frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 20 September 2022
DOI 10.3389/fphy.2022.957128

OPEN ACCESS

EDITED BY
Alexander Scheinker,
Los Alamos National Laboratory (DOE), United States

REVIEWED BY
Nhan Tran,
Fermi National Accelerator Laboratory (DOE), United States
Saurabh Kulkarni,
Graphcore, United States

*CORRESPONDENCE
Berthié Gouin-Ferland,
berthie.gouin-ferland@usherbrooke.ca

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 30 May 2022
ACCEPTED 29 August 2022
PUBLISHED 20 September 2022

CITATION
Gouin-Ferland B, Coffee R and Therrien AC (2022), Data reduction through optimized scalar quantization for more compact neural networks. *Front. Phys.* 10:957128. doi: 10.3389/fphy.2022.957128

Data reduction through optimized scalar quantization for more compact neural networks

Berthié Gouin-Ferland[1]*, Ryan Coffee[2] and Audrey C. Therrien[1]

[1]Interdisciplinary Institute for Technological Innovation - 3IT, Sherbrooke, QC, Canada, [2]SLAC National Accelerator Laboratory, Menlo Park, CA, United States

Raw data generation for several existing and planned large physics experiments now exceeds TB/s rates, generating untenable data sets in very little time. Those data often demonstrate high dimensionality while containing limited information. Meanwhile, Machine Learning algorithms are now becoming an essential part of data processing and data analysis. Those algorithms can be used offline for post processing and post data analysis, or they can be used online for real time processing providing ultra low latency experiment monitoring. Both use cases would benefit from data throughput reduction while preserving relevant information: one by reducing the offline storage requirements by several orders of magnitude and the other by allowing ultra fast online inferencing with low complexity Machine Learning models. Moreover, reducing the data source throughput also reduces material cost, power and data management requirements. In this work we demonstrate optimized nonuniform scalar quantization for data source reduction. This data reduction allows lower dimensional representations while preserving the relevant information of the data, thus enabling high accuracy Tiny Machine Learning classifier models for online fast inferences. We demonstrate this approach with an initial proof of concept targeting the CookieBox, an array of electron spectrometers used for angular streaking, that was developed for LCLS-II as an online beam diagnostic tool. We used the Lloyd-Max algorithm with the CookieBox dataset to design an optimized nonuniform scalar quantizer. Optimized quantization lets us reduce input data volume by 69% with no significant impact on inference accuracy. When we tolerate a 2% loss on inference accuracy, we achieved 81% of input data reduction. Finally, the change from a 7-bit to a 3-bit input data quantization reduces our neural network size by 38%.

KEYWORDS

machine learning - ML, neural network, quantization, classification, free electron lasers, data acquisition, ultra high rate detectors

1 Introduction

Detectors for large physics and light source experiments now produce data much faster than acquisition systems can collect, triage and store it [1, 2]. The current approach of saving all raw data requires a large amount of cabling, power and downstream storage, beyond what the architecture or budget can allow [1]. Thus, several current and planned experiments would benefit from data reduction at the source. Furthermore, the initial data preparation steps before analysis tend to be very similar over time - deletion of invalid events, baseline corrections and initial information extraction, such as calculating timestamps or energy. Moving these steps at the edge–near the detector–would reduce data at the source and thus lightening the load of the high speed communication system and high-speed storage. Even so, several of this initial analysis requires complex mathematical operations which require many sequential steps, an iterative approach, and significant computational resources. This limits the capacity for true real-time data reduction [3, 4].

One strategy exploits ultra low latency Edge Machine Learning (edgeML); the deployment of inference models near the detector capable of real-time analysis, veto and compression of incoming data. Machine Learning (ML) models like neural networks (NN) can be trained to emulate arbitrary mathematical operations while using simpler addition and multiplication operations that can be greatly accelerated using appropriate hardware [5]. This strategy of moving much of the data preparation steps at the source enables to reduce both data velocity and data volume, resulting in resource savings in term of data transfer, processing and storage.

The LCLS-II built at SLAC National Accelerator Laboratory is capable of generating coherent x-ray shots at a 1 MHz rate [1, 6]. The experimental hutches host several dozen different instruments to capture the maximum information about each event. However, the system must be run at a lower rate to collect the data from all these instruments and send it to disk [1]. To achieve continuous full rate experiments, a first proof of concept targeting the Cookiebox detector demonstrated that deploying ML inference models on FPGA can reduce data velocity in real-time [7].

