TinyML and Novel AI Workflow Enables Smarter Wireless Low Power Sensors Managed and Deployed at Large Scale at the Far Edge

Mark Stubbs and Kabir Manghnani
Shoreline IoT Inc., 1671 Dell Ave. Suite 208, Campbell, CA 95008

ABSTRACT

Significant challenges prevent large scale deployment of sensors at the far edge in utilities, industry, energy production, transportation/infrastructure, and many other applications. For example, nearly half a billion motors are unmonitored due to challenges such as large upfront equipment cost, labor intensive installation, ease of access to power, and cost-effective wide area communications.

Advances in ML, low power microcontrollers, battery technology, and narrowband IoT cellular communications have opened the door to address these large markets. Over the next few years, the combination of TinyML, low power MCUs, long battery life, and narrowband cellular IoT networks will lead to an explosion in deployment of smarter sensors at the far edge providing greater insights enabling predictive maintenance and process improvements.

We will present an end-to-end architecture & market ready Industrial AI+IoT solution, which includes:

- Peel & Stick sensors, with 5 year battery life which utilizes TinyML running on ARM M4 MCU to detect and classify conditions at the far edge and communicate important events using built-in narrowband cellular modem.
- IoT Cloud platform for management and deployment of a large scale network of sensors.
- Novel TinyML AI Workflow to train, build, deploy, and adapt TinyML models at scale to millions of sensors with tiny MCUs

EXAMPLE PROBLEM – MOTOR HEALTH MONITORING

- 500M+ industrial motors
  - A tiny fraction are actively monitored
  - Motor failure is expensive
- Manual data logging (on-site)
  - $300 to $1,000/machine/year service fees
- Battery powered sensor
  - Still not real-time; streaming data drains battery
  - Still expensive; $500/machine/year cellular data

But how do we train and deploy a network of battery powered sensors to affordably monitor these assets and achieve very long battery life?

Enabling the Peel & Stick Battery Powered Sensor

TinyML, specifically Google's Tensorflow Lite for Microcontrollers, is a key technological component to enable a long life battery powered sensor.

Shoreline's iCast Sense product integrates the TinyML inference engine with a combination of sensors including:

- 3-axis Accelerometer (vibration)
- Temperature
- Humidity
- MEMs Microphones

In addition, the device includes narrowband LTE wireless and Bluetooth low energy.

Sensors are always on and monitoring for problems. When the inference engine detects an anomaly, the wireless network is powered up and the anomaly plus sensor data are sent to the cloud and the end user is notified.

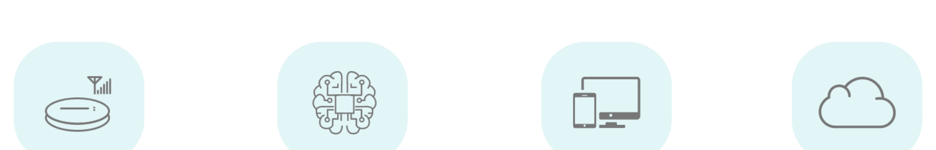
This lower power always on sensing and powering the radios only when required results in significantly longer battery life, resulting in up to 5 years on a single set of batteries under certain conditions.

TinyML + the low power MCU and Sensors enable the peel and stick battery powered sensor solution at the far edge, preventing the need for an always on radio connection to run analysis of the sensor data in the cloud.

But how do we install, monitor, and manage this network of intelligent sensors at massive scale, in billions of units?

Deploying TinyML Sensors at Scale

Massively Scalable End-to-end AIoT Solution

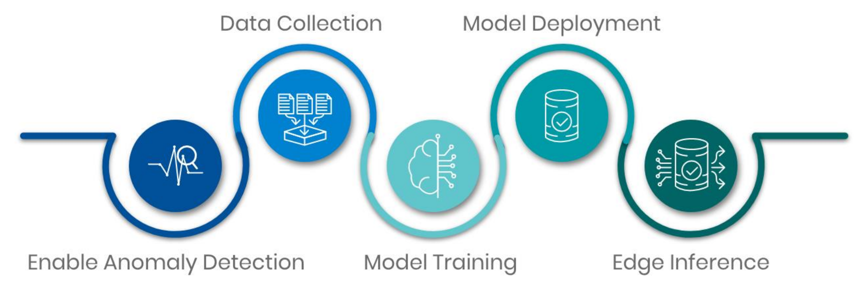


PEEL & STICK SENSOR	LOCAL ML ENGINE	WEB/ MOBILE INTERFACE	INTEGRATED IOT CLOUD
6 sensors, Fast on-boarding, Built-in Cellular → No gateways or additional infra	Anomaly detection w/ TinyML enables 5+ year Battery Life	Real-Time alerts, and asset monitoring configuration	Includes device and user management as well as AI Workflows

How do we train and develop these TinyML models?

TinyML AI Workflow

Managing a large scale deployment of intelligent sensors requires an AI workflow.



Once the sensor is installed on the asset, the sensors to train are selected and anomaly detection is enabled. Data collection begins and the information is uploaded to the cloud.

