tinyML® Summit

Miniature dreams can come true…

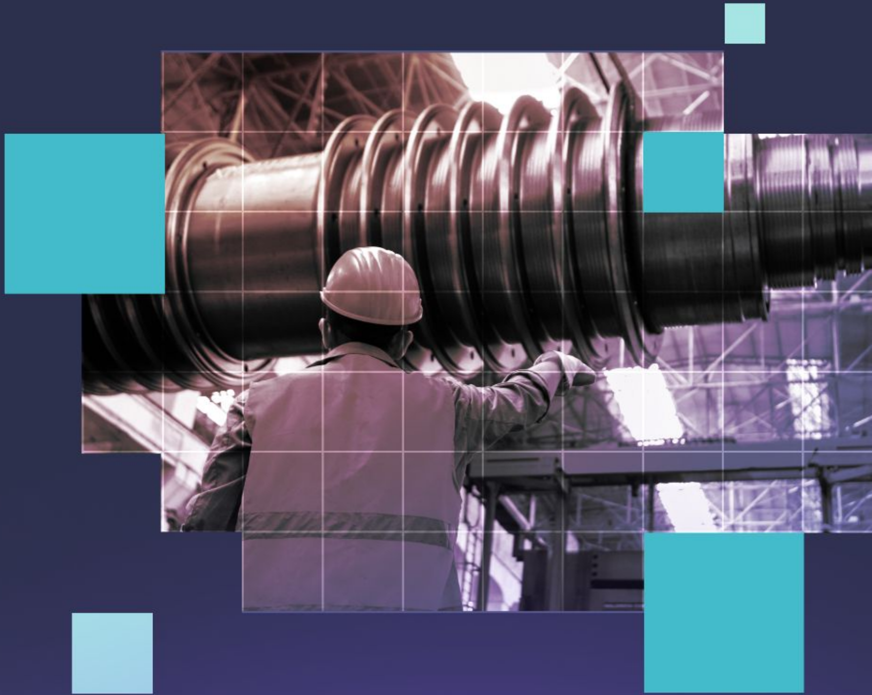
March 28-30, 2022 | San Francisco Bay Area

TINY ML

www.tinyML.org

EDGE IMPULSE

Building data-centric AI tooling for embedded engineers



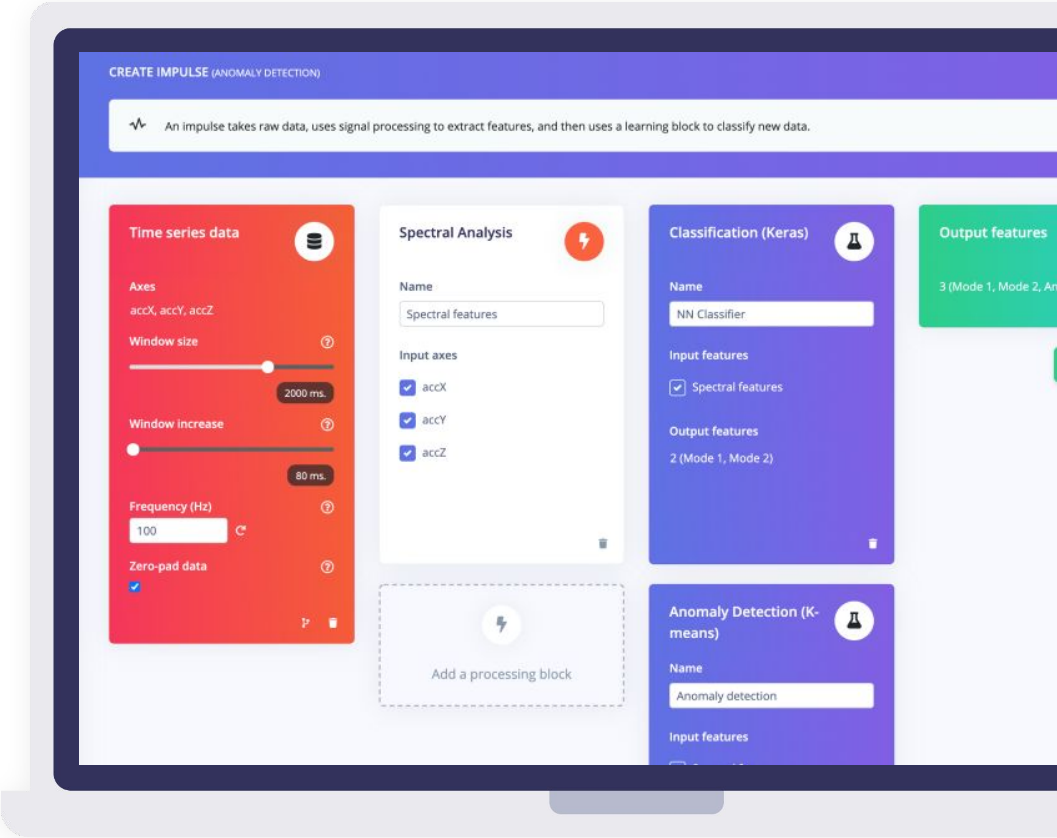
Daniel Situnayake (@dansitu)
Founding TinyML Engineer

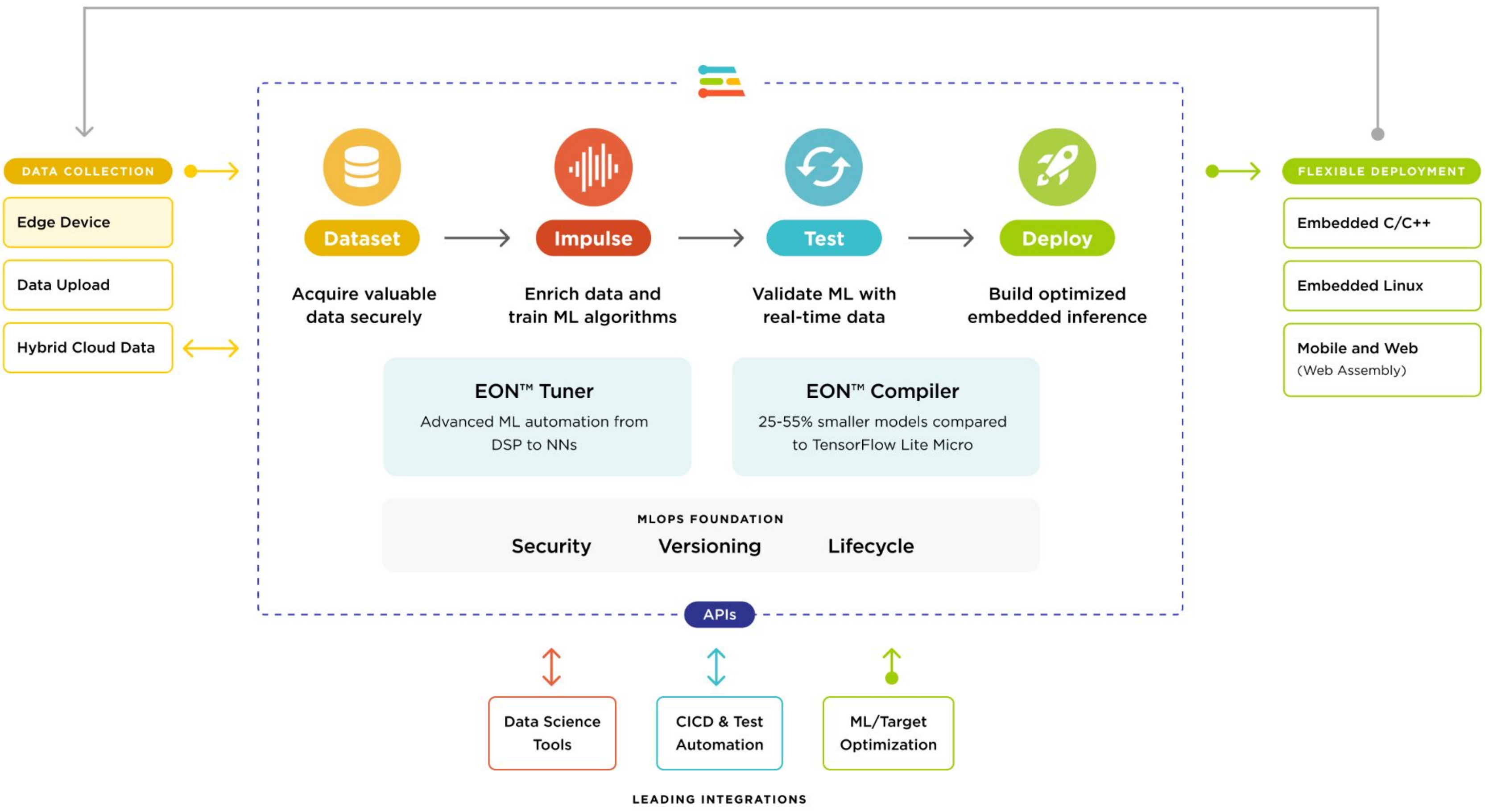
It’s good to see you!