The Cookiebox is a diagnostic detector which non-destructively samples each x-ray shot to reconstruct the single shot time–energy profile via the method of attosecond angular streaking [3]. The reconstructions are to be used to select which x-ray shots fit particular experimental objectives, rejecting invalid shots, aggregating simple reference shots, or reserving complicated shots for deeper covariance-based analysis. Such a streaming shot evaluation system significantly reduces the raw data rate from other instruments before it is written to persistent storage. However, to achieve this, each x-ray shot must be analyzed with very low latency, within about 100 μs, to avoid overly large raw data ring buffers. Such low-latency capability of edgeML has been demonstrated [7] and further work is ongoing to provide a fully working system. The Cookiebox detector produces a large volume of data, on the order of 100's of GB/s, which itself becomes a challenge when designing low complexity ML algorithms suited for limited capacity edgeML accelerators. For that reason, the compression and analysis must be distributed all along the data path, including prior to the ML algorithms. In this work, we suggest to optimally quantize the Cookiebox data before feeding it to our NN inference model. This compression strategy reduces throughput while preserving relevant information which enabled for leaner and more accurate NNs.

The Materials and Methods section begins with an overview of the CookieBox diagnostic detector, followed by a description of the quantization algorithm and the NN developed for the CookieBox. We then present the results for both the optimal quantizer model and for the ML model. Finally, we conclude with a discussion of how the quantization impacts the data and the ML model.

2 Materials and methods

2.1 Cookiebox

The diagnostic detector that we take as our demonstration use case is an attosecond angular streaking instrument composed of an array of 16 electron time-of-flight (ToF) spectrometers, illustrated in Figure 1 [8]. The spectrometers are placed on a plane perpendicular to the x-ray propagation. A micrometer wavelength infra-red laser with a circular polarization modulates the central electrical field with a period of 10–30 fs [3]. A low pressure gas is present in the center chamber. When the atoms or molecules are hit by x-rays, their electrons are ejected and collected by the electron spectrometers. This instrument can measure the polarization and the time-energy spectrum of each individual x-ray shot produced by LCLS-II (diagnostic mode), or be used to measure numerous features associated with a given target atom or molecule (experimental mode).

Each spectrometer signal is fed to a 12 bit, 6.4 GS/s analog-to-digital converter. Thus, the total instrument generates data at a rate of 1.229 Tb/s. The first data reduction step consists of identifying the time at which electrons hit the spectrometer using a peak finding algorithm. The digitized signal is thus converted into timestamps corresponding to each electron "hit". For each x-ray shot, each spectrometer collects approximately 100 hits, which are converted into 16 bit timestamps, which results in a data rate of 26 Gb/s. In experimental mode, this data is collected, however in diagnostic mode, the data must be analyzed within 100 μs to select the correct processing for each shot while avoiding large rapid memory buffers.

We choose this detector since it has recently been shown compatible with both the signal rates and the energy resolution

