper describes RespiraTech, a prototype multimodal breath monitoring system based on ESP32, to be used as a platform for further research in healthcare screening. The platform consists of an SGP30 TVOC/eCO₂ sensor, an MQ-135 metal oxide sensor, a DHT22 temperature and humidity sensor, an INMP441 I²S microphone and a MAX30102 pulse oximeter combined with a Python/FastAPI backend, SQLite storage, a web dashboard, and PDF report generator. The contribution of our paper is of methodological rather than diagnostic nature. It includes a specification of capabilities and limitations of each channel, a provenance scheme distinguishing between hardware and synthesized recordings, validity criteria for saturated, zero, and missing channels, and a dataset preparation procedure including a train-test split by participants. We also present honest benchmark results: unstable behavior of SGP30 and MAX30102 sensors, a reading of the MQ-135 sensor at ADC full scale, and a problem with the ESP32 ADC driver interfering with integrated firmware operation.

Keywords: Breath analysis; Electronic nose; ESP32; Internet of Things; Multimodal sensing

1 Introduction

The exhaled breath consists of nitrogen, oxygen, carbon dioxide, and water vapor as well as thousands of volatile organic compounds (VOCs) that can be either produced by human metabolism or ingested from the environment [1]. The possibility to analyze breath in a simple, painless and repeatable way makes breath analysis a promising tool for diagnosing and monitoring respiratory and metabolic disorders [2, 3]. For accurate analysis of exhaled gases laboratory equipment like gas chromatography-mass spectrometry is required [4]. Such devices are expensive, not portable and require specially trained personnel.

Chemical sensor arrays, called electronic noses, represent an alternative technology allowing for analyzing breath through pattern recognition based on level detection, enables classification of sample groups [5, 6]. Electronic nose experiments have shown the classification of asthma patients from healthy people [7] as well as that of COPD from asthma patients [8]. Moreover, a low-cost instrument assembled from Arduino and MQ-sensors has been applied to detect the levels of ammonia, hydrogen sulfide, acetone and alcohol in 105 patients [9]. However, although the above achievements seem to be promising, they were carried out using a specific protocol and did not solve problems with practical application. Metal-oxide sensors show signal drift due to moisture sensitivity and simultaneous response to multiple gases [10], while composition of exhaled air can be affected by external factors such as diet and smoking [11].

One lesser-known problem is engineering reliability. A multisensor prototype implements analog, I²C, single-wire, and I²S interfaces to various peripherals on a single microcontroller. A saturated analog-to-digital converter channel, a bus peripheral that suddenly stops responding or a driver clash can generate readings that appear real. Such readings mixed into a dataset train models on data artefacts, not physiology.

This paper introduces RespiraTech, a multimodal breath-monitoring prototype running on an ESP32. Gas sensing (SGP30, MQ-135), environmental sensing (DHT22), respiratory sound recording (INM-P441) and optional pulse oximetry (MAX30102) form the basis along with the Python backend and web interface; machine learning is intended for the future. Our contribution is presented as the working prototype, with individual components tested on the bench but not yet integrated. The rest of the paper is structured as follows. The problem statement is presented first, followed by the objectives and contributions of the work and a review of related literature. The architectural design, hardware design, methodology and implementation are then described, followed by the experimental setup, results and discussion, limitations, ethics and safety, and conclusions.

2 Problem statement and motivation

The use of spirometry, imaging, and laboratory tests is still the way that respiratory and metabolic diseases are usually confirmed. However, these require equipment, trained personnel, and the presence at the clinic, making it difficult to screen early in resource-limited conditions. The development of a portable breath analysis device can make screening more accessible, provided that the data is reliable and the claims are reasonable.

We have identified three issues. P1: accessibility and cost due to the resource consumption involved in the traditional approach to diagnosis. P2: data reliability, due to the fact that low-cost sensors are non-selective, drift with age and humidity, and may be corrupted by wiring and driver failures in a multi-sensor build. P3: evaluation hygiene, due to the optimism of machine-learning studies using small cohorts because of the overlap between training and test set or corruption of the dataset by faulty or synthetic data points [12].

RespiraTech is driven by the need for a low-cost research platform that will solve P2 and P3 already at the design stage, ensuring that any data collected afterwards and any claim based on that data can be substantiated.

