



FOSDEM'22

beer
open source
lightning talks



devrooms
8000+ hackers
600+ lectures

Online / 5 & 6 February 2022

FOSDEM'22

Car Whispering: the AI Mechanic TinyML Audio Event Detection

Eoin Jordan¹, Dr Martin Serrano², and Pearse Gough³

¹ Data Science Institute, National University of Ireland Galway, Ireland

² Data Science Institute, National University of Ireland Galway, Ireland

³ Google, Ireland



 **Infinitech**



My Background

- ✓ Software & Electronics Engineer 10+ years
- ✓ NUIG DSI (PhD Researcher on semantic technologies supervised by Dr Martin Serrano) 4 Months
- ✓ CoderDojo (Mentor) 4+ years
- ✓ Galway National Park City (Maker Champion) 2+ years
- ✓ RethinkWaste.ie (Co-Founder, Voluntary Director and Product Designer) 3+ Years
- ✓ Driver of 15+ year old cars for longer than I care to say.





FOSDEM '22

Introduction

- Starting my journey with ML and have a history in Electronics
- To make peace with my limitations as a mechanic “Knowing what you don't know”
- Mysterious sounds coming from my car I cannot identify
- Fault Codes were not helping for certain issues
- Advent of TinyML via the TinyML Foundation and the contributions to the movement by TensorFlow Lite (Google), and newer companies lowering the bar to entry such as Edge Impulse.



Aim

- To generate interest in our AI Mechanic project.
- To demonstrate the low barrier into TinyML.
- To also demonstrate an easy route to building prototypes with sensors.
- To gather feedback from the audience on this topic, and how to best start it as an open source hardware software project.

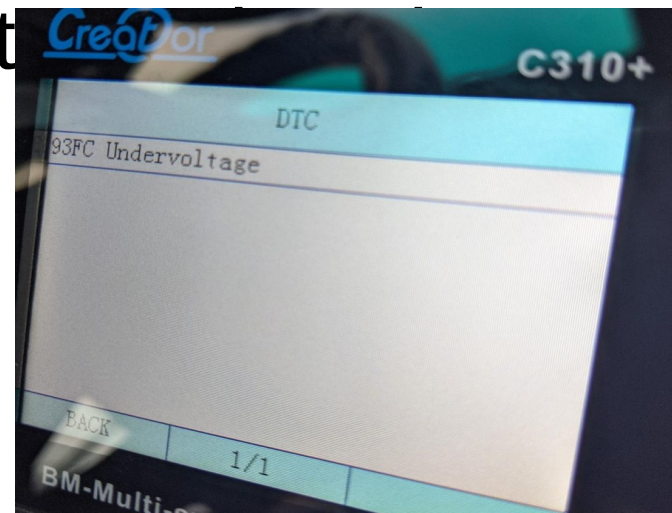




FOSDEM '22

The Problem..

- A little knowledge is a dangerous *thing*.
(a recurring theme today). Lets try to grow ours.
- After fixing an oil pressure issue in my car I introduced a new self oiling feature to the system, and the only error I had was not a



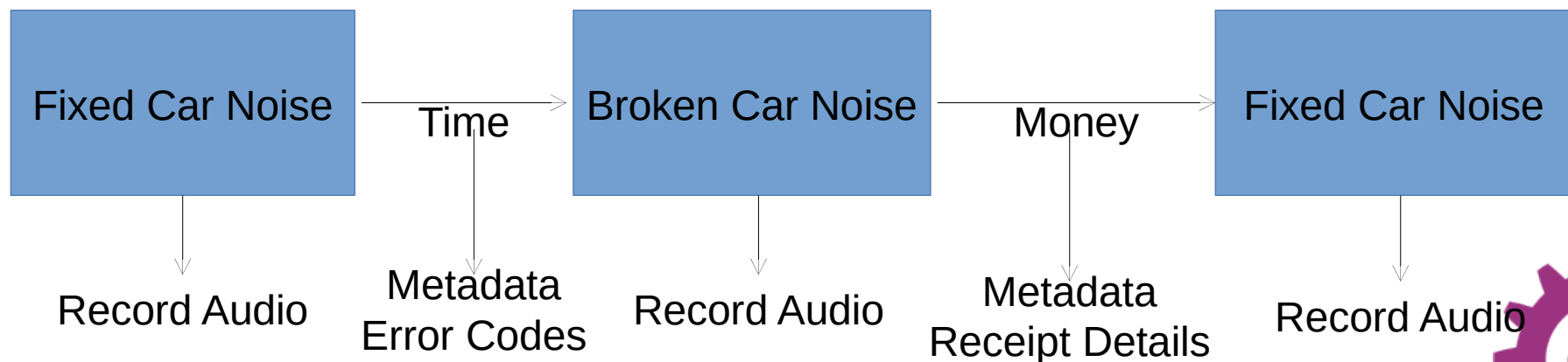


FOSDEM '22

The Solution?

Record the desired state and wait for the next issue to arise (fortunately my car is old enough to provide useful data)

- Allow the professionals to do the work and try to learn from them.
- Capture audio from a car with an issue get it fixed and record the audio of the fixed car.
- Label the data.



