Quantum Autoencoders for Anomaly Detection to Identify Gamma-Ray Bursts in the Ratemeters of the AGILE Anti-Coincidence System

Author:
Alessandro Rizzo

Co-authors:
Nicolò Parmiggiani, PhD
Farida Farsian, PhD
Andrea Bulgarelli, PhD

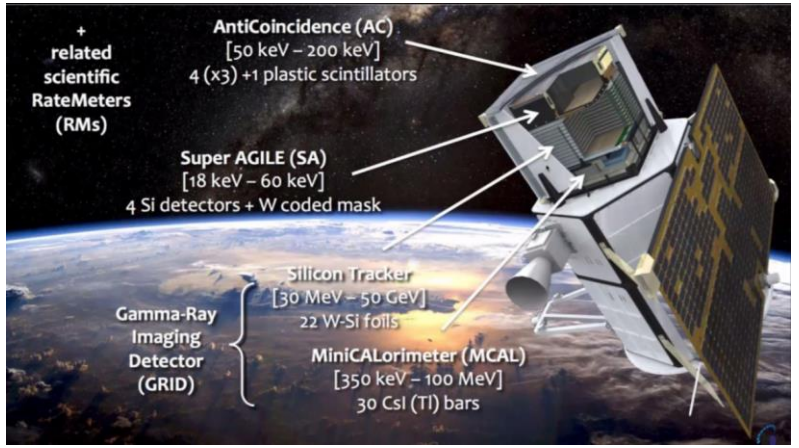
INAF – Spoke 10

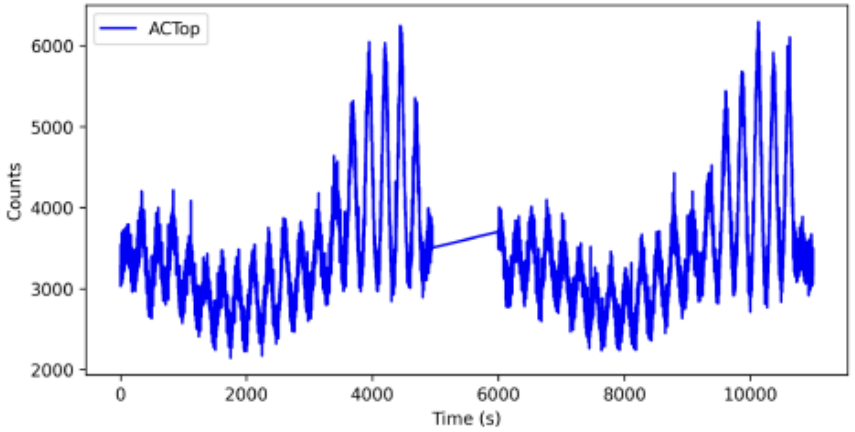
15 October 2024 – INAF Central Scientific Unit VIII-Computing

A novel approach to Gamma-Ray Burst detection using quantum computing and deep learning

AGILE Gamma-Ray Burst Detection

- **AGILE** is a space mission launched from the Italian Space Agency (ASI) in 2007 to study X-ray and gamma-ray phenomena through data acquired by different instruments onboard the satellite.
- **AGILE** ended operations in January 2024.
- The **Anti-Coincidence System** (ACS) is part of the **Gamma-Ray Imaging Detector** (GRID). It is composed of five panels and it can detect photons. The ACS records each panel count rate in telemetry as ratemeters (RM) data, with 1.024 seconds resolution → Each ACS panel RM count rate constitutes a different time series.