- I lead the ML team at **Edge Impulse**
- Previously on TFLM @ **Google**
- Co-authored **TinyML** with Pete Warden
- Co-writing **AI at the Edge** with Jenny Plunkett ✍🏽

How our product works

- Engineering toolkit for embedded machine learning and AI
- MLOps reimagined for embedded hardware
- End-to-end integration with devices and data science tools




DATA COLLECTION
Edge Device
Data Upload
Hybrid Cloud Data
Dataset
Acquire valuable data securely
Impulse
Enrich data and train ML algorithms
Test
Validate ML with real-time data
Deploy
Build optimized embedded inference
EON™ Tuner
Advanced ML automation from DSP to NNs
EON™ Compiler
25-55% smaller models compared to TensorFlow Lite Micro
MLOPS FOUNDATION
Security
Versioning
Lifecycle
APIs
FLEXIBLE DEPLOYMENT
Embedded C/C++
Embedded Linux
Mobile and Web
(Web Assembly)
Data Science Tools
CICD & Test Automation
ML/Target Optimization
LEADING INTEGRATIONS

Used in production every day

40,000+
Developers

70,000+
Projects

1,000+
Enterprises

TRUSTED BY LEADING ENTERPRISES

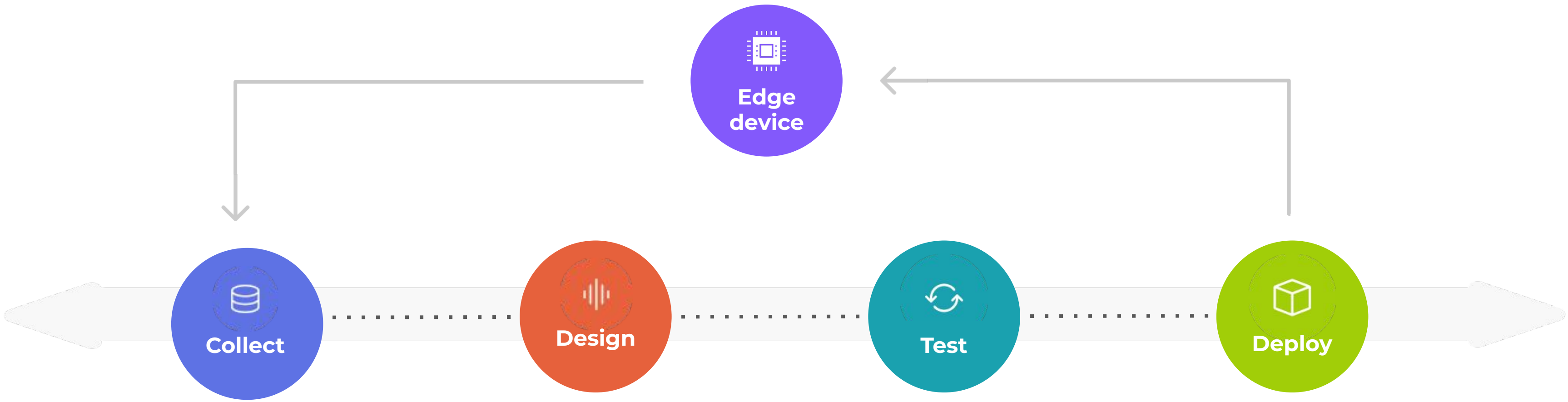
ŌURA ADVANTECH poly NASA SONY

1. Feedback loops
2. Data-centric AI
3. Rapid prototyping

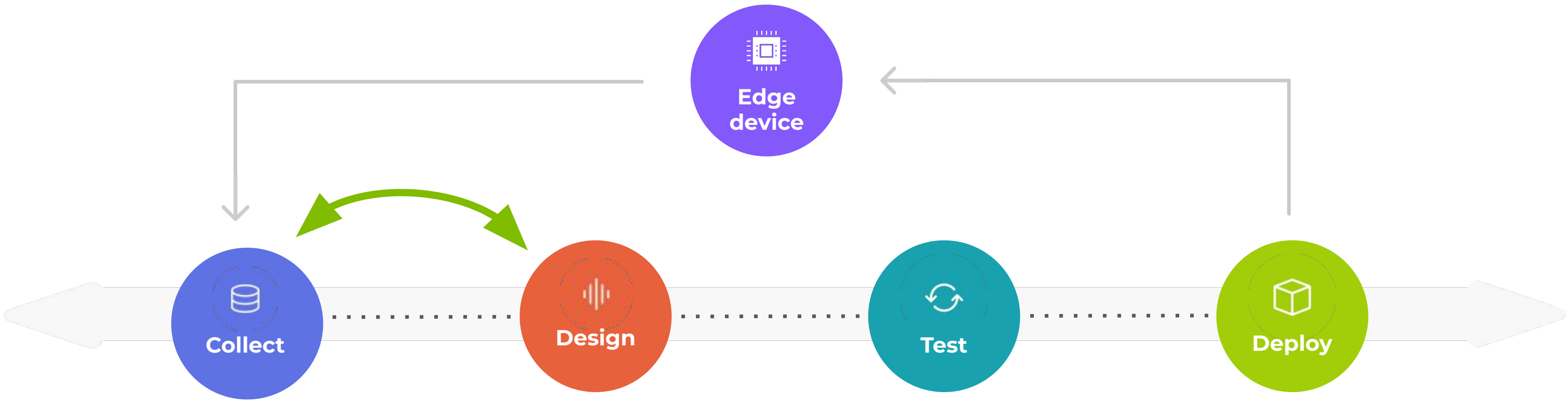
Feedback loops

Continuous improvement

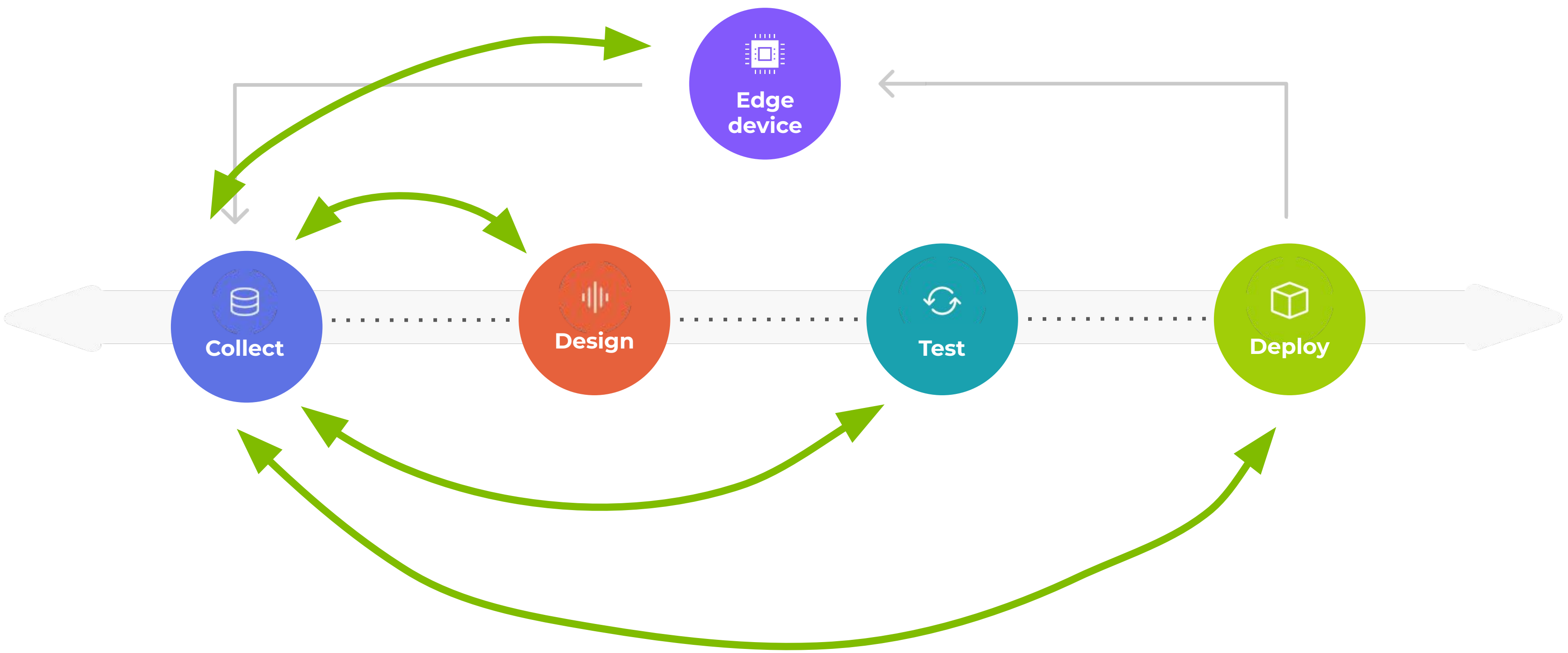
The edge ML workflow



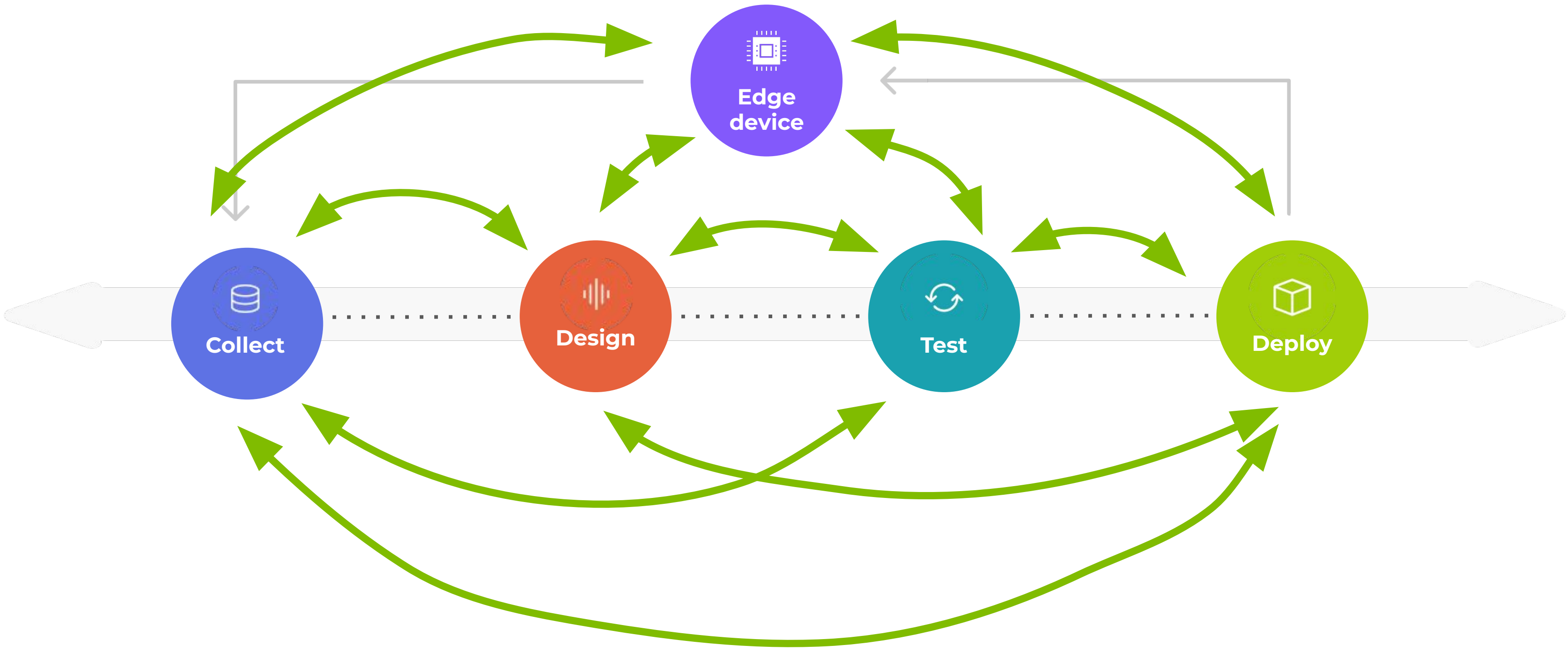
The edge ML workflow: Feedback loops



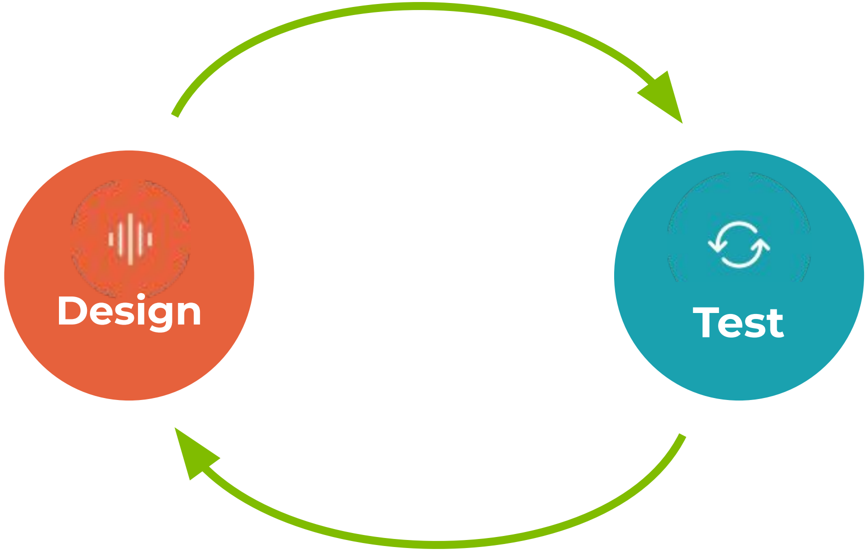
The edge ML workflow: Feedback loops



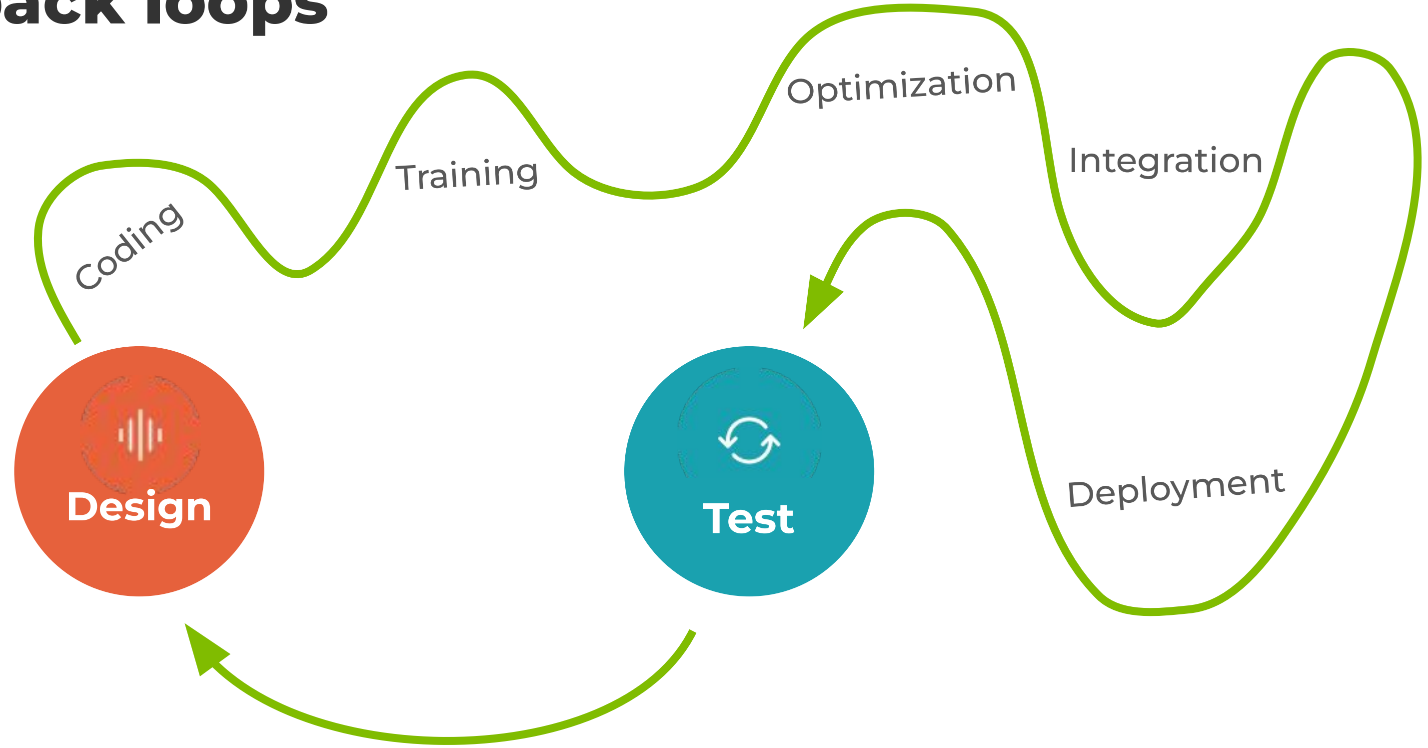
The edge ML workflow: Feedback loops



Tight feedback loops



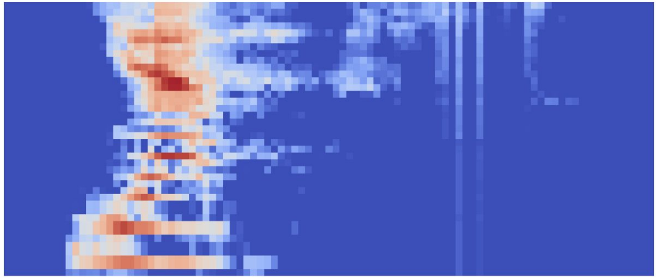
Tight feedback loops



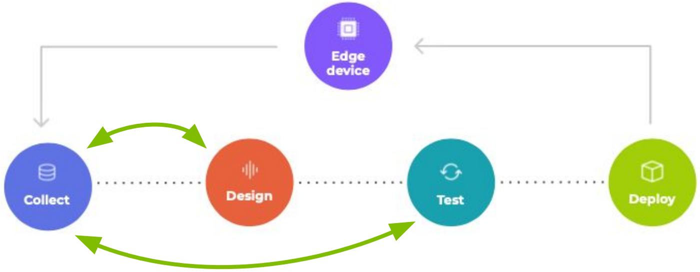
Tight loops: Reduced data requirements

Transfer learning for few-shot keyword spotting

- Feature extractor trained on Multilingual Spoken Words corpus (23.4M samples)
- Train a KWS model with 5 examples
- https://www.seas.harvard.edu/news/2021/12/voice-technology-rest-world



Harvard John A. Paulson School of Engineering and Applied Sciences



Tight loops: Low training time