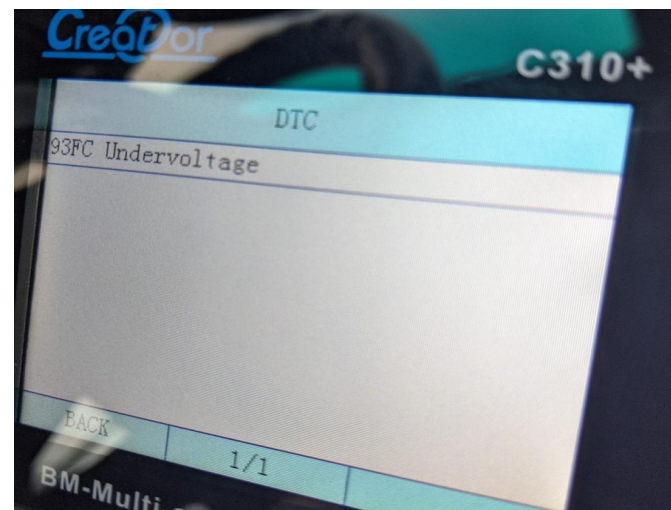
Considering the options

- Cloud based connected via phone hotspot
- TinyML[1] finally brings independent Intelligence to the Edge of the Network



Why on a Microcontroller?

- Inexpensive
- Low Power
- Handheld
- No Connectivity Required
- Accessible and easy to deploy to
- Privacy* ...When not connected to a network



Edge Intelligence Main Challenges^[3]

- Open Architecture
- Modelling and Performance Analysis
- Heterogenous Wireless Networking
- Resource Allocation and Energy Efficiency
- QoS and QoE Provisioning
- Security and Privacy Concerns
- Federation and Cross Platform Service Supply





FOSDEM '22

TinyML 

- “Tiny machine learning is broadly defined as a fast growing field of machine learning technologies and applications including hardware, algorithms and software capable of performing on-device sensor data analytics at extremely low power, typically in the mW range and below, and hence enabling a variety of always-on use-cases and targeting battery operated devices.”^[5]
- “TinyML, it’s both a concept and an organization — and it has acquired significant momentum over the last year or two.”^[4]
- Options Considered:
- TensorFlow Lite
- Automated Workflow for Generation of Models (Edge Impulse^[6] as it has the Apache 2.0 license)



^[4]<https://semiengineering.com/why-tinymml-is-such-a-big-deal/>

^[5]<https://www.tinymml.org/>

^[6]<https://www.edgeimpulse.com/>



FOSDEM '22

Selecting the Hardware

The following development boards are supported by Tensorflow Lite Officially^[6]:

- Arduino Nano 33 BLE Sense^[7]
- SparkFun Edge
- STM32F746 Discovery kit
- Adafruit EdgeBadge
- Adafruit TensorFlow Lite for Microcontrollers Kit
- Adafruit Circuit Playground Bluefruit
- Espressif ESP32-DevKitC
- Espressif ESP-EYE
- Wio Terminal: ATSAMD51
- Himax WE-I Plus EVB Endpoint AI Development Board
- Synopsys DesignWare ARC EM Software Development Platform
- Sony Spresense
- I chose to use:
 - Arduino Tiny Machine Learning Kit (Arduino Nano 33 BLE Sense)
 - M5 Echo Atom (ESP32)



[8]

[7]<https://store.arduino.cc/collections/kits/products/arduino-tiny-machine-learning-kit>

[8]<https://docs.m5stack.com/en/atom/atomecho>

[9]<https://www.tensorflow.org/lite/microcontrollers>

YAMNet and AudioSet

- YAMNet is an audio event classifier suggested in the audio event example on TensorFlow's tutorial, which samples a given audio waveform and makes predictions the given Audio scenarios are described by the AudioSet ontology.
[10]

- This is the most relevant area for my research but may not be of interest widely in the audience. I see myself contributing mainly on this part of the project. Engine Starting, Engine Knocking. Heavy Engine "We estimate this quality"[12].

AudioSet

HOME ONTOLOGY DATASET
DOWNLOAD ABOUT

Dataset >

Heavy engine (low frequency)

The sound of a large, heavy-duty internal combustion engine, such as that which powers a large truck, locomotive, or ship. Jet engines, steam engines, rocket engines etc. are excluded.

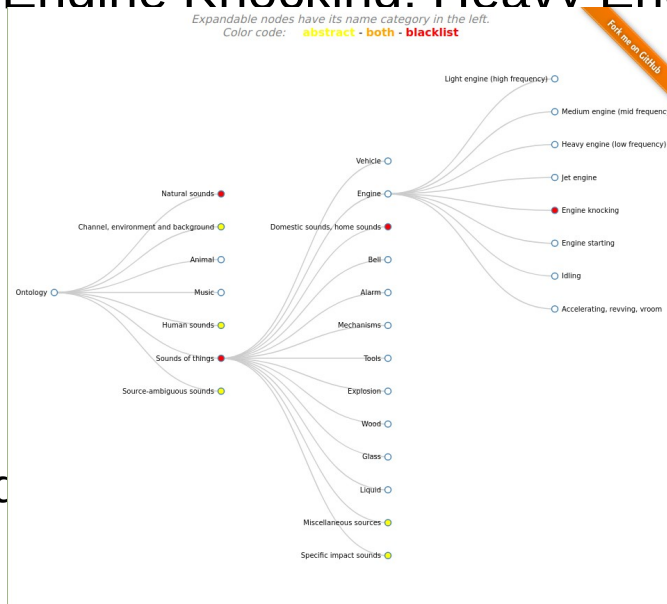
[View full entry in ontology](#)