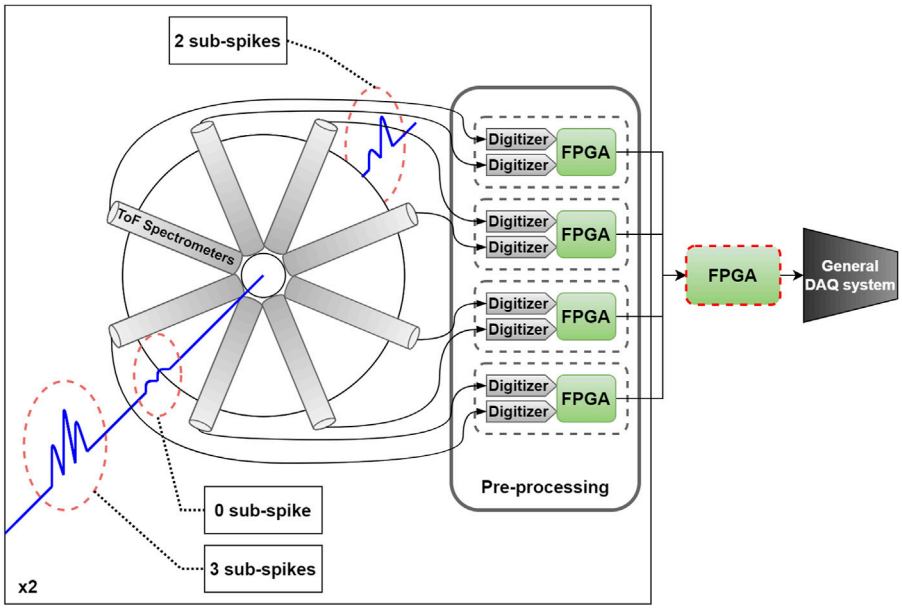

FIGURE 1
Diagram of the data flow of the CookieBox. The ToF spectrometers signals are processed by analog electronics before being digitized. Each 2 digitizers feed into an FPGA that will be hosting the neural network discussed in this article. The central FPGA, circled with a red dotted line, collects the extracted features from all 8 peripheral FPGA and forwards the inference results to the rest of the DAQ system.

required for the x-ray pulse reconstruction algorithms. The facility wide interest at LCLS-II has further motivated that such an instrument provide continuous streaming diagnoses of x-ray pulses, both time-energy distribution as well as polarization, at the full MHz repetition rate of the facility. As such, this instrument is now capable of working with the full data rate as soon as the LCLS-II ramps up to the highest quasi-continuous rate to provide offline diagnostic. The present work aims to move the diagnostic capability on the edge and in real-time.

2.2 Dataset (CookieSimSlim)

To create the initial training datasets, we use a simple simulation [9] to generate data via Monte-Carlo simulation of attosecond angular streaking. The simulation begins with a Poissonian choice of so-called self-amplified spontaneous emission (SASE) sub-spikes, forming the x-ray shot, each with an energy consistent with the few % SASE bandwidth of the FEL process and a relative temporal delay that is chosen as an even random choice across the 2π period of the angular optical cycle.

The period of the optical cycle is chosen experimentally by the choice of dressing optical laser field and is typically in the regime of few femtoseconds total period for addressing SASE structures as targeted here [3]. The data generator also allows for sub-spike polarization variation such that our model is fully compatible with the recent developments in time-dependent polarization shaping in SASE FELs [10].

The resulting dataset from the CookieSimSlim generator is an HDF5 formatted tree of events, or "x-ray shots", each with a list of electron hit energies (Xhits) for each detector angle. This list of hit energies itself is a sampling from a smooth probability distribution Y_{pdf} that is the sum of the Gaussian energy distributions for each of the sub-spikes (offset *via* $\kappa \sin(\phi)$ where κ is the streaking kick strength and $\phi \in [0, 2\pi)$ is the random phase associated with the sub-spike relative arrival timing. The shot-dependent parameters such as kick strength, phase, dark-count rate, SASE width and so on are all produced as attributes of the particular shot in the HDF5 file. For convenience the output file also includes an "image" representation of the energy hit histogram X_{img}.

The hit energies in X_{hits} are represented as 32 bit floats from the generator. This bit depth is considered a "conventional" representation since in the experiment, the data will be represented as an energy conversion of an integer arrival time that is typically only of 16 bit resolution. Allowing 32 bit floating precision for the energy mapping result is therefore considered a convenient precision for sake of the arithmetic in producing that calibrated energy for each hit.

2.3 Quantization

All of the information on the x-ray shot is contained in the timing of the electron hits; that is the 16 bit timestamps obtained

after the pulse finding step. With CookieSimSlim, this data is provided in a 32 bit float format. We encode this large set of data to a small set of optimized values modelled on the probability distribution function (PDF) of the source detector with a nonuniform scalar quantizer [11, 12].

The mean-square error (MSE) is used to judge of the quality of the quantization. The MSE is obtained with Eq. 1:

$$\mathrm{MSE}(Y, Q(Y)) = \frac{1}{N}\sum_{i=0}^{N-1}\left(y_i - Q(y_i)\right)^2 \tag{1}$$

where Y is the original discrete dataset and $Q(Y)$ is the quantized dataset.

The first order entropy of the datasets is used to mesure the amount of information it contains. The first order entropy is obtained with Eq. 2:

$$\mathrm{H}(Y) = -\sum_{i=0}^{N-1} P(y_i)\log_2 P(y_i) \tag{2}$$

where Y a discrete variable, that represent our dataset or quantized dataset, with possible outcomes $y_0 \ldots y_{N-1}$ which occur with probability $P(y_0) \ldots P(y_{N-1})$ [13]. The base two of the logarithm function is to gives the entropy in bits.