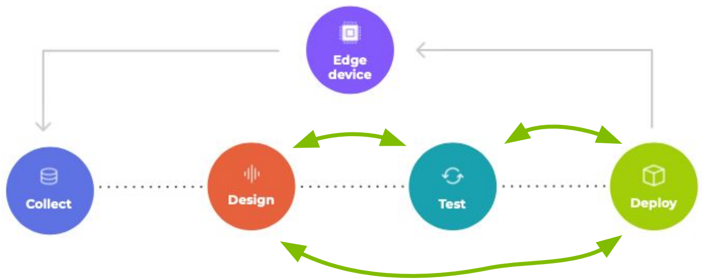
Auto-scaling cloud infrastructure

- Kubernetes-based jobs system
- Automatic scaling in AWS and Azure clouds
- Distributed batch processing
- GPU acceleration on-demand

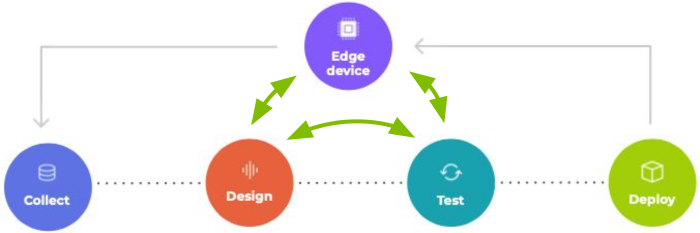
kubernetes

aws

Azure



Tight loops: EON Tuner



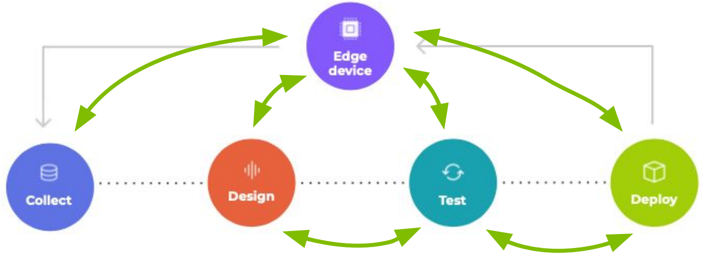
Co-optimization of DSP and ML

- Automatic hyperparameter search
- Estimates performance metrics using hybrid simulation and benchmarking
- Loss function incorporates target specs (RAM/ROM/Latency)

Tight loops: Performance calibration

Test and tune on real world data

- Models real world performance on streaming data (FAR/FRR)
- Uses GA for multi-objective optimization in post-processing configuration space
- Pareto front plotted; selected config integrated into model



Data-centric AI

Great data, great performance

01000101 01100100 01100111 01100101 00100000 01001001 01101101 01110000 01110101
01101100 01110011 01100101 00100000 01110010 01101111 01100011 01101011 01110011

