Neuro Mem Smart!™

Application Note

Predictive Monitoring

Accelerating the modeling of novel events for predictive maintenance.

Keywords: neural network, novelty detection, predictive maintenance, anomaly classification, knowledge bases, non-linear classifier, lifelong learning, smart sensors

The Challenge

Smart sensors are driving the deployment of monitoring systems in our everyday lives from wearables tracking our mobility to complex sensor hubs ensuring the quality of a production line and the proper operation of its machinery. Their "smartness" comes from a software running a pattern recognition engine and associated decision logic. In the case of predictive monitoring, the sensor needs to be paired with some higher intelligence capable of learning the novelties (objects or events) which are not recognized by the always-on recognition engine. Their detection cannot just call for a warning or actuation signal. Their recording would generate a large amount of redundant data and this solution is not practical for most sensory devices lacking storage capacity. Learning the novelties will enable the modeling of a spurious change, a temporary drift or an irreversible trend, paving the way to intelligent decision making.

This is where a NeuroMem® neural network becomes a true problem solver. **Unlike a Deep Learning engine, this network will admit when it does not know, which is the information of interest in predictive maintenance. It will be capable of learning in real-time (including new categories) and classify novelties.**

Condition monitoring of humans and machines with the fusion of cognitive sensors

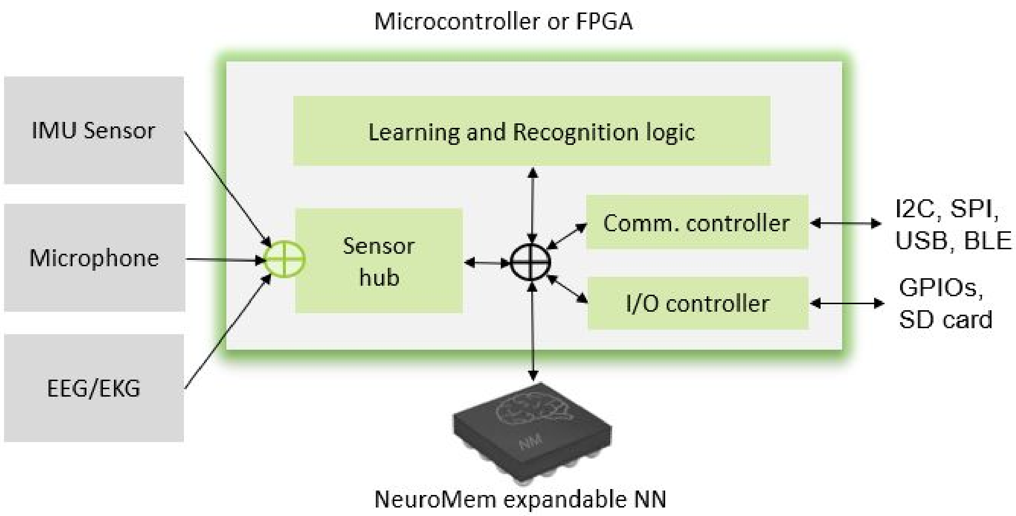
(Video, Audio, Acceleration, Pressure, Temperature, Biosensors, …)

This application note explains how you can take advantage of a NeuroMem neural network to learn new conditions while running an always-on recognition, making it a **unique enabler for predictive maintenance**.

The Solution

The hardware architecture

The minimum hardware is composed of a sensor hub interfaced to one or multiple sensors, a microcontroller or a Field Programmable Gate Array (FPGA), a NeuroMem chip, and some GPIOs for communication and access to an SPI memory bank or SD card. The module can feature an expansion connector to support additional NeuroMem chips and scale the neural network if needed (Refer to the example of the NeuroShield board with NeuroBrick expansion modules).



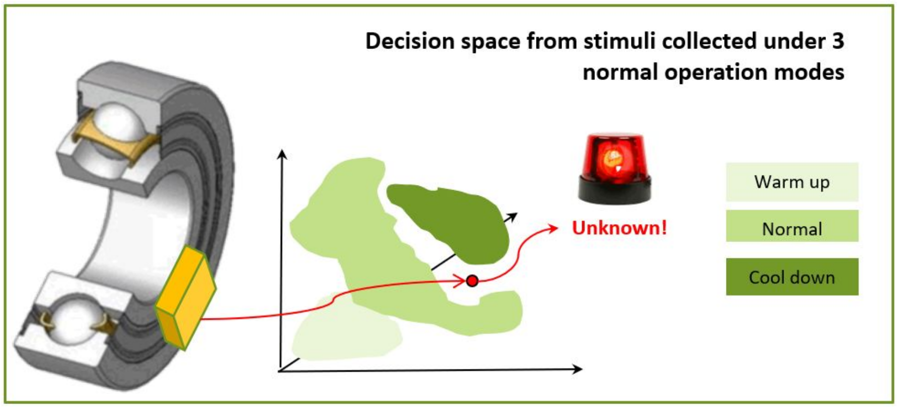
The FPGA acts as a glue between all the components. At runtime, its two main tasks are to (1) convert the signal from the sensors into features for the neurons and (2) convert the neurons' response into actionable items. Depending on the application, the actions can be to stay idle, turn a warning signal on, switch a relay, write the features and time stamp to memory, or learn the features as "Novelty."

Learning the normalcy

Let's take the example of the monitoring of a ball bearing (an helicopter has between 20 to 60 ball bearings). In this case, the sensor hub integrates a microphone and an Intertial Measurement Unit. Its size is small enough to be affixed to the side of the bearing.

