The Intelligence of Things enabled by Syntiant's TinyML board

Alireza Yousefi, Luiz Franca-Neto, Will McDonald, Atul Gupta, Mallik Moturi, and David Garrett.

Syntiant Corp, 7555 Irvine Center Dr Ste 200, Irvine, CA 92618.

Abstract

In this poster, we present our new TinyML development board designed for battery-powered always-on edge-AI applications. The board contains an IMU sensor for motion sensing, a MEMS microphone for audio applications, and a uSD card slot for data collection. Having an ultra-low-power NDP101 chip at its core, the dream of having a sub-mW edge-AI system can readily come true. DNN models can be trained and uploaded to the board using the Edge Impulse platform. The poster will also present two use cases for the board in which we demonstrate how to build audio and motion detection models and deploy them on the board. The board can be seen as a step toward democratizing "Tiny" machine learning.

Introduction

- **TinyML**– the intersection of embedded systems and ML [1]
  - Deploying ML models at the edge where the data exists.
  - Lower latency, better privacy, lower power, and higher reliability.
  - **Deep learning** [2]: Learning a hierarchical representation with increasing levels of abstraction.
- **DL/ML for time-series data**
  - Keyword spotting (KWS) - Traditional approaches (e.g., HMMs) vs. DNNs[3] : less computational complexity and superior performance.
  - Extending to variety of audio/voice and sensor applications.
- **Always-on** intelligence with Syntiant NDPs (Figure 1)
  - Event-driven processing for different sensing modalities.

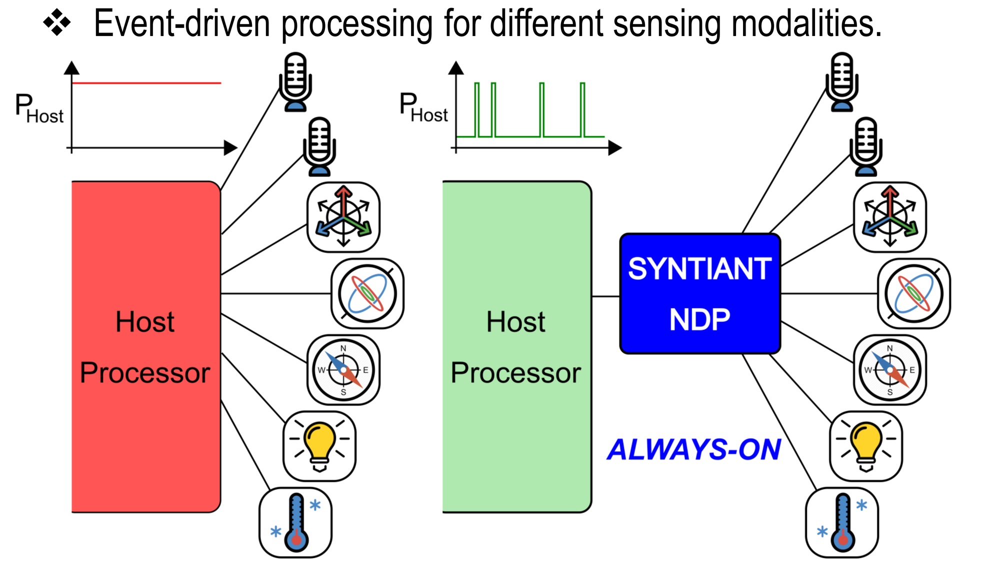

Figure 1: Saving the power consumption in an AIOT system by using the always-on intelligence.

