

● WHITEPAPER · VAHA VOICE CAPTURE

# Wake-word and stop-phrase detection for VAHA: dataset, validation accuracy and first field check

---

---

## AUTHORS

Ritesh Bonthalakoti

## REVIEWER

Bharat Meda

## VERSION

v1.0

## DATE

2026-10-21

## STATUS

Published

## DISTRIBUTION

Public

## CONTENTS

|                                        |                                       |
|----------------------------------------|---------------------------------------|
| Abstract                               | 6 Analysis and discussion             |
| 1 Introduction and objective           | 7 Limitations and threats to validity |
| 2 Requirements and acceptance criteria | 8 TRL assessment                      |
| 3 System under test                    | 9 Conclusions and next steps          |
| 4 Test setup and method                | 10 References                         |
| 5 Results                              | Appendices                            |

## ABSTRACT

# Abstract

VAHA is Dharanova's offline voice-capture device. It starts recording when it hears the wake word "Marvin" and stops on a stop phrase (class `im_done`) or on silence. Detection uses an Edge Impulse keyword-spotting model, MFCC features and a Keras classifier, that runs on an Arduino UNO Q. This paper reports what the model was trained on and how well it classifies. The training project holds 8,466 one-second samples in three classes, split 80/20 into training and test. The `marvin` class comes from the public Google Speech Commands dataset and `bg` was imported from another Edge Impulse project; `im_done` was recorded for VAHA. On the platform's validation split the int8 model scored 97.6 % (95 % interval 96.6–98.3 %, calculated), below the 98.7 % quoted earlier for this model. Three unseen test samples, one per class, were all classified correctly with over 99.9 % confidence. A 34-trial check in a bathroom passed every trial, but individual trials were not logged. Every measured accuracy so far comes from the cloud platform, not from the shipped model on the device. The work is an experimental proof of concept (TRL 3).

### KEY POINT

#### Key findings

- **Accuracy:** 97.6 % on the validation split, int8 model (96.6–98.3 %, calculated). Recall: `bg` 94.7 %, `im_done` 99.3 %, `marvin` 97.4 %.
- **Earlier figure:** the 98.7 % quoted earlier for this model lies outside that interval. The run it came from has not been identified.
- **Dataset:** 8,466 samples. `marvin` is public Google Speech Commands data and `bg` was imported from another project; `im_done` was recorded for VAHA.
- **Unseen samples:** 3 of 3 correct at 99.9 % or more, 19.9 to 25.0 points above the firmware thresholds.
- **Field check:** 34 of 34 passed in a bathroom, without a per-trial log. Logged, the same result would show at least 91.6 %.
- **Readiness:** TRL 3. Next: the shipped model on the UNO Q over the 1,689 test samples.

## SECTION 1

# Introduction and objective

VAHA exists to catch a thought before it is lost, so recording has to start with no button and no phone. The wake-word detector therefore has to trigger on a softly spoken word and stay silent through the sounds of a bathroom: an exhaust fan, running water and a tiled room's echo. A stop phrase ends the recording, with silence as a fallback.

The detector is a three-class keyword-spotting model: `marvin` (wake), `im_done` (stop) and `bg` (everything else). This paper answers four questions:

1. What is the model trained and tested on, and where did each class come from?
2. How accurately does the model separate the three classes, on data it was not trained on?
3. Does it agree with the 98.7 % quoted earlier for this model?
4. What does the evidence show about the shipped model on the device, in a bathroom?

**In scope:** dataset composition and provenance; the platform-reported validation accuracy and confusion matrix; live classification of unseen test samples; the bathroom field check; and the detection thresholds in the shipped firmware.

**Out of scope:** speech-to-text after recording starts (a separate publication); characterisation of the microphone; model retraining and long-term drift.

## SECTION 2

# Requirements and acceptance criteria

**No measurable acceptance criteria were set before this work.** The product's design intent, hands-free start and stop in a noisy bathroom, was the only target. Table 2.1 lists the objectives this paper evaluates afterwards; they are qualitative and are assessed in section 6.

Table 2.1: Objectives (qualitative, evaluated after testing)

| ID | Objective                                                                                                        |
|----|------------------------------------------------------------------------------------------------------------------|
| O1 | The model separates <code>marvin</code> , <code>im_done</code> and <code>bg</code> on data it was not trained on |
| O2 | Unseen test samples are classified correctly, above the firmware's detection thresholds                          |
| O3 | The shipped detector starts and stops recording reliably in a bathroom with fan and water noise                  |
| O4 | The training data is documented: source and licence for every class                                              |

Measurable limits will be set before the next round of tests: accuracy on the test set, the missed-wake and missed-stop rates, and false wakes per hour of background noise (section 9).

SECTION 3

System under test

The detector listens continuously, classifies overlapping one-second windows and acts on the confidence scores (Figure 1).