3 Objectives and contributions

The goals of the study are:

1. Design an inexpensive prototype for IoT breath monitoring with ESP32 microcontroller.
2. Investigate multimodal gas and environment sensing for breath-related measurements.
3. Explore recording respiratory audio for possible signal processing in the future.
4. Add supplementary pulse and SpO₂ measurements.
5. Derive communication protocol between hardware and software.
6. Create healthcare-oriented dashboard with history of tests and reports generation.
7. Design a systematic path to prepare datasets for future machine learning.
8. Perform evaluation of the integration reliability, sensor limitations and data quality.

The contributions of this work are the following.

- **C1.** Multimodal acquisition framework for breath-related studies where for each channel is specified what can be measured and what cannot be measured (see Table 2).
- **C2.** A data-provenance and quality-flag scheme that separates hardware records from synthetic demonstration records and defines validity rules for saturated, zero-valued and missing channels.
- **C3.** Recorded bench report on failed integrations (overloaded ADC, I²C communication errors, no optical signals, and driver incompatibilities), along with possible causes and methods to prevent them, reported as observations instead of being left out.
- **C4.** Data preparation and assessment methodology for future breath research requiring baseline adjustment, reference markers, informed consent and splitting of training and testing data per individual.

4 Related work and research gap

4.1 Breath analysis and electronic noses

The development of electronic-nose and related technologies is discussed in [5], while the pattern-analysis sequence for sensor arrays is discussed in [6]. In respiratory medicine, array-based instruments have distinguished between asthma patients and healthy individuals [7] and between COPD patients and asthmatic patients [8]. In the case of diabetes, breath acetone is the most researched biomarker, and there have been specific sensors developed for measuring acetone [13]. The general-purpose gas sensors like SGP30 or MQ-135 are not acetone-selective, and thus, any claims about acetone are unjustified for these channels. Table 1 provides an overview of the literature most relevant to this paper.

Table 1. Related work and relevance to this study.

Ref.	Work	Relevance
[5]	Review of electronic-nose applications	Sensor-array background
[6]	Review of pattern analysis for machine olfaction	Basis for feature and classifier choices
[7]	E-nose, asthma versus controls	Array-based discrimination in a controlled study
[8]	E-nose, COPD versus asthma	Disease-versus-disease discrimination
[13]	Acetone-selective sensor for diabetes	Shows acetone needs selective sensing
[9]	Low-cost MQ-based device, 105 subjects, lung-function-labelled	Closest low-cost precedent; ammonia and H ₂ S useful, acetone and alcohol weak
[11]	Endogenous versus exogenous confounders	Motivates ambient baseline recording
[10]	Gas-sensor drift compensation	Motivates calibration and drift handling
[14]	Lung auscultation fundamentals	Clinical basis of respiratory sounds
[15]	Open respiratory-sound database	Public data for audio methods

4.2 Audio, pulse oximetry, and IoT health care systems

Respiratory sounds have sound clinical foundations [14], and there are also publicly available databases for training audio classification algorithms [15]. The pulse oximeter measures the arterial

oxygen saturation based on absorption of red light and infrared [16]; however, the accuracy of the readings is influenced by blood flow, movement, and skin color [17], and so it is not considered here as a breath measurement. Internet-of-Things healthcare surveys highlight the sensing, gateway, cloud, and application layers along with challenges related to security and data quality [18].

4.3 Research gaps

There are three research gaps from among the literature that is reviewed above which would be relevant for building an individual's own research system. First, inexpensive equipment usually only uses one sensor and the use of combination sensors (gas, acoustic and physiological sensors), is infrequent in published literature in such a way that others can replicate it. Second, most papers focus on classification output while paying little attention to saturation, sensor failure or driver failures even though they influence the quality of data. Third, the distinction between real and simulated data in demo software..

5 Proposed system architecture

The overall architecture is a layered structure (Figure 1). The sensing layer has five sensing modules and optionally includes the display unit. An ESP32 MCU represents the edge layer, where sampling is carried out at the specific speed for each channel, timestamping and flagging are performed, and one record is output per acquisition cycle. In the development process, the record is transferred via the USB serial interface to the laptop, while the target networking protocol is WiFi. The backend in Python validates records, marks them with the mode, and saves them. The presentation layer includes the web dashboard and PDF reporting. Machine learning pipeline is indicated by dashed lines in Figure 1 since it is still planned.