Syntiant TinyML Board

- Syntiant TinyML board (Figure 2)
  - A **self-contained edge-AI inference system** for **audio and motion** applications (Figure 3).
- **An ideal platform for data collection**
  - With a 32GB micro-SD card
    - > 3 days of uncompressed audio data (Fs = 16kHz)
    - > 300 days of 6-axis IMU sensor data (Fs = 100Hz)

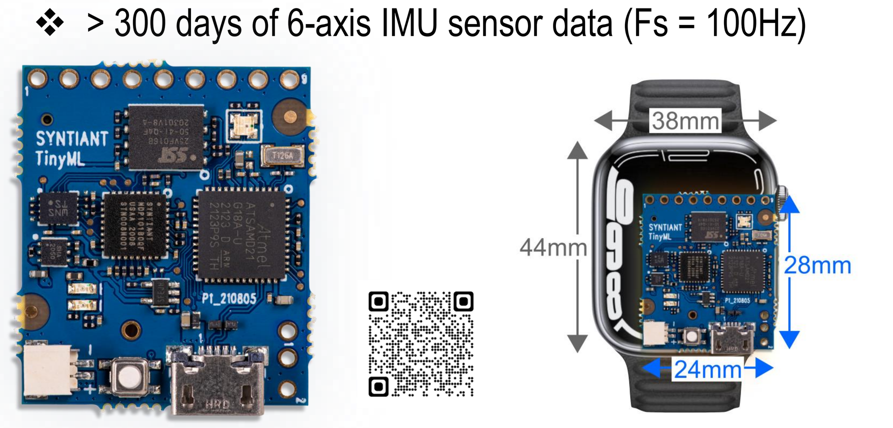

Figure 2: Syntiant TinyML board (left), Link for purchasing TinyML boards (center), Comparing a TinyML board with an Apple Watch Series 7 (right)

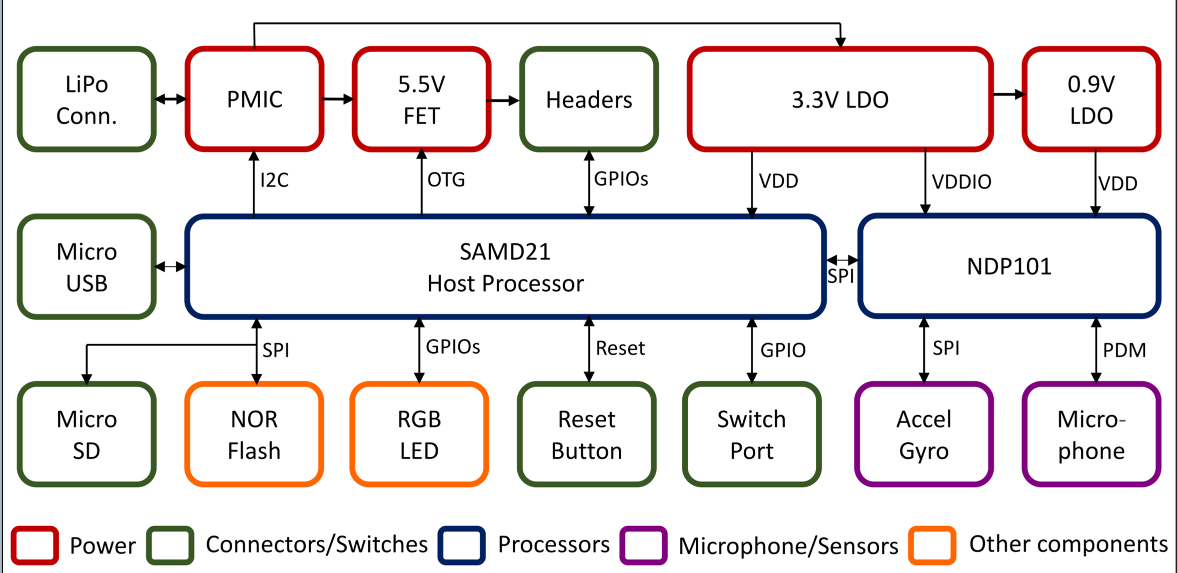

Figure 3: Syntiant TinyML board – block diagram.

Syntiant NDP101

- **Purpose-built** to run deep neural network models (DNN) (Figure 4)
  - **At-memory computation** – Exploits the **inherent parallelism** of DNNs while computing **at required numerical precision**.
  - Compared to CPU/MCUs and DSPs, NDP10x delivers **20x more throughput** and consumes **200x less energy per inference** [4].