2.4 The neural networks

The data quantization allows for a much reduced input size for the convolutional neural network (CNN) which contributes to its size reduction. The CNN type of architecture is used to reduce the impact of the input data dimensionality on the model dimensionality itself. We also design our model to be the smallest as possible. For that, we use strategies inspired from the SqueezeNet CNN architecture [14]. However, since our CNN only has three convolution layers, our strategies boil down to using as few and as small as possible filters. We only use 3×3 filters which is the smallest kernel size to capture the notion of relative dependencies in all direction within a 2D space. We gradually double the numbers of filters between each convolution layer from the beginning to the end of the network like in the VGG model [15]. We use 10 filters in the first convolution layer allowing for below or close to 10,000 network parameters while potentiating the accuracy. The two last layers of the CNN are fully connected layer with 5 neurons each. For all layers, except the last one, the activation function is the rectified linear unit (ReLU) activation function. For the last layer, the Softmax activation function is used for classification.

A specific CNN is dedicated for each corresponding bit depth because of the model input size changes according to the number of bit used for the quantization. Except the input size, all the other configurations are the same for all CNNs. Figure 4 shows examples of the CNN input heatmap images in regards of the number of quantization levels. In this example, each input heatmap images (Figures 4A–C) requires a specific CNN.

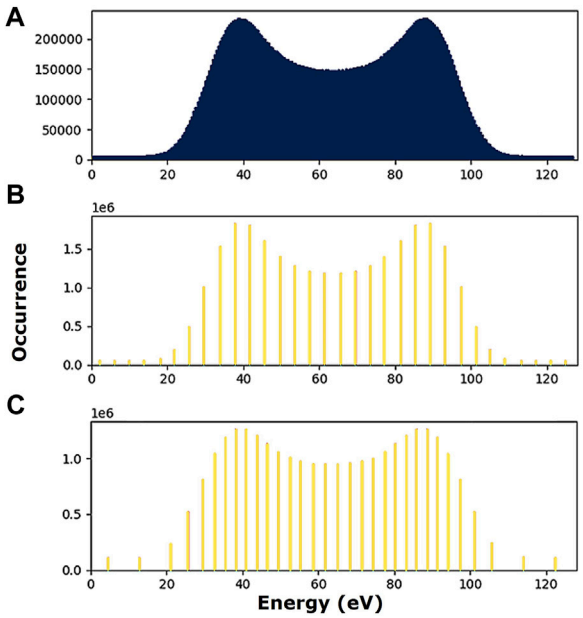

FIGURE 2
5-bit uniform quantization **(B)** and 5-bit optimized nonuniform quantization **(C)** effect on the original data distribution **(A)**. The Lloyd-Max algorithm gives more granularity to the ranges where the data are more occurring within the distribution which maximizes the entropy and minimizes the mean-square error when quantizing.

3 Results

3.1 Quantization effect on dataset

Uniform quantization allows for a quick and simple design. However, optimized nonuniform quantization minimizes the MSE, but also requires to train the quantizer beforehand. For comparison, the quantization is done using a uniform quantizer and a PDF-optimized nonuniform quantizer. In both cases, we quantized to obtain 5 strategic and realistic bit depths from 3 to 7 bits. This yields M quantization levels with $M = 2^n$ and n the bit depth.

The quantization levels for the uniform quantizer are uniformly placed within the distribution. For training the PDF-optimized nonuniform quantizer, we used the Lloyd-Max (LM) algorithm [11, 12]. The LM algorithm uses an iterative k-means clustering approach to determine which quantization level locations minimize the MSE. The initial estimate for the quantization levels are uniformly placed within the distribution. The tolerance for change in MSE after which the LM algorithm converged is set to $1e-5$. For both quantizer designs, the actual quantization is done by mapping the input value to its nearest quantization levels. Figure 2A shows the original data distribution while Figures 2B,C respectively show the distribution with uniform quantization and optimized nonuniform quantization.

