Online Machine Learning on Microcontrollers
Online Clustering on ESP32 use case

Christophe Cérin, Mamadou Sow

LIPN, USPN – Paris 13

JRAF'23 – 13/12/2023

Table of Contents

Part. 1 Introduction

Introduction

First elements of context

- Online machine learning is a method of machine learning in which data becomes available in a sequential order and is used to update the best predictor for future data at each step,
- Batch learning techniques that generate the best predictor by learning the entire training data set at once.
- Continual learning means constantly improving the learned model by processing continuous streams of information:
  - Challenging since the continual acquisition of incrementally available information from non-stationary data distributions generally lead to catastrophic forgetting.
  - Catastrophic forgetting: the tendency of an artificial neural network to abruptly and drastically forget previously learned information upon learning new information;

Introduction

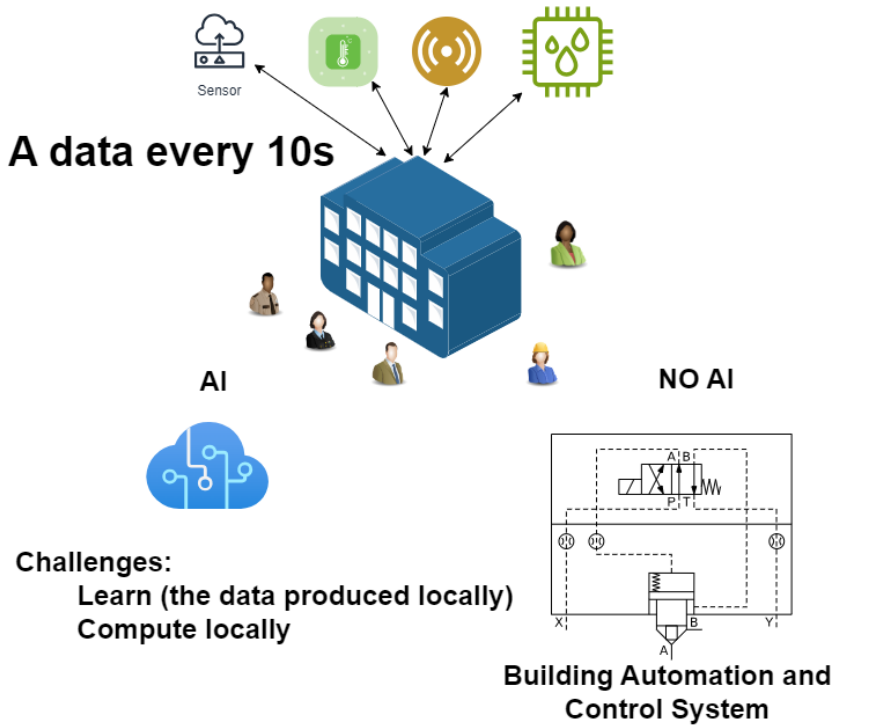
First elements of context: existing Python libraries

- Scikit-learn: Provides **out-of-core** implementations of algorithms for
  - Classification: Perceptron, SGD classifier, Naive bayes classifier.
  - Regression: SGD Regressor, Passive Aggressive regressor.
  - Clustering: Mini-batch k-means.
  - Feature extraction: Mini-batch dictionary learning, Incremental PCA.
- River (https://github.com/online-ml/river). It aims to be the most user-friendly library for doing machine learning on streaming data. River is the result of a merger between Creme and scikit-multiflow:
  - Family of algorithms: Linear models, with a wide array of optimizers, Decision trees, and random forests, (Approximate) nearest neighbors, Anomaly detection, Drift detection, Recommendation systems, Time series forecasting, Bandits, Factorization machines, Imbalanced learning, Clustering, Bagging/boosting/stacking, Active learning.
  - Other online utilities: Feature extraction and selection, Online statistics and metrics, Preprocessing, Built-in datasets, Progressive model validation, Model pipelines.

Introduction

Motivations

- We want a model that can learn from new data without revisiting past data.
- We want a model which is robust to concept drift.
- We want to develop your model in a way closer to what occurs in a production context, which is usually event-based.
- It plays nicely with the rest of the programming ecosystem for the embedded systems.



https://github.com/CampusIoT/datasets/tree/main/BuildPred/notebooks

Part. 2 Problem definition and solutions

Problem definition

Clustering is the target

- Given online data produced at *low frequency* from a *small number* of sensors, learn on it.