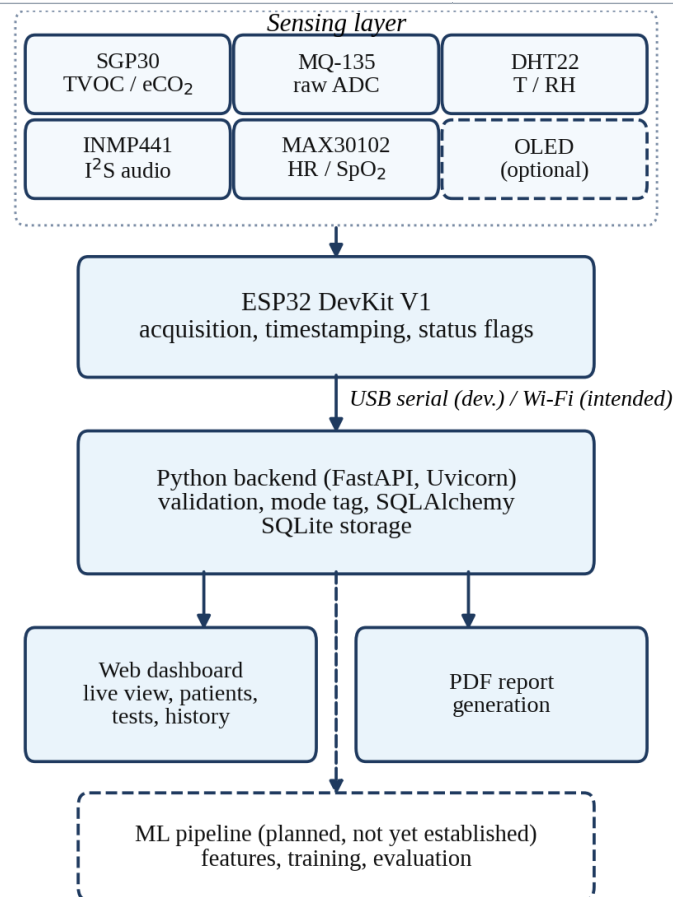


Figure 1. Layered architecture of RespiraTech. Dashed elements are optional or planned.

There are two design principles that affect the software. One principle states that hardware and synthetic data can be distinguished in any time: the backend reveals its mode of operation and, if there is no hardware connected, the dashboard will be in demo mode (synthetic data) until manual change. Another principle is that all channels should be considered untrusted until their validation succeeds (see Data provenance and quality gating). The status endpoint of the backend shows application name, operational state, mode of the prototype and connectivity of the sensor data.

6 Hardware and sensor specifications

6.1 Controller

Controller is ESP32 DevKit V1 with ESP32-D0WD-V3 as a core chip offering two cores processor, Wi-Fi communication, and a 12-bit successive approximation ADC and I²S devices [19]. The ADC2 pins cannot be used when the Wi-Fi module is operating. Hence, analog output of MQ-135 sensor is connected to the pin 34, which is an ADC1 input-only pin.

6.2 Sensors

The sensors, their interface and main limitations are presented in Table 2 below.

Table 2. Sensors, outputs and interpretation limits.

Sensor	Output used	Interface	Principal limitation
SGP30 [20]	TVOC (ppb), eCO ₂ (ppm)	I ² C, 0x58	Indoor-air metal-oxide sensor. eCO ₂ is an estimate derived from raw signals, not exhaled CO ₂ . Not acetone-selective.
MQ-135 [21]	Raw ADC count	Analog, GPIO 34	Broad cross-sensitivity. Raw count is not a calibrated concentration. Needs heater warm-up.
DHT22 [22]	°C, %RH	Single wire, GPIO 4	Reads no faster than about once per 2 s. Near-saturated exhaled air risks condensation.
INMP441 [23]	Digital audio, audio level	I ² S	Amplitude alone does not diagnose disease. Sensitive to ambient noise.
MAX30102 [24]	Red/IR counts; HR, SpO ₂	I ² C, 0x57	Supplementary, not a breath sensor. Affected by motion, perfusion and skin tone.
OLED (optional)	Status display	I ² C	May share the bus only if address and logic level are compatible.