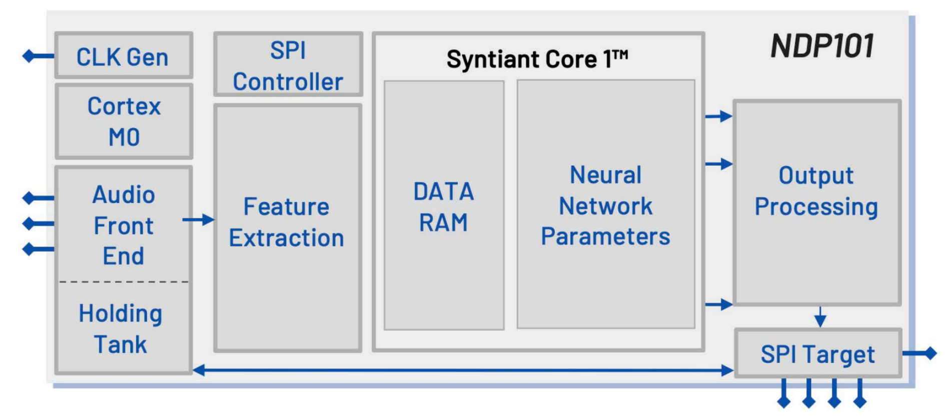

Figure 4: Syntiant Neural Decision Processor (NDP) chip – block diagram.

- **Syntiant Core 1**
  - Configurable Fully-connected layers (FC) (Figure 5)
  - 590k parameters, ReLU and softmax activations, Programmable interlayer scaling
  - Max frame rate: 200Hz

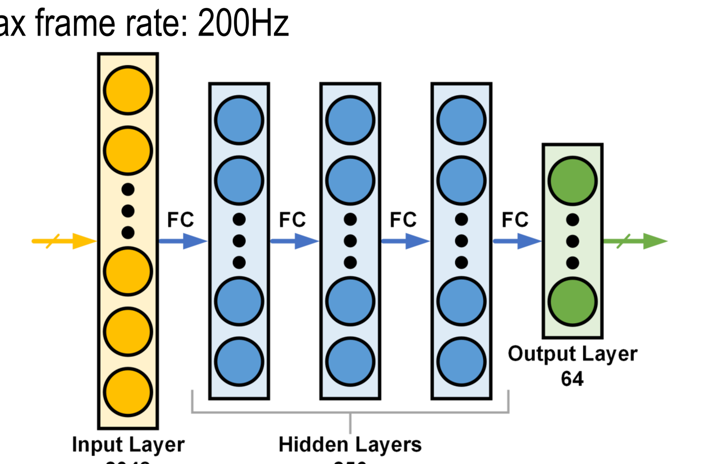

Figure 5: Four configurable FC layers in Syntiant Core 1

- Other features:
  - 2 PDM microphones, I2S or PCM-over-SPI input
  - SPI / I2C interface for sensors
  - **Supports frequency-domain and time-domain inputs**
    - **Configurable** FFT-based feature extraction (can be used for **speech audio** and **non-speech event** detection.)
  - 96kB holding tank
  - Embedded ARM Cortex-M0 processor
    - Can be used for preprocessing the input data, posterior handling (to improve FAR and FRR), etc.
- **Always-on power consumption**
  - **140uW** for audio/voice applications.
  - **100uW** for sensor data (by-passing the feature extractor).
- Modeling and deployment process for NDP101 (Figure 6)
  - **It always starts with data!**
    - Open-source datasets if available, otherwise the data needs to be collected, cleaned and properly labeled.
    - **Data-augmentation**