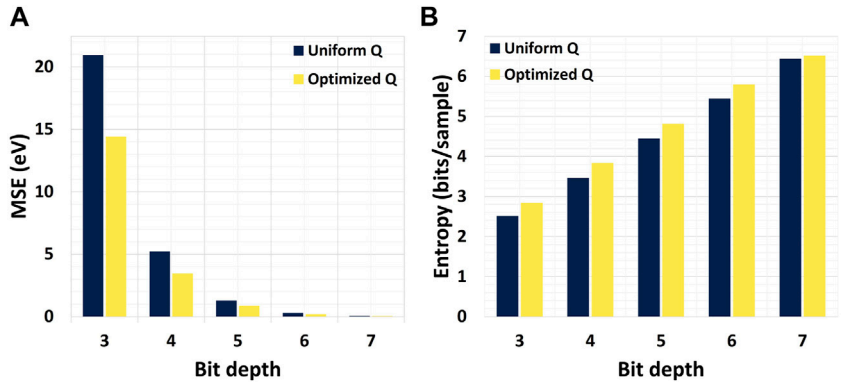

FIGURE 3
Mean-square error **(A)** and entropy **(B)** as a function of the quantization. We interpret low mean-square error with high entropy as a better information representation within data.

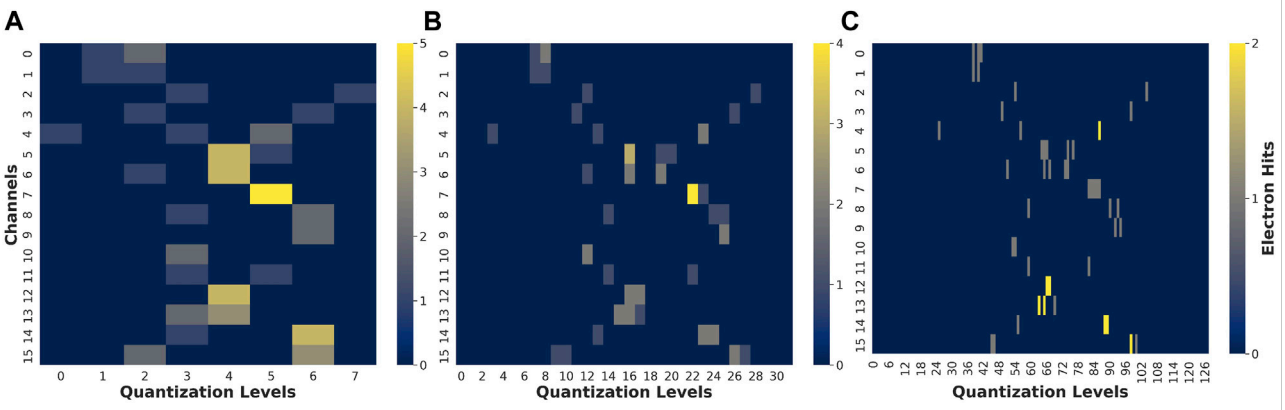

FIGURE 4
3, 5 and 7 bits quantized data heatmaps for a 2 pulses event [respectively **(A–C)**]. For a human eye, the 5 bits quantized data heatmaps **(B)** makes the two pulses distinction less ambiguous over 3 bits quantized data heatmaps **(A)**. However, the 7 bits quantized data heatmaps does not improve the distinction for a human eye.

Figure 3A shows how nonuniform quantization minimized the MSE compared to a uniform quantization. In addition, Figure 3B shows how nonuniform quantization also maximized the entropy. The data are quantized using the uniform and nonuniform quantizer and then converted as an heatmap with the bin intervals being the quantization levels. This result is an input heatmap image of size 16 ×M and it is the input of the CNN. Each pulse within an x-ray shot will create a vertical sinusoidal wave where the relative phase between waves reflects the time interval between the pulses. Figure 4 shows how quantizing with 5 bits over 3 bits makes the pulse count less ambiguous, but also how quantizing with 7 bits does not drastically simplify the pulse counting task (for a human eye).

3.2 Classification accuracy and model size

We trained the CNNs to classify the pulse count in every x-ray shot event (i.e. heatmap image). A unique and dedicated CNN is trained for each bit depth and corresponding quantized heatmap image size. This is because the quantized heatmap image size determine the input size of the CNN which then impact the overall CNN dimensionality. However, all the model parameters (kernel size, number of filters. . .) and initial weights are kept steady for all CNNs.

The desired pulse count per x-ray shot may change between LCLS-II experiments. For that reason, we trained the CNNs on a local GPU (RTX3090) to classify 0, 1, 2, 3 and “many” pulses for every shots. The “many” class correspond to all events with

TABLE 1 CNN training Configurations.