Dataset	Number of videos	Duration hours
Overall	4,233	11.7
Evaluation	60	0.2
Balanced train	60	0.2
Unbalanced train	4,113	11.4

[Learn more about the dataset parts.](#)

✖ We estimate this class has low quality

In a random sample of videos for this class, we found 2 / 10 (20%) were accurate. Note that quality in the unbalanced training set may be significantly lower. [Learn more.](#)



- Air Leak
- Fly Wheel
- Perished bushing
- Damaged Shock
- Brake Discs
- Wheel Balance

<http://research.google.com/audioset/>

[_low_frequency.html](http://www.jordipons.me/apps/audioset/_low_frequency.html)

<http://www.jordipons.me/apps/audioset/>



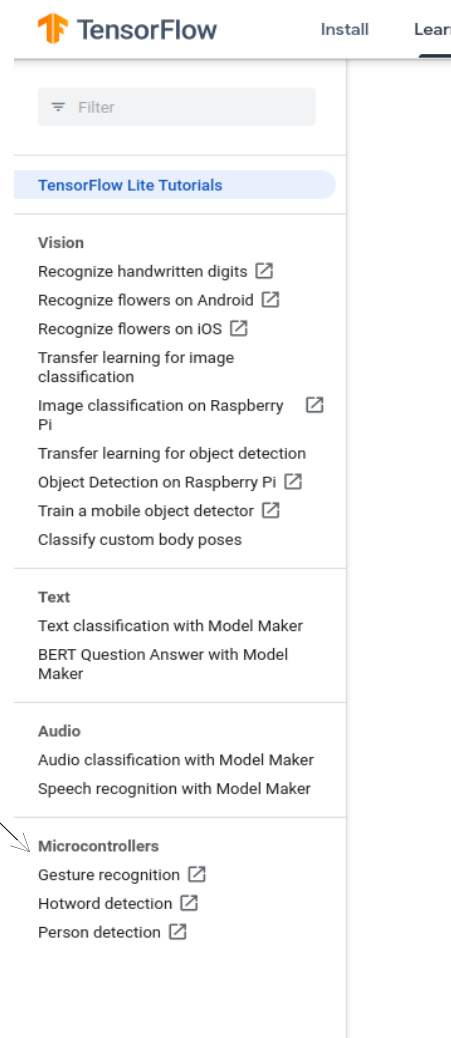
[10] <https://tfhub.dev/google/yamnet/1>

[11] https://www.tensorflow.org/lite/examples/audio_classification/overview?authuser=2

[12] http://research.google.com/audioset/ontology/dataset/heavy_engine_low_frequency.html

Tensorflow Lite Micro

- TensorFlow Lite for Microcontrollers is a port of TensorFlow Lite, designed to run machine learning models on DSPs, microcontrollers and other devices with limited memory.^[14]



[14]https://www.tensorflow.org/lite/microcontrollers#why_microcontrollers_are_important

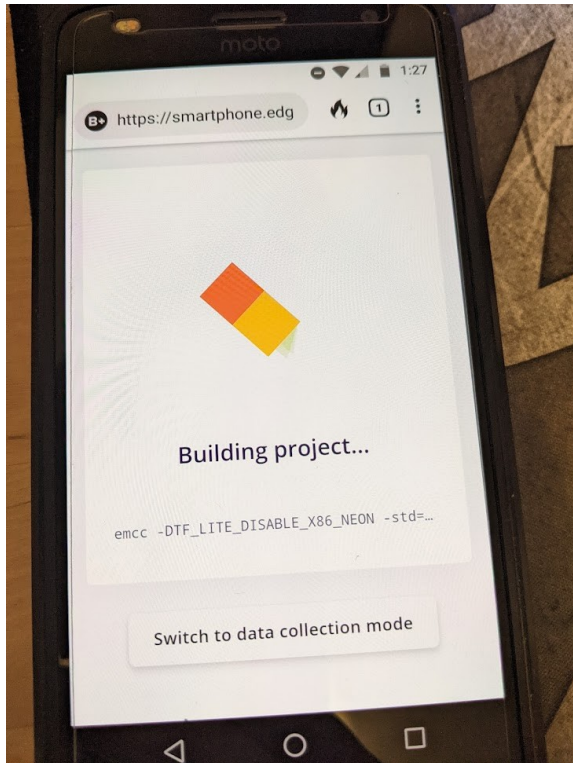
[13]<https://github.com/tensorflow/tflite-micro.git>

[14]https://www.tensorflow.org/lite/microcontrollers#why_microcontrollers_are_important



Capturing some Data

- Recording the good and bad states and annotating them. I used Edge Impulses collection app (which is also open source Apache License 2.0)



SAMPLE NAME	LABEL	ADDED	LENGTH
background noise.2qsjj947	background noise	Feb 03 2022, 14:14:03	3m 18s
Background Noise.2qsj7vbd	Background Noise	Feb 03 2022, 14:07:52	21s
Background Noise.2qsj6mi0	Background Noise	Feb 03 2022, 14:07:11	21s
air leak.2qsj66p9	air leak	Feb 03 2022, 14:06:55	3m 18s
air leak.2qsj4349	air leak	Feb 03 2022, 14:05:45	21s
air leak.2qsj2lmb	air leak	Feb 03 2022, 14:04:59	21s
air leak engine inside cabin.2qsj1...	air leak engine inside cabin	Feb 03 2022, 14:04:18	21s
air leak engine inside cabin.2qsiv...	air leak engine inside cabin	Feb 03 2022, 14:03:15	21s
air leak engine inside cabin.2qsiu...	air leak engine inside cabin	Feb 03 2022, 14:02:34	21s
air leak.2qsitsbp	air leak	Feb 03 2022, 14:02:22	3m 18s
air leak engine inside cabin.2qsis...	air leak engine inside cabin	Feb 03 2022, 14:01:48	21s
air leak engine inside cabin.2qsir...	air leak engine inside cabin	Feb 03 2022, 14:01:05	21s