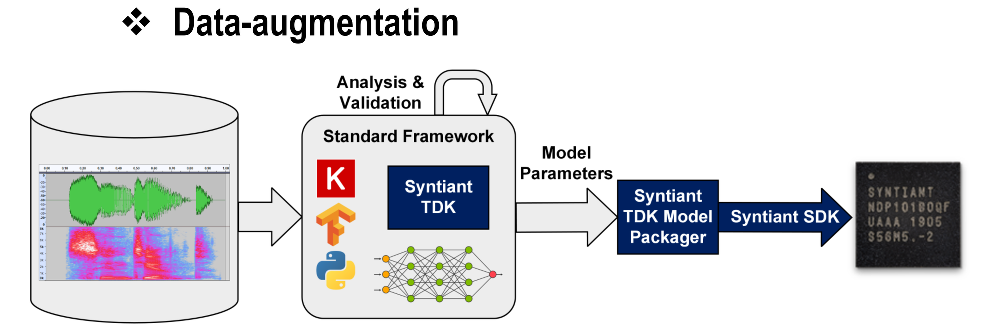

Figure 6: Training a DNN model and deploying it on NDP101

Use Cases

- **Use case #1: key word speech interface** (Table 1)
  - Edge Impulse platform
  - Dataset: Google's Speech Commands
    - Training to detect to two keywords : "Go" and "Stop"
  - 3 output classes: "**Go**", "**Stop**", and "**Unknown**" (Figure 6)
    - Separation between the classes (Figure 7)
  - Model accuracy on the test set: **97.17%** (Figure 8)

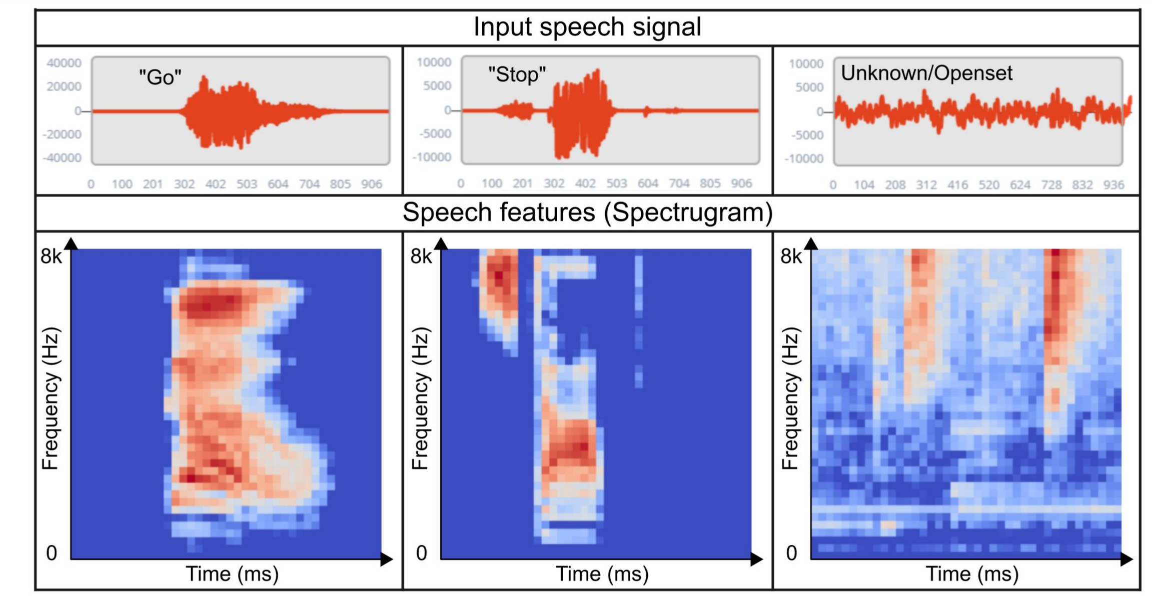

Figure 7: Samples from the training set – time-domain (input to the feature extractor) and frequency-domain (input to the DNN) [from Edge Impulse platform]

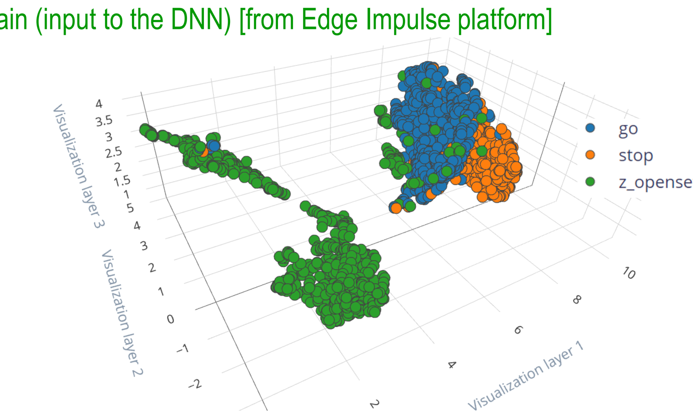

Figure 8: The training set visualization (from Edge Impulse platform).