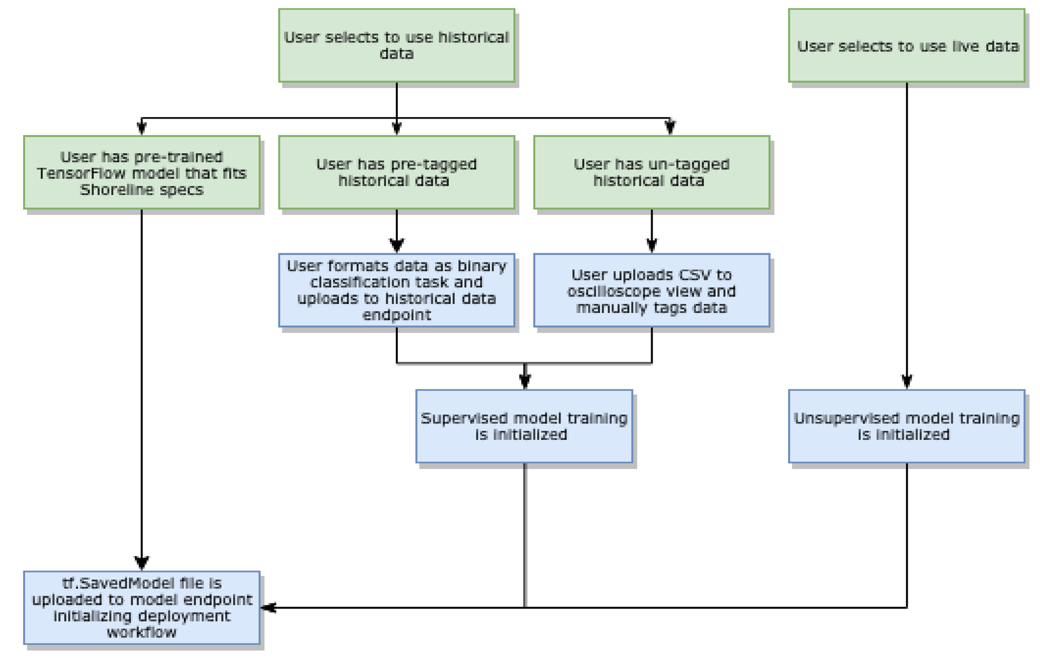
Once enough data is collected, the model is trained and deployed to the inference engine in the far edge device.

How does AI Workflow handle supervised vs unsupervised models?

Flexible Model Development Workflows

Model may be developed two ways:

- Historical data provided by the user
- Live data recorded from the device



Data Collection

Data collection for the unsupervised model begins when the sensor is installed. The AI workflow configures the sensors to use, the sampling rate, and the duration of time to record data for building the model.

The length of time is determined by the type of asset being monitored and the data requirements determined experimentally to provide best results for that type of asset.

Once the data collection is complete, the model is trained and is ready to deploy.

Now that we have a model, how do we deploy it at the far edge?

Deploying The Model

Once the model is trained and tested on the data, it is packaged and compressed and sent using over-the-air (OTA) update to the far edge device.

Since the device is battery operated, the device checks in at the configured update interval and receiving a notification that the model update is available. The model is downloaded, the inference engine is started, and the training data collection is shut down.

Now the inference engine takes over and runs the model on live data at the far edge and will only transmit significant events classified by the inference engine to the cloud..

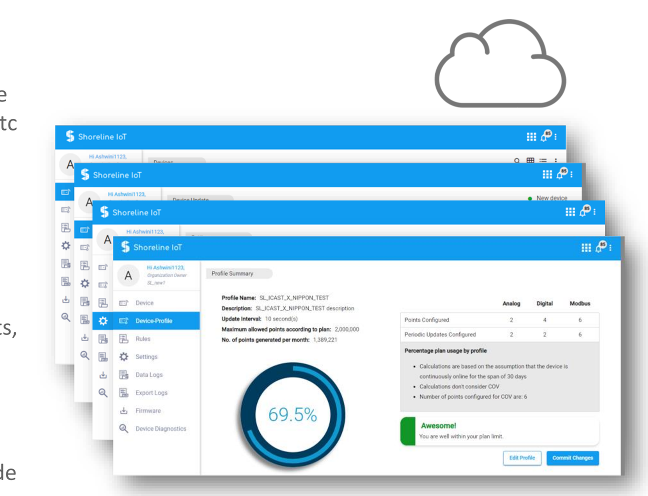
Cloud Platform

Our cloud platform houses the AI workflow, as well as device management and functions to notify the end user when an anomaly is present.

When an anomaly condition is present, the anomaly trigger is sent to the cloud. From there, a chain of actions may be configured including notifying key personnel to address the problem.

Predictive models allow scheduling repairs preventing costly emergency repairs.

- DEVICE MANAGEMENT
  - Device provisioning, sensor profile configuration, diagnostics, OTA, etc
- USER MANAGEMENT
  - Account registry, multi tenancy, configuration & authentication
- ASSET MONITORING
  - Real-time data, rules engine, alerts, dashboard & Osc view
- AI WORKFLOW FOR TF LITE MICRO
  - Training, model development, Inference, deploy on far-edge node



Real-life Use Case

Predictive Maintenance Demo

Anomaly Detection w/ TF Lite Micro on ARM M4

Real-time Anomalies Detected

- Bearing wear
- Shaft misalignment
- Noise from binding, bushings and bearings
- Over temperature, Flow variance, Air Leaks

Local Anomaly States

- Normal Operation
  Cellular radio OFF
  No cloud connection
- Anomaly Detected
  Cellular radio ON
  Upload data to cloud