The NeuroMem neurons are initially trained with examples of sounds and vibrations of the ball bearing under normal operation modes of the helicopter. This training can be done on the final hardware or a remote system also equipped with a NeuroMem network and running the same learning and recognition logic.

As soon as a new example is taught to a NeuroMem neural network, it is taken into account in the next recognition. This real-time modeling and feedback are efficient to quickly validate a recognition engine. Thanks to their physical interconnectivity, the neurons can produce a collective response in microseconds and decide if the last example should be learned by a new neuron or simply discarded because it is redundant with what they already know. Examples can be simple or aggregated features extracted from single or multiple signals. They have two additional attributes, which are a category and a context. For a predictive maintenance application, the context can be set as either “Normal” or “Novelty”. The category can be either a fixed value or an index to an operation mode table (for example, Warm-Up, Normal, Cool Down as shown in the figure below).



Understanding the conditions of a normal operation mode is not always simple. In real-life applications, variability is rarely linear, and this is why the non-linear model generator of the NeuroMem neural network becomes very powerful. Also, in the context of predictive monitoring, we want to refrain the neurons from over-generalizing since the final goal is to detect subtle variations from the normal operating mode. In a NeuroMem network, a single setting called Maximum Influence Field can adjust the neurons’ conservatism.

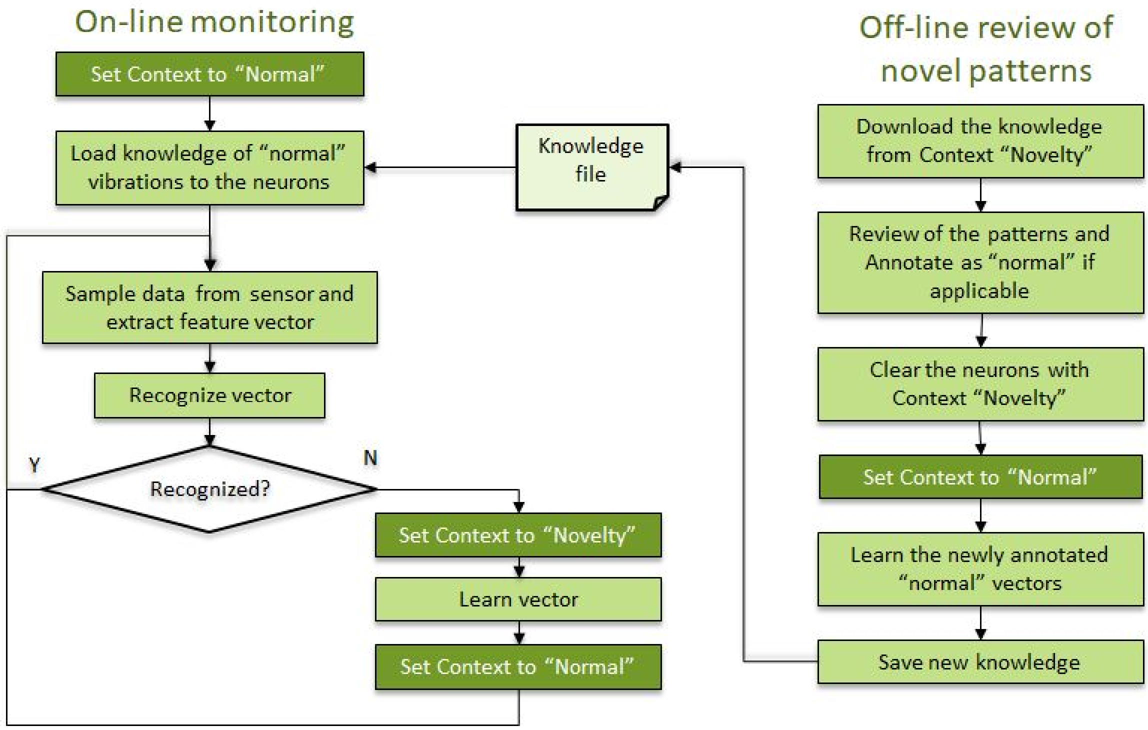
Learning novelty from normalcy

Once the neurons have modeled the normal operation modes of a ball bearing, monitoring can begin. The recognition logic samples the signals generated by the sensors, extracts features, sends them to the neurons and reads their global response. If it reports a positive recognition, the monitoring continues. Otherwise, the learning logic is activated. It sets the context to “Novelty”, learns the features with a category set to a timestamp or other parameter relevant for the post-analysis. If the hardware has storage, the corresponding signal samples can be stored to help with data post-correlation. Finally, the context of the network is then reset back to “Normal” so the monitoring of the normal operation mode can resume.

Analyzing the novelties

Upon landing, the new neurons committed during the flight contain the features of the encountered novelties and their timestamps. Upon download and reviewing this data, a human supervisor can realize that some novelties are normal but had simply not been part of the initial training. He can then decide to learn them under the “Normal” context.

The following diagram summarizes the workflow for the deployment of continuous monitoring with novelty recording and classification.



Conclusion

NeuroMem has unique features for continuous monitoring and novelty detection:

- Non-linear classifier, with a global response in nanoseconds
- Model generator (500 nanoseconds to commit a new example, no duplication, intrinsic correction, and degeneration)
- Knowledge traceability (the novel features reside in the neurons' memory and can be retrieved for visualization and analysis)

Its small footprint and low-power consumption make it the perfect candidate to design cognitive sensors.