	GO	STOP	Z_OPENSET	UNCERTAIN
GO	98.3%	0.4%	0.1%	1.2%
STOP	0.9%	96.2%	0.1%	2.7%
Z_OPENSET	1.1%	0%	97.1%	1.8%
F1 SCORE	0.98	0.98	0.98	

Figure 9: Model performance – confusion matrix (from Edge Impulse platform)

- **Use case #2: hand/wrist gesture detection** (Table 2)
  - Trained and deployed using Syntiant TDK/SDK tool-chain.
  - Dataset : collected by Harvey Mudd Clinic Team [6].
  - 4 output classes: "**watch-check**, "**outward-flick**", "**inward-flick**" and "**Unknown**" (Figure 10)
    - **Time-domain** 6-axis IMU sensor data (Figure 11)
      - the feature extractor was bypassed in the NDP chip.
    - Data frame and window structures – creating the input vector fed into the DNN (1440 features) (Figure 12).
  - Model accuracy on the test set: **99.64%** (Figure 13).

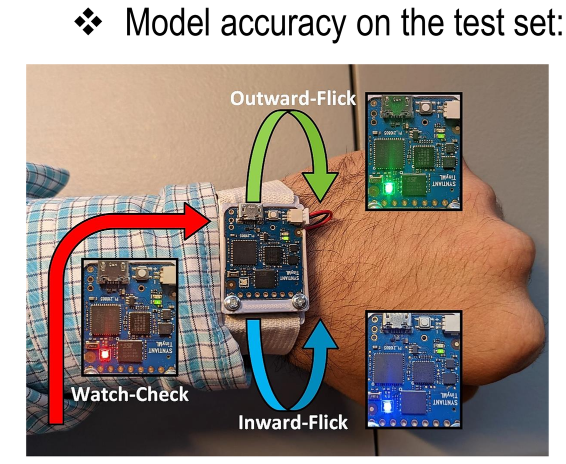

Figure 10: Wrist band with TinyML board - The gestures definition.

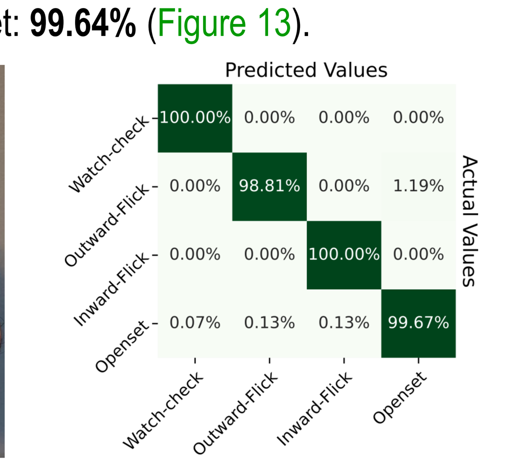

Figure 12: The hand/wrist gesture detection model performance confusion matrix.