Problem definition

Clustering is the target

- Given online data produced at *low frequency* from a *small number* of sensors, learn on it.
- Today: learning = clustering;
- Smart building systems use sensors and data analytics to monitor energy usage in real-time, allowing building managers to optimize energy consumption and reduce costs.
- Applications of clustering in our context: visualization tool (building dashboard) $\Leftarrow$ simplify tasks like controlling building temperature, equipment maintenance through mobile devices and computers;
- Lessons learned from Louis Closson PhD. (Adeunis - LIG): AI is not always useful to get a model of the building – clustering, anomaly detection $\Rightarrow$ eyes are enough to trigger an alarm;

Problem definition

Clustering is the target

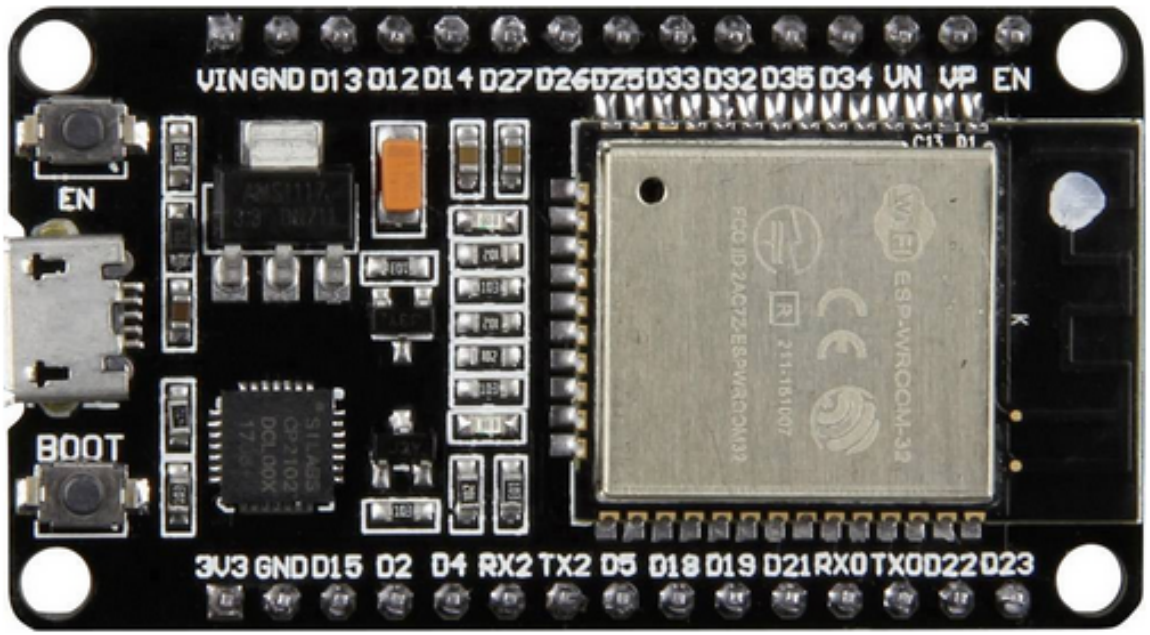
- Given online data produced at *low frequency* from a *small number* of sensors, learn on it.
- Today: learning = clustering;
- Smart building systems use sensors and data analytics to monitor energy usage in real-time, allowing building managers to optimize energy consumption and reduce costs.
- Applications of clustering in our context: visualization tool (building dashboard) $\Leftarrow$ simplify tasks like controlling building temperature, equipment maintenance through mobile devices and computers;
- Lessons learned from Louis Closson PhD. (Adeunis - LIG): AI is not always useful to get a model of the building – clustering, anomaly detection $\Rightarrow$ eyes are enough to trigger an alarm;
- Mandatory:
  - Rule on an embedded processor (ESP32);

Problem definition

Online clustering is the target application

- Mandatory :
  - ESP32 microcontroller with WROOM-32D chip: processor frequency up to 240 MHz, Wi-Fi 2.4 GHz, Bluetooth LE, 38 GPIO pins, 520 KB RAM, 4 MB flash memory, Additional Options: USB to UART bridge, buttons, LEDs

- CPU ESP-WROOM-32 (Tensilica Xtensa LX6)
- Voltage : 3.3V
- Flash : 4000 kB
- RAM : 520 kB
- EEPROM : 448 kB
- Clock speed : 240MHz
- WiFi : Yes
- Bluetooth : Yes
- SD : No



Online learning (clustering)

Algorithm

The algorithm is based on offline clustering (traditional Kmeans or kmeans++ or KNN can be used);

Memory: W = M/c, where c is a constant factor;

Each sort in step 4 can be done in parallel (if the hardware does permit it);

The idea of eliminating p data by taking them in a regular way corresponds to the idea of preserving diversity in the input.