Loss Function	Sparse Categorical Crossentropy
Optimizer	Adam
Learning Rate	0.001
Batch Size	2042
Validation Split	0.2

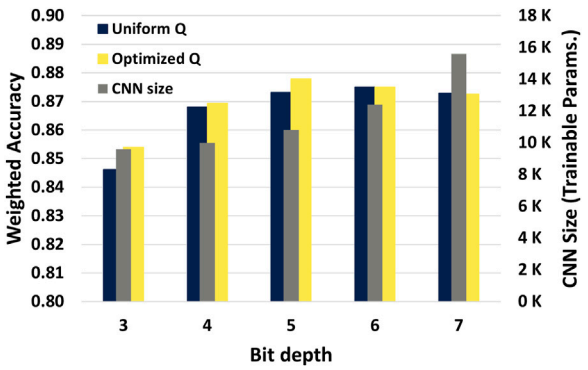

FIGURE 5
CNN weighted accuracy and size. Grey bars read on the right axis and the blue and yellow bars on the left axis. The model shows higher accuracy with 5-bit depth over 7-bit depth while requiring fewer model parameters.

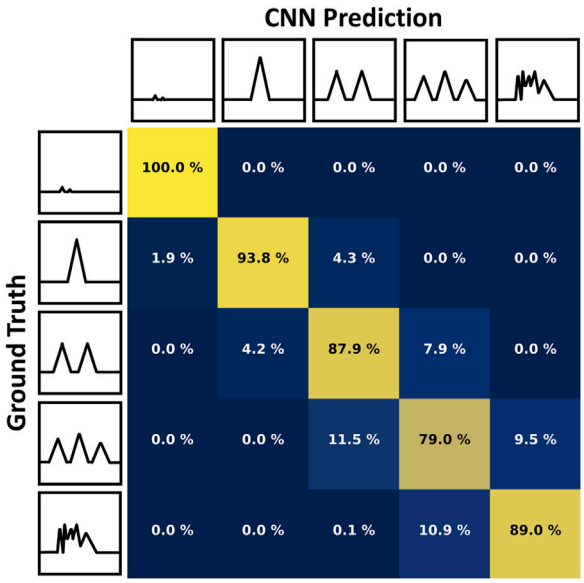

FIGURE 6
Confusion matrix showing the performance of the 5-bit dedicated CNN in predicting the number of pulses (0, 1, 2, 3 or "Many") in a single x-ray shot with data optimally quantized using 5 bits. A perfect CNN would have 100% (yellow) in every box on the diagonal and 0% (dark blue) everywhere else.

4 pulses or more, which have little value in most experiments and are generally rejected. The training setup is described in Table 1. The training set includes 400,000 events and the test set 100,000 events. All the data are generated using the CookieSimSlim generator. Figure 5 shows the relation between the CNN weighted prediction accuracy as well as the CNN size when applying the method in simulation with the test set. We see that this optimized quantization scheme allows for data reduction on 5 bits while allowing for more accurate and leaner inference models then when using 7 bits. Figure 6 shows the confusion matrix of the 5-bit dedicated CNN in predicting the number of pulses.

4 Discussion

4.1 Quantization

Our goal for using optimized nonuniform quantization over uniform quantization was to maximize the information representation of the Cookiebox source on a lower bit budget. This is what Figure 3 suggest. We saw that while a nonuniform quantization minimized the MSE compare to a uniform quantization, it also maximized the entropy which represent the amount of information carried out by the data. Nonetheless, we also saw that the gap in performance tends to shrink for a larger bit depth. This is because the optimized quantization levels converge towards a uniform distribution as more levels are created within the same limited interval. We noticed that this nonuniform quantization is sensitive to changes in the data distribution; if the source statistic changes overtime a mismatch effect could occur and change the quantizer performance. We recommend training the nonuniform quantizer on a dataset that includes those variations or to include a calibration step to train the quantizer before the data acquisition to ensure the quantizer representativity. Note that the same mismatch effect would occur to a standalone NN (i.e. without the prior optimized nonuniform quantization).

Nevertheless, quantization allows to pass from a 16-bit scalar data representation to a 7, 5 and even 3-bit representation. This yields a data reduction of 56%, 69 % and 81%. Even if quantization is a lossy coding, let's recall that optimized nonuniform quantization allows for data reduction while maximizing the information retention. This method avoids the computational load of lossless coding, which reduces the acquisition system latency and overall resources usage.

