ATLAS PUB Note

ATL-DAPR-PUB-2024-002

2nd December 2024

Autoencoder-based time series anomaly detection for ATLAS Liquid Argon calorimeter data quality monitoring

The ATLAS Collaboration

This note introduces a prototype autoencoder-based algorithm designed to identify detector anomalies in ATLAS Liquid Argon calorimeter data. The data is represented as a multidimensional time series, corresponding to statistical moments of topological calorimeter clusters. The model is trained in an unsupervised fashion on good-quality data and is evaluated to detect anomalous intervals of data-taking. The liquid argon noise burst phenomenon is used to validate the approach. The potential of applying such an algorithm to detect arbitrary transient calorimeter detector issues is discussed.

1 Introduction

This note serves as a proof-of-concept study of autoencoder anomaly detection for the ATLAS Liquid Argon (LAr) calorimeter data quality monitoring. The LAr calorimeter [1, 2] and its data-taking performance are crucial to the ATLAS physics program. To ensure high quality data is collected, a suite of data quality monitoring (DQM) and assessment procedures is followed, both during data-taking (online) and afterwards (offline). While the LAr calorimeter has excellent data quality efficiency [3], there remain several motivations for investigating such techniques. First, the current DQM algorithms primarily focus on identifying data suffering from known types of detector issues. An anomaly detection algorithm could thus serve as an early warning system for rare, previously undiscovered detector issues. This is especially important with the changing detector conditions and the expected increase of the instantaneous luminosity provided by the Large Hadron Collider (LHC). In addition, performing DQM checks requires substantial human input. Anomaly detection algorithms could also serve in partially relieving this burden.

In the studies presented in this note, the calorimeter data is represented as a multidimensional time series. This assumes that the detector operates normally for most of the time, interrupted by (ideally) brief periods of anomalous activity. Such an approach complements conventional data quality monitoring techniques within ATLAS, which are typically based on monitoring distributions in the form of histograms.

The structure of the note is as follows: Section 2 briefly introduces the LAr calorimeter. Section 3 summarizes the overall DQM concepts in ATLAS. Section 4 describes specific LAr calorimeter DQ issues and their current mitigation, with a particular focus on the noise burst phenomenon, which is used to benchmark the anomaly detection algorithm. Section 5 introduces the autoencoder-based approach for anomaly detection. Section 6 discusses how the input time series are prepared and the algorithm training. Section 7 details the results of using the anomaly detection algorithm. Finally, Section 8 summarizes the main findings of this note.

2 Liquid Argon calorimeter

2.1 Introduction

The ATLAS [4] LAr calorimeter is a key component of the ATLAS detector, designed to trigger on and to provide precision measurements of electrons, photons, jets, and missing transverse momentum. Liquid argon is used as the active material due to its radiation hardness, linear energy response, and fast readout.

The LAr calorimeter is divided into four sections: the electromagnetic barrel (EMB) and endcap calorimeters (EMEC), the hadronic endcap calorimeter (HEC), and the forward calorimeter (FCal), as shown in Fig. 1(a). The EM calorimeter is the calorimeter part located closest to the collision point. EMB covers the pseudorapidity range of $|\eta| < 1.5$, while the endcap sections cover the range of $1.4 < |\eta| < 3.2$.[1] The three layers contain lead as the passive material, constructed in an accordion shape, to ensure complete azimuth coverage, as shown in Fig. 1(b). The large number of channels (173 312) in the EM calorimeter allows for

---

[1] ATLAS uses a right-handed coordinate system with its origin at the nominal interaction point (IP) in the centre of the detector and the z-axis along the beam pipe. The x-axis points from the IP to the centre of the LHC ring, and the y-axis points upwards. Polar coordinates (r, ϕ) are used in the transverse plane, ϕ being the azimuthal angle around the z-axis. The pseudorapidity is defined in terms of the polar angle θ as $\eta = -\ln\tan(\theta/2)$ and is equal to the rapidity $y = \frac{1}{2}\ln\left(\frac{E+p_z c}{E-p_z c}\right)$ in the relativistic limit. Angular distance is measured in units of $\Delta R \equiv \sqrt{(\Delta y)^2 + (\Delta\phi)^2}$.
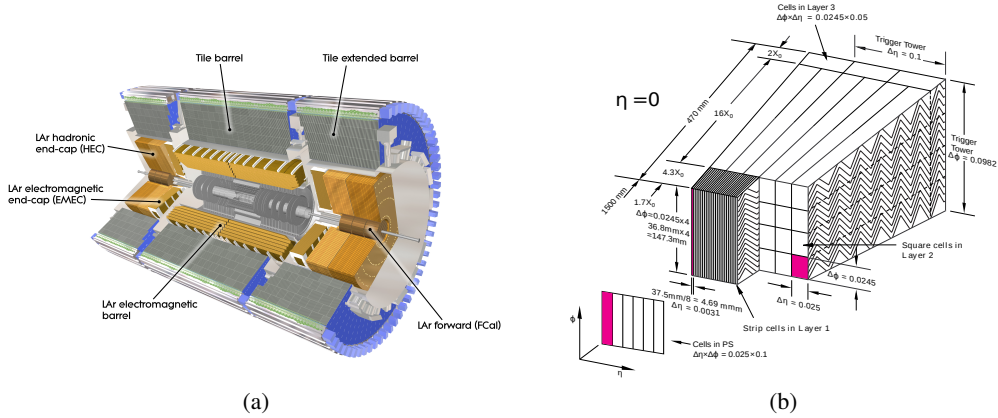
Figure 1: In (a), an image of the ATLAS calorimeter and its main components is shown (taken from Ref. [2]). The accordion-like folding of the absorber and the longitudinal readout segmentation is shown in (b) (taken from Ref. [1]).

an accurate position determination. The HEC is located outside the EM calorimeter and is composed of copper and liquid argon, providing 5632 channels. It consists of two endcap sections, covering the region of $1.5 < |\eta| < 3.2$. In the high pseudorapidity range $3.1 < |\eta| < 4.9$, the FCal is installed, using layers of copper and tungsten absorber matrices, resulting in 3524 channels. An additional letter, A or C, is added to distinguish the positive and negative pseudorapidity regions, respectively.

Complementing the LAr calorimeter, ATLAS also has a Tile calorimeter in the outer central region $|\eta| < 1.7$, which is used to measure the energy of hadrons where the radiation levels are low. While the Tile calorimeter is essential for measuring hadronic showers in the barrel region, it is not the main focus of the work presented in this note, due to a different technology to the LAr calorimeter, thus being susceptible to different types of data quality issues.

2.2 Cell measurements

Particles passing the LAr calorimeter create showers, which induce ionization in the liquid argon. The resulting electrons are collected in electric fields of about 1 kV/mm, with a drift time of around 450 ns. As the drift time is longer than the 25 ns interval between filled bunches, the calorimeter response is sensitive to collision activity from previous and subsequent bunch crossings. The 1524 front end boards (FEBs) amplify and shape the signal to a bipolar shape that integrates to zero over the entire duration. This is done to improve the signal-to-noise ratio and mitigate effects of multiple proton–proton interactions within the same (adjacent) bunch crossings, referred to as in-time (out-of-time) pile-up. An example of a typical signal pulse shape is shown in Fig. 2. Four samples of the pulse are used to evaluate the energy E_{cell}, timing τ_{cell} and quality factor Q_{cell} for a given channel via a set of linear equations:

$$E_{\text{cell}} = \sum_{i=1}^{4} a_i(s_i - P), \quad E_{\text{cell}}\tau_{\text{cell}} = \sum_{i=1}^{4} b_i(s_i - P), \quad Q_{\text{cell}} = \sum_{i=1}^{4} \left(E_{\text{cell}}(g_i - g_i'\tau_{\text{cell}}) - (s_i - P)\right), \quad (1)$$

where s_i are the sampled signal values, g_i is the ideal pulse shape and g_i' its time derivative. The signal pedestal value, P, is measured during calibration runs. The filtering weights a_i, b_i are calculated for each
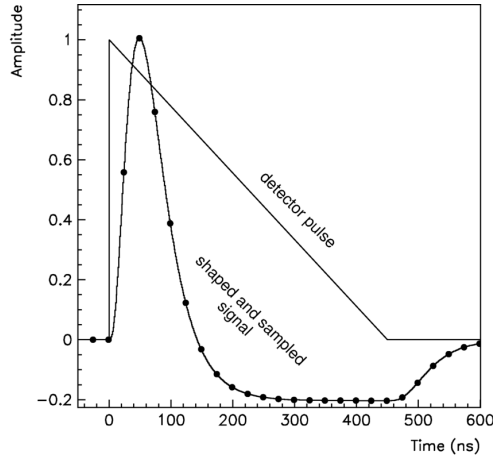
3

Figure 2: Triangular pulse of the current in the LAr cell and the FEB output signal after bipolar shaping. Also indicated are the sampling points every 25 ns. Figure taken from Ref. [2].

cell to minimize the contribution of noise and pile-up. The Q-factor reflects how much a pulse shape resembles an ideal argon ionization pulse shape, its value is limited in hardware to 16 bits (65 535). A small Q-factor indicates a good agreement of the extracted pulse with the ideal one. The timing resolution of the readout electronics is found to be below 1 ns, allowing to accurately measure out-of-time signals, veto cosmic backgrounds, reject beam-induced backgrounds and search for long-lived particles. The Q-factor is a crucial quantity in LAr data quality monitoring and is likewise used in the algorithm presented in this note.