^[15]<https://smartphone.edgeimpulse.com/classifier.html>

^[16]<https://studio.edgeimpulse.com/studio/69300/learning/keras/7>



FOSDEM '22

Training the neural network

EDGE IMPULSE

- Dashboard
- Devices
- Data acquisition
- Impulse design
 - Create impulse
 - MFE
 - Spectrogram
 - NN Classifier
- EON Tuner
- Retrain model
- Live classification
- Model testing
- Versioning
- Deployment

GETTING STARTED

- Documentation
- Forums

NN CLASSIFIER (AI MECHANIC) Eoin Jordan

#1 Click to set a description for this version

Neural Network settings

Training settings

Number of training cycles 30

Learning rate 0.0005

Validation set size 20 %

Auto-balance dataset ☐

Neural network architecture

Input layer (10,395 features)

Dense layer (20 neurons)

Dense layer (10 neurons)

Add an extra layer

Output layer (7 classes)

Start training

Training output

Model

Model version: Quantized (int8)

Last training performance (validation set)

ACCURACY 61.4% LOSS 1.23

Confusion matrix (validation set)

	BACKGF	AIR LEA	AIR LEA	BACKGF	IDLING	NORMA	OIL CAP
BACKGF	61.5%	0%	38.5%	0%	0%	0%	0%
AIR LEAK	0%	88.3%	11.1%	0.6%	0%	0%	0%
AIR LEAK	0%	0%	100%	0%	0%	0%	0%
BACKGF	0%	96.7%	0%	2.2%	1.1%	0%	0%
IDLING	0%	32.1%	42.9%	0%	25%	0%	0%
NORMAL	0%	0%	42.3%	0%	0%	53.8%	3.8%
OIL CAP C	0%	0%	0%	0%	0%	0%	100%
F1 SCORE	0.76	0.70	0.68	0.04	0.39	0.70	0.96

Feature explorer (full training set)

The feature explorer is only supported when you have a single DSP block.

On-device performance

INFEREN... 38 ms. PEAK RA... 11.8K FLASH US... 221.7K

Add a processing block

DESCRIPTION	AUTHOR	RECOMMENDED
Audio (MFCC) Extracts features from audio signals using Mel Frequency Cepstral Coefficients, great for human voice.	EdgeImpulse Inc.	★ Add
Audio (MFE) Extracts a spectrogram from audio signals using Mel-filterbank energy features, great for non-voice audio.	EdgeImpulse Inc.	★ Add
Flatten Flatten an axis into a single value, useful for slow-moving averages like temperature data, in combination with other blocks.	EdgeImpulse Inc.	Add
Image Preprocess and normalize image data, and optionally reduce the color depth.	EdgeImpulse Inc.	Add
Spectral Analysis Great for analyzing repetitive motion, such as data from accelerometers. Extracts the frequency and power characteristics of a signal over time.	EdgeImpulse Inc.	Add
Spectrogram Extracts a spectrogram from audio or sensor data, great for non-voice audio or data with continuous frequencies.	EdgeImpulse Inc.	Add
Audio (Syntiant) EXPERIMENTAL Syntiant only. Compute log Mel-filterbank energy features from an audio signal.	EdgeImpulse Inc.	Add
Raw Data Use data without pre-processing. Useful if you want to use	EdgeImpulse Inc.	Add



FOSDEM '22

Training the Model

**EDGE IMPULSE**

Dashboard

Devices

Data acquisition

Impulse design

- Create impulse
- MFE
- Spectrogram
- NN Classifier

EON Tuner

Retrain model

Live classification

Model testing

Versioning

Deployment

GETTING STARTED

- Documentation
- Forums

RETRAIN MODEL (AI MECHANIC)

Eoin Jordan

Retrain model with known parameters

MFE

Spectrogram

NN Classifier

Training...

Build output

Cancel

no options or files have changed.

Scheduling job in cluster...
Job started
Reducing dimensions for visualizations...
UMAP(n_components=3, verbose=True)
Construct fuzzy simplicial set
Sat Feb 5 23:19:30 2022 Finding Nearest Neighbors
Still running...
Sat Feb 5 23:19:32 2022 Finished Nearest Neighbor Search
Sat Feb 5 23:19:34 2022 Construct embedding
Still running...
completed 0 / 500 epochs
completed 50 / 500 epochs

© 2022 Edgimpulse Inc. All rights reserved





FOSDEM '22

Accuracy

Training output

Model

Model version: [?](#) Quantized (int8) ▾

Last training performance (validation set)

 ACCURACY
61.4%

 LOSS
1.23

Confusion matrix (validation set)