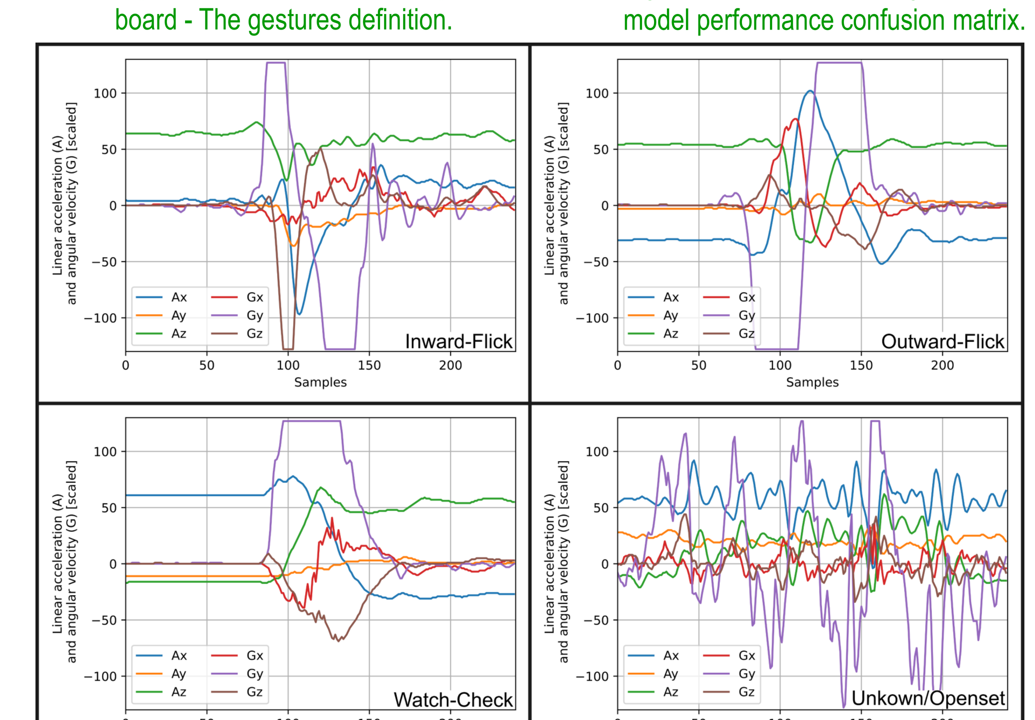

Figure 11: Samples from the training set (2.4s input window to the DNN) – Ax,y,z: Linear accel., Gx,y,z: Angular velocity. Scaled from +/-8g m/s² and +/- 2000 degrees/s to 8-bit values [-128,128).

Modeling Parameters (Table 1)	
Data pre-processing	
Sampling frequency	16kHz
Frame length and stride	32 ms, 24 ms
Window length (input to DNN)	986 ms
# DCT features, FFT length	40, 512
Dataset size [samples]	9868
Dataset split: (training, val., test)	(72%, 8%, 20%)
Data augmentation	
SpecAugment [5]: time mask param. (T)	1
Additive gaussian noise: stddev	0.2
DNN model	
# of FC layers	4
# of input features	1600
# of output classes	3
Hidden layers width	256
Activation function	ReLU, Softmax
Training	
Epochs, Batch size	50, 32
Optimizer	Adam
Learning rate	0.0005
Initial decay rates: (β_1,β_2)	(0.9,0.999)
Loss function	Crossentropy
Regularization	
Dropout	0.2

Modeling Parameters (Table 2)	
Data pre-processing	
Sampling frequency	100Hz
Frame length and stride	60 ms, 60 ms
Window length (input to DNN)	2.4 s (40 frames)
Dataset size [samples]	25000
Dataset split: (training, val., test)	(65%, 23%,12%)
Data augmentation	
Time shift [ms]	[-400, 400]
DNN model	
# of FC layers	4
# of input features	1440
# of output classes	4
Width of hidden layers	256
Activation function	ReLU, Softmax
Training	
Epochs, Batch size	25, 32
Optimizer	SGD
Learning rate	0.001
Learning rate decay	10^{-7}
Momentum (+ Nesterov momentum)	0.9
Loss function	Crossentropy
Regularization	
Dropout	0.2