4.2 Convolutional neural network inference model

The goal of limiting the Cookiebox data dimensionality was also to reduce the input dimensionality for our inference model. Fully connected NNs are really sensitive to their input size and tend to become very large if not contained. We used a simple CNN architecture to minimize this effect, but Figure 5 still shows the benefits of a small input size in term of model dimensionality. For instance, with a straightforward 16-bit representation and the same architecture, this CNN would require approximately 2.6 million parameters. By contrast, our 7, 5 and 3-bit representations shrunk the CNN size to 15,595, 10,795 and 9595 parameters respectively, a reduction of two orders of magnitude compared to the 16 bit input model. Within these smaller models, the change from a 7-bit to a 5-bit input data quantization reduces the CNN size by 31%, with no significant impact on inference accuracy. When we tolerate a 2% loss, the change from a 7-bit to a 3-bit input data quantization reduces the CNN size by 38%. Note that the CNN size reduction is only due to the input dimensionality reduction and that no optimization (weight pruning, weight quantization. . .) is done on the model itself.

We saw small improvements in accuracy between uniform and nonuniform quantization in Figure 5, but as for the MSE and entropy, the difference tend to plateau beyond 6 bits. Our simulation data exhibit bimodality within the distribution for all 16 channels dimensions and we expect a better gain of nonuniform over uniform quantization for more complex multimodal distributions. If the dimensions exhibit different distributions, we recommend to train and quantized in respect to each dimensions. With that said, our model still demonstrates better performances then the first iteration of NN that tackled the Cookiebox problem while being almost two orders of magnitude smaller [7].

The drop in accuracy from 6 to 7 bits correlates with a significant input size growth. Because the number of filters and kernel size are constant for all bit depth, it limits the learning potential of the CNN when having larger and more complex input. A solution to that is to use a bigger CNN. However, we would also need a bigger dataset to maintain the model generalization ability. Because our goal was to designed a small NN and to reduce data generation, we do not consider going bigger and deeper a viable, sustainable and elegant solution for edgeML.

Finally, we used scalar quantization as a proof of concept, but the next step is to use vector quantization with the Linde-Buzo-Gray algorithm to compress even more multidimensional data while conserving relevant information [16]. This could be even more promising for data source reduction and for smaller edgeML models.

5 Conclusion

Large physics experiments now produce more and more data at an ever-increasing throughput. Simultaneously, ML is becoming more popular among the community for its ability to model complex systems and the growing ML hardware accessibility. In addition to that, edgeML is a promising tool for large science experiment online data reduction. However, edgeML applications face challenges in terms of power efficiency and for hardware implementation. Moreover, some applications like the Cookiebox diagnostic detector require ultra low-latency inference.

In this work, we combined optimized data quantization with the generalization capacities of NN to reduce data source throughput while preserving relevant information and thus reducing material cost, power and data management requirements. This approach also enables smaller NN for fast real-time–on the edge–inferencing. The real-time diagnostic function would be a huge boon for upcoming LCLS-II experiments.

Beyond energy efficiency for data management system, it is worth mentioning that the cheapest data is the data which is never generated. The Jevons paradox showed us that in many technological area, increasing a process efficiency only tend to rise its absolute usage. That is already something addressed by the communication technology community [17]. The scientific community now have a real opportunity to save in development, infrastructure and energy cost by using previously developed models directly at the source to generate as much useful information and less data.

With the development larger and faster detectors planned over the next decades in several disciplines such a medical imaging, particle physics and quantum computing, the data velocity problem will not go away. There is no universal approach; each application presents a different set of challenges. Yet, edgeML is a powerful tool that can take advantage of the inherent structure of many data types which makes it a perfect candidate for real-time data reduction in many fields.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

BGF, RC, and AT contributed to conception and design of the study. RC designed the simulation and BGF adapted it for manuscript. BGF performed the results analysis supervised by AT. BGF wrote the first draft of the manuscript. RC wrote

sections of the manuscript about the CookieBox detector. All authors contributed to manuscript revision, read, and approved the submitted version.