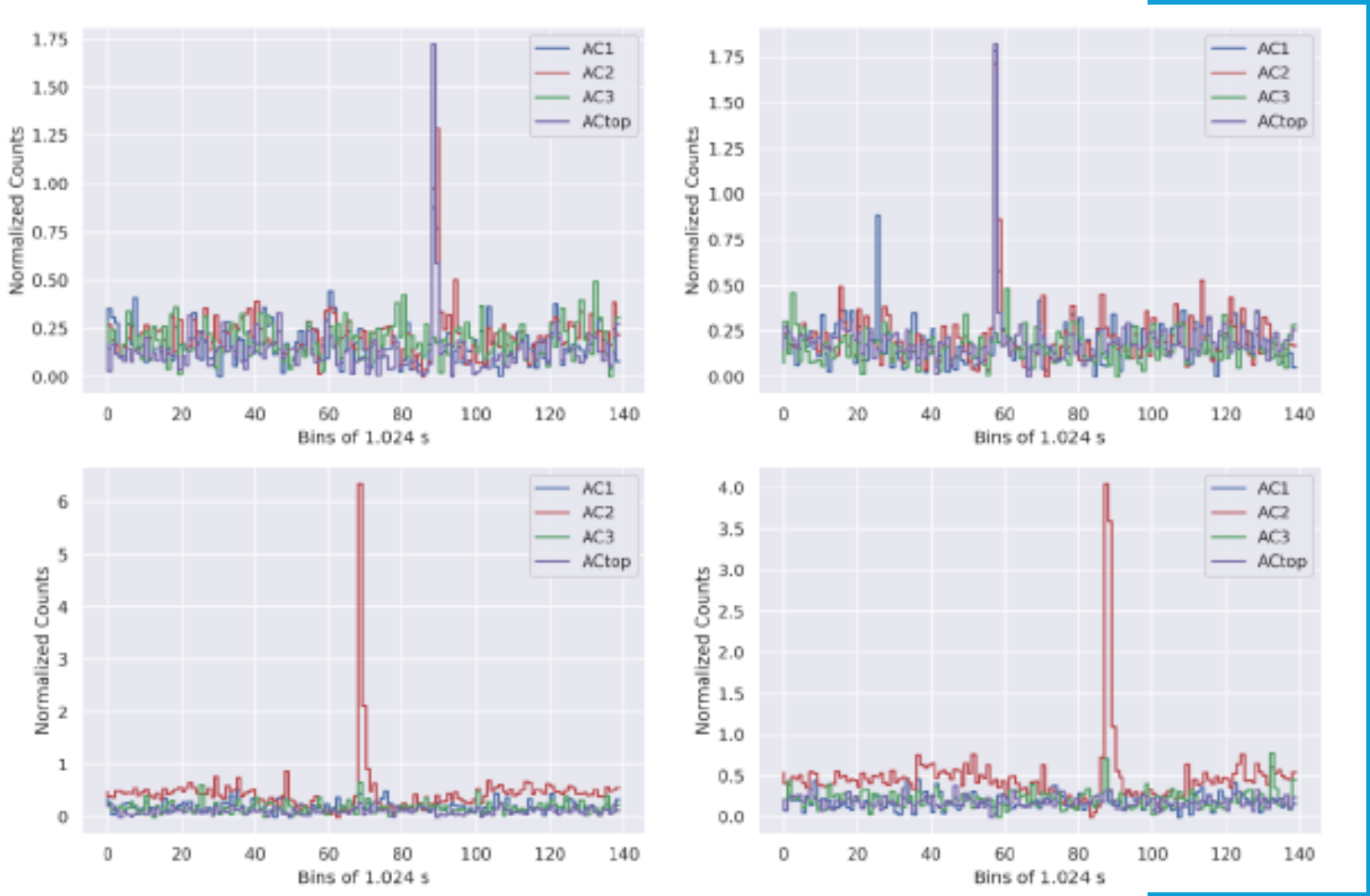




N. Parmiggiani et al, **The AGILE real-time analysis software system to detect short-transient events in the multi-messenger era**. Astronomy and Computing, Volume 44, 2023.