DETECTION PATH ON THE DEVICE

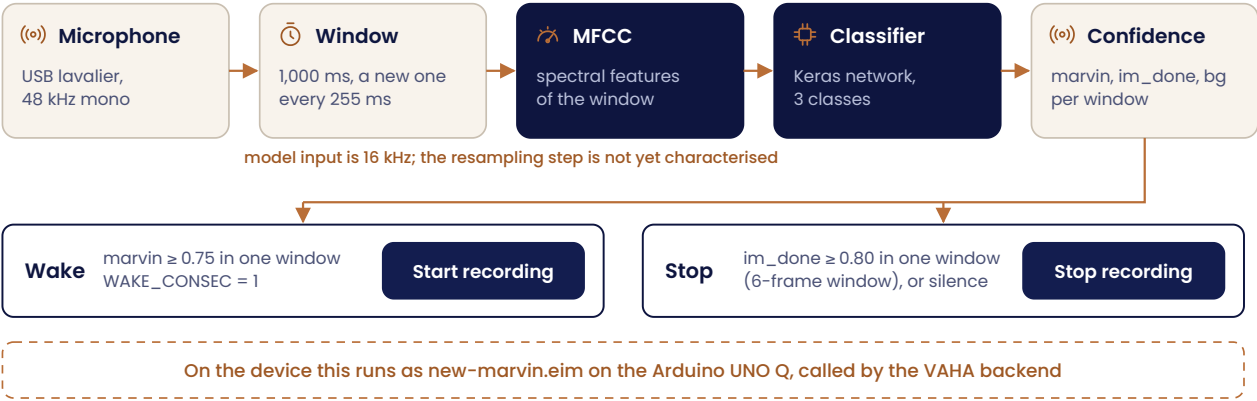
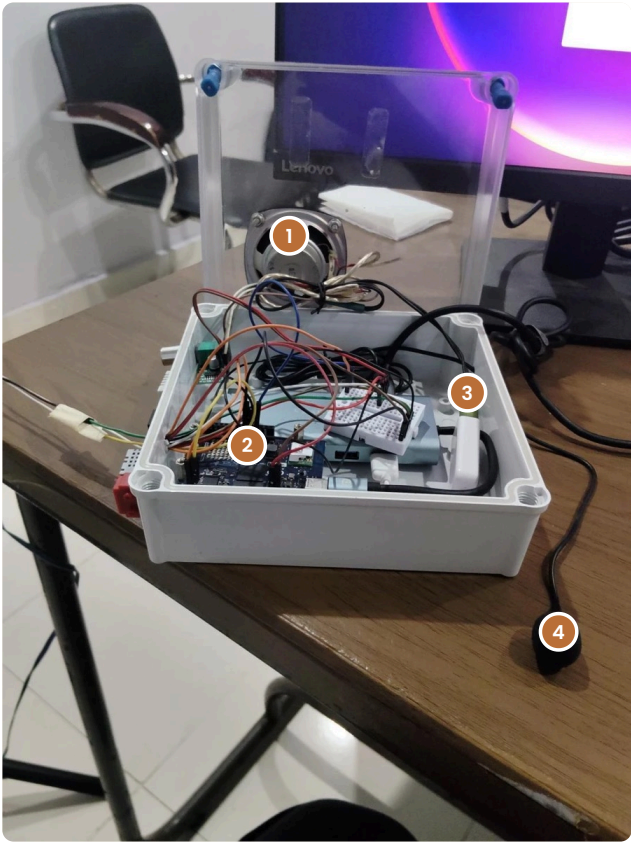


**Figure 1** The detection path. Audio is classified in 1,000 ms windows, a new one every 255 ms (3.92 per second, calculated). A single window above threshold is enough to start or stop recording.

THE AUDIO HARDWARE, ENCLOSURE OPEN



- 1 Speaker**  
PAM8403 amplifier and driver; chime and speech output, not in the detection path
- 2 Arduino UNO Q**  
Runs new-marvin.eim and the wake and stop logic
- 3 3.5 mm microphone input jack**  
Part of the audio input path
- 4 USB lavalier microphone**  
The input the model listens on, 48 kHz mono

**Figure 2** The VAHA audio hardware with the enclosure open.

Table 3.1: Components

| Item                | Detail                                                                                                                                                                                                                                                                                             |
|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Hub                 | Arduino UNO Q, 4 GB RAM / 32 GB storage, Linux alongside a microcontroller core. Board revision and serial not recorded                                                                                                                                                                            |
| Microphone          | USB lavalier microphone, 48 kHz mono. Model and serial not recorded                                                                                                                                                                                                                                |
| Model               | Edge Impulse Studio project 1079833 ("Marvin WakeWord"). Impulse: 1,000 ms window, 255 ms increase, 16 kHz → MFCC (DSP block, implementation v4) → Keras classifier, trained on CPU                                                                                                                |
| Model on the device | <code>new-marvin.eim</code> , called by VAHA's Python backend. The training run it was exported from, and whether it is the int8 or float32 model, have not yet been confirmed                                                                                                                     |
| Classes             | <code>marvin</code> (wake), <code>im_done</code> (stop), <code>bg</code> (background: everything else)                                                                                                                                                                                             |
| Detection rules     | Wake: <code>marvin</code> $\geq 0.75$ in one window ( <code>WAKE_THRESHOLD</code> = 0.75, <code>WAKE_CONSEC</code> = 1). Stop: <code>im_done</code> $\geq 0.80$ in one window ( <code>STOP_THRESHOLD</code> = 0.80, <code>STOP_CONSEC</code> = 1, <code>STOP_WINDOW</code> = 6 frames), or silence |
| Audio path          | Captured at 48 kHz, classified at 16 kHz. The resampling step between the two has not yet been characterised                                                                                                                                                                                       |