2.3 Topo-clusters

The studies presented in this note use clusters of topologically connected cells (topo-clusters) as inputs [5]. The clustering algorithm starts with a seed cell, for which the absolute energy measurement is at least four times above the expected noise threshold, denoted by $\varsigma > 4$. The topo-cluster is expanded by adding all adjacent cells, and this expansion iteratively repeats for the neighbours of all newly added cells with $\varsigma > 2$, at which point a last layer of all adjacent cells is included. If a neighboring cell happens to be another seed cell, the two topo-clusters are merged. If the protocluster has two or more local cell maxima, defined by the additional requirement $E > 500$ MeV, it is split between the corresponding signal peaks. At this stage, the final topo-cluster is obtained. At each step, the cells under consideration can be a mixture of those from the LAr and Tile calorimeters.

Using this approach, a significant signal from a background of electronic noise and other sources of fluctuations, such as pile-up, can be obtained. Topo-clusters are routinely used as inputs for the reconstruction of higher-level physics objects, for example, jets and missing transverse momentum. Based on the clustered cells, topo-cluster moments are obtained by taking a weighted average of the cell values. The moments used in this study, namely the timing and the quality factor, are calculated by weighting the contribution of each cell by its energy squared, E_i^2 [5]. In addition, the topo-cluster timing uses only a subset of cells satisfying the noise threshold to optimize the timing resolution:

4

$$Q_{\text{cluster}} = \frac{\sum_{\{i|E_{\text{cell},i}>0\}} E_{\text{cell},i}^2 Q_{\text{cell},i}}{\sum_{\{i|E_{\text{cell},i}>0\}} E_{\text{cell},i}^2}, \quad \tau_{\text{cluster}} = \frac{\sum_{\{i|\varsigma_{\text{cell},i}>2\}} E_{\text{cell},i}^2 \tau_{\text{cell},i}}{\sum_{\{i|\varsigma_{\text{cell},i}>2\}} E_{\text{cell},i}^2}. \tag{2}$$

3 Data quality monitoring and assessment

The LHC is designed to collide proton or heavy ion bunches. A single period with beams circulating in the LHC is called a fill. Data-taking for physics is declared once the LHC achieves stable beams, at which point ATLAS ramps up the high voltage (HV) in several of its system and declares its readiness for physics data-taking. A single continuously-recorded data sample is called a run, assigned a unique six-digit identifier, and further subdivided into multiple luminosity blocks (lumiblocks, LBs). One LB typically corresponds to a time period of 60 s, during which the instantaneous luminosity, detector, and trigger configurations are considered stable. A single LB serves as a standard unit of granularity for DQM purposes. Nonetheless, rejection of poor-quality data is possible in intervals finer than a single LB, as discussed in Section 4.1.5.

ATLAS data that satisfy trigger requirements are organized into streams, based on various trigger selections and prescales [6, 7]. Calibration streams monitor the electronic noise, beam-spot position, detector alignment, among others. Physics streams contain all events satisfying trigger decisions of interest to physics analyses. The main physics stream was the primary stream that was used for developing and validating the anomaly detections algorithm described in this note. However, its reconstruction takes a significant time. Hence, to perform offline DQ validation as soon as possible, the express stream is used. The express stream contains a representative subset of data collected in the physics and calibration streams, and is promptly reconstructed at the ATLAS Tier 0 computing facility, providing offline data such as monitoring histograms. Of particular interest for LAr DQM is the `CosmicCalo` stream, which triggers on events on the basis of low-threshold calorimeter activity during empty bunch crossings, where no proton–proton collisions are expected. This provides a sample enriched in electronic noise from the calorimeter.

The DQ operational workflow [3] is illustrated in Fig. 3. Data are monitored online, focusing on minimizing data losses. The recorded data are then assessed offline in a two-stage approach. The first-pass consists of analyzing data from rapidly reconstructed streams that record only a selected subset of the data, shown in orange and green. Based on the initial assessment, updates are made to the ATLAS conditions database [8, 9]. For the LAr calorimeter, these involve setting noisy channel masks, rejecting noise bursts and identifying data corruption issues. This is discussed in greater detail in Section 4. Finally, after the calibration loop is complete, the entire data are reconstructed in bulk processing. A second update to the conditions database is possible at this stage, for example, to provide a more precise rejection of LAr noise, leading to increases in DQ efficiency. These changes are taken into account in further bulk data reprocessings, which are requested as needed.

The detector and trigger status, configuration, and other information such as calibration constants, are stored in the ATLAS conditions database. A number of applications record the detector conditions in the database, either real-time during data-taking (online) or during Tier 0 data reconstruction or later (offline). The conditions database allows for the indication of intervals where detector conditions are not nominal. These are recorded in a defect database, a subcomponent of the conditions database, at the LB granularity. Defects can be set manually by DQ experts during DQ assessment procedures, as happens for the LAr
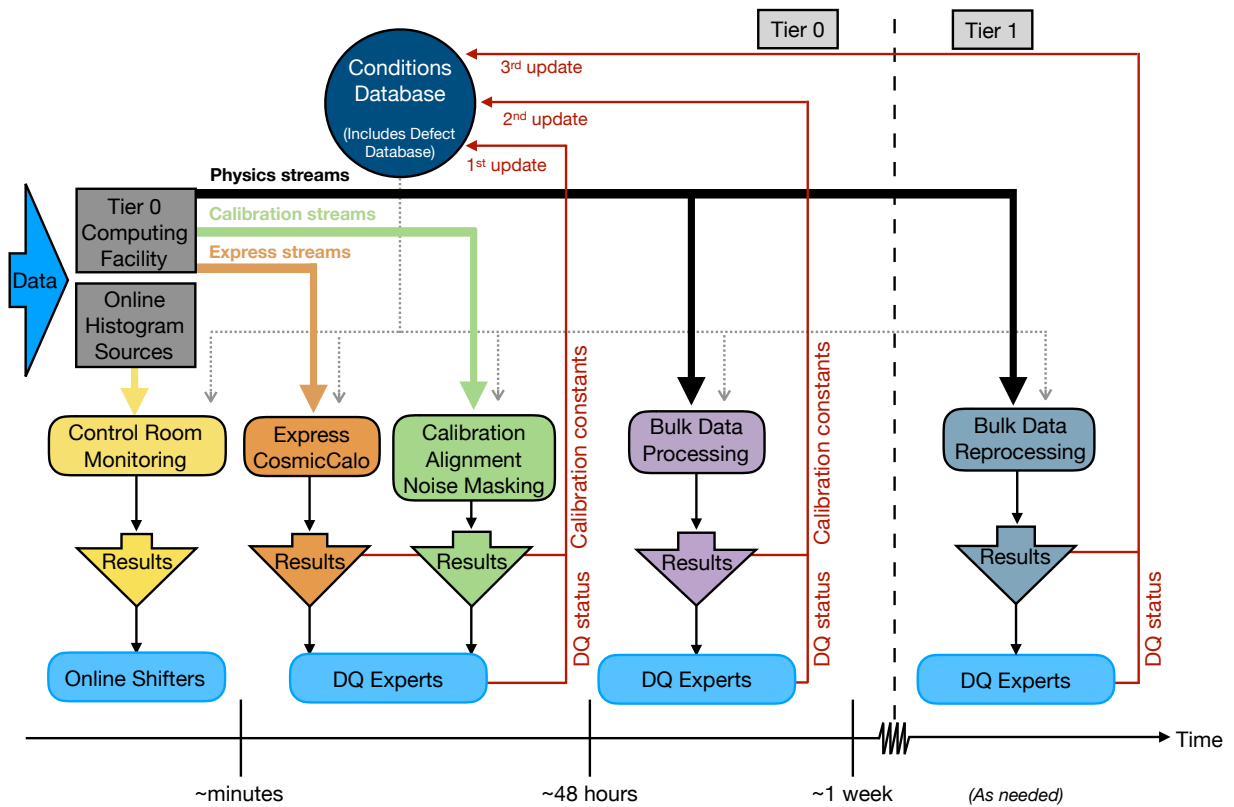
Figure 3: Schematic diagram illustrating the nominal operations workflow for the data quality assessment of ATLAS data. Online histogram sources include the high-level trigger farm, the data acquisition system, and full reconstruction of a fraction of events accepted by the trigger. Figure taken from Ref. [3].

system. Defects can also be assigned automatically by various detector systems, indicating suboptimal status of the detector hardware. The final product resulting from the DQ monitoring and assessment is the so-called Good Runs List (GRL), a set of XML files that contain the list of (Run, LB) pairs that are certified for use in physics analyses.