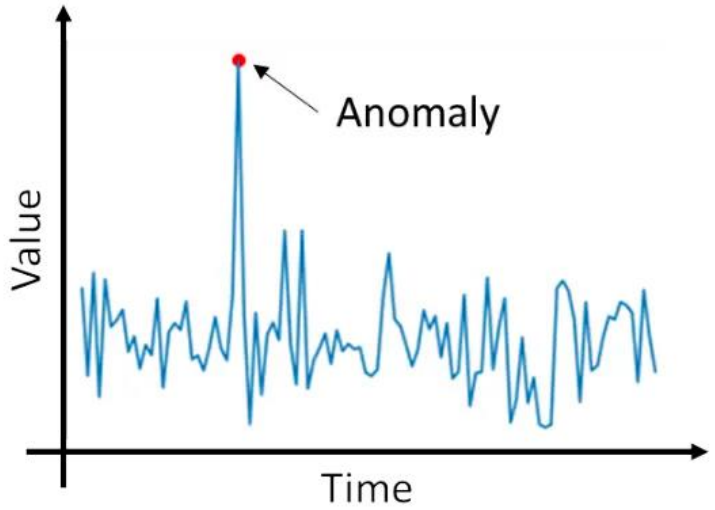
Gamma-Ray Bursts

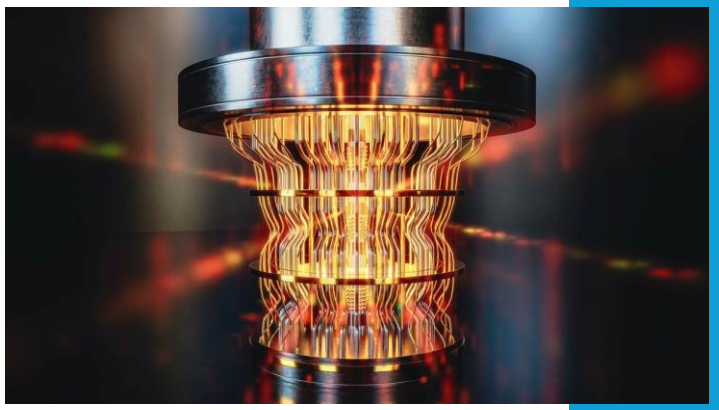
- **Gamma-ray bursts** (GRBs) are energetic explosions that have been observed in distant galaxies.
- They can last from a few milliseconds to several minutes and can be hundreds of times brighter than an average supernova.
- A **light curve** is a time series representing the count rate detected by the ACS, plotted as a function of time. The number of photons received is shown on the y-axis, while time is displayed on the x-axis.



The goal of this work

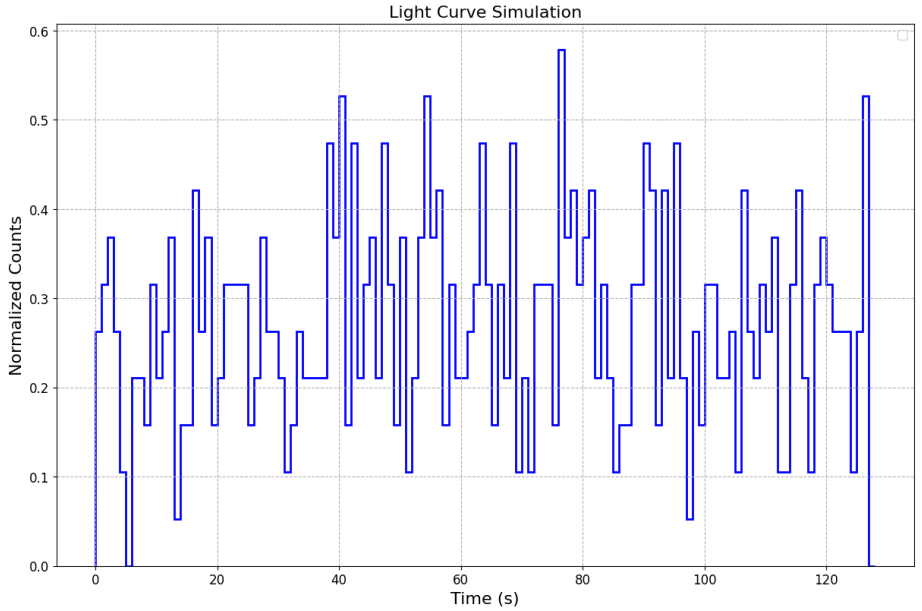
- **Anomaly detection** involves identifying data points or patterns that significantly deviate from the expected or normal behavior within a dataset. In this case, the task is to detect anomalies in background light curves, where such anomalies may indicate the occurrence of a GRB.
- **Quantum approach** → Quantum Autoencoder
- Reproduce neural networks developed by **AGILE** team to compare quantum and classic machine learning algorithms.
- **Final objective** → Verify if a quantum model can replicate the results obtained by a classic machine learning model.
- **Quantum machine learning** can exploit properties of quantum mechanics such as **superposition** or **entanglement** to process and represent data in ways that classical models cannot, potentially leading to more efficient algorithms for certain tasks.