## SECTION 4

# Test setup and method

## 4.1 Equipment

| Instrument          | Detail                                                                      | Used for                                 |
|---------------------|-----------------------------------------------------------------------------|------------------------------------------|
| Edge Impulse Studio | Project 1079833, through its REST API with a project-scoped key, 2026-10-21 | Dataset composition, live classification |
| Edge Impulse Studio | Training-output page for the same project                                   | Validation accuracy and confusion matrix |
| VAHA device         | Arduino UNO Q with the USB lavalier microphone (Table 3.1)                  | Bathroom field check                     |

No calibrated instrument was used; none is needed for counting classifications.

## 4.2 Environment

The dataset count, validation and live classification ran on the Edge Impulse cloud platform. The field check ran in one bathroom with the exhaust fan running and a tap open. Its date, the speaker, the distance to the microphone and the noise level were not recorded.

## 4.3 Procedures

Each source of evidence covers a different model and a different set of audio (Figure 3).

### ● THREE SOURCES OF EVIDENCE, AND WHAT EACH ONE COVERS

| Platform validation                                                             | Live classification                                                  | Bathroom field check                                 |
|---------------------------------------------------------------------------------|----------------------------------------------------------------------|------------------------------------------------------|
| <b>WHERE IT RAN</b><br>Edge Impulse Studio (cloud)                              | <b>WHERE IT RAN</b><br>Edge Impulse Studio (cloud)                   | <b>WHERE IT RAN</b><br>VAHA device, in a bathroom    |
| <b>MODEL</b><br>int8 model of the training run shown in Studio                  | <b>MODEL</b><br>The project's current model                          | <b>MODEL</b><br>new-marvin.eim on the UNO Q          |
| <b>DATA</b><br>Validation split: 20 % of the training data, about 1,355 samples | <b>DATA</b><br>3 test samples, one per class, never used in training | <b>DATA</b><br>34 spoken trials, fan on, tap running |
| <b>RECORD</b><br>Training-output screenshot                                     | <b>RECORD</b><br>Raw API responses                                   | <b>RECORD</b><br>Summary result, no per-trial log    |

Not yet done: the shipped model, on the device, over the 1,689-sample test set

**Figure 3** The three sources of evidence in this paper. Only the field check ran on the device, and it was not logged.

**A. Dataset composition** (2026-10-21). Every training and test sample in the project was counted by class and split, and filenames and storage paths were checked to confirm where each class came from.

**B. Validation accuracy.** Edge Impulse holds back 20 % of the training split as a validation set during training (Appendix A) and reports accuracy, a confusion matrix and F1 scores on it. The values were read from the training-output page with the int8 (quantised) model selected (Appendix C).

**C. Live classification** (2026-10-21). One test sample per class, never used in training, was classified by the platform's own inference on its servers, which returns a confidence score for each class.

**D. Field check.** With VAHA in a bathroom, fan on and tap open, the wake word and the stop phrase were spoken in 34 trials in total. The result was noted as a summary; per-trial results and the split between the two phrases were not recorded.

**No test-set evaluation yet.** The platform's model-testing run over the 1,689-sample test set was not available for this project, so no accuracy is reported on the test set as a whole.

## 4.4 Runs and sample size

| Test                | n                                               | Comment                                    |
|---------------------|-------------------------------------------------|--------------------------------------------|
| Dataset composition | 8,466 samples                                   | Complete count                             |
| Validation accuracy | About 1,355 samples (20 % of 6,777, calculated) | One training run                           |
| Live classification | 3 samples, one per class                        | Functional check, not an accuracy estimate |
| Field check         | 34 trials                                       | Summary result only                        |

**CAUTION****Small and unlogged samples**

Only the validation split is large enough to estimate accuracy. The live classifications and the field check show that the detector works, not how often it fails (section 5.5).

**4.5 Standards followed**

No standard for keyword-spotting accuracy was identified as directly applicable. The test documentation follows the structure of ISO/IEC/IEEE 29119-3 loosely. No accredited testing was done.

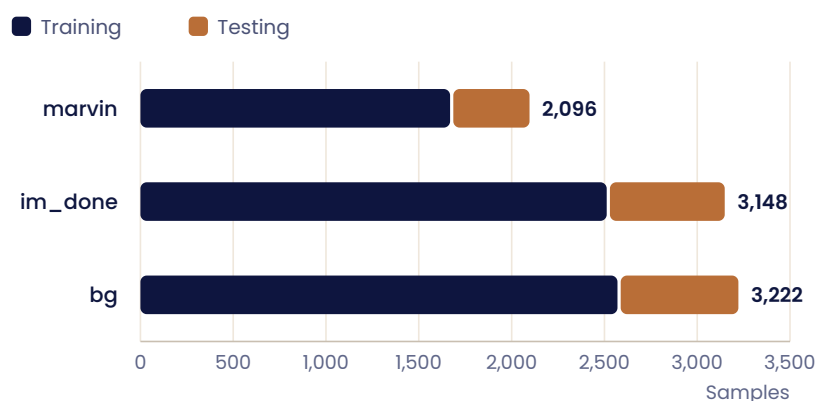
**SECTION 5****Results**

Values are **measured** (platform-reported) or counted unless marked calculated. Every calculated value, and every chart, is produced by the analysis scripts published with this paper.