For the monitoring of LAr noise bursts, an alternative approach to defect rejection was developed, called the time window veto. This was done as the duration of the large majority of noise bursts is well below the typical 60 s duration of a single LB. On the other hand, the time window veto allows for the rejection of data at a much finer granularity. The timestamp when the noise burst occurred is identified and all data collected within a time window of the order of milliseconds is marked. By uploading these time windows to the conditions database, the corresponding data can be marked during the bulk reconstruction stage and later rejected in physics analyses. However, this is less flexible than the defect approach, as it requires the time windows to be defined during the initial calibration loop. The anomaly detection algorithms presented here would also be able to benefit from such functionality to reject short intervals of data.

6

4 Liquid Argon calorimeter data quality monitoring

4.1 Data quality issues

In this section, several known issues that affect the performance of the LAr calorimeter are described together with their impact on the data quality efficiency. The description is largely based on Ref. [7].

4.1.1 High voltage trips

As the electrode voltage affects the amount of signal collected, the HV lines are constantly monitored. The most common issue affecting the HV supply is a sudden drop in HV due to a current spike. When the spike occurs, the HV is reduced and gradually ramped back up to nominal values. The HV trips are logged in dedicated databases.

4.1.2 Coverage

The LAr calorimeter design nominally provides complete coverage in azimuth and up to $|\eta| = 4.9$. If this is degraded, for example due to several HV lines going down in a faulty sector, the corresponding data are rejected by marking a coverage defect.

4.1.3 Data corruption

The FEB output is the basic building block for detector information. Basic online and offline checks are performed on the FEB output to ensure data integrity. The data might be corrupted, due to, for example, a desynchronization between the FEB and the central LHC clock, among other issues.

To ensure uniform readout coverage, any event containing a corrupted block is discarded. To limit the offline rejection when a permanent corruption error is observed during data taking, the run must be paused to reconfigure the problematic FEBs.

4.1.4 Noisy channels

Special calibration runs are performed between LHC fills to extract electronic calibration constants for the calorimeter channels. These runs are likewise used as the primary means of identifying and classifying problematic channels. Depending on their behavior, cells can be masked, and their energy estimated from the average energy of eight neighboring cells in the same layer. The topo-clusters, used in the studies presented in this note, directly benefit from noisy cell subtraction. Therefore, this type of phenomenon is not explicitly targeted here.
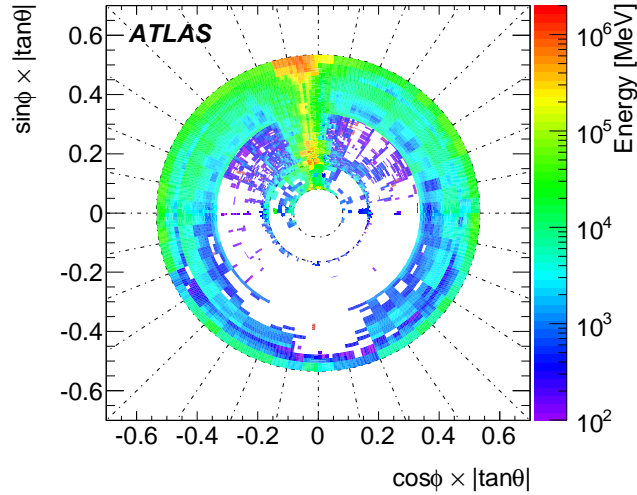
Figure 4: A typical calorimeter noise burst event, obtained during an empty bunch crossing, corresponding to an energy deposit of a few TeV. The transverse plane of the electromagnetic calorimeter endcap A (EMECA) is shown. Figure taken from Ref. [7].

4.1.5 Noise bursts

The LAr calorimeter periodically suffers from large bursts of coherent noise, the intensity of which was found to be correlated with the instantaneous luminosity. Figure 4 shows an example of a typical event containing a noise burst in one of the endcaps with a total energy of 2 TeV, recorded during an empty bunch crossing.

A simple but robust algorithm is used to flag the coherent noise, named the `LArNoisyRO` algorithm. The algorithm defines two boolean flags:

- The standard flag requires more than five FEBs containing more than 30 channels each with $Q > 4000$.

- The saturated flag requires more than 20 channels with an energy greater than 1 GeV and a saturated Q-factor (65 535).

The standard flag is defined to be sensitive to noise bursts widely spread over a partition, whereas the saturated flag can possibly be confined to a much smaller area.

The noise bursts were found to consist of a peak of hard events, i.e. those identified by the standard flag, surrounded by soft events, the latter not always identified by the standard flag. A time window veto procedure was therefore devised around the hard events. Technically, this is achieved by clustering noisy events within a period of 1 s and using the timestamps of the first and last flagged events to define a time interval. The time window is extended by ±100 ms to account for peripheral events of the noise burst. The final time window is then stored in the conditions database during the calibration loop, and read back during bulk reconstruction.

Studies of the noise burst duration revealed that most bursts last less than 0.5 s, and more than 90% of them are shorter than 0.5 μs, as shown in Fig. 5. The noise burst frequency was found to be correlated with the
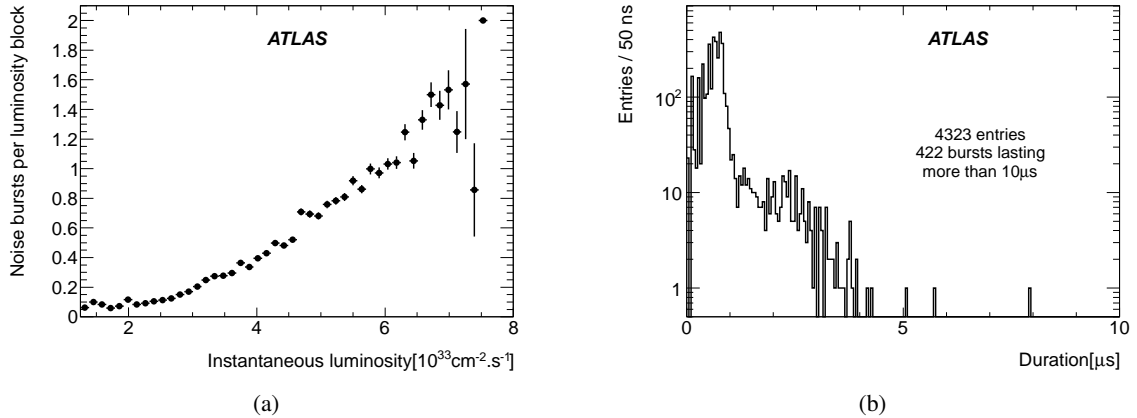
Figure 5: Distributions of (a) the number of noise bursts detected as a function of instantaneous luminosity and (b) noise burst durations (for noise bursts lasting less than $10\,\mu$s). Figures taken from Ref. [7].

instantaneous luminosity as shown in the same figure. Noise bursts serve as an excellent benchmark for generic detector anomaly detection techniques due to their transient and readily identifiable nature.

A subset of noise bursts, referred to as mini noise bursts, affect a smaller region of the detector and for an even shorter time period (tens of ns).

4.1.6 Data quality performance

Data can be marked unsuitable for physics analysis by either the defect rejection or time window veto approaches. Figure 6 shows the fraction of data lost in the 2015–2018 data-taking period in the LAr calorimeter using the two approaches, broken down by year of data-taking and type of problem. Overall, the LAr calorimeter has excellent performance, with the various data issues typically contributing to data losses below the percent level.

5 Autoencoder anomaly detection

The goal of anomaly detection studies in DQM presented here is to classify a certain time series sample with features $\vec{X}_i$, where i is the numbers of dimensions. Depending on the goal, intervals spanning from individual events to entire LBs could be identified as anomalous.

Supervised classification algorithms could in principle be used for DQM. Supervised learning requires truth sample labels, e.g. a binary label for the presence of a noise burst in the case of noise burst detection. However, due to the excellent detector performance (see Fig. 6), detector issues are rare, leading to an imbalanced class problem. Moreover, there are no reliable ground truth labels that can be used for DQ issues, and it is necessary to rely on the output of existing algorithms. This restricts the supervised algorithm to monitoring problems that are already well understood.