	BACKGROUND NO	AIR LEAK	AIR LEAK ENGINE	BACKGROUND NO	IDLING	NORMAL ENGINE	OIL CAP OFF ENGINE
BACKGROUND NO	61.5%	0%	38.5%	0%	0%	0%	0%
AIR LEAK	0%	88.3%	11.1%	0.6%	0%	0%	0%
AIR LEAK ENGINE	0%	0%	100%	0%	0%	0%	0%
BACKGROUND NO	0%	96.7%	0%	2.2%	1.1%	0%	0%
IDLING	0%	32.1%	42.9%	0%	25%	0%	0%
NORMAL ENGINE	0%	0%	42.3%	0%	0%	53.8%	3.8%
OIL CAP OFF ENGINE	0%	0%	0%	0%	0%	0%	100%
F1 SCORE	0.76	0.70	0.68	0.04	0.39	0.70	0.96

Feature explorer (full training set) [?](#)

The feature explorer is only supported when you have a single DSP block.

On-device performance [?](#)

 INFERRENCING TIME
38 ms.

 PEAK RAM USAGE
11.8K

 FLASH USAGE
221.7K





FOSDEM '22

Implementing the Code

```
nano_ble33_sense_microphone_RGB_LCD | Arduino 1.8.16
File Edit Sketch Tools Help
nano_ble33_sense_microphone_RGB_LCD
#define RED 22
#define BLUE 24
#define GREEN 23
#define LED_PWR 25

// If your target is limited in memory remove this macro to save 10K RAM
#define EIDSP_QUANTIZE_FILTERBANK 0

/* Includes ----- */
#include <PDM.h>
#include <Wire.h>
#include "rgb_lcd.h"

rgb_lcd lcd;

/** Audio buffers, pointers and selectors */
typedef struct {
    int16_t *buffer;
    uint8_t buf_ready;
    uint32_t buf_count;
    uint32_t n_samples;
} inference_t;

static inference_t inference;
static signed short sampleBuffer[2048];
static bool debug_nn = false; // Set this to true to see e.g. features generated from the raw signal

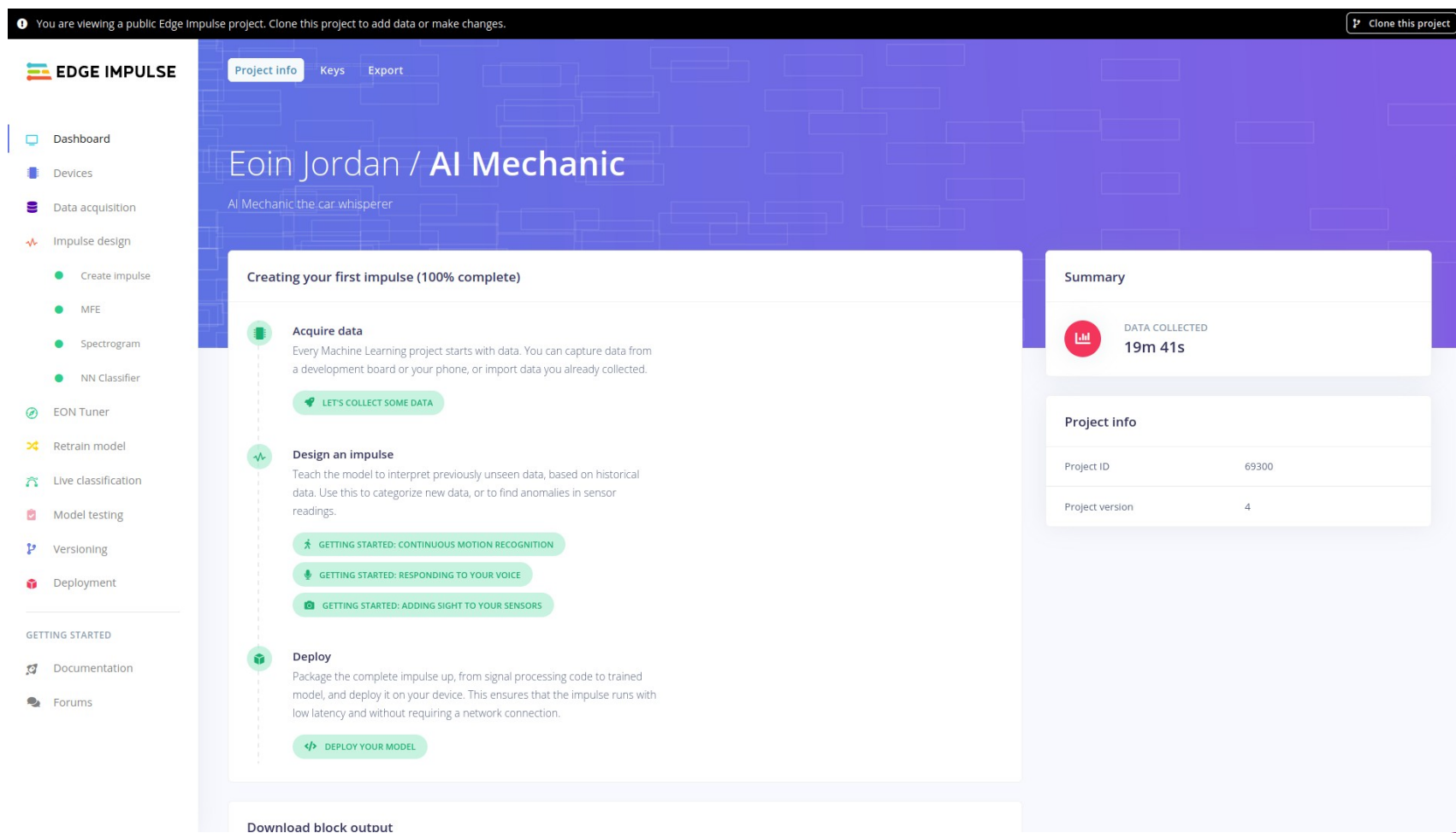
/**
 * @brief Arduino setup function
 */
void setup()
{
    // initialize the digital Pin as an output
    pinMode(RED, OUTPUT);
    pinMode(BLUE, OUTPUT);
    pinMode(GREEN, OUTPUT);
    pinMode(LED_PWR, OUTPUT);
    // put your setup code here, to run once:
    Serial.begin(115200);
    // set up the LCD's number of columns and rows:
    lcd.begin(16, 1);
    // Print a message to the LCD.
    lcd.print("hello, fosdem!");

    // summary of inferencing settings (from model_metadata.h)
    ei_printf("Inferencing settings:\n");
    ei_printf("\tInterval: %.2f ms.\n", (float)EI_CLASSIFIER_INTERVAL_MS);
    ei_printf("\tFrame size: %d\n", EI_CLASSIFIER_DSP_INPUT_FRAME_SIZE);
    ei_printf("\tSample length: %d ms.\n", EI_CLASSIFIER_RAW_SAMPLE_COUNT / 16);
    ei_printf("\tNo. of classes: %d\n", sizeof(ei_classifier_inferencing_categories) / sizeof(ei_classifier_inferencing_categories[0]));

    if (microphone_inference_start(EI_CLASSIFIER_RAW_SAMPLE_COUNT) == false) {
        ei_printf("ERR: Failed to setup audio sampling\r\n");
        return;
    }
}
```