**5.1 Dataset composition and provenance**

Table 5.1: Samples by class and split (counted, 2026-10-21)

| Class        | Training     | Testing      | Total        | Share of all samples (calculated) |
|--------------|--------------|--------------|--------------|-----------------------------------|
| bg           | 2,579        | 643          | 3,222        | 38.1 %                            |
| im_done      | 2,521        | 627          | 3,148        | 37.2 %                            |
| marvin       | 1,677        | 419          | 2,096        | 24.8 %                            |
| <b>Total</b> | <b>6,777</b> | <b>1,689</b> | <b>8,466</b> |                                   |

**● DATASET BY CLASS AND SPLIT, 8,466 SAMPLES****Provenance**

Public Google Speech Commands  
identified from filenames

Recorded for this project  
stored in this project

Imported from another project  
original source not yet traced

**Figure 4** The dataset by class and split, with what is known of each class's source.

Each class is split almost exactly 80/20: the test share is 19.96 % for `bg`, 19.92 % for `im_done` and 19.99 % for `marvin` (calculated).

**Provenance.** All 419 `marvin` test filenames follow the `<hash>_nohash_<n>` pattern of the public Google Speech Commands dataset, which is where the class comes from. The classified `marvin` and `bg` samples are stored under another Edge Impulse project (342217), so both classes were imported into this one. The `im_done` sample is stored under this project (1079833), as expected for audio recorded for VAHA. Where the `bg` audio originally came from has not yet been traced.

## 5.2 Live classification of unseen test samples

Table 5.2: Three test samples, one per class, 2026-10-21

| Test sample                         | True class           | Top class, confidence           | Firmware threshold | Margin (calculated) | Result |
|-------------------------------------|----------------------|---------------------------------|--------------------|---------------------|--------|
| <code>ff63ab0b_nohash_1</code>      | <code>marvin</code>  | <code>marvin</code> , 99.999 %  | 0.75               | 25.0 points         | Met    |
| <code>im_done_001799</code>         | <code>im_done</code> | <code>im_done</code> , 99.924 % | 0.80               | 19.9 points         | Met    |
| <code>000020a912-99_segment1</code> | <code>bg</code>      | <code>bg</code> , 99.991 %      | No action          | —                   | Met    |

All three exceed the platform's own minimum confidence of 0.6 as well as the firmware thresholds. The classifications ran on the platform's servers, with the project's current model. They show that the model in the project classifies unseen audio correctly. They do not test `new-marvin.eim` on the device.

## 5.3 Validation accuracy and confusion matrix

Table 5.3: Validation metrics, int8 model (platform-reported unless marked)

| Metric                                                                  | Value              |
|-------------------------------------------------------------------------|--------------------|
| Overall accuracy                                                        | 97.6 %             |
| 95 % interval for the accuracy (calculated, Wilson, $n \approx 1,355$ ) | 96.6–98.3 %        |
| Loss                                                                    | 0.06               |
| F1 score: <code>bg</code> / <code>im_done</code> / <code>marvin</code>  | 0.96 / 0.99 / 0.97 |
| Weighted average precision / recall                                     | 0.98 / 0.98        |
| Area under ROC curve                                                    | 1.00               |
| Accuracy quoted earlier for this model (product README, R&D log)        | 98.7 %             |

### ● VALIDATION CONFUSION MATRIX, INT8 MODEL, ROW-NORMALISED

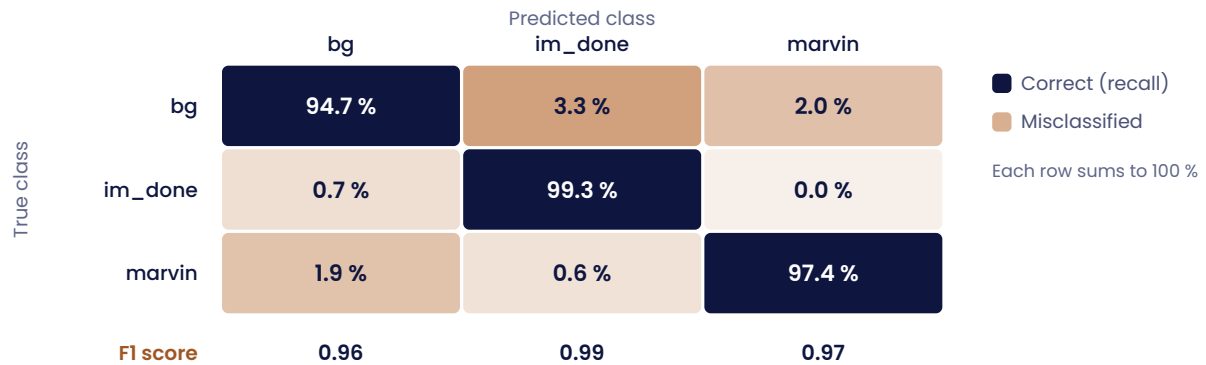


**Figure 5** Validation confusion matrix for the int8 model, row-normalised. The diagonal is each class's recall.

bg has the lowest recall: 94.7 % of background windows are classified correctly, 3.3 % as im\_done and 2.0 % as marvin. im\_done has the highest, 99.3 %.