For the previously mentioned reasons, unsupervised algorithms, which do not rely on labelled data, are better suited for anomaly detection purposes. One type of unsupervised anomaly detection algorithm

9
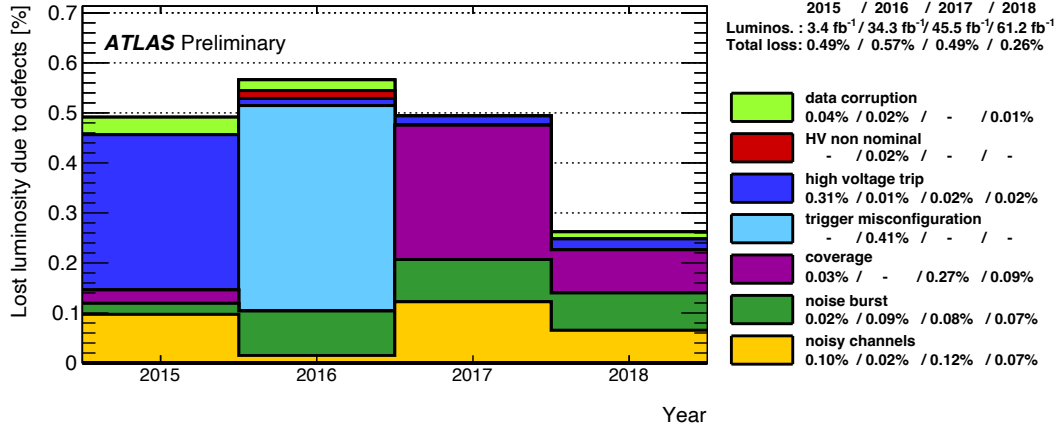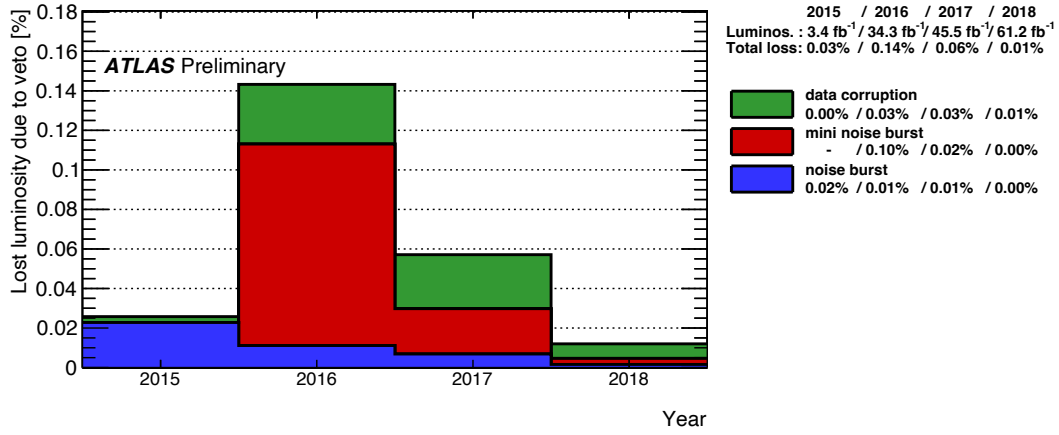
Figure 6: Fraction of data lost between 2015–2018 in the LAr calorimeter, rejected using the (a) defect-based (b) time window veto approach. Figures taken from Ref. [10]. The detector issues are explained in more detail in the text.

explored in this note is the autoencoder (AE) [11]. The AE consists of two neural networks, one called an encoder f and the other the decoder g, which try to compress and reconstruct the input with the highest possible fidelity. Crucially, the encoder needs to reduce the dimensionality of the input, otherwise it will just learn the identity matrix. During training, a mean-squared error (MSE) loss function is minimized:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} \|\vec{X}_i - g(f(\vec{X}_i))\|^2.$$ (3)

The AE learns to reconstruct the bulk of the data distribution, corresponding to the good training data. As detector issues are rare, they are more likely to be poorly reconstructed, i.e. have a higher reconstruction loss. The main challenge of unsupervised techniques is that they are generally not as powerful as supervised approaches for a given well-defined task. However, they become useful when truth labels do not exist or are noisy.

In the studies presented in this note, a long short-term memory (LSTM) [12] block is used as both the encoder and decoder. LSTM is a type of recurrent neural network, a cyclical network allowing output from some nodes to affect subsequent input to the same nodes. The main advantage of LSTM is its ability to capture long-term dependencies in the input data. A hidden state size of 64 was used for the latent layer. The network performance was not found to depend significantly on the hidden state size, although no significant hyperparameter optimization was performed beyond that. The architecture of the AE is shown in Fig. 7. The hidden state at the final time step of the encoder LSTM corresponds to the latent state. The hidden state is repeated 100 times to reproduce the length of the original time series, and is fed to a decoder LSTM. After a final dense layer, the reconstructed time series are obtained. Such techniques have been previously applied to various scenarios, for instance, for manufacturing time series data in Ref. [13].
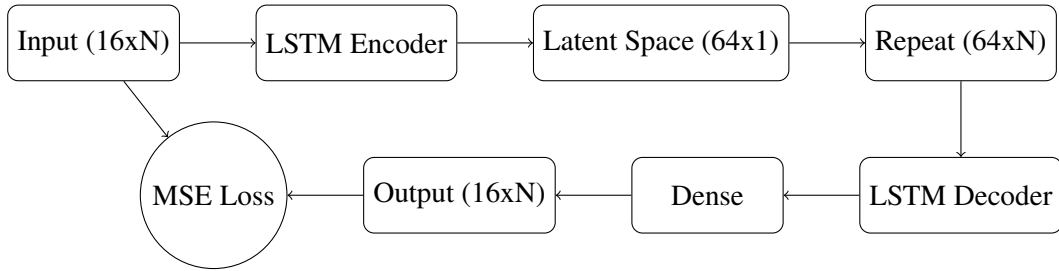

Figure 7: Autoencoder architecture with LSTM layers processing time series data. For training samples, $N = 2000$, while for inference it can be arbitrary. The loss between the input and reconstructed time series is used as an anomaly score.

6 Data preparation and network training

A sketch of the data processing pipeline is shown in Fig. 8, which aggregates and averages topo-cluster properties to obtain a set of event-level variables.

To build the time series, two topo-cluster moments are used: the Q-factor and the timing, τ. These variables are picked due to their sensitivity to different types of detector issues and similarity under different data streams. The inputs are first broken down by four detector partition: barrel C: $-1.5 \leq \eta \leq 0$, barrel A: $0 < \eta \leq 1.5$, endcap C: $-3.2 \leq \eta < -1.5$ and endcap A: $1.5 < \eta \leq 3.2$. The forward calorimeter is not included in these preliminary studies.

Two event-level quantities, the mean and the standard deviation, are computed by considering the quality factor and the timing of all topo-clusters. The mean and the standard deviation are obtained in an energy-squared-weighted approach, in the same way as the topo-cluster-level variables themselves are calculated from individual cells in Eq. 2. This results in a $2 \cdot 2 \cdot 4 = 16$ dimensional time series.

As a final step of data preprocessing, each dimension is normalized to ensure it contributes a similar weight to the reconstruction loss. The normalization constant for the Q-factor is $65\,536$, i.e. the maximum possible hardware-limited value. The normalization factor for the timing is $100\,\text{ns}$, corresponding to four times the duration between subsequent proton–proton bunch crossings at the LHC. The normalized input values are restricted to the range $[-10, 10]$ to avoid extreme outliers. The normalized input feature distributions for the training data are shown in Fig. 9. The endcaps, being more susceptible to noise, have longer tails in the timing and Q-factor distributions.
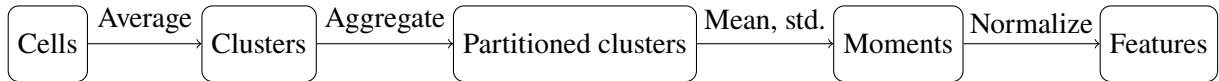
Figure 8: A sketch of the data processing pipeline.

For obtaining the AE training dataset, various runs of data-taking were considered. The first one was a proton–proton run collected in 2018 at $\sqrt{s} = 13$ TeV and recorded in the main stream. A single heavy ion run, collected in 2023 at $\sqrt{s_{NN}} = 5.36$ TeV with a bunch spacing of 50 ns in two streams, one corresponding to hard probes and the other to ultra-peripheral collisions (UPC), was also added to the training dataset. This combination was performed to obtain a model more robust to changing detector, beam and data-taking conditions. The different data samples were combined and randomly sampled.

For a more efficient training of the LSTM blocks, the data was split into uniform samples of 2000 consecutive events each. This granularity was chosen to capture most noise bursts within the training sample. A sliding window with a 50% stride was used to augment the training data, meaning the same event was typically used in two consecutive windows. The final dataset consisted of around 120 000 samples, all satisfying the GRL.