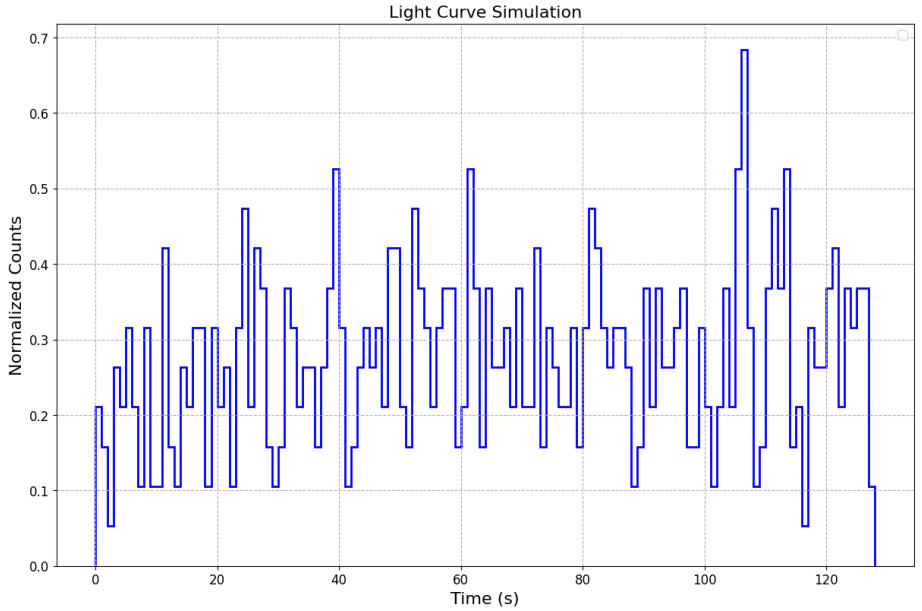




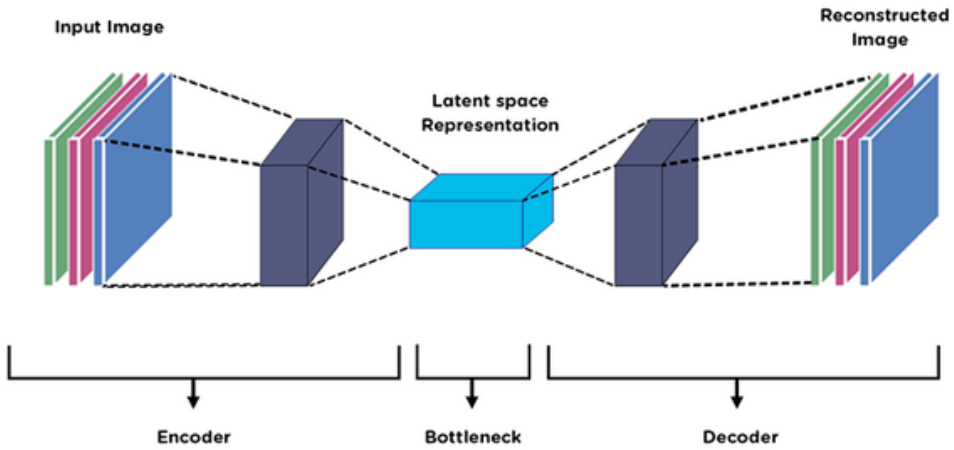
Dataset

- Simulated background-only light curves to detect anomalies, such as GRBs.
- Time in seconds on the x-axis → The light curve contains 128 features in total.
- Count rate on the y-axis → It represents the number of photons detected within each bin. The values are normalized between 0 and 1 using the Min-Max scaling algorithm on the full dataset.





Autoencoder



- In Deep Learning, an **autoencoder** is a type of neural network used to compress and reconstruct data, often used for tasks like data denoising or dimensionality reduction.

- It consists of three main parts:
  - **Encoder** → It compresses the input data into a lower-dimensional representation.
  - **Bottleneck Layer** → It holds the compressed representation of the input data and captures the most essential features.
  - **Decoder** → It reconstructs the original data from its compressed form.

- This network is trained to minimize the difference between the original input and the reconstructed output, enabling it to detect anomalies when the reconstruction error significantly deviates from the expected range.

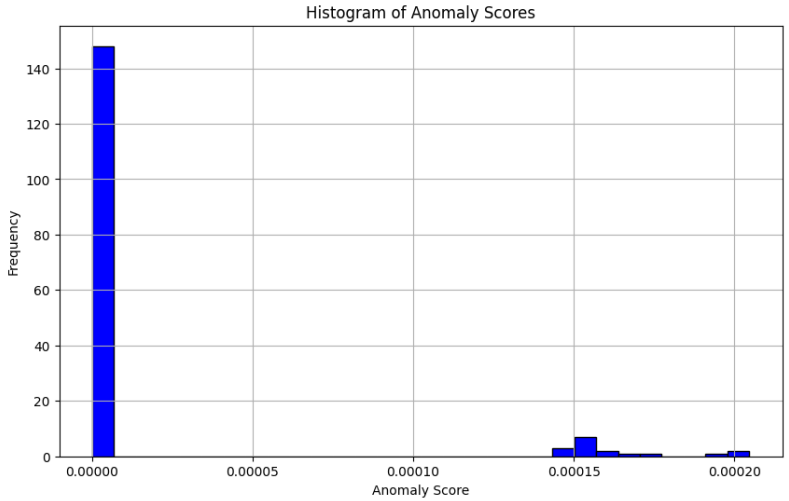
- **Applications**:
  - Image Compression
  - Anomaly Detection
  - Feature Extrection