## 5.4 Bathroom field check

**Table 5.4: Field check**

| Item    | Value                                                                                                   |
|---------|---------------------------------------------------------------------------------------------------------|
| Setting | Bathroom, exhaust fan running, tap open                                                                 |
| Trials  | 34, wake word and stop phrase combined                                                                  |
| Result  | 34 of 34 passed at the firmware thresholds                                                              |
| Record  | Summary result in the VAHA repository README (reference 3). No per-trial log, date, speaker or distance |

## 5.5 Uncertainty

### ● WHAT EACH PIECE OF EVIDENCE CAN SHOW, AT 95 % CONFIDENCE

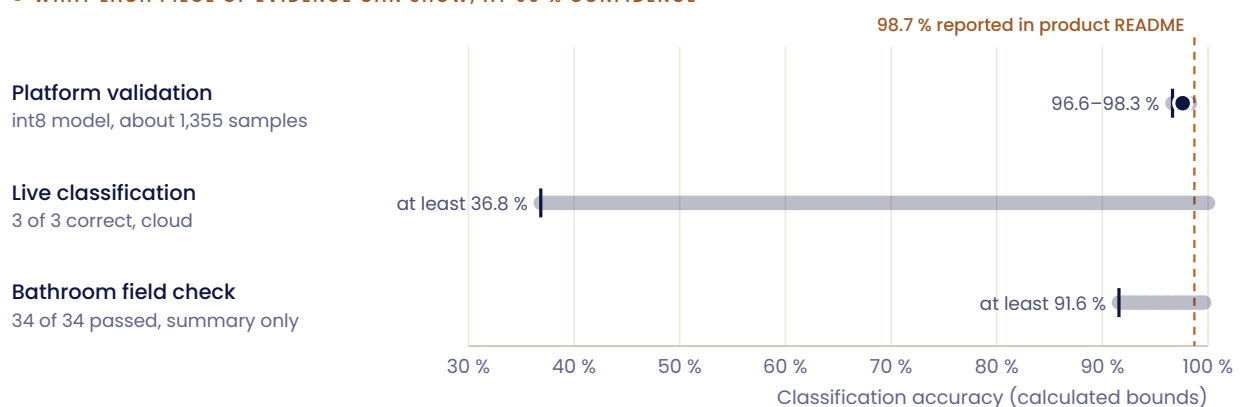


**Figure 6** What each piece of evidence can show about accuracy at 95 % confidence (calculated). The field-check bound treats the 34 passes as logged results.

- **Validation accuracy:** 96.6–98.3 % at 95 % (Wilson score interval, calculated). The validation set size, about 1,355, is taken as 20 % of the training split; the platform does not report it exactly. The interval covers sampling only, from one training run.

- **Live classification:** with 3 of 3 correct, accuracy can only be said to exceed 36.8 % at 95 % confidence (exact binomial, calculated).
- **Field check:** if all 34 trials were logged as correct, accuracy would exceed 91.6 % at 95 % confidence. Showing at least 95 % takes 59 consecutive correct trials, and 99 % takes 299 (calculated).
- **Transcription:** the validation metrics were transcribed by hand from the training-output page (Appendix C). The confusion-matrix rows sum to 100 % within rounding, as checked by the analysis script.

## SECTION 6

# Analysis and discussion

## 6.1 Results against the objectives

| ID | Objective                                     | Result                                                                                                                   | Met?                       |
|----|-----------------------------------------------|--------------------------------------------------------------------------------------------------------------------------|----------------------------|
| O1 | Separates the three classes on unseen data    | 97.6 % on the validation split (96.6–98.3 %); recall 94.7–99.3 % per class. Measured on the platform, for the int8 model | Met                        |
| O2 | Unseen test samples correct, above thresholds | 3 of 3 correct, 19.9 and 25.0 points above the wake and stop thresholds. Too few to estimate a rate                      | Met                        |
| O3 | Reliable start and stop in a bathroom         | 34 of 34 passed; no per-trial log, date or conditions recorded                                                           | Partly met (not logged)    |
| O4 | Training data documented                      | marvin source identified from filenames; bg imported, original source not yet traced                                     | Partly met (bg not traced) |

## 6.2 Where the detector is most likely to fail

**Background audio is the weak point.** bg has the lowest recall, 94.7 %. Of all background windows, 2.0 % are taken for the wake word, a false wake, and 3.3 % for the stop phrase, which ends a running recording early. Per window, a false stop is therefore more likely than a false wake, which the 97.6 % headline does not show.

**Overlapping windows help sensitivity.** The marvin and im\_done test samples are 0.94 s and 0.84 s long, and a new 1 s window starts every 255 ms, so each spoken phrase falls into several overlapping windows. With WAKE\_CONSEC = 1, any one of them above 0.75 starts a recording, so a single missed window does not mean a missed wake.