The network was trained using the Adam [14] optimizer for 100 epochs using the MSE loss (Eq. 3), with an initial learning rate of 10^{-4}, decaying by 10% each epoch. The pytorch [15] library was used for training. The training and validation loss was monitored to check for convergence and overtraining.

For inference, there is no requirement to split the data and therefore entire LBs of data can be evaluated at once. This is possible as LSTM blocks allow processing arbitrarily long inputs. For testing the algorithm, several additional datasets are employed. These include proton–proton collision data at both $\sqrt{s} = 13$ TeV and $\sqrt{s} = 13.6$ TeV. The former includes a run heavily enriched in noise bursts that is used to tune the anomaly threshold in Section 7. A run from the `CosmicCalo` stream is also considered.
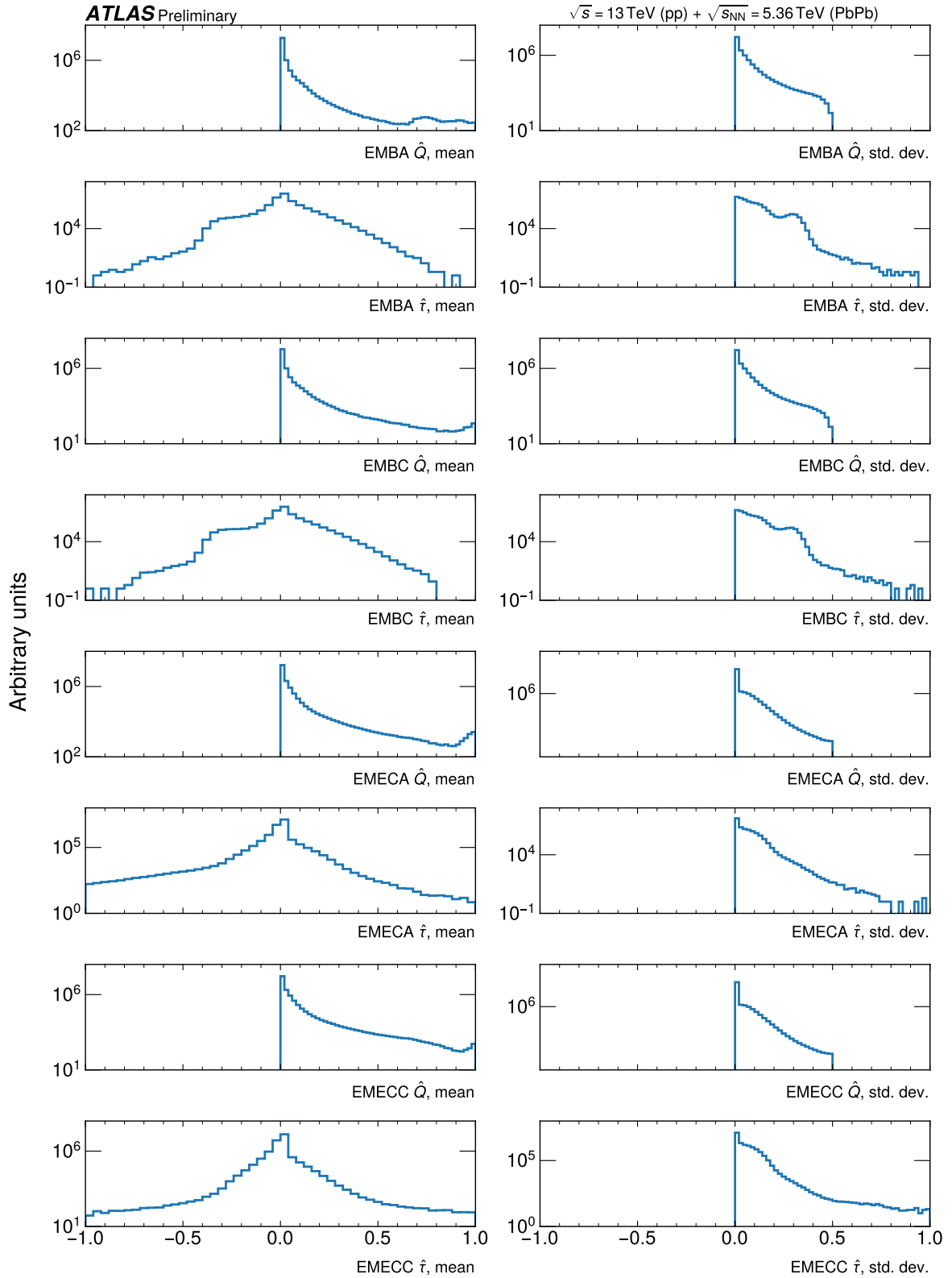
Figure 9: Distributions of normalized input features used in the autoencoder training.

7 Results

In this section, different approaches of identifying anomalous time windows are discussed. An important constraint is not to tag individual events as being anomalous, as these could be caused by rare or new physics processes.

7.1 Time window identification

Similarly to the noise burst detection algorithm, anomalous time windows can be identified in a two-stage process. First, a requirement that the reconstruction loss exceeds a pre-defined threshold, $L > L_{\text{th}}$ is imposed to define anomalous events. The choice of L_{th} would need to be optimized to achieve a good trade-off between the anomaly reporting rate and the number of false positives. In the second stage, the anomalous events can be clustered to identify anomalous time windows, taking advantage of the procedure discussed in Section 4.1.5.

As a baseline, L_{th} is optimized by considering the L distributions for events identified by the reference noise burst algorithm. These distributions are shown in Figure 10. The histograms are plotted separately for three classes of events: those not identified as noisy by the `LArNoisyRO` algorithm (label 0), those inside vetoed time windows (which are removed from physics analysis, label 3), and those that are identified as noisy but fall outside any vetoed time window (label 1). In the longer term, the threshold could be adapted by considering shifter feedback and evolving data-taking conditions.
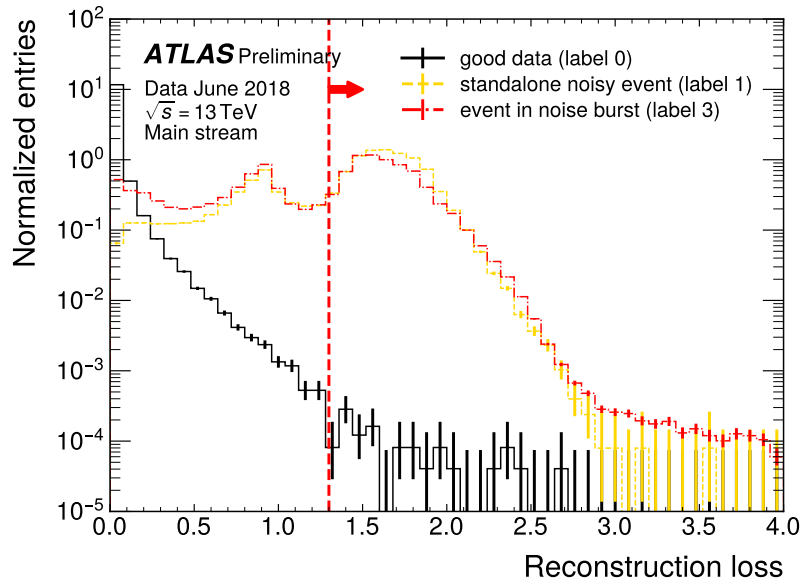

Figure 10: Distribution of event reconstruction losses for a run in the main physics stream enriched in noise bursts. The losses are broken down based on the event label returned by the reference noise burst algorithm. The chosen anomaly threshold $L_{\text{th}} = 1.3$ separates good data from the bulk of the noisy events.

Figures 11, 12, 13, and 14 show examples of anomalous time windows flagged by the AE in various LBs. The plots show the input and reconstructed time series for the different model features. The

reconstruction loss, anomaly decisions $L > L_{th}$ and the output of the reference `LArNoisyRO` algorithm are also displayed.

The model is able to identify noise bursts in the `CosmicCalo` stream as shown in Fig. 11, even though it was not exposed to that data during training. It likewise performs well for pp data (Fig. 12), where it correctly identifies the noise burst. The standalone noisy event rate is significantly reduced relative to the reference algorithm. The rate can be tuned by adjusting the cut on L_{th}.

The autoencoder complements `LArNoisyRO`, occasionally identifying additional time windows that the reference algorithm does not. In Fig. 13, this likely corresponded to the temporary failure of a vacuum diffusion pump, resulting in sporadic noise being picked up by cables feeding the signals into the FEBs several times throughout the year. The model also generalises well for lead–lead collisions, as shown in Fig. 14.