SGP30 sensor detects VOC and CO₂-eq in indoor air and requires sampling at a one-second interval to generate the default value [20]. MQ-135 sensor is a metal oxide sensor with resistive change to many gases, but the output of unshielded sensor module cannot measure concentrations [21]. The sensor DHT22 provides temperature and relative humidity data [22]. The INMP441 is a digital microphone that outputs audio samples using I²S interface [23]. The MAX30102 sensor has two LEDs (red and infrared), a photodetector, and FIFO memory [24].

6.3 Pin assignments

Table 3 shows the pin assignment that was used in previous tests. Wiring and electrical compatibility, along with logic levels and MQ-135 sensor output, have to be checked for any assertion about the functionality of the entire circuit.

Table 3. ESP32 pin assignment used in earlier tests.

Signal	GPIO	Note
I ² C SDA	21	Shared by SGP30, MAX30102, optional OLED
I ² C SCL	22	Shared bus clock
DHT22 data	4	Single-wire protocol
MQ-135 analog output	34	ADC1 channel, input-only
MQ-135 digital output	27	Optional threshold output
INMP441 BCLK	26	I ² S bit clock
INMP441 WS/LRCLK	25	I ² S word select
INMP441 SD	33	I ² S serial data

6.4 Breath chamber design

A portable box of approximate dimension 200 x 80 x 155 mm is being suggested, having a disposable mouthpiece, a sample tube, a sensing box, and a vent. These dimensions are tentative. The aim is to

isolate the heated MQ-135 sensor from the user's breathing pathway, reduce moisture condensation on sensors, and measure ambient air at the start of each breath cycle.

7 Methodology

7.1 Phased development

Development occurs in phases: isolated testing of each sensor, testing pairs of sensors on the common bus, firmware integration, backend connection, and finally measurement control. A phase is deemed completed only if the output of the phase is plausible and consistent. The integration phase is yet to be completed (see Results and discussion).

7.2 Measurement record

Every acquisition process results in a record containing the following fields shown in Table 4. This table outlines the schema target and validity constraints; they are design constraints of the data path, not a guarantee that all the constraints apply to the existing codebase.

Table 4. Measurement record and intended validity rules.

Field	Source	Intended validity rule
tvoc_ppb, eco2_ppm	SGP30	Missing if sensor not detected; initialisation defaults excluded
mq135_raw	MQ-135	Range 0–4095; flagged saturated at 4095
temperature_c	DHT22	Missing on read failure; plausibility range check
humidity_pct	DHT22	Range 0–100; high values flagged as condensation risk
spo2_pct, heart_rate_bpm	MAX30102	Invalid if red or IR count is zero
audio_level	INMP441	Frame level; silence and clipping flagged
mode	Backend	Hardware or synthetic; never mixed

7.3 Data provenance and quality gating

Data path is illustrated in Figure 2 below. Each unfiltered frame undergoes presence and plausibility tests, is tagged with a mode and per channel quality, and gets saved. Synthetic demonstrative data have the synthetic tag and are not used for training nor measurements. High ADC value and zero optical reading get flagged and never evaluated as clinical.

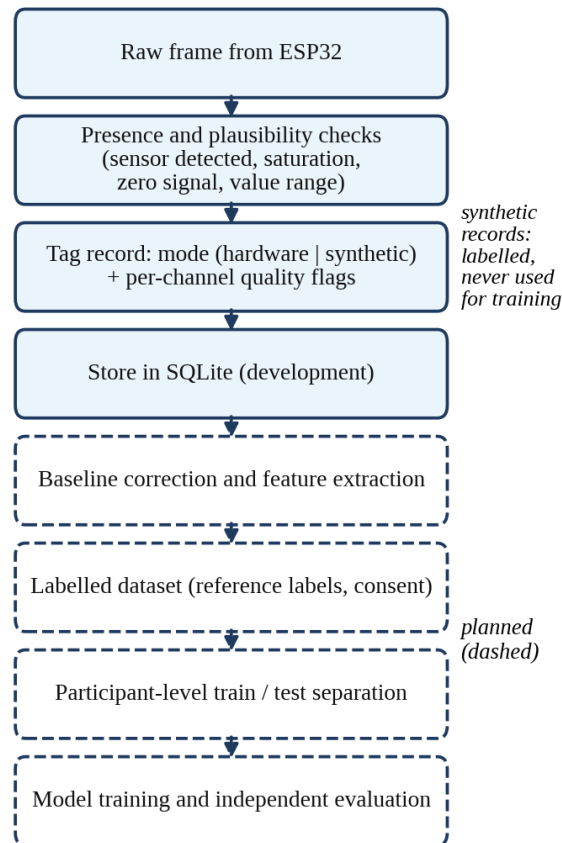


Figure 2. Data-quality gating and planned dataset pipeline. Dashed steps are planned.