Sharing the Model



The screenshot displays the Edge Impulse Studio web interface. At the top, a black banner contains a notification: "You are viewing a public Edge Impulse project. Clone this project to add data or make changes." with a "Clone this project" button. The left sidebar features the "EDGE IMPULSE" logo and a navigation menu with options: Dashboard, Devices, Data acquisition, Impulse design (with sub-items: Create impulse, MFE, Spectrogram, NN Classifier), EON Tuner, Retrain model, Live classification, Model testing, Versioning, and Deployment. Below these are "GETTING STARTED" links for Documentation and Forums. The main content area has a blue header with "Project info", "Keys", and "Export" tabs. The project name "Eoin Jordan / AI Mechanic" is prominently displayed, with the subtitle "AI Mechanic the car whisperer". The central section, titled "Creating your first impulse (100% complete)", outlines three steps: 1. "Acquire data" (with a "LET'S COLLECT SOME DATA" button), 2. "Design an impulse" (with three "GETTING STARTED" buttons: "CONTINUOUS MOTION RECOGNITION", "RESPONDING TO YOUR VOICE", and "ADDING SIGHT TO YOUR SENSORS"), and 3. "Deploy" (with a "DEPLOY YOUR MODEL" button). On the right, a "Summary" box shows "DATA COLLECTED 19m 41s". Below it, a "Project info" table lists "Project ID: 69300" and "Project version: 4". At the bottom, a "Download block output" button is visible.

You are viewing a public Edge Impulse project. Clone this project to add data or make changes. [Clone this project](#)

EDGE IMPULSE

- Dashboard
- Devices
- Data acquisition
 - Create impulse
 - MFE
 - Spectrogram
 - NN Classifier
- EON Tuner
- Retrain model
- Live classification
- Model testing
- Versioning
- Deployment

GETTING STARTED

- Documentation
- Forums

Project info Keys Export

Eoin Jordan / AI Mechanic

AI Mechanic the car whisperer

Creating your first impulse (100% complete)

- Acquire data**
Every Machine Learning project starts with data. You can capture data from a development board or your phone, or import data you already collected.
[LET'S COLLECT SOME DATA](#)
- Design an impulse**
Teach the model to interpret previously unseen data, based on historical data. Use this to categorize new data, or to find anomalies in sensor readings.
[GETTING STARTED: CONTINUOUS MOTION RECOGNITION](#)
[GETTING STARTED: RESPONDING TO YOUR VOICE](#)
[GETTING STARTED: ADDING SIGHT TO YOUR SENSORS](#)
- Deploy**
Package the complete impulse up, from signal processing code to trained model, and deploy it on your device. This ensures that the impulse runs with low latency and without requiring a network connection.
[DEPLOY YOUR MODEL](#)

Summary

DATA COLLECTED
19m 41s

Project info

Project ID	69300
Project version	4

[Download block output](#)





FOSDEM '22

Deploying the Model

Grove: "Grove is a modular, standardized connector prototyping system. Grove takes a building block approach to assembling electronics."

Compared to the jumper or solder based system, it is easier to connect, experiment and build and simplifies the learning system, but not to the point where it becomes dumbed down.

Some of the other prototype systems out there takes the level down to building blocks.

Good stuff to be learned that way, but the Grove system allows you to build real systems.

It requires some learning and expertise to hook things up."



Seeed Studio have great resources and library support for their models with useful examples.

Supply Chain withstanding you may need to search in multiple stores for your parts regardless.