7.2 Lumiblock classification

Another possibility, more similar to the defect based DQM approach, is to use the average reconstruction loss as a summary anomaly score for an entire LB. While the average statistics are not sensitive to rare, transient effects (clustering of anomalous events is a much stronger constraint), they remain a useful metric. Figure 15 shows the evolution of the LB-averaged reconstruction loss, showing spikes in the average loss corresponding to a long-lasting noise burst or a HV trip. While the reconstruction loss shows a small overall trend as a function of the LB number (and hence the LHC delivered instantaneous luminosity), the trend is not significant compared to the anomalous LBs. Such effects could be mitigated by adding an offset to the reconstruction loss threshold, where the offset is defined based on a rolling average of the previous LB. This would allow for the AE to naturally evolve with the bulk of the data, while still identifying anomalous LB.

Figure 16 compares the LB-averaged losses from different datasets in the form of a violin plot. The noise-burst enriched LBs are separated from the training data, with significantly higher events losses. The reconstruction loss is generally low for different kinds of GRL-satisfying data: proton–proton collision data at different center-of-mass energies and heavy-ion collision data, indicating a degree of robustness to a different set of data-taking conditions. Even for GRL-satisfying data, there are individual LBs with high reconstruction losses, showing the potential of the model to discover issues in data previously flagged as physics-analysis-ready. The `CosmicCalo` dataset has a higher than average reconstruction loss, as it is biased to record events with significant calorimeter activity during empty bunch crossings.
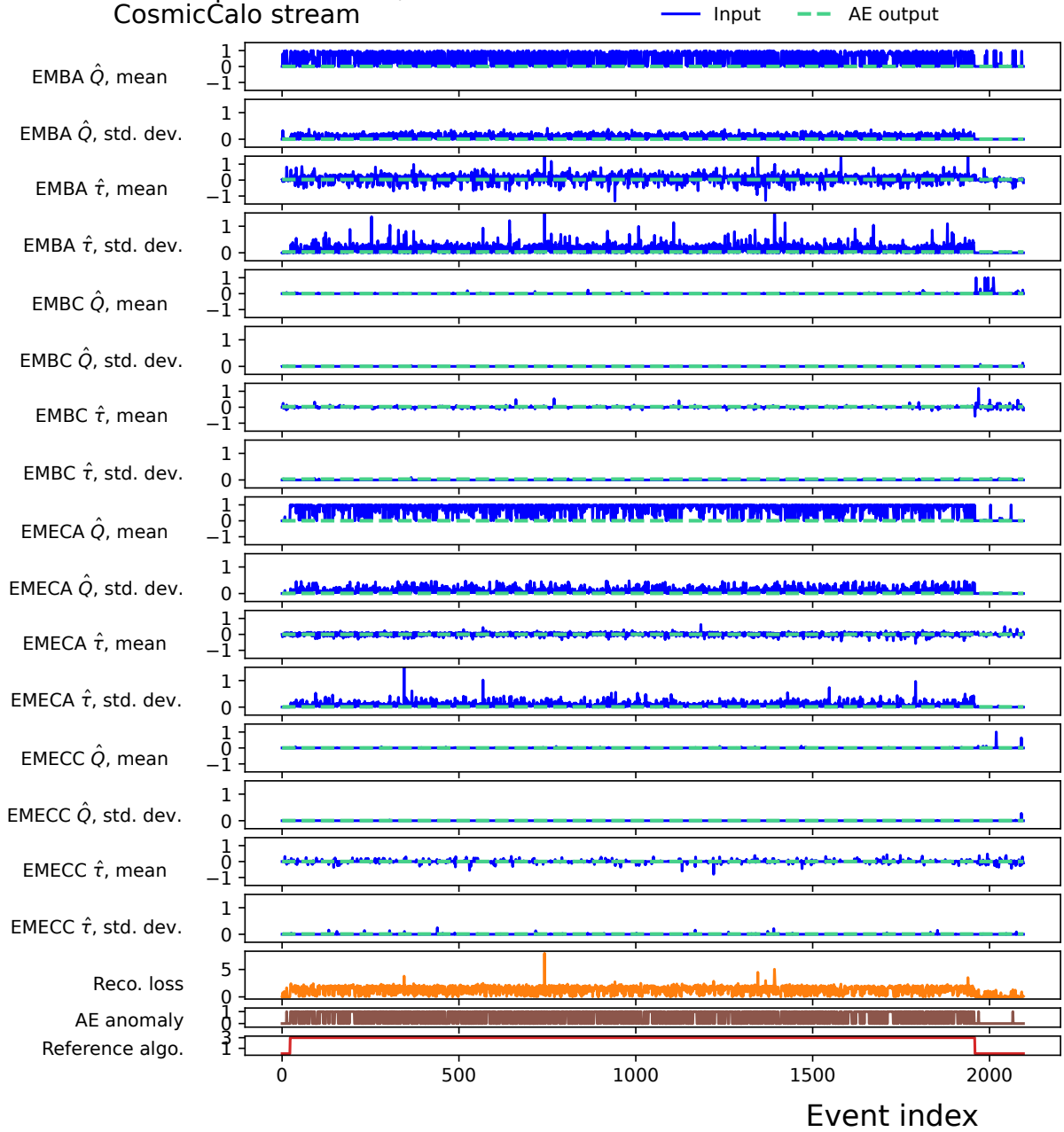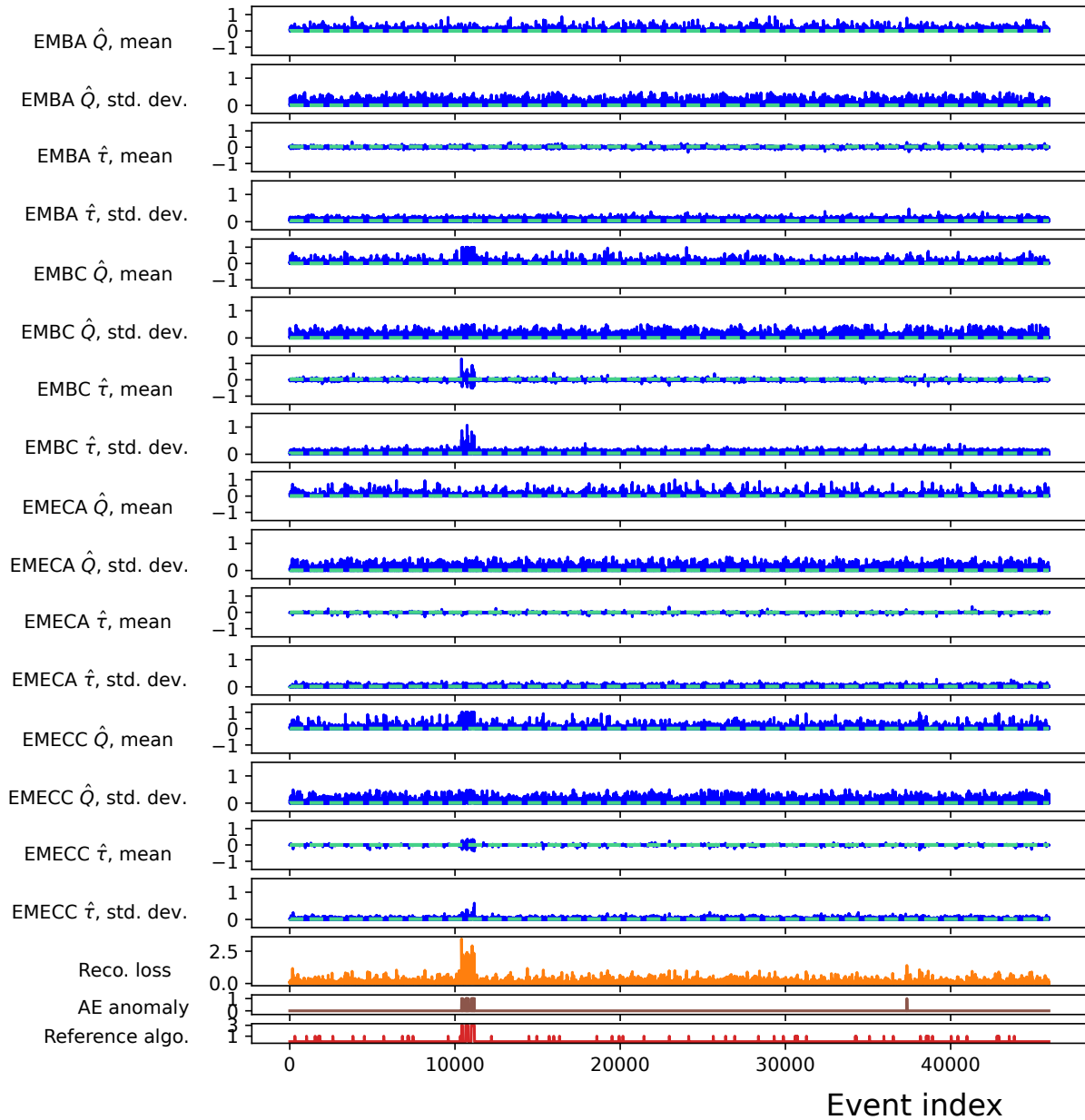
Figure 11: The original (blue solid line) and autoencoder reconstructed (cyan dashed line) time series for different input features in the different calorimeter partitions, shown in the top sixteen rows. The three bottom rows correspond to the reconstruction loss, L, events where $L > 1.3$, and the noisy event labels assigned by the reference `LArNoisyRO` algorithm, respectively. The algorithm identifies an anomalous time window that lasts for most of the lumiblock.
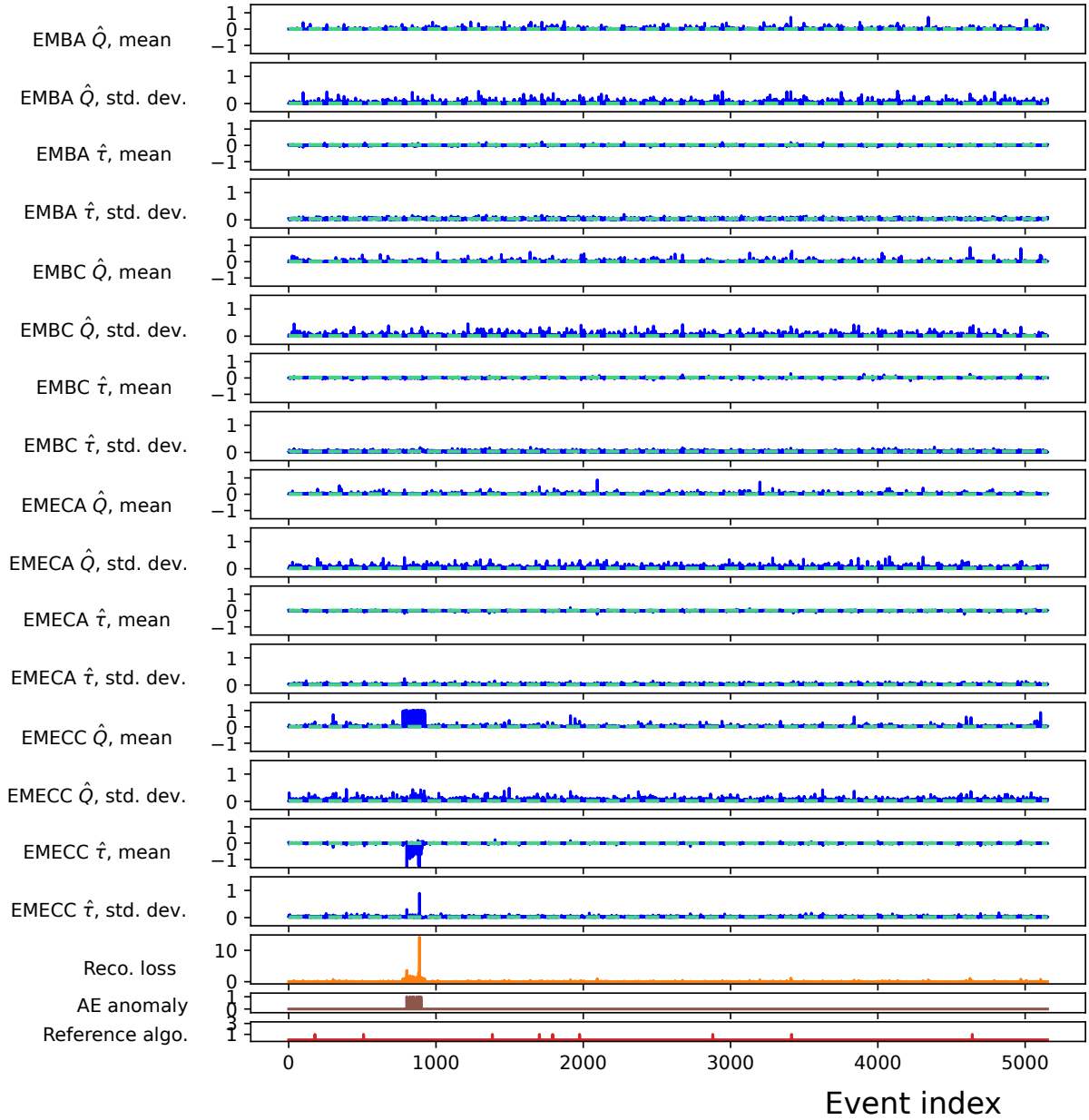
Figure 12: The original (blue solid line) and autoencoder reconstructed (cyan dashed line) time series for different input features in the different calorimeter partitions, shown in the top sixteen rows. The three bottom rows correspond to the reconstruction loss, L, events where $L > 1.3$, and the noisy event labels assigned by the reference `LArNoisyRO` algorithm, respectively. The autoencoder identifies a significant anomalous time window at an event index of around 11 000, which coincides with the reference algorithm.