7.4 Planned sampling protocol

Sampling in the future will adhere to the following sampling protocol: an initial ambient baseline reading prior to each sampling event, a standardised sitting and breathing procedure, a number of repetitions for exhaling by participants, and an abstention period prior to sampling from food, drink and smoking, determined in accordance with the standardisation guidelines described in [3]. The conditions are recorded to account for exogenous confounders [11]. The sensors will have had their recommended time to warm up, and breath signals will be measured relative to the session baseline [10].

7.5 Planned analytics

After calibration and cleaning, features such as baseline-corrected response amplitude, peak, slope and area for gas channels, and level and spectral descriptors for audio, will be extracted. A random-forest classifier [25] implemented in scikit-learn [26] is a candidate baseline model. Data will be split by participant, never by sample, so that repeated exhalations from one person cannot appear in both training and test sets [12]. Sensitivity, specificity and area under the ROC curve with confidence intervals will be reported only against reference labels such as clinical lung-function testing. None of this analysis has been performed.

8 Firmware and software implementation

8.1 Firmware

Firmware is programmed in the Arduino environment for the ESP32. Its data-acquisition functions consist of individual routines per sensor. SGP30 gets sampled every second, DHT22 is sampled not quicker than the minimal interval required for it to be read, MQ-135 uses averaged ADC1 readings, INMP441 uses DMA buffered I²S frame samples to compute the level value and MAX30102 uses its FIFO buffer. The measurements get timestamps and are transmitted to the backend bridge in text form via serial connection.

8.2 Integration fault

The problem of resetting followed by the message “ADC: CONFLICT! driver_ng is not allowed to be used with the legacy driver” occurred when all the sensor drivers were included in one build. The occurrence of such a message means that the legacy driver and the new ADC driver have been compiled together into the same binary file. We consider the reason for that to be a combination of a legacy I²S driver and the new ADC driver for reading analog signals [19]. The proposed solution is to port the microphone driver to the new I²S API so that only one generation of ADC driver stays in the build.

8.3 Backend

Backend is implemented in Python using FastAPI [27] with the aid of Uvicorn, SQLAlchemy, and an SQLite database. During development, the backend runs in loopback mode (IP address: 127.0.0.1, port: 8765). The last entry can be retrieved via an API endpoint /api/latest. Loopback service with an SQLite database is appropriate only during development, while a deployment for real-life users will require transport-level encryption, authentication, and server-level database management.

8.4 Dashboard and reports

The dashboard aims for white and sky blue design with responsive desktop and mobile interfaces along with clear connection status. There will be eight views as follows: desktop view, live monitoring, patients, breath test, history, report, mobile view and mobile monitoring. The workflow aims to connect patients, tests creation, sensor monitoring, history and PDF report. Table 5 presents the results of the bench test with the current build.

9 Experimental setup and testing

All the tests performed were bench tests. The ESP32 DevKit V1 board and sensor module components were mounted using a breadboard and jumper wires and programmed using Arduino IDE from a laptop. The serial output was observed using USB, and an I²C scan was done to check for the presence of the addresses 0x58 and 0x57. The test was done separately using each of the sensors and then together and lastly with the backend running and querying the status endpoint.

10 Results and discussion

10.1 Observations

Table 5 summarises the sample readings obtained from the sensors during the bench tests.

Table 5. Bench observations and sample readings.

Sensor	Parameter	Sample reading
DHT22	Temperature	Approximately 28 °C
DHT22	Relative humidity	Varying readings in %RH during the test
SGP30	TVOC, eCO ₂	Values returned at times; later ERROR and no response in I ² C scan (expected address 0x58)
MQ-135	Raw ADC count	4095 (12-bit full scale) in the latest reading
INMP441	Audio level	Audio captured in the stand-alone test
MAX30102	Red and IR counts	Zero at times (expected address 0x57)
Integrated firmware	Reset behaviour	Repeated resets with the ADC driver conflict message

10.2 Discussion of individual findings

DHT22. Readings around 28 °C with different humidity levels could be expected under room conditions, yet without a reference thermometer and hygrometer no precision of the readings can be confirmed. As exhaled air is humid, it needs to be studied how the sensor behaves in case of high humidity.