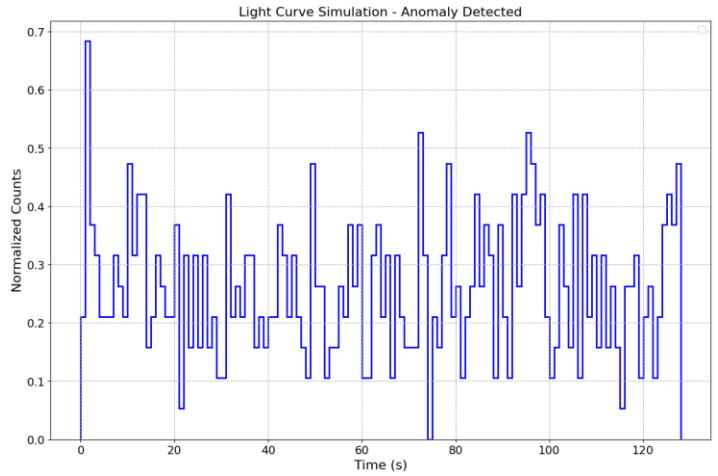
Classical Approach (1)

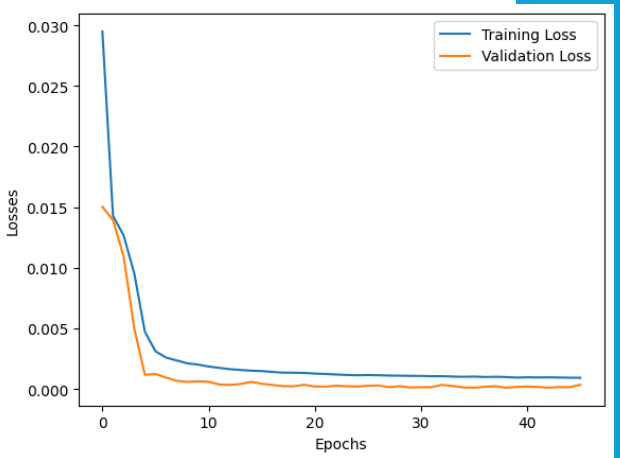
- **Vanilla Autoencoder** → Composed of 1D Convolutional layers for the encoder and 1D Transposed Convolutional layers for the decoder.
- **Variational Autoencoder (VAE)** → Same structure as the vanilla autoencoder, but includes a probabilistic representation of the latent space between the encoder and the decoder.

Layer (type)	Output Shape	Param #
conv1d (Conv1D)	(None, 64, 250)	1,500
dropout (Dropout)	(None, 64, 250)	0
conv1d_1 (Conv1D)	(None, 32, 500)	625,500
dropout_1 (Dropout)	(None, 32, 500)	0
conv1d_transpose (Conv1DTranspose)	(None, 64, 500)	1,250,500
dropout_2 (Dropout)	(None, 64, 500)	0
conv1d_transpose_1 (Conv1DTranspose)	(None, 128, 250)	625,250
dropout_3 (Dropout)	(None, 128, 250)	0
conv1d_transpose_2 (Conv1DTranspose)	(None, 128, 1)	1,251

Total params: 2,504,001 (9.55 MB)
Trainable params: 2,504,001 (9.55 MB)
Non-trainable params: 0 (0.00 B)