Traditional AI

The focus is on this part

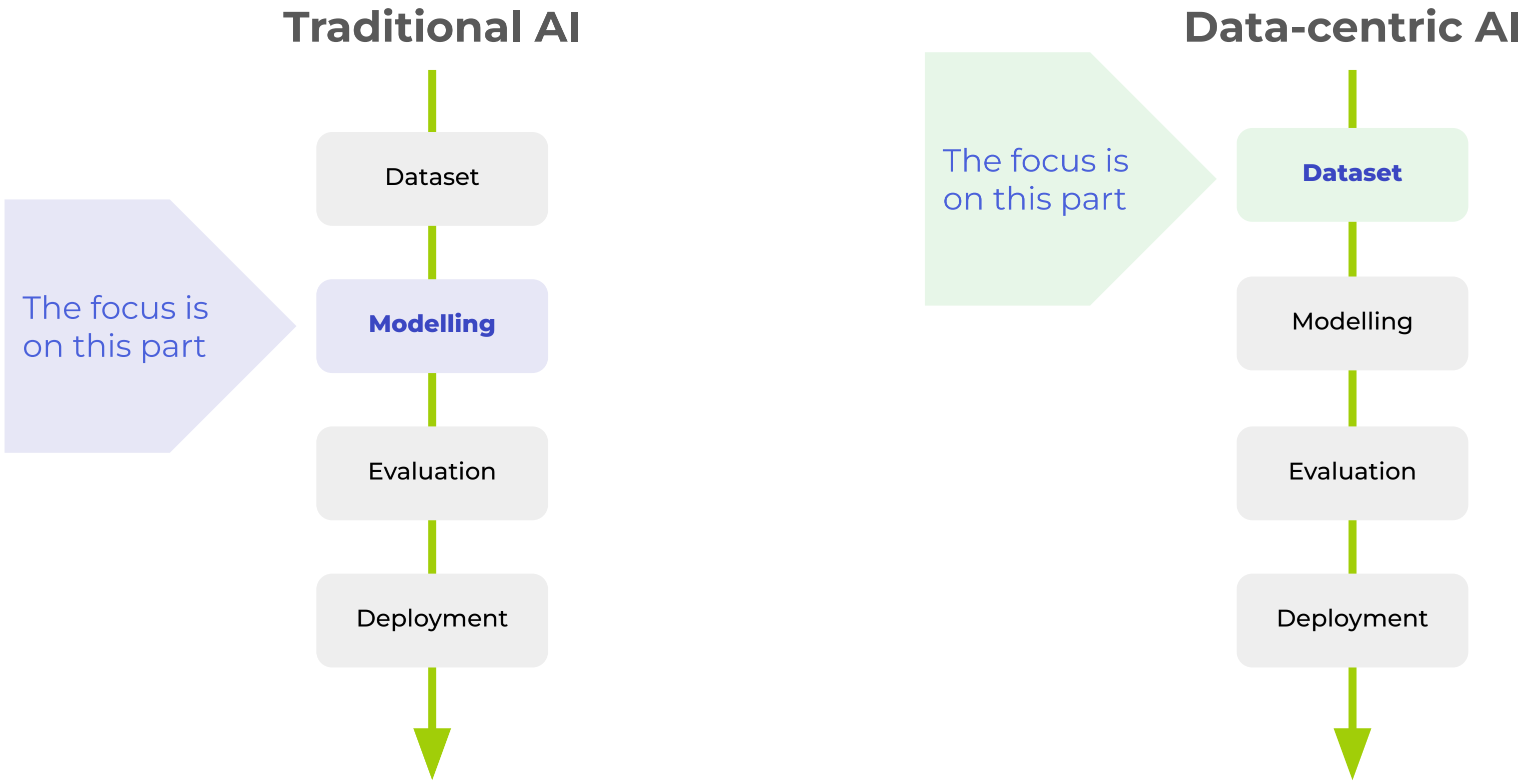
Dataset

Modelling

Evaluation

Deployment

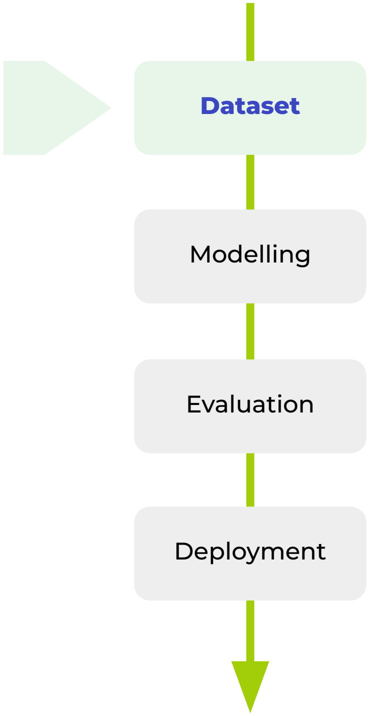
How do we train the best model for this dataset?


Traditional AI
Dataset
The focus is on this part
Modelling
Evaluation
Deployment
Data-centric AI
The focus is on this part
Dataset
Modelling
Evaluation
Deployment

Data-centric AI

How do we create the best dataset for this problem?

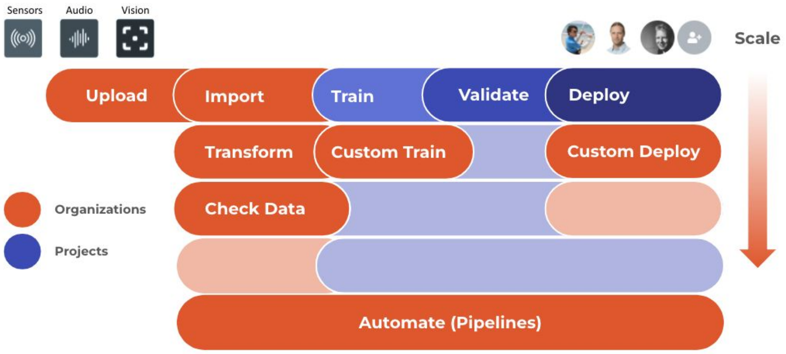
- Focus on dataset quality and data collection
- Downstream benefits to modelling, evaluation, and real world performance
- “Garbage in, garbage out”



Data-centric: Ingestion, cleaning & transformation

Data pipeline definition

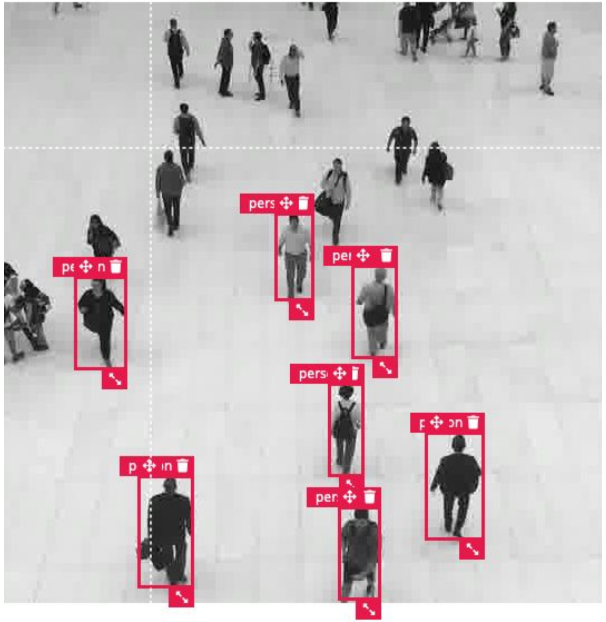
- Specify data transformations in Docker containers (file in, file out)
- Develop locally using preferred data science tools
- Distributed batch processing using cloud compute



Data-centric: Assisted labelling

Better labels for better performance

- Import an unlabelled dataset via API or cloud bucket
- Attempt to label using model
  - Tracking (OpenCV CSRT)[1]
  - Imagenet/COCO/Megadetector
  - User trained model (not-quite-active-learning)



[1]Alan Lukezic, Tom'as Voj'ir, Luka Cehovin Zajc, Jir'i Matas, and Matej Kristan. Discriminative correlation filter tracker with channel and spatial reliability. *International Journal of Computer Vision*, 2018.