Algorithm 1 Main algorithm of our generic framework

Require: W: window size in points' number that fit in memory;
Require: p: number of points to remove at each iteration; $p \ll W$;
1: n = Read(W inputs from the data stream);
2: **loop**
3: Cluster the n input data (for instance with k-means or k-means++);
4: Sort the n data;
5: Evict, in the sorted data, p data (for example by considering regular intervals);
6: Read p new data from the data stream to form a new window of size W;
7: **end loop**

Steps 4, 5, 6 form the incremental model updates

Note: a) overlapping between the production T_p of data and the consumption (sequential execution) b) No pressure on the input buffer if $T(p) > 50.W_p + W_p \log W_p + W_p \Rightarrow$ hint to calibrate the size of W_p.

Online learning (clustering)

Performance metric and example of results

Table 1: Quality metrics with no recalculation of centroids between iterations.

W	n	k	Centroids with data stream algorithm	Centroids with Scikit learn kmeans	Jaccard index	Gini index for data stream algorithm	Gini index for Scikit learn kmeans	Ratio Gini indexes (%)
256	12256	5			0	18165972.03	1746156257.56	-98.95
256	12256	5			0	11216278.13	1647259205.64	-99.31
256	12256	5			0	19471853.63	1696275260.49	-98.85
256	12256	5			0	8301303.31	878905310.71	-99.05
256	12256	5			0	17447215.28	1680150040.83	-98.96
256	12256	5			0	15332205.36	863360772.02	-98.22
256	12256	5			0	21623806.32	2429909814.65	-99.11

Metrics of performance over the set of centroids:

- Gini index is (approximately) the sum of the squares of the distances of the points from the center of gravity => Dispersion
- Jaccard index (dissimilarity)

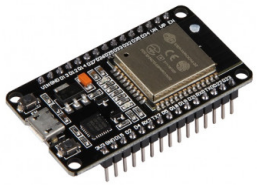
$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

Online learning (clustering)

Platform – Implementation details

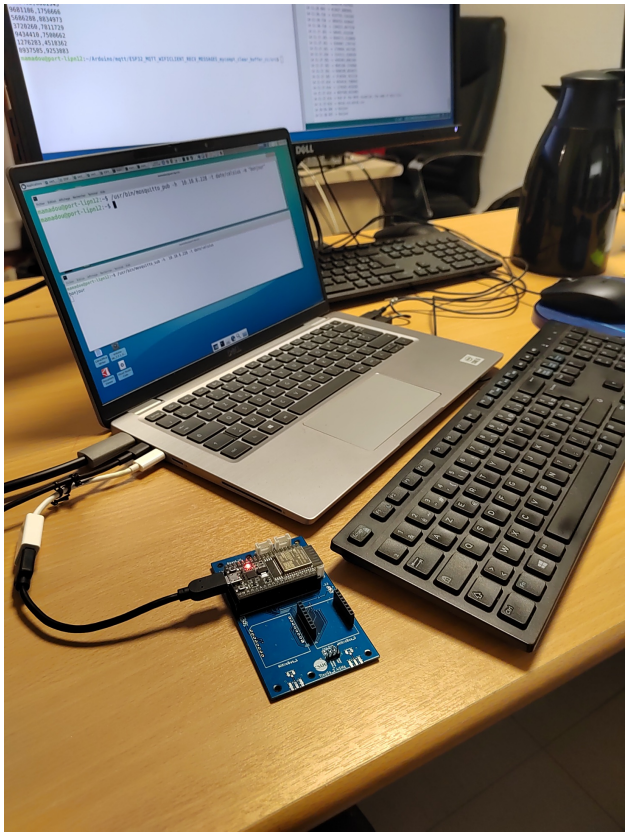
- Requirements to use the ESP-WROOM-32D microcontroller:
  - PC running Ubuntu 22.04.1 or 18.04.1 LTS operating system ;
  - ESP-IDF or Arduino which contains the API for ESP32 and scripts to operate the toolchain ;
  - The ESP32 board itself and a USB cable to connect it to the PC.
- IDE (Integrated Development Environment): ARDUINO ;
- Install the ESP32 board drivers in the Arduino IDE ;
- Use the MQTT protocol combined with the Kmeans algorithm on a data flow.