I recommend avoiding cloned devices for your comfort building as drivers can cause

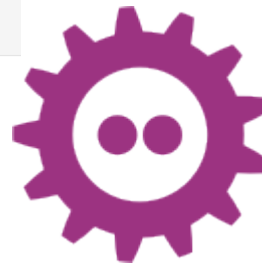
issues on windows. Use reputable electronics stores (See references, please suggest

any you may know). To avoid supporting potentially dangerous work environments

for their producers, and help future developments.

☐	 Enlarge	Mfr. Part No. 101020452 Mouser Part No 713-101020452	Seeed Studio	Display Development Tools Grove - OLED Display 1.12" V2	Datasheet	169 In Stock	1: €11.95
☐	 Enlarge	Mfr. Part No. 104020250 Mouser Part No 713-104020250	Seeed Studio New Product	Display Development Tools Grove - OLED Display 1.12 (SH1107) V2.0 - SPI/IIC - 3.3V/5V	Learn More	16 In Stock	1: €10.94
☐	 Enlarge	Mfr. Part No. 104020248 Mouser Part No 713-104020248	Seeed Studio New Product	OLED Displays & Accessories Grove - OLED Display 0.66 (SSD1306), V1.0	Learn More	38 In Stock 30 On Order View Dates	1: €4.33
☐	 Enlarge	Mfr. Part No. 104020129 Mouser Part No 713-104020129	Seeed Studio	Display Development Tools Grove - Triple Color E-Ink Display 1.54"	Learn More	107 In Stock	1: €19.64
☐	 Enlarge	Mfr. Part No. 104020130 Mouser Part No 713-104020130	Seeed Studio	Display Development Tools Grove - Triple Color E-Ink Display 2.13"	Datasheet Learn More	102 In Stock	1: €16.95

3.3v compatible grove displays:
<https://www.mouser.ie/c/?q=grove%20display>





FOSDEM '22

Edge Intelligence - Live Demo..





FOSDEM '22

Testing Via Edge Impulse



[20]<https://studio.edgeimpulse.com/public/69300/latest>

[21]<https://docs.edgeimpulse.com/docs/running-your-impulse-arduino>



FOSDEM '22

Future Work

- Replace the display with a 3.3v one, parts ordered already.
- Combine Error Codes from OBD scanner to tag audio collected in real time by collecting via a streaming service.
- Share collected events via a network of users, interested in suggestions of how to implement that??
- Extend the AudioSet Ontology to include more mechanical terms. (Relevant to my research)
- Perform reasoning on the minified AI Mechanic Ontology
- Monitor Device Battery Performance
- Seeking contributors: <https://github.com/eoinjordan/CarWhispering>
- More target MCUs: BBC:Micro(My sons), ESP32-C3 (RISC-V), and of course the RP2040:



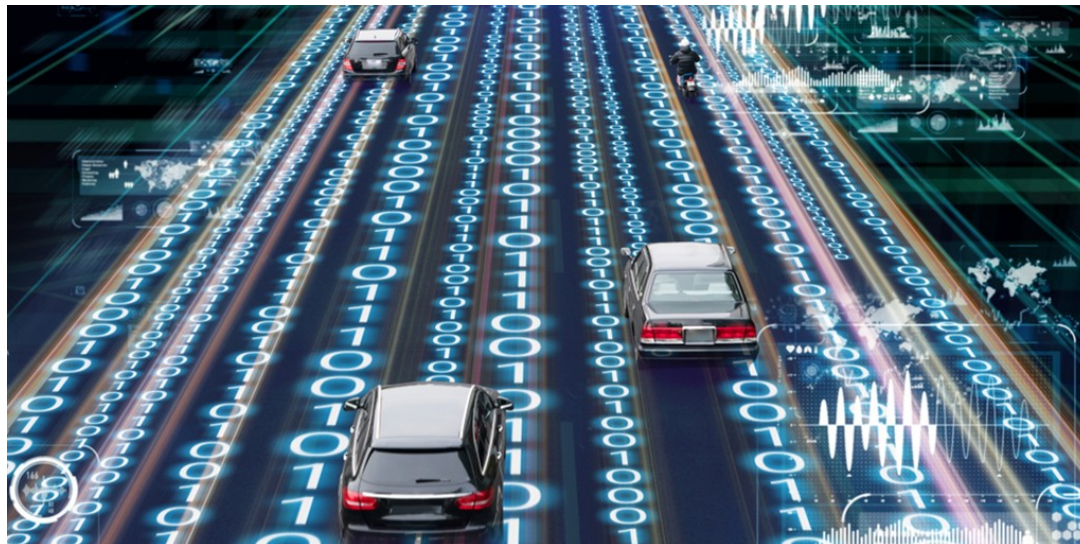


FOSDEM '22

Future Work on i3-MARKET



- Automotive Data is yet a potential area for further exploration.
- Automotive Data Owners may want to share the collected data if there are incentives to do so.
- Data Sharing model by selling data sets via Data Marketplaces can act as an incentive.
- Creation and Support of Automotive Data Space is yet a requirement we will explorer.
- Perform Data Cleaning and Define Data Formats for Sharing Automotive Collected Data is important
- AI Edge Intelligence will benefit from more Available End User Automotive Data.
- Seeking contributors: <https://github.com/eoinjordan/CarWhispering>



Future Work on INFINITECH

- Detect a coercive event via a **microphone** and embedded model on an **MCU**
- Call an api via the **Modbus** or other secure wired system communication on the ATM that signals an investigation needs to take place on that persons account for possible human trafficking
- TAH external system is also triggered for *redflag indicators*^[22] in the area.