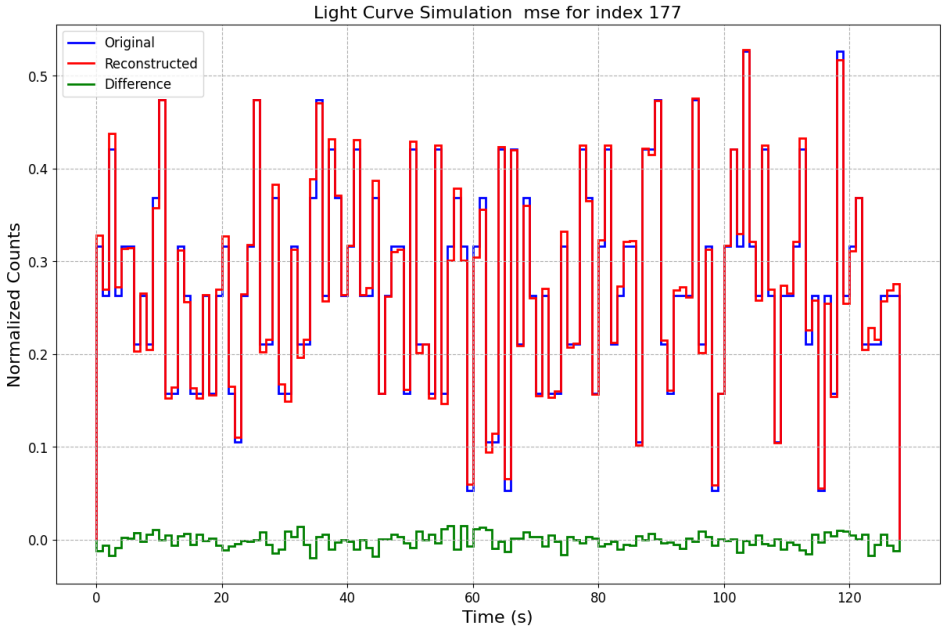
- The main difference between a Vanilla Autoencoder and a Variational Autoencoder is the structure of the latent space. In a VAE, the latent space is continuous and probabilistic. This feature makes VAEs particularly useful for generative modeling, as they can generate new data points by sampling from the learned distribution in the latent space.

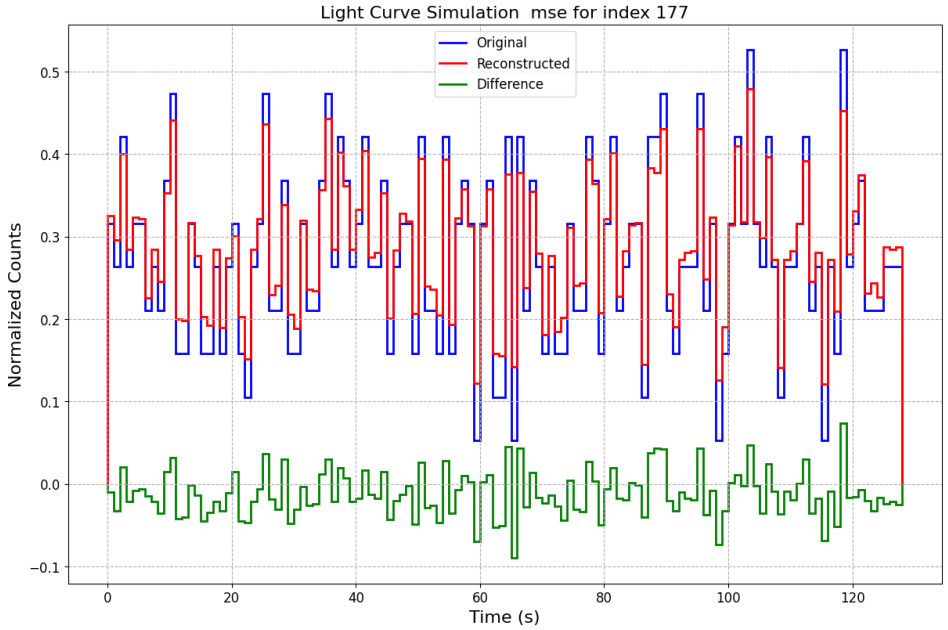
N. Parmiggiani et al, **A Deep Learning Anomaly Detection Method to Identify Gamma-Ray Bursts in the Ratemeters of the AGILE Anti-coincidence System**. The Astrophysical Journal, Volume 945, Number 2, 2023.

Classical Approach (2) – Reconstructed Lightcurves

- On the left, a reconstructed light curve from the test set using the vanilla autoencoder.
- On the right, the same light curve but reconstructed by the variational autoencoder.

- The vanilla autoencoder might perform better because it is focused on minimizing the reconstruction error, enabling it to learn a more precise mapping for the reconstruction task.