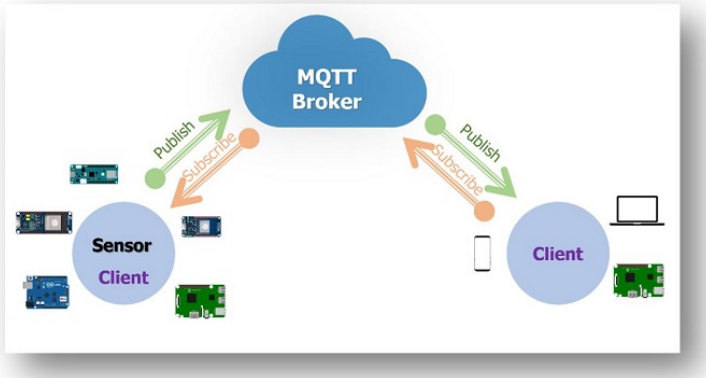


Material

Platform – Overview



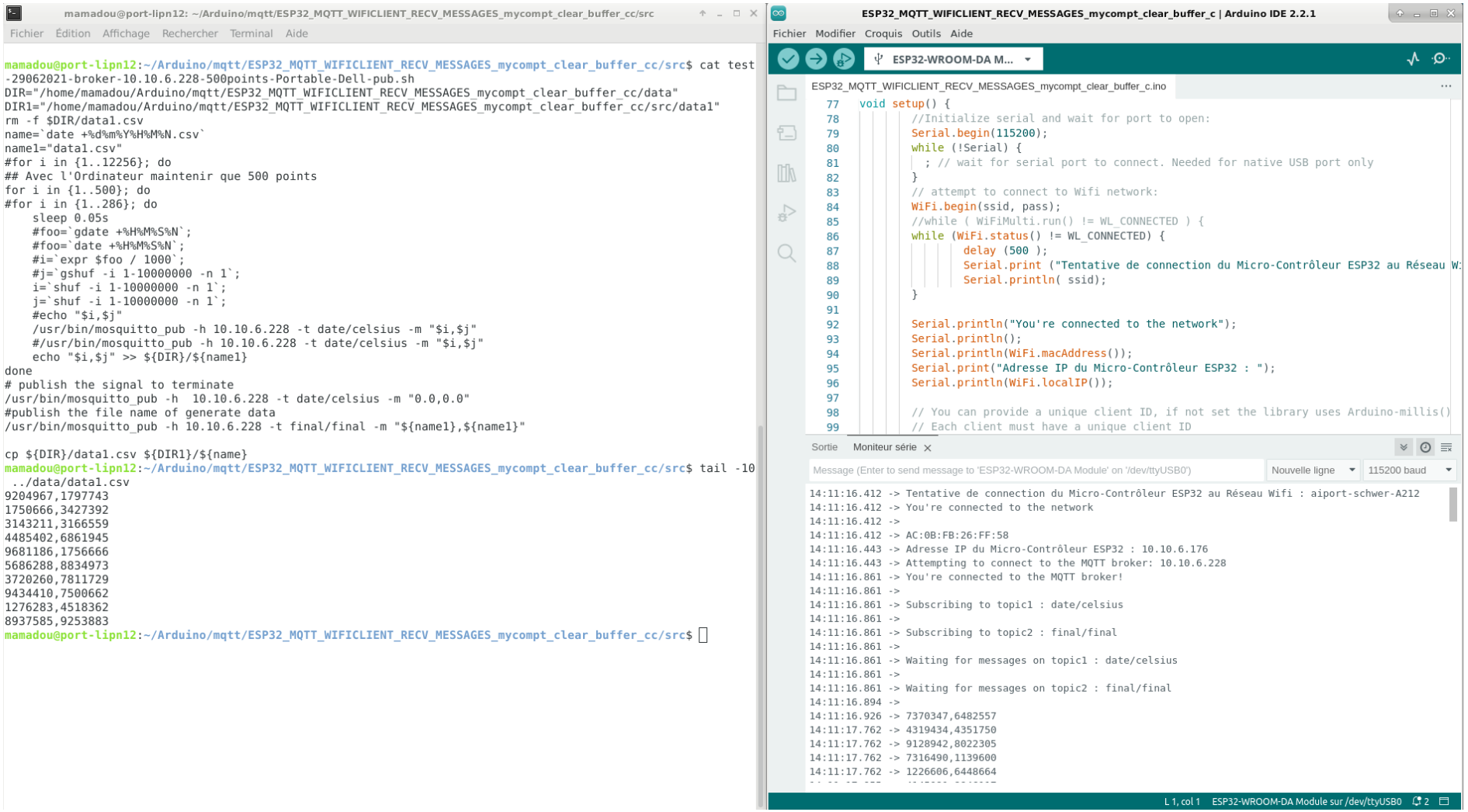
(a) Physical view



(b) Logical view

Material

Platform – Data Acquisition and Operation with Arduino



Part. 3 Conclusion and future works

Conclusion and Future works

Embedded systems and frugal AI

- Although these connected objects are far less powerful than servers in data centers, they are also, above all, less expensive and less energy-consuming.
- It's true that complex AI models can't be trained in them, but already operational algorithms can be run on them.
- If hardware requirements are too stringent, then design new approaches:
  - With simplicity in mind;
  - Motivated by the targeted ecosystem; (here the Smart Building sector);
  - With the usual rigor of experimental science:
    - Context and purpose of the experiment;
    - Describes what was done in the experiment. Includes materials used and procedures followed.
    - Presents the findings of the experiment.
    - Interprets and explains the findings.
    - Conclusion: Summarises findings and interpretations.

Conclusion and Future works

Embedded systems and frugal AI

- In our case, challenging the AI model in a data center for Smart-Building $\Longleftrightarrow$ enriching the library with solutions useful to all.

Conclusion and Future works

Embedded systems and frugal AI

- In our case, challenging the AI model in a data center for Smart-Building $\Longleftrightarrow$ enriching the library with solutions useful to all.
- Is your AI justified for your context? Perhaps you can also often do "well" with "less". We don't always have "big science" problems to deal with. We also have concrete problems to solve.

Conclusion and Future works

Embedded systems and frugal AI

- In our case, challenging the AI model in a data center for Smart-Building $\Longleftrightarrow$ enriching the library with solutions useful to all.
- Is your AI justified for your context? Perhaps you can also often do "well" with "less". We don't always have "big science" problems to deal with. We also have concrete problems to solve.