Figure 13: The original (blue solid line) and autoencoder reconstructed (cyan dashed line) time series for different input features in the different calorimeter partitions, shown in the top sixteen rows. The three bottom rows correspond to the reconstruction loss, L, events where $L > 1.3$, and the noisy event labels assigned by the reference `LArNoisyRO` algorithm, respectively. The algorithm identifies a significant anomalous time window at an event index of around 900, which the reference algorithm does not.
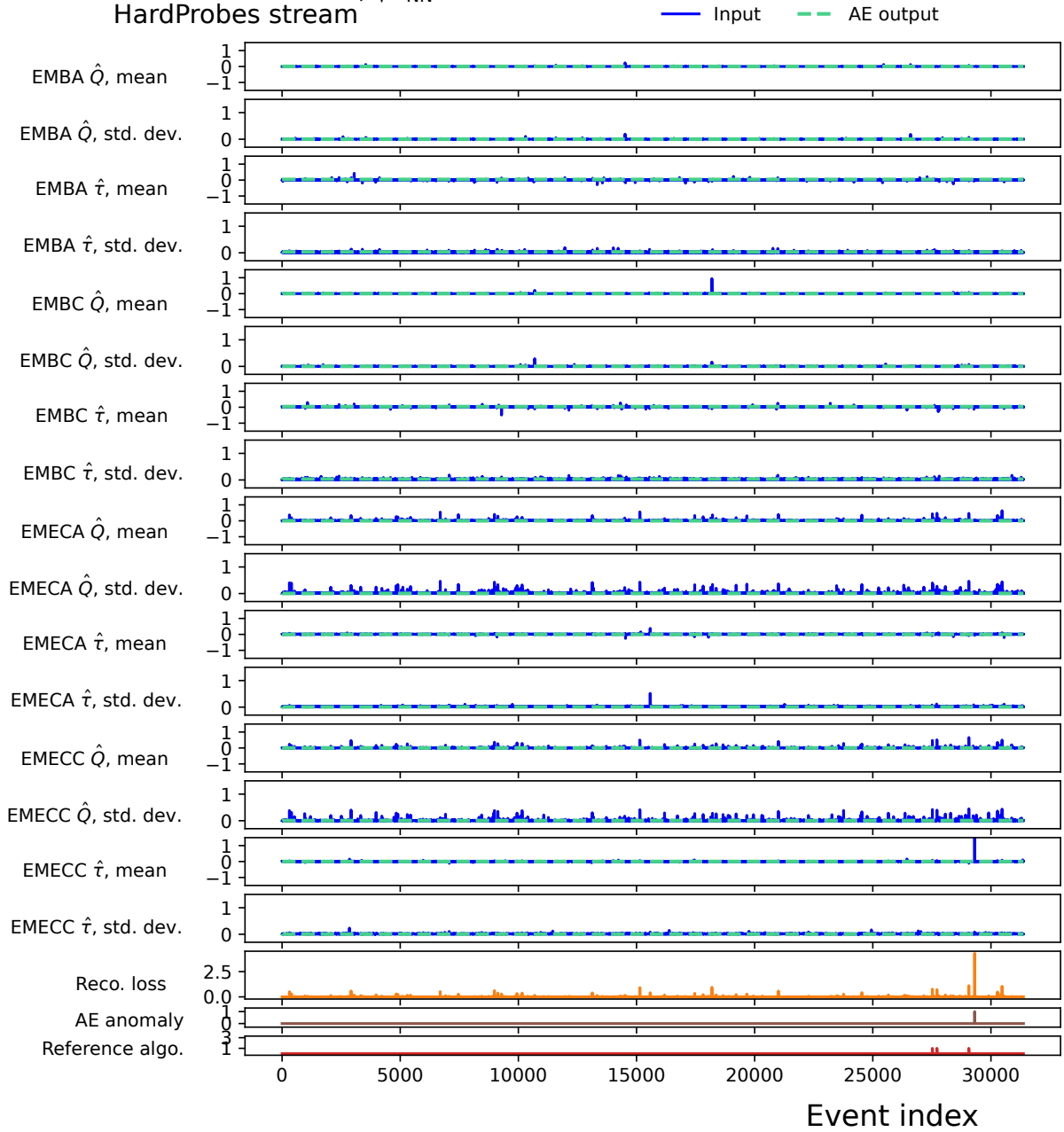
Figure 14: The original (blue solid line) and autoencoder reconstructed (cyan dashed line) time series for different input features in the different calorimeter partitions, shown in the top sixteen rows. The three bottom rows correspond to the reconstruction loss, L, events where $L > 1.3$, and the noisy event labels assigned by the reference `LArNoisyRO` algorithm, respectively.
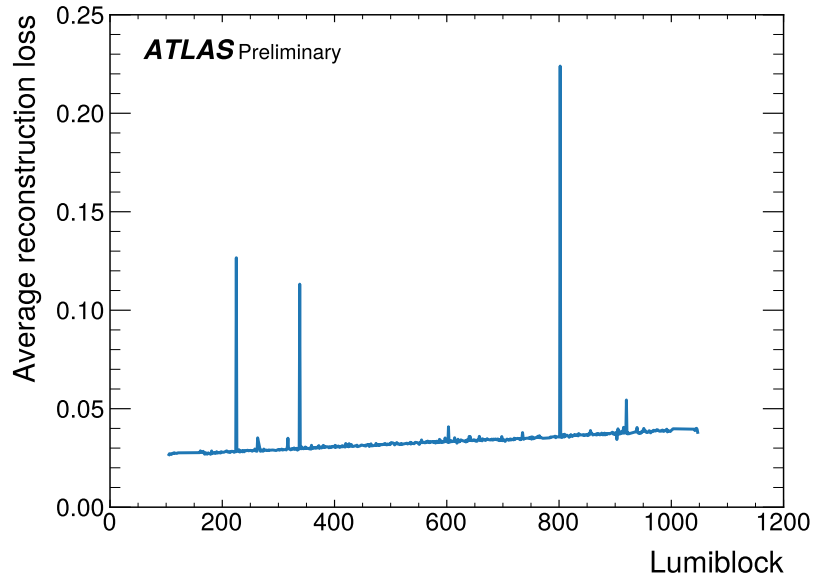
Figure 15: Evolution of average reconstruction loss as a function of the lumiblock number. The spikes in the average reconstruction loss correspond to severe noise bursts or a high voltage trip.
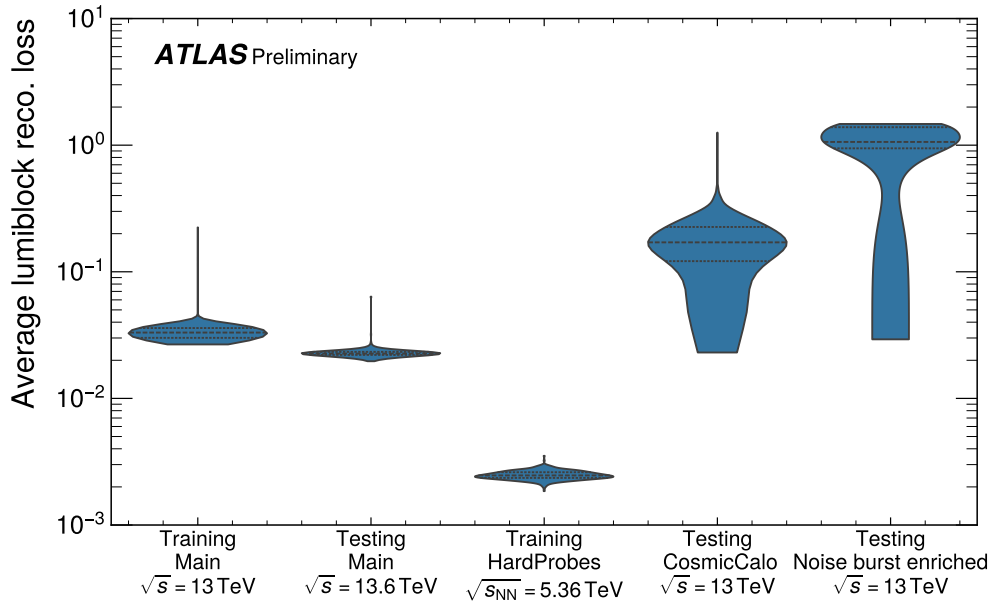