Data-centric: Dataset exploration

Dimensionality reduction for feature visualization

- Uses Uniform Manifold Approximation and Projection (UMAP)

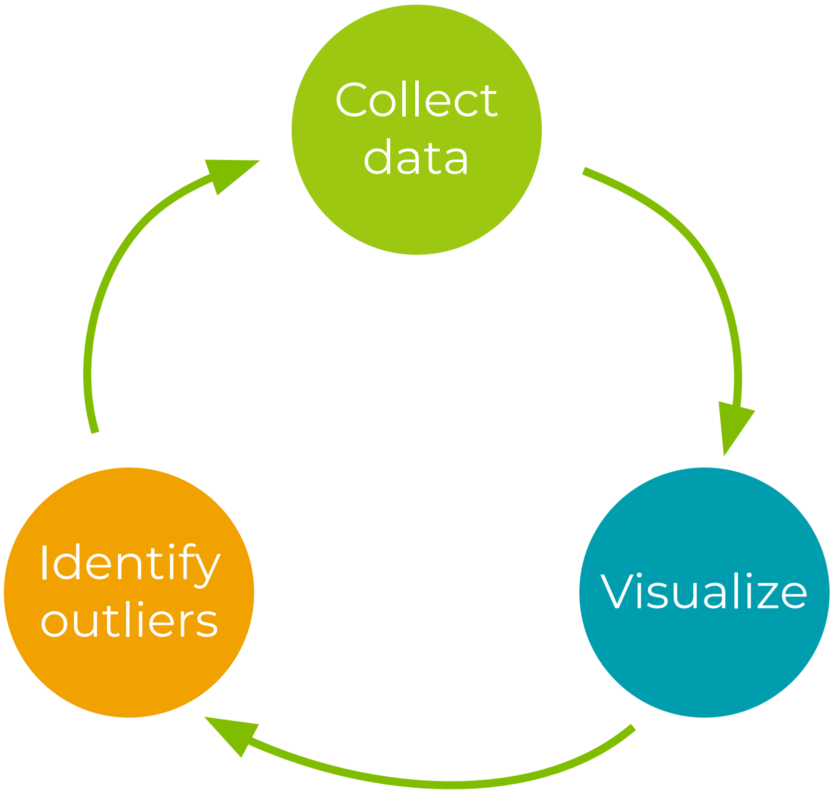
- Highly effective for processed signal (e.g. after feature engineering), less so with raw data

Data-centric: Dataset exploration II

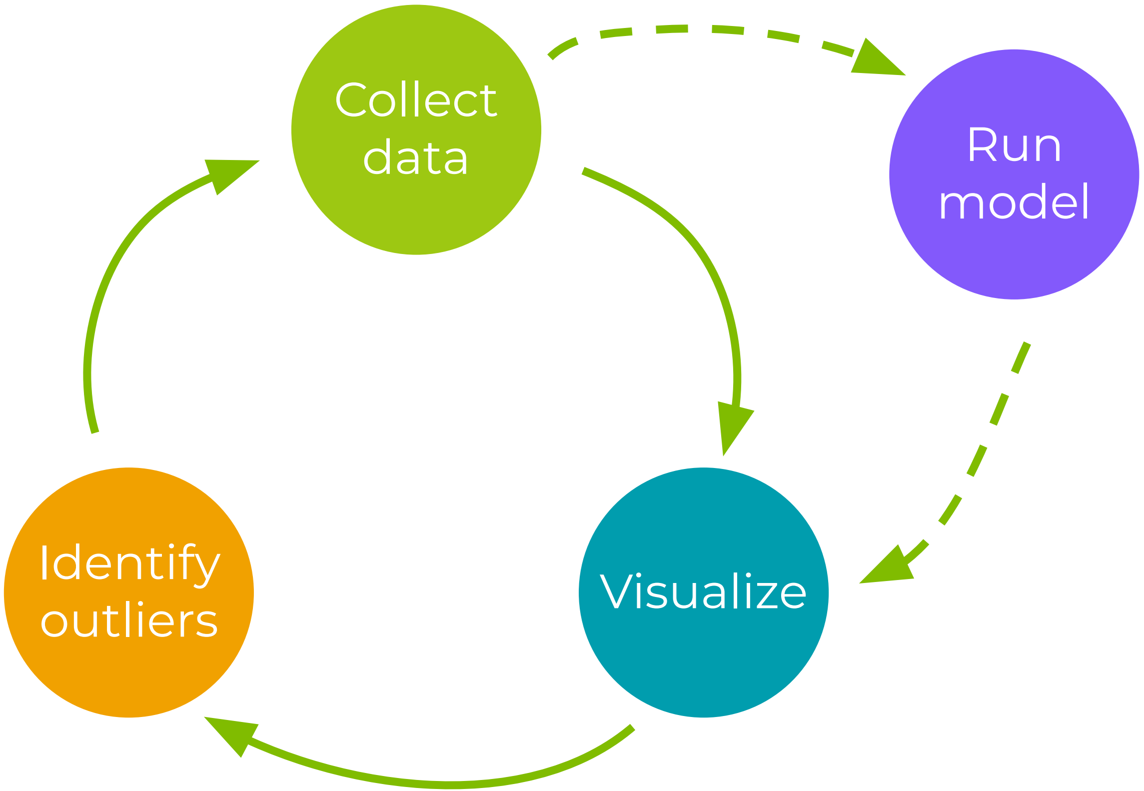
Feature extraction and dimensionality reduction for raw data exploration

- Uses modality-specific feature extractor and dimensionality reduction
- Works with very high-dimensionality features (image, spectrogram)