- **The vision is more important than the vision:** admittedly, these "emerging" models don't scale up and can't compete with conventional AI models in production, but the quest is not for conventional performance!

Conclusion and Future works

Embedded systems and frugal AI

- What is the future of AI for IoT in the smart building sector?
  - Is it MLops? (a set of practices aimed at deploying and maintaining machine learning models in production reliably and efficiently)
  - Is it DataOps? (a collaborative data management practice focused on improving the communication, integration, and automation of data flows between data managers and data consumers across an organization)
  - Is it AIops? (strategies that leverage artificial intelligence for IT operations)

Conclusion and Future works

Embedded systems and frugal AI

- What is the future of AI for IoT in the smart building sector?
  - Is it MLops? (a set of practices aimed at deploying and maintaining machine learning models in production reliably and efficiently)
  - Is it DataOps? (a collaborative data management practice focused on improving the communication, integration, and automation of data flows between data managers and data consumers across an organization)
  - Is it AIops? (strategies that leverage artificial intelligence for IT operations)
- Anyway, the quest is to be aware of what we are doing for the planet, how we make it ... for those who will succeed us;

Thank you for your attention. Do you have any questions?

Techies or not?

https://spectrum.ieee.org/ai-supercomputer-2662304872 (Cerebras Introduces Its 2-Exaflop AI Supercomputer)

https://spectrum.ieee.org/joy-buolamwini (Why AI Should Move Slow and Fix Things – Joy Buolamwini of the Algorithmic Justice League fights for the "excoded")

https://spectrum.ieee.org/solid-state-refrigerator (Electrocaloric Material Makes Solid-State Fridge Scalable – Refrigerant-free refrigerators would be portable and efficient)