Figure 16: Average lumiblock reconstruction losses for different datasets used in training and testing the model in the form of a violin plot. The dashed lines show the quartiles.

8 Summary

A prototype autoencoder-based anomaly detection algorithm for Liquid Argon calorimeter data quality monitoring was introduced in this note. The algorithm was trained on good quality data in an unsupervised manner. Using multidimensional time series data corresponding to calorimeter cluster quality factors and timing, the algorithm is able to identify anomalous data-taking intervals. The approach works well for LAr noise burst detection, where it complements and could be integrated with the existing time-window veto procedure. The model shows promising results for different types of detector issues that it was not explicitly designed to target, thus potentially acting as a generic approach for identifying transient calorimeter issues. The algorithm was designed with different data-taking conditions in mind, incorporating both proton–proton and heavy-ion collision data in its training. It could help reduce the shifter burden and act as a warning system for previously unknown or new detector issues, which could be provoked by changing detector and data-taking conditions.

References

[1] ATLAS Collaboration, *ATLAS Liquid Argon Calorimeter: Technical Design Report*, ATLAS-TDR-2; CERN-LHCC-96-041, 1996, URL: https://cds.cern.ch/record/331061 (cit. on pp. 2, 3).

[2] ATLAS Collaboration, *ATLAS Liquid Argon Calorimeter Phase-I Upgrade: Technical Design Report*, ATLAS-TDR-022; CERN-LHCC-2013-017, 2013, URL: https://cds.cern.ch/record/1602230 (cit. on pp. 2–4).

[3] ATLAS Collaboration, *ATLAS data quality operations and performance for 2015–2018 data-taking*, JINST **15** (2020) P04003, arXiv: 1911.04632 [physics.ins-det] (cit. on pp. 2, 5, 6).

[4] ATLAS Collaboration, *The ATLAS Experiment at the CERN Large Hadron Collider*, JINST **3** (2008) S08003 (cit. on p. 2).

[5] ATLAS Collaboration, *Topological cell clustering in the ATLAS calorimeters and its performance in LHC Run 1*, Eur. Phys. J. C **77** (2017) 490, arXiv: 1603.02934 [hep-ex] (cit. on p. 4).

[6] ATLAS Collaboration, *Performance of the ATLAS trigger system in 2015*, Eur. Phys. J. C **77** (2017) 317, arXiv: 1611.09661 [hep-ex] (cit. on p. 5).

[7] ATLAS Collaboration, *Monitoring and data quality assessment of the ATLAS liquid argon calorimeter*, JINST **9** (2014) P07024, arXiv: 1405.3768 [hep-ex] (cit. on pp. 5, 7–9).

[8] E. J. Gallas et al., *Conditions and configuration metadata for the ATLAS experiment*, J. Phys. Conf. Ser. **396** (2012) 052033, ed. by M. Ernst, D. Düllmann, O. Rind and T. Wong (cit. on p. 5).

[9] ATLAS Collaboration, *Software and computing for Run 3 of the ATLAS experiment at the LHC*, (2024), arXiv: 2404.06335 [hep-ex] (cit. on p. 5).

[10] ATLAS Collaboration, *LArCaloPublicResults2015*, CERN, 2015, URL: https://twiki.cern.ch/twiki/bin/view/AtlasPublic/LArCaloPublicResults2015 (visited on 07/05/2023) (cit. on p. 10).

[11]   G. E. Hinton and R. R. Salakhutdinov, *Reducing the Dimensionality of Data with Neural Networks*, Science **313** (2006) 504 (cit. on p. 10).

[12]   S. Hochreiter and J. Schmidhuber, *Long Short-Term Memory*, Neural Computation **9** (1997) 1735, ISSN: 0899-7667 (cit. on p. 11).

[13]   R.-J. Hsieh, J. Chou and C.-H. Ho, 'Unsupervised Online Anomaly Detection on Multivariate Sensing Time Series Data for Smart Manufacturing', *2019 IEEE 12th Conference on Service-Oriented Computing and Applications (SOCA)*, 2019 90 (cit. on p. 11).

[14]   D. P. Kingma and J. Ba, *Adam: A Method for Stochastic Optimization*, 2017, arXiv: 1412.6980 [cs.LG] (cit. on p. 12).

[15]   A. Paszke et al., *PyTorch: An Imperative Style, High-Performance Deep Learning Library*, 2019, arXiv: 1912.01703 [cs.LG] (cit. on p. 12).