**The same overlap drives false wakes.** The detector makes 14,118 decisions an hour (calculated), and one window above threshold is enough. A per-window false-accept rate of only 0.01 % would give about 1.4 false wakes an hour (illustrative, calculated), which a user would notice. Requiring two windows in a row (WAKE\_CONSEC = 2) should cut false wakes from isolated noisy windows, at some cost in missed wakes. The balance between the two has to be measured on real background audio.

**The thresholds are not yet tuned from data.** The three live samples scored 99.9 % or more, far above 0.75 and 0.80, but they were correct, confident samples. Setting the thresholds needs the spread of scores across the whole test set and over hours of background audio, so the trade-off between missed wakes and false wakes can be read off directly.

### 6.3 What the training data means for use in VAHA

- **marvin** is public data. Google Speech Commands (reference 2) holds short single-word clips from many speakers, so the wake word was learned from voices other than VAHA's users, recorded on other microphones in other rooms. That should help it generalise across speakers. It also means no **marvin** training sample was recorded through VAHA's own microphone or in a bathroom.
- **Licence.** Speech Commands is released under the Creative Commons Attribution 4.0 licence, which requires attribution in the product's documentation.
- **bg** is the class that matters most for false wakes, and where its audio came from has not yet been traced. Whether it contains bathroom noise (fans, running water, echo) decides how far the 94.7 % **bg** recall carries over to a real bathroom.

### 6.4 Which accuracy to quote

This paper quotes 97.6 %, the only figure with platform evidence behind it. The 98.7 % used earlier, in the product README and the R&D Development Log (RM-RD-01, 22 September 2026), is 1.1 points higher and outside the 95 % interval, so sampling does not explain it. Two explanations fit: an earlier training run, or the float32 model. The figure will be confirmed or withdrawn once the run shipped as `new-marvin.eim` is identified.

## SECTION 7

# Limitations and threats to validity

- **Platform, not device.** Validation and live classification ran in the cloud, on the project's current model. `new-marvin.eim` has run on the UNO Q only in the field check, which was not logged.
- **Validation split, not test set.** The 97.6 % comes from 20 % of the training split, held back by the platform during training. The 1,689-sample test set has not been evaluated as a whole.
- **One training run.** The validation interval covers sampling, not run-to-run variation.
- **Hand-transcribed metrics.** The validation figures were read from the training page, not exported.
- **Three live samples.** Enough to show the model works end to end, not to estimate a rate.
- **Field check not logged.** No per-trial record, date, speaker, distance or noise level.
- **Untraced source for bg**, and no VAHA-recorded **marvin** samples.
- **Audio path untested.** The device captures at 48 kHz and the model expects 16 kHz. The resampling step, and any gain or filtering before it, were not part of any test.

- No false-wake measurement over hours of real background audio.

SECTION 8

# TRL assessment

TECHNOLOGY READINESS: WHERE THIS WORK SITS

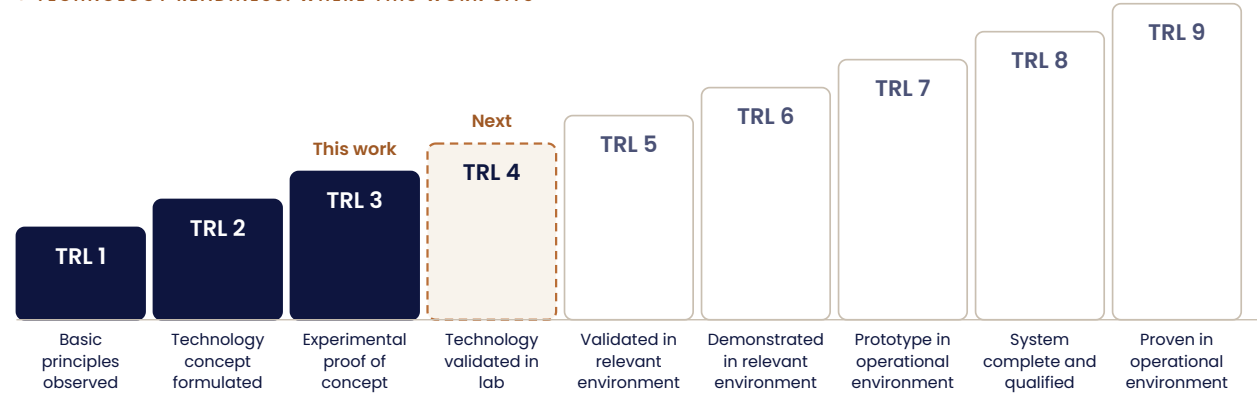


**Figure 7** Where this work sits on the EU Horizon TRL scale. Definitions from the Dharanova publishing guidelines (EU Horizon TRL scale, ISO 16290 criteria).