SGP30. This sensor generated TVOC and eCO₂ values initially, but then gave an error and disappeared from the bus, so it was not found in I²C scan. A disappeared sensor is primarily associated with problems with wiring, power supply, or bus loading issues like poor contact with the breadboard, and not with sensing errors. In addition, systematic checks of power voltage, pull-up resistors, and connections are necessary. Default initialisation at start-up should not be confused with readings [20].

MQ-135. Raw value of 4095 corresponds to the total number of 12-bit ADC samples. This is a signal of saturation or input voltage exceeding the limits of converter. Possible reasons can be an output from the module exceeding the input range of the ESP32 without using a divider, the lack of warming up, or the incorrect wiring of the device [19, 21]. Without excluding these options, the channel is considered invalid.

INMP441 and MAX30102. Since the microphone worked separately in the test, the wiring is probably correct, and the problem occurs at the integration stage (see Integration fault). The sensor MAX30102 gives the result of zero counts of red and infrared. Zero counts mean that no valid pulses were detected. Possible reasons for this are unconfigured LED current value, the lack of finger contact or bus error.

10.3 Implications for data handling

From the bench tests, it has been established that individual sensors can be raised, that the backend and the dashboard with labels work, and that various failure modes are generated upon integration of the system. The above findings confirm the contribution of C2, which is, the lack of per-channel validity gating means that saturated values or zero values will be saved along with true values for training the model.

11 Limitations, ethical considerations and safety

11.1 Limitations

SGP30 gives readings of TVOC and CO₂ equivalent and is neither an acetone detector nor a measurement of clinical exhaled CO₂. The MQ-135 has wide cross-sensitivity and the uncalibrated output reading is not a concentration reading. Amplitude of sound does not determine the respiratory disease condition, and MAX30102 is not a breath sensor. Future areas of research include respiratory disorders, metabolic disorders, renal disorders or hepatic disorders, and telemedicine.

11.2 Ethics and data protection

Sampling of breath from the subjects needs approval of the institutional ethical review board as well as giving consent where participants will be made aware that the device is a research prototype and not a diagnostic tool. Subjects will be coded and will not provide personal identifiers while access to the information will be limited only to the research group. For any application outside the scope of research, there will be need for encryption during transmission of data, authentication and compliance with the regulations governing health data. It is important to have references, calibration and independent validation of results before any claim about diseases is made.

11.3 Hygiene and electrical safety

A one-time-use mouthpiece, together with a one-way valve where necessary, helps prevent cross contamination from one person to another. Since the MQ-135 heater gets hot, it has to be protected from the user. In the development phase, the device gets powered from the USB port of the laptop, and in case the device uses batteries in future, proper measures should be taken. The outputs displayed on the dashboard should have a clear indication of what they are.

12 Conclusion and future work

This paper presented RespiraTech, a multimodal IoT breath-monitoring prototype, together with a data-integrity methodology that distinguishes real from synthetic data and defines validity rules for each channel. The prototype combines gas, environmental, acoustic and physiological sensing around an ESP32 with a Python backend and a labelled demonstration dashboard. Bench tests showed that individual sensors can operate, but they also exposed ADC saturation, intermittent I²C devices, zero optical signals and a driver conflict that currently prevent reliable integrated operation.

Future work proceeds in this order: resolve the driver conflict and validate every bus connection; add level conditioning for the MQ-135 and complete sensor warm-up and calibration; test the chamber with controlled reference conditions; obtain ethics approval; collect a pilot dataset with reference labels under the protocol described in the Methodology; and train and evaluate models with participant-level separation. Edge inference and field trials are later steps that depend on all of the above.

Acknowledgment

The authors thank Prof. Fayaz Ahamed Shaikh, Department of Computer Science and Engineering (IoT, Cybersecurity Including Blockchain), for guidance and support throughout this work.