Funding

This research was undertaken, in part, thanks to funding from the Canada Research Chairs Program. AT hold the Canada Research Chair in Real-Time Intelligence Embedded for High-Speed Sensors, which funded the current work, materials and personnel, including BGF. RC thanks the Department of Energy, Office of Science, Office of Basic Energy Sciences for funding the development of the detector array itself under Grant Number FWP 100498 "Enabling long wavelength Streaking for Attosecond X-ray Science" and for funding Field Work Proposal 100643 "Actionable Information from Sensor to Data Center" for the development of CookieNetAE as well as the associated algorithmic methods, EdgeML computing hardware, and personnel. RC also acknowledges synergistic funding for the computational method development by the Office of Fusion Energy Science under Field Work Proposal 100636 "Machine Learning for Real-time Fusion Plasma Behavior Prediction and Manipulation".

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

References

1. Thayer JB, Carini G, Kroeger W, O'Grady C, Perazzo A, Shankar M, et al. Building a data system for lcls-ii. In: 2017 IEEE Nuclear Science Symposium and Medical Imaging Conference, NSS/MIC 2017 - Conference Proceedings. Piscataway, NJ, USA: Institute of Electrical and Electronics Engineers Inc. (2018). p. 1–4. doi:10.1109/NSSMIC.2017.8533033

2. Guglielmo GD, Fahim F, Herwig C, Valentin MB, Duarte J, Gingu C, et al. A reconfigurable neural network ASIC for detector front-end data compression at the HL-LHC. *IEEE Trans Nucl Sci* (2021) 68:2179–86. doi:10.1109/TNS.2021.3087100

3. Hartmann N, Hartmann G, Heider R, Wagner MS, Ilchen M, Buck J, et al. Attosecond time–energy structure of x-ray free-electron laser pulses. *Nat Photon* (2018) 12:215–20. doi:10.1038/s41566-018-0107-6

4. Li S, Guo Z, Coffee RN, Hegazy K, Huang Z, Natan A, et al. Characterizing isolated attosecond pulses with angular streaking. *Opt Express* (2018) 26:4531. doi:10.1364/OE.26.004531

5. Duarte J, Han S, Harris P, Jindariani S, Kreinar E, Kreis B, et al. Fast inference of deep neural networks in FPGAs for particle physics. *J Instrum* (2018) 13:P07027. doi:10.1088/1748-0221/13/07/P07027

6. Schoenlein R. *New science opportunities enabled by lcls-ii x-ray lasers*. SLAC Report SLAC-R-1053 (2015). p. 1–189.

7. Therrien AC, Herbst R, Quijano O, Gatton A, Coffee R. Machine learning at the edge for ultra high rate detectors. In: 2019 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC) (2019). p. 1–4. doi:10.1109/NSS/MIC42101.2019.9059671

8. Walter P, Kamalov A, Gatton A, Driver T, Bhogadi D, Castagna JC, et al. Multi-resolution electron spectrometer array for future free-electron laser experiments. *J Synchrotron Radiat* (2021) 28:1364–76. doi:10.1107/S1600577521007700

9. [Dataset] Coffee RN. *Cookiesimslim: A simple simulation and data generator that approximates attosecond x-ray angular streaking results for lcsl-ii algorithm development* (2022).

10. Sudar N, Coffee R, Hemsing E. Coherent x rays with tunable time-dependent polarization. *Phys Rev Accel Beams* (2020) 23:120701. doi:10.1103/PhysRevAccelBeams.23.120701

11. Lloyd S. Least squares quantization in pcm. *IEEE Trans Inf Theor* (1982) 28:129–37. doi:10.1109/TIT.1982.1056489

12. Max J. Quantizing for minimum distortion. *IEEE Trans Inf Theor* (1960) 6:7–12. doi:10.1109/TIT.1960.1057548

13. Sayood K. *Introduction to data compression, fourth edition*. 4th ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. (2012).

14. Gholami A, Kwon K, Wu B, Tai Z, Yue X, Jin P, et al. SqueezeNext: Hardware-Aware neural network design. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) (2018). p. 1719–171909. doi:10.1109/CVPRW.2018.00215

15. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. In: 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings (2015). p. 1–14. Cited By 15336.

16. Linde Y, Buzo A, Gray R. An algorithm for vector quantizer design. *IEEE Trans Commun* (1980) 28:84–95. doi:10.1109/TCOM.1980.1094577

17. Liao HT, Chen KS. *Mapping the landscape of green communications and green computing: A review based on bibliometric analysis*. In: 2021 IEEE 21st International Conference on Communication Technology (ICCT) (2021). p. 565–9. doi:10.1109/ICCT52962.2021.9658091