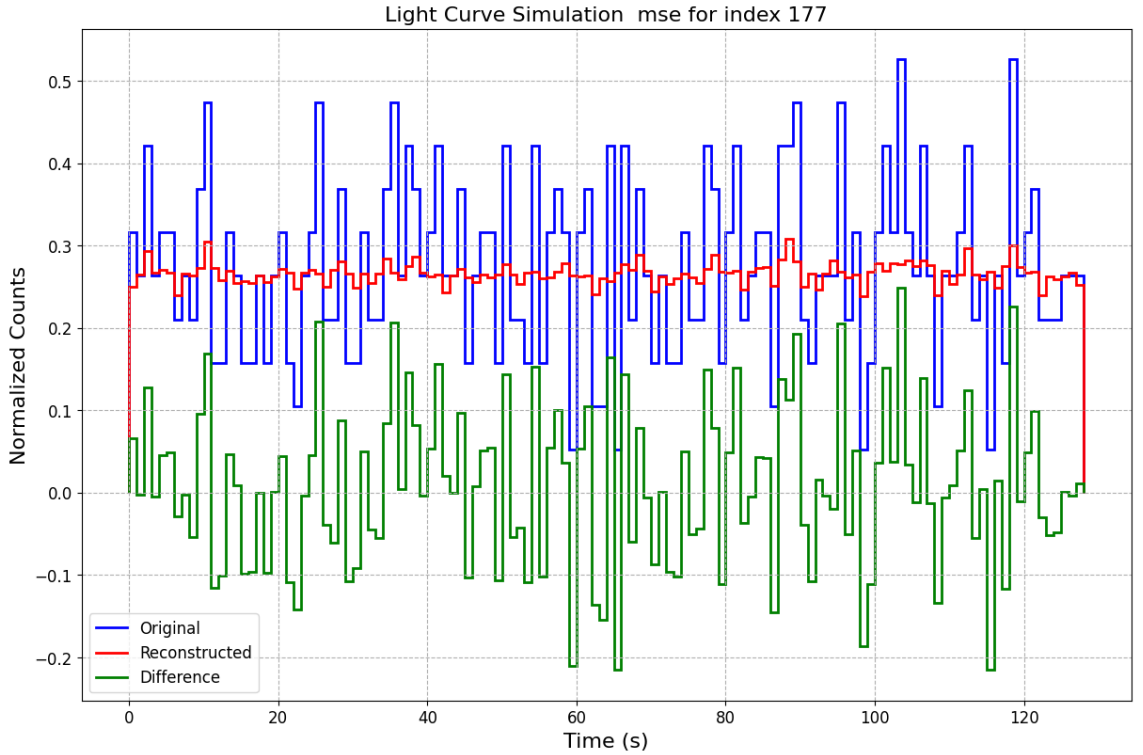


Classical Approach (3) – Model's parameters

- To compare the classical approach with the quantum model, we gradually reduced the **number of parameters** in the classical network until it could no longer perform the reconstruction task effectively. This approach was taken to reflect the parameter limitations inherent in the quantum models due to computational constraints.

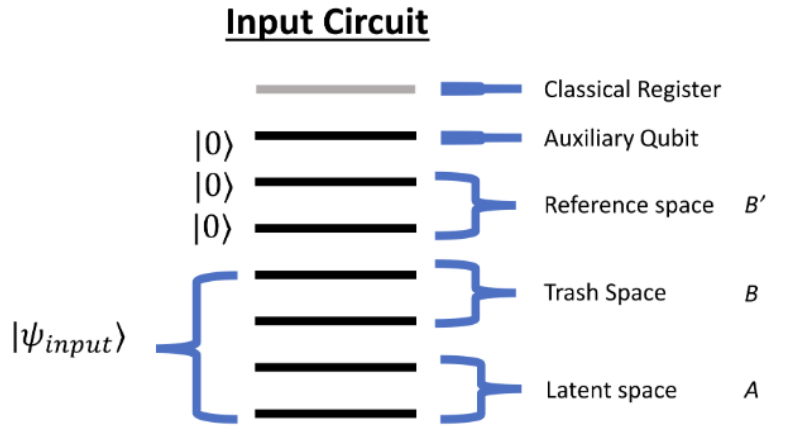
Layer (type)	Output Shape	Param #
conv1d_8 (Conv1D)	(None, 64, 4)	24
dropout_16 (Dropout)	(None, 64, 4)	0
conv1d_9 (Conv1D)	(None, 32, 8)	168
dropout_17 (Dropout)	(None, 32, 8)	0
conv1d_transpose_12 (Conv1DTranspose)	(None, 64, 8)	328
dropout_18 (Dropout)	(None, 64, 8)	0
conv1d_transpose_13 (Conv1DTranspose)	(None, 128, 4)	164
dropout_19 (Dropout)	(None, 128, 4)	0
conv1d_transpose_14 (Conv1DTranspose)	(None, 128, 1)	21