Data-centric: Data feedback loop



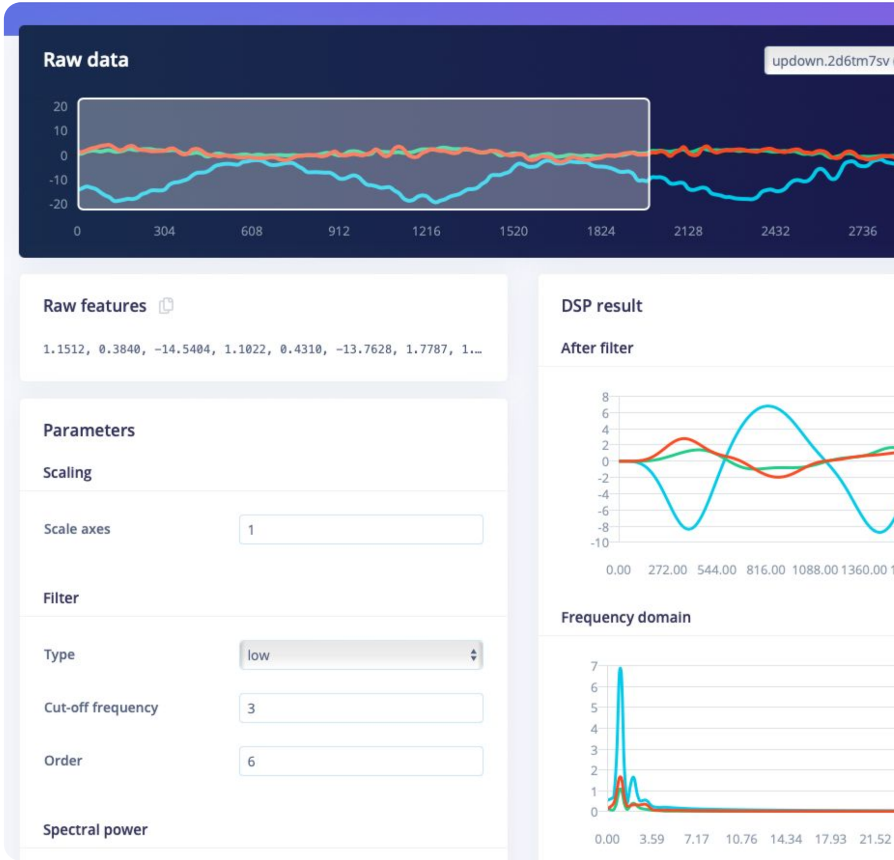
Data-centric: Data feedback loop



Data-centric: Interactive feature engineering

Real-time visualization of DSP

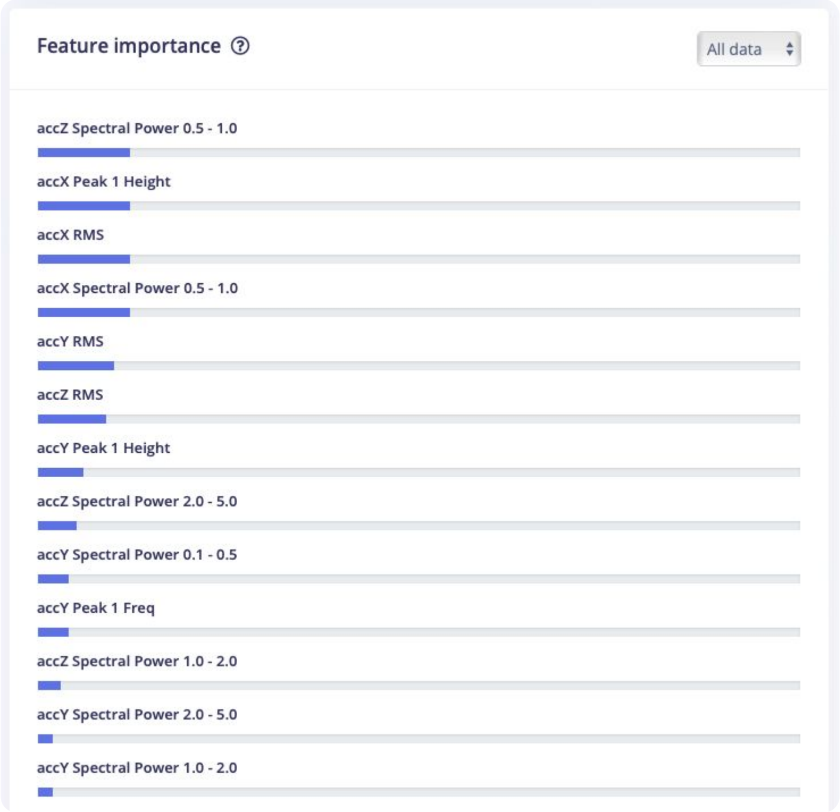
- Immediate feedback loop enabling tactile exploration by domain expert
- Service-based architecture for real time DSP on individual samples (separate from job-based system for batched data)



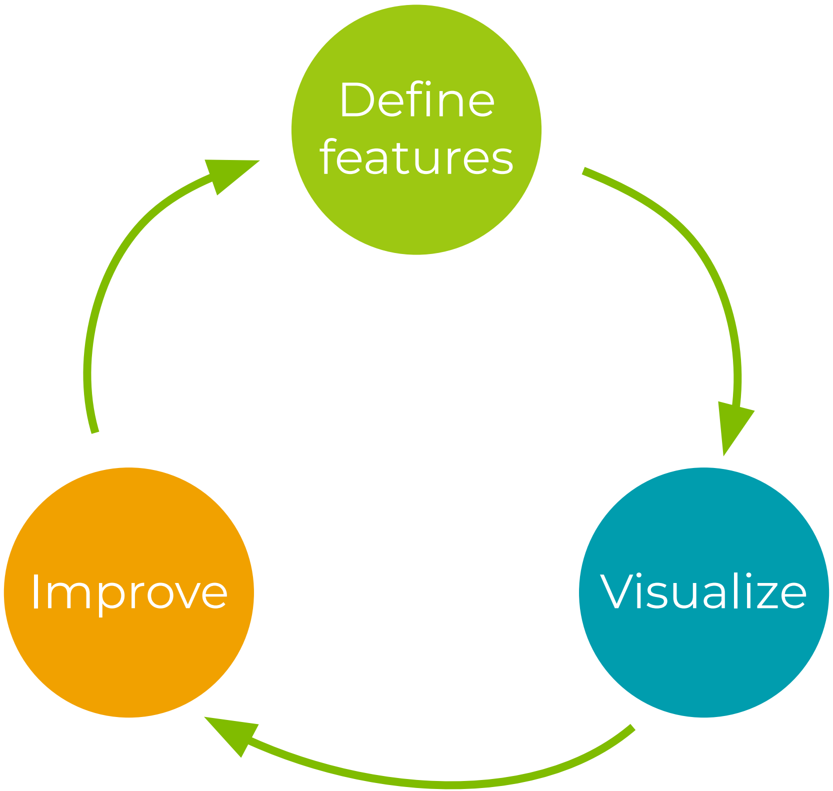
Data-centric: Feature importance

Don’t use everything

- Uses recursive feature elimination with cross-validation (RFECV)
- Only computed for relatively low-dimensionality data



Data-centric: Feature engineering feedback loop



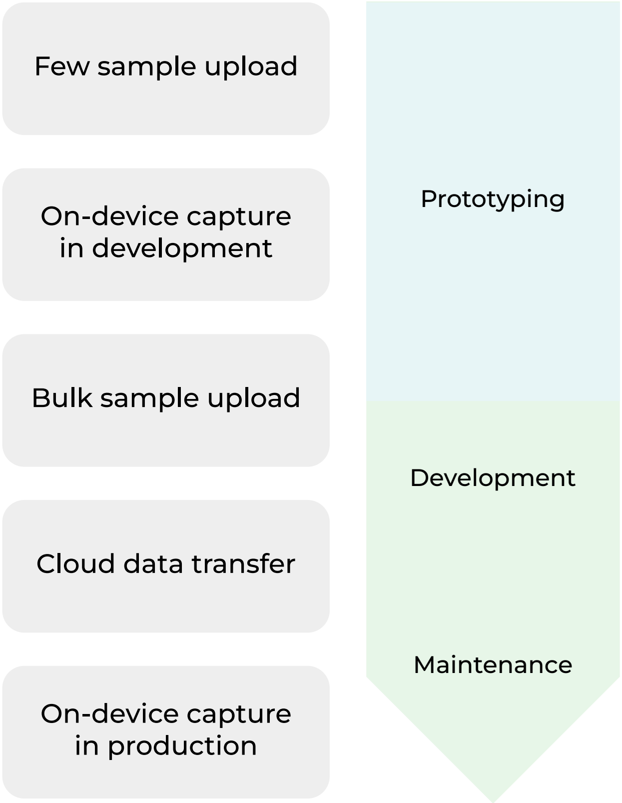
Rapid prototyping

Hardware at the speed of software

Rapid proto: Data ingestion

Technical requirements change throughout workflow

- Low-overhead data format (CBOR) & protocols for on-device use
- Data forwarder for lab work
- Pre-built firmwares with drivers



Rapid proto: EON Tuner

Establish a baseline quickly

- Search space based on prior knowledge of data modalities
- Reusable workers to minimize startup cost
- Customize and retrain any Tuner-discovered model