|                   |                                                                                                                                                                                                                                                                                                                                                     |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TRL demonstrated  | <b>Below TRL 4.</b> The evidence supports an experimental proof of concept (EU Horizon TRL 3) for the wake-word and stop-pharse detector                                                                                                                                                                                                            |
| Integration level | Component: the detector alone                                                                                                                                                                                                                                                                                                                       |
| Fidelity          | Prototype: the model runs on the UNO Q in VAHA's enclosure, but all measured evidence comes from the cloud platform                                                                                                                                                                                                                                 |
| Environment       | Cloud platform and lab. One unlogged bathroom check                                                                                                                                                                                                                                                                                                 |
| Evidence          | Sections 5.1 to 5.3                                                                                                                                                                                                                                                                                                                                 |
| Why not TRL 4     | TRL 4 requires the component to be validated in the lab <i>against defined criteria</i> . No criteria were set beforehand, the shipped model has not been tested on the device, the validation split is part of the training data, and the field check was not logged                                                                               |
| Gaps to TRL 4     | (1) Set measurable limits first. (2) Record which training run and model version ship as <code>new-marvin.eim</code> . (3) Run <code>new-marvin.eim</code> on the UNO Q over all 1,689 test samples. (4) Log field trials: at least 59 per phrase, plus hours of background audio for false wakes. (5) Record the source and licence of every class |
| Gaps beyond TRL 4 | TRL 5 needs the relevant environment defined (a candidate: a home bathroom with fan and water noise, speakers at typical distances) and the detector tested there                                                                                                                                                                                   |

## SECTION 9

# Conclusions and next steps

The VAHA detector separates its three classes well on the platform: 97.6 % on the validation split, with the stop phrase recognised most reliably and background audio least reliably. Three unseen samples were classified correctly with wide margins over the firmware thresholds. The wake-word class comes from a public dataset and the background class was imported from another project. The 98.7 % quoted earlier is not supported by the platform evidence. Every measured accuracy so far comes from the cloud, so this is a proof of concept, below TRL 4.

### ● PATH FROM THIS WORK TO TRL 4

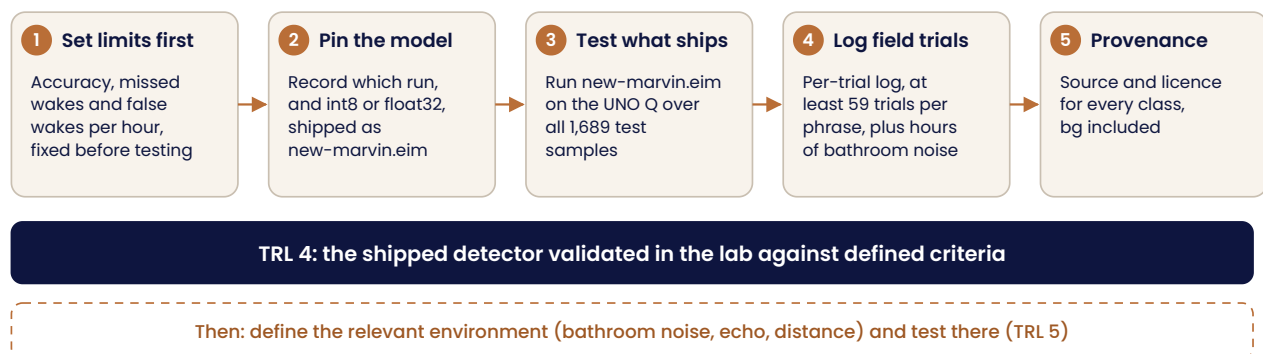


**Figure 8** Path from this work to TRL 4, from the gaps listed in section 8.

Next steps, in priority order:

- 1. Set measurable limits** for test-set accuracy, missed wakes, missed stops and false wakes per hour, before testing.
- 2. Pin the shipped model:** record the training run and model version (int8 or float32) behind `new-marvin.eim`. This also settles the 97.6 % and 98.7 % question.
- 3. Test what ships:** run `new-marvin.eim` on the UNO Q over all 1,689 test samples, recording each window's scores, not only right or wrong. This replaces the platform's model-testing run, tests the device and its audio path at the same time, and gives the score spread needed to set the thresholds and `WAKE_CONSEC` (section 6.2).
- 4. Log the field trials:** per trial, the phrase, result, confidence, distance, speaker and noise conditions; at least 59 trials per phrase; and several hours of bathroom background audio to count false wakes.
- 5. Record provenance:** trace where the `bg` audio came from, and add attribution for the Speech Commands data.
- 6. Consider VAHA-recorded wake-word samples** in the bathroom and through the device microphone, if the field trials show missed wakes.

SECTION 10

# References

1. Edge Impulse Studio project 1079833 ("Marvin WakeWord"), training and test data, training-output page and classify results, accessed 2026-10-21.
2. P. Warden, *Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition*, arXiv:1804.03209, 2018. Released under the Creative Commons Attribution 4.0 licence.
3. Dharanova, *VAHA* source repository, <https://github.com/dharanova-code/vaha>.
4. Dharanova Private Limited, *R&D Development Log RM-RD-01*, 22 September 2026 (internal document).
5. ISO/IEC/IEEE 29119-3:2021, *Software and systems engineering: Software testing, Part 3: Test documentation*.
6. ISO 16290:2013, *Space systems: Definition of the Technology Readiness Levels (TRLs) and their criteria of assessment*.