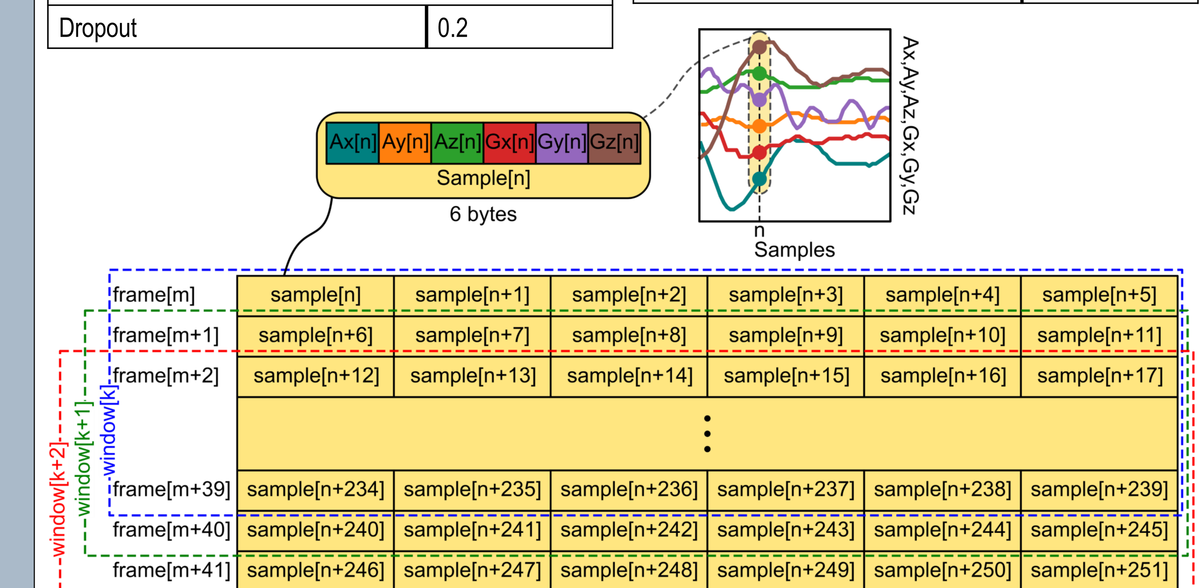

Figure 12: Constructing data frames and windows as the input of the DNN model from time-domain 6-axis IMU data samples.

Conclusion

- **The Syntiant TinyML board can enable variety of ML use cases at the edge** including keyword speech interface, acoustic event detection, sensor applications, and condition-based monitoring.
- TinyML models can be easily trained on the Edge Impulse platform and deployed on the board.
- **NDP101 is tailored to run DNN models – The most efficient solution.**
  - Small foot-print fully-connected models are **effective** and **computationally efficient** for audio and sensor applications– **more advanced models** such as RCNNs, DS-CNNs [7] and Transformers [8] can be deployed on our **Syntiant Core 2** available in NDP120/200.
- The two models presented here did not have production-level FAR/FRR because of using relatively small datasets.
  - The performance can be improved by collecting more data that captures the application environment, using more advanced data augmentation techniques and applying hard negative mining.

References

[1] Janapa Reddi, V., Plancher, B., Kennedy, S., Moroney, L., Warden, P., et al. (2022). Widening Access to Applied Machine Learning With TinyML. Harvard Data Science Review, 4(1).

[2] LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. "Deep learning." nature 521.7553 (2015): 436-444.

[3] Chen, G., Parada, C., & Heigold, G. (2014, May). Small-footprint keyword spotting using deep neural networks. In 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).

[4] Jeremy Holleman. (2019). "The Speed and Power Advantage of a Purpose-Built Neural Compute Engine.", Syntiant, accessed on 18 March 2022, https://www.syntiant.com/post/keyword-spotting-power-comparison .

[5] Park, D. S., et al. (2019). Specaugment: A simple data augmentation method for automatic speech recognition. arXiv preprint arXiv:1904.08779.

[6] Vicki Moran, and Will McDonald. "Training Neural Networks for Sensors. " TinyML Talks (2020).

[7] Zhang, Yundong, et al. "Hello edge: Keyword spotting on microcontrollers." arXiv:1711.07128 (2017).

[8] Berg, Axel, et al. "Keyword transformer: A self-attention model for keyword spotting." arXiv:2104.00769 (2021).

Acknowledgements

The authors would like to thank Aurelien Lequertier from Edge Impulse for his technical support, and our student senior-clinic team from Harvey Mudd College: Henry Limm, Yaqub Mahsud, Vicki Moran, Taylor Sloop, Maxime Vienne for helping us with the hand gesture detection use case (data collection and modeling).