Rapid proto: Containerized learning

Existing models and preferred tooling

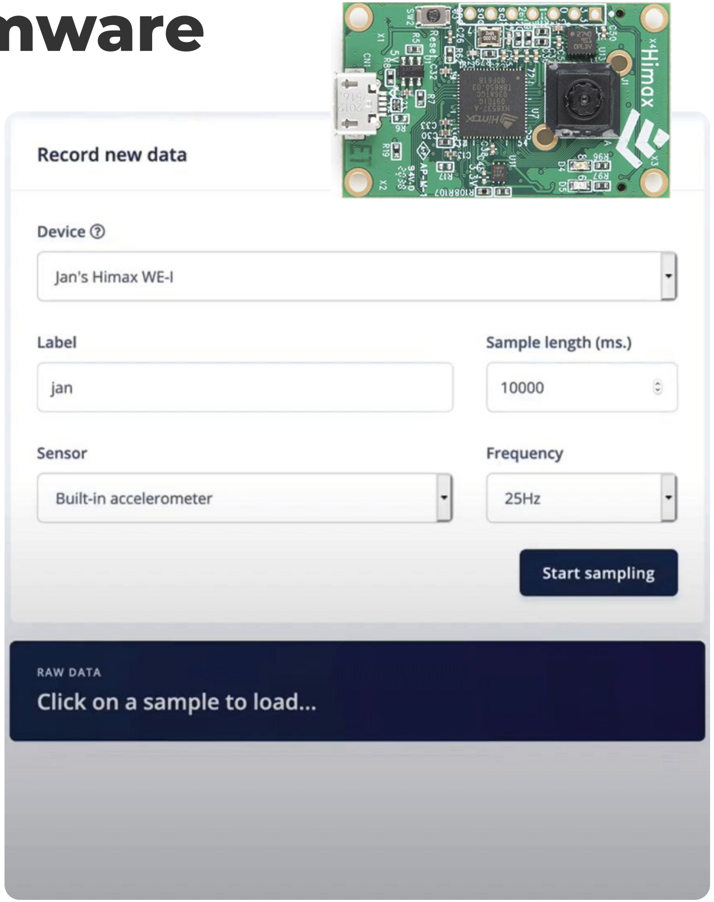
- Define training in Docker container (file in/file out)
- Runs automatically in infrastructure
- Trained model is evaluated and compiled downstream

```
# syntax = docker/dockerfile:experimental
FROM ubuntu:20.04
WORKDIR /app

ARG DEBIAN_FRONTEND=noninteractive

# Install base packages (like Python and pip)
RUN apt update && apt install -y curl zip git lsb-release software-properties-commo
RUN python3 -m pip install --upgrade pip==20.3.4

# Install TensorFlow (separate script as this requires a different command on M1 Mac
COPY dependencies/install_tensorflow.sh install_tensorflow.sh
RUN /bin/bash install_tensorflow.sh && \
    rm install_tensorflow.sh

# Install CMake (separate script as this requires a different command on M1 Macs)
COPY dependencies/install_cmake.sh install_cmake.sh
RUN /bin/bash install_cmake.sh && \
    rm install_cmake.sh

RUN apt update && apt install -y protobuf-compiler

# Copy Python requirements in and install them
COPY requirements.txt ./
RUN pip3 install --no-use-pep517 -r requirements.txt

# Copy the rest of your training scripts in
COPY . ./

# And tell us where to run the pipeline
ENTRYPOINT ["python3", "-u", "train.py"]
```

Rapid proto: Development firmware

Integrates with sensors and API

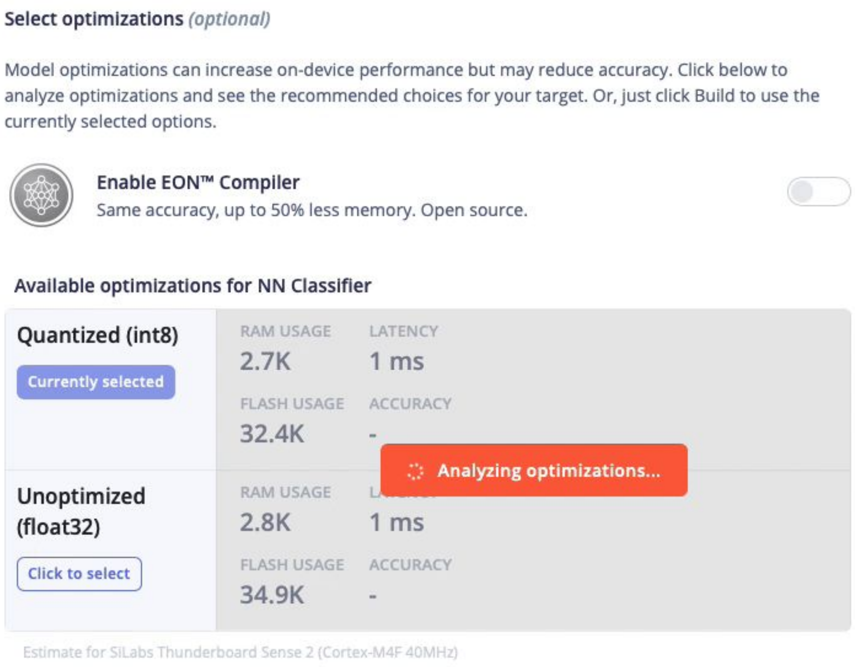
- Provides an AT command interface to collect data and run inference
- Rapidly deploy and test model with real sensors
- Daemon forwards commands and data between Studio and device



Rapid proto: Portable lightweight SDK

C++11 as lowest common denominator

- Open source C++ SDK compatible with majority of relevant devices
- Toggle optimizations via macros, “let the linker do the work”
- Minimize code size using model compilation and code generation (EON Compiler)



Rapid prototyping: Containerized deployment

Cloud builds for embedded software

- Define build in a Docker container (file in/file out)
- Rapid, universal access to latest build
- No development machine dependencies

```
FROM ubuntu:18.04

WORKDIR /ei

# Install base dependencies
RUN apt update && apt install -y build-essential software-pr

# Install LLVM 9
RUN wget https://apt.llvm.org/llvm.sh && chmod +x llvm.sh &&
RUN rm /usr/bin/gcc && rm /usr/bin/g++ && ln -s $(which clan

# Install Python 3.7
RUN add-apt-repository ppa:deadsnakes/ppa && apt install -y

# Copy the base application in
COPY app ./app

# Copy any scripts in that we have
COPY *.py ./

# This is the script our application should run (-u to disab
ENTRYPOINT [ "python3", "-u", "build.py" ]
```

The next five years of edge AI tooling

Hardware and software closer together 💞

- Automatic tailoring of algorithms for hardware
- Selection and config of hardware based on algorithms
- Compilation into HDL, not machine code

Closing the loop with deployed devices 🔄

- On-device out of distribution and drift detection
- Deep insights into model performance in the field
- Management of heterogeneous compute

Better use of precious resources 💎

- Tools for working with unlabelled data
- Edge-specific human-in-the-loop learning
- Collaborative tooling for community development

Thank you! 🙏🏽

We're hiring — edgeimpulse.com/careers

@edgeimpulse

@dansitu

edgeimpulse.com

TINY ML SUMMIT

tinyML Summit 2022 Sponsors

ANALOG DEVICES AHEAD OF WHAT'S POSSIBLE™

AONdevices

arm

ASPINITY

brainchip The Neuromorphic Computing Company

CEVA®

Deeplite

EDGE IMPULSE

emza visual sense

FotaHub

GREENWAVES TECHNOLOGIES

Grovety Inc.

Himax

HOTG

imagimob

infineon

itemis

KLIKA TECH GLOBAL IOT SOLUTIONS

LatentAI

LATTICE SEMICONDUCTOR

Micro.ai

OmniML

NXP

PX!

Plumerai

PROPHESEE

Qeexo

Qualcomm

Rackner

RealityAI® Engineering Solutions for the Edge

REEXEN technology

RENESAS

SAP

seeed studio The IoT Hardware Enabler

SensiML™

Sony Semiconductor Solutions Corporation

ST life.augmented

SA STREAM ANALYZE

synaptics®

SynSense

SYNTIANT

Tensil.ai

TensorFlow

XMOS

Copyright Notice

www.tinyML.org