APPENDICES

# Appendices

## A. Configuration

Table A.1: Impulse and training settings (from the training page and the classify results)

| Setting                     | Value                            |
|-----------------------------|----------------------------------|
| Window size / increase      | 1,000 ms / 255 ms                |
| Sampling frequency          | 16 kHz                           |
| DSP block                   | MFCC, implementation version 4   |
| Learning block              | Keras classifier, trained on CPU |
| Training cycles             | 100                              |
| Learning rate               | 0.005                            |
| Batch size                  | 32                               |
| Validation set size         | 20 % of the training split       |
| Model version reported      | Quantised (int8)                 |
| Platform minimum confidence | 0.6                              |

Table A.2: Detection constants in the VAHA backend (source repository, reference 3)

| Constant       | Value |
|----------------|-------|
| WAKE_THRESHOLD | 0.75  |
| WAKE_CONSEC    | 1     |

| Constant       | Value    |
|----------------|----------|
| STOP_THRESHOLD | 0.80     |
| STOP_CONSEC    | 1        |
| STOP_WINDOW    | 6 frames |

## B. Data published with this paper

- **Dataset composition:** sample counts by class and split, as counted on 2026-10-21.
- **Live classification:** the three raw responses from the platform, with every class's confidence score and the sample's metadata.
- **Validation metrics:** the accuracy, confusion matrix and F1 scores transcribed from the training page.
- **Calculated values:** every interval, bound and share quoted in this paper, with its formula.

## C. Platform screenshots

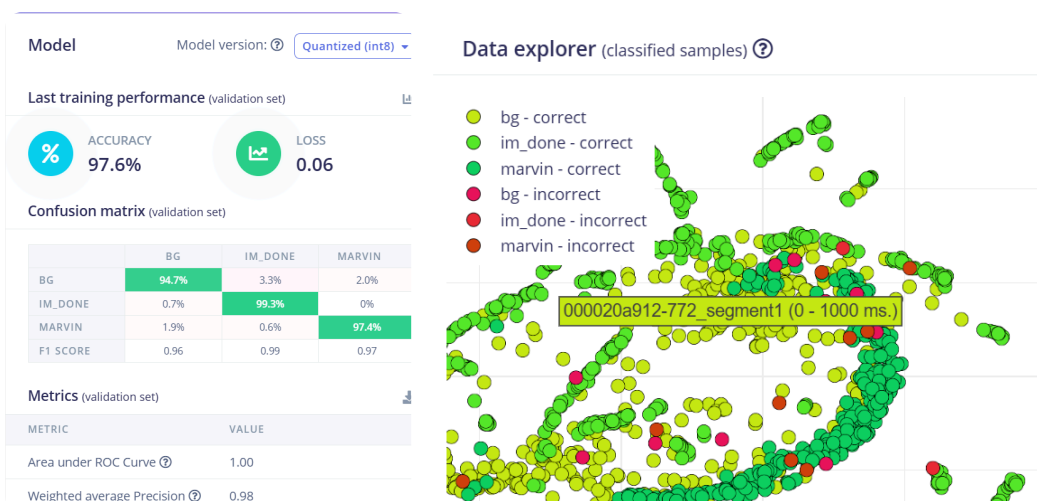


**Figure 9** Left: the training-output page for the int8 model, the source of Table 5.3 and Figure 5. Right: the data explorer from the same page, which places samples by feature similarity and colours them by class and by whether the model classified them correctly.

## D. Revision history

All versions by Ritesh Bonthalakoti.

| Version | Date       | Changes                                                                                                                                                                                                                                                                                                                                                                                                              |
|---------|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| v1.0    | 2026-10-21 | First published version, reviewed by Bharat Meda. Released as Public                                                                                                                                                                                                                                                                                                                                                 |
| v0.6    | 2026-10-21 | Retitled. Detection path, audio hardware and the three sources of evidence described. Dataset composition and provenance by class added. Validation accuracy given with its 95 % interval; accuracy bounds for the live classifications and the field check, and false wakes per hour, added. Objectives, TRL assessment and the path to TRL 4 added. Configuration and platform screenshots moved to the appendices |

| Version | Date       | Changes                                                                                                            |
|---------|------------|--------------------------------------------------------------------------------------------------------------------|
| v0.5    | 2026-10-19 | Re-scoped to the two Edge Impulse screenshots; other images moved to the System Overview publication               |
| v0.4    | 2026-10-19 | Open-enclosure photo of the microphone and speaker added                                                           |
| v0.3    | 2026-10-19 | Companion app screenshots added as illustrative context                                                            |
| v0.2    | 2026-10-19 | Edge Impulse screenshots, transcribed confusion matrix and summary metrics; 97.6 % and 98.7 % discrepancy surfaced |
| v0.1    | 2026-10-19 | First draft, from the product README and the R&D log                                                               |



We build intelligence into everyday objects.

● HOW TO CITE

Ritesh Bonthalakoti. *Wake-word and stop-phrase detection for VAHA: dataset, validation accuracy and first field check*. Dharanova whitepaper, v1.0, 2026.

[dharanova.com](https://dharanova.com) · [info@dharanova.com](mailto:info@dharanova.com)