How do cash machines work?

Find out how ATMs access your money, while keeping your details safe

Keypad
The keypad is where users enter their four-digit PIN. This is then sent to the cardholder's bank in an encrypted form in case someone intercepts it.

Cash dispenser
Once the transaction is complete, the requested amount of cash will travel up from the vault through the roller mechanism to the dispenser, where it can be collected.

Receipt slot
Your transaction details are printed on thermal paper, which uses heat rather than ink to turn the paper black and form the necessary text.

"The magnetic strip on the back of your card is actually composed of millions of tiny magnets"

Rollers
The rollers check the thickness of each note to make sure that two notes aren't stuck together, so the cardholder receives the correct amount of money.

Suction cups
The suction cups pick up notes individually, before the rollers process them.

Screen
The screen, commonly an LCD on modern machines, will prompt the cardholder through the transaction in a step-by-step format. It will either be touch screen or have buttons on either side.

Card slot
Once a card has been entered, its black magnetic stripe is read to authenticate the card and its owner.

Reject box
If the machine pulls the wrong number of notes or detects that one is damaged, they will be dropped into a reject box and the process will be repeated.

Deposit box
Once cheques or deposits are made they are stored in the deposit box, where the bank will collect them at a later stage.

Cash chambers
Each chamber, or currency box, may hold thousands of notes. Most ATMs will stock each denomination depending on the currency used in its location.

HOW IT WORKS



<https://www.howitworksdaily.com/how-do-cash-machines-work/>

<https://www.howitworksdaily.com/how-do-cash-machines-work/>^[23]

<https://www.traffikanalysis.org/>^[22]

<https://www.infinitech-h2020.eu/>^[24]





FOSDEM '22

Conclusions

- AI has become more accessible
- AI at the Edge or Edge Intelligence make for makers with TinyML(TensorFlow Lite), and everyone via intuitive tools such as Edge Impulse is a possibility
- Offline enhancements to existing erroring systems are now possible to implement cheaply.
- It is now possible to build a device that can give us a pocket car whispering mechanic. Very exciting!

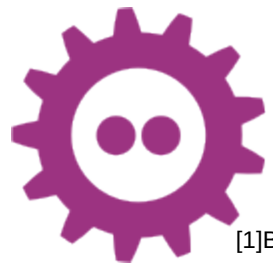


 Infinitech



Questions and Comments

- Thank you so much for your attention!
- Accepting contributors from all areas:
- Mechanics, automation, build, electronics, graphics, documentation, test, development, ML, no qualifications necessary, beginners, novice drivers or otherwise all are welcomed with open arms:
 - <https://github.com/eoinjordan/CarWhispering>



FOSDEM '22 References

- [1] Bryon Moyer., "Why TinyML Is Such A Big Deal" September 2nd, 2021. Accessed: Jan. 04, 2022. [Online]. Available: <https://semiengineering.com/why-tinyml-is-such-a-big-deal/>
- [2] https://www.tensorflow.org/lite/microcontrollers#why_microcontrollers_are_important
- [3] J. C. Nobre et al., "Vehicular software-defined networking and fog computing: Integration and design principles," *Ad Hoc Networks*, vol. 82, pp. 172–181, Jan. 2019, doi: 10.1016/J.ADHOC.2018.07.016.
- [4] <https://semiengineering.com/why-tinyml-is-such-a-big-deal/>
- [5] <https://www.tinyml.org/>
- [6] <https://www.edgeimpulse.com/>
- [7] <https://store.arduino.cc/collections/kits/products/arduino-tiny-machine-learning-kit>
- [8] <https://docs.m5stack.com/en/atom/atomecho>
- [9] <https://www.tensorflow.org/lite/microcontrollers>
- [10] <https://tfhub.dev/google/yamnet/1>
- [11] https://www.tensorflow.org/lite/examples/audio_classification/overview?authuser=2
- [12] http://research.google.com/audioset/#!/dataset/heavy_engine_low_frequency.html
- [13] <https://github.com/tensorflow/tflite-micro.git>
- [14] https://www.tensorflow.org/lite/microcontrollers#why_microcontrollers_are_important
- [15] <https://smartphone.edgeimpulse.com/classifier.html>
- [16] <https://studio.edgeimpulse.com/studio/69300/learning/keras/7>
- [Recording the good and bad states and annotating them. I used Edge Impulses collection app (which is also open source Apache License 2.0)
<https://smartphone.edgeimpulse.com/classifier.html>
- [17] <https://studio.edgeimpulse.com/studio/69300/learning/keras/7>
- [18] <https://github.com/eoinjordan/CarWhispering>
- [19] https://wiki.seeedstudio.com/Grove_System/
- [20] <https://studio.edgeimpulse.com/public/69300/latest>
- [21] <https://docs.edgeimpulse.com/docs/running-your-impulse-arduino>



 Infinitech



Transfer Learning

- Audio events detected in other models can help identify similar types in another e.g. engine sounds used in the type of vechical may help identify my idling one.

https://colab.research.google.com/github/tensorflow/tensorflow/blob/master/tensorflow/lite/g3doc/tutorials/model_maker_audio_classification.ipynb?authuser=2#scrollTo=wbMc4vHjaYdQ

https://www.tensorflow.org/tutorials/audio/transfer_learning_audio?authuser=2