References

- [1] Amann A, et al. The human volatilome: Volatile organic compounds (VOCs) in exhaled breath, skin emanations, urine, feces and saliva. *J. Breath Res.* 2014;8(3):034001.
- [2] Miekisch W, Schubert JK, Noeldge-Schomburg GFE. Diagnostic potential of breath analysis—Focus on volatile organic compounds. *Clin Chim Acta.* 2004;347(1–2):25–39.
- [3] Horváth I, et al. A European Respiratory Society technical standard: Exhaled biomarkers in lung disease. *Eur. Respir. J.* 2017;49(4):1600965.
- [4] Deng C, Zhang J, Yu X, Zhang W, Zhang X. Determination of acetone in human breath by gas chromatography–mass spectrometry and solid-phase microextraction with on-fiber derivatization. *J. Chromatogr. B.* 2004;810(2):269–275.
- [5] Wilson AD, Baietto M. Applications and advances in electronic-nose technologies. *Sensors.* 2009;9(7):5099–5148.
- [6] Gutierrez-Osuna R. Pattern analysis for machine olfaction: A review. *IEEE Sensors J.* 2002;2(3):189–202.
- [7] Dragonieri S, et al. An electronic nose in the discrimination of patients with asthma and controls. *J. Allergy Clin. Immunol.* 2007;120(4):856–862.
- [8] Fens N, et al. Exhaled breath profiling enables discrimination of chronic obstructive pulmonary disease and asthma. *Am. J. Respir. Crit. Care Med.* 2009;180(11):1076–1082.
- [9] Shahzad AA, et al. A low-cost device for measurement of exhaled breath for the detection of obstructive lung disease. *Biosensors.* 2022;12(6):409. <https://doi.org/10.3390/bios12060409>
- [10] Vergara A, et al. *Chemical gas sensor drift compensation using classifier ensembles.* Sens Actuators B Chem. 2012;166–167:320–329.
- [11] Pleil JD, Stiegel MA, Risby TH. Clinical breath analysis: Discriminating between human endogenous compounds and exogenous (environmental) chemical confounders. *J. Breath Res.* 2013;7(1):017107.
- [12] Saeb S, Lonini L, Jayaraman A, Mohr DC, Kording KP. *The need to approximate the use-case in clinical machine learning.* GigaScience. 2017;6(5):gix019.
- [13] Righettoni M, Tricoli A, Pratsinis SE. Si:WO₃ sensors for highly selective detection of acetone for easy diagnosis of diabetes by breath analysis. *Anal. Chem.* 2010;82(9):3581–3587.
- [14] Bohadana A, Izbicki G, Kraman SS. Fundamentals of lung auscultation. *N. Engl. J. Med.* 2014;370(8):744–751.
- [15] Rocha BM, et al. An open access database for the evaluation of respiratory sound classification algorithms. *Physiol. Meas.* 2019;40(3):035001.
- [16] Jubran A. Pulse oximetry. *Crit. Care.* 2015;19:272.
- [17] Sjoding MW, Dickson RP, Iwashyna TJ, Gay SE, Valley TS. Racial bias in pulse oximetry measurement. *N. Engl. J. Med.* 2020;383(25):2477–2478.
- [18] Islam SMR, Kwak D, Kabir MH, Hossain M, Kwak KS. The Internet of Things for health care: A comprehensive survey. *IEEE Access.* 2015;3:678–708.
- [19] Espressif Systems. *ESP32 series datasheet; ESP-IDF programming guide: ADC, I2S and GPIO peripherals.* Shanghai, China: Espressif Systems. <https://docs.espressif.com>
- [20] Sensirion AG. *SGP30 datasheet: Indoor air quality sensor for TVOC and CO₂eq measurements.* Stäfa, Switzerland: Sensirion AG. <https://sensirion.com>
- [21] Zhengzhou Winsen Electronics Technology Co., Ltd. *MQ-135 gas sensor technical data sheet.* Zhengzhou, China.
- [22] Aosong Electronics Co., Ltd. *DHT22 (AM2302) digital temperature and humidity sensor datasheet.* Guangzhou, China.
- [23] InvenSense (TDK). *INMP441: Omnidirectional microphone with bottom port and I²S digital output, datasheet.* San Jose (CA), USA.
- [24] Maxim Integrated. *MAX30102: High-sensitivity pulse oximeter and heart-rate sensor for wearable health, datasheet.* San Jose (CA), USA.
- [25] Breiman L. Random forests. *Mach. Learn.* 2001;45(1):5–32.
- [26] Pedregosa F, et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* 2011;12:2825–2830.
- [27] Ramírez S. *FastAPI documentation.* <https://fastapi.tiangolo.com>