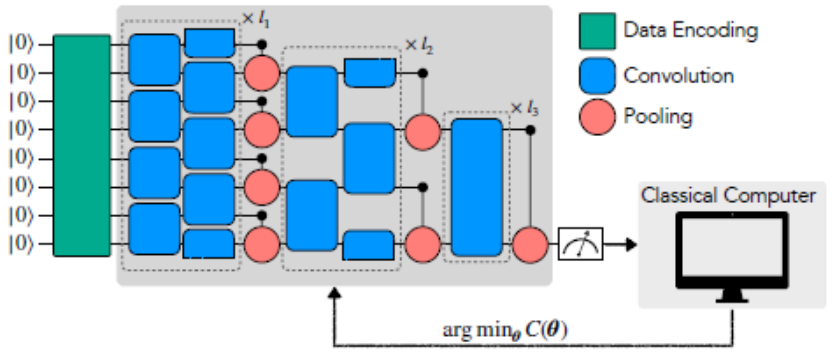
Total params: 705 (2.75 KB)
Trainable params: 705 (2.75 KB)
Non-trainable params: 0 (0.00 B)



Quantum Approach (1)

- The aim of this work is to replicate the results achieved with classical methods using quantum deep learning networks.
- We utilized two different frameworks, IBM's **Qiskit** and Xanadu's **PennyLane**, to design and simulate various quantum circuits.
- For encoding classical data into quantum states, we employed two techniques: **data re-uploading** and **amplitude embedding**.
- We implemented both a standard quantum autoencoder and a quantum convolutional autoencoder to explore the potential of quantum architectures in this context.





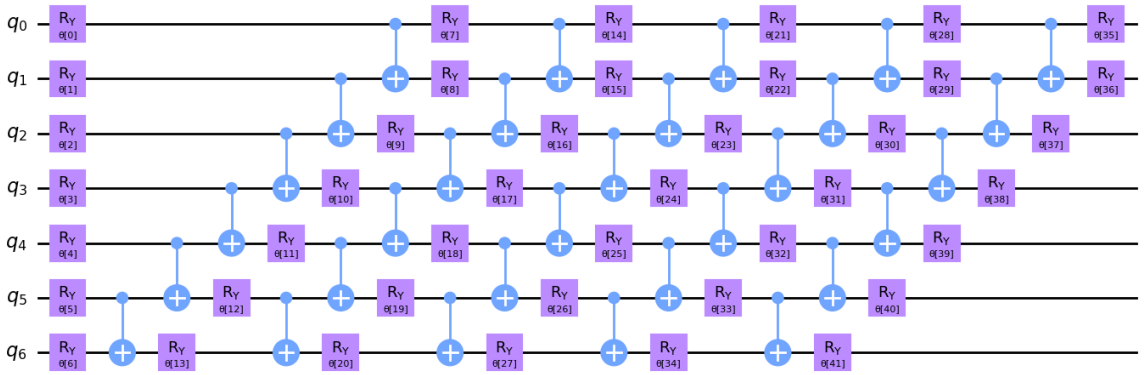
Romero J., et al. **Quantum Autoencoder for Efficient Compression of Quantum Data**. Quantum Science and Technology, Volume 2, Number 4, 2017.
T. Hur, et al. **Quantum convolutional neural network for classical data classification**. Quantum Machine Intelligence, Volume 4, 2022.

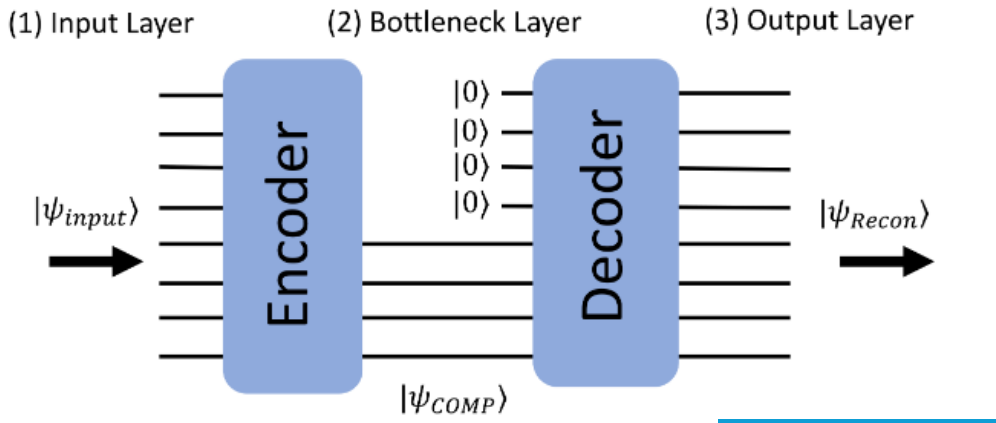
Quantum Approach (2)

- The **quantum autoencoder** is composed of three layers:
  - Input layer
  - Bottleneck layer
  - Output layer

- **Parametrized quantum circuit** for the training process.

- As loss function, we used the **fidelity score** between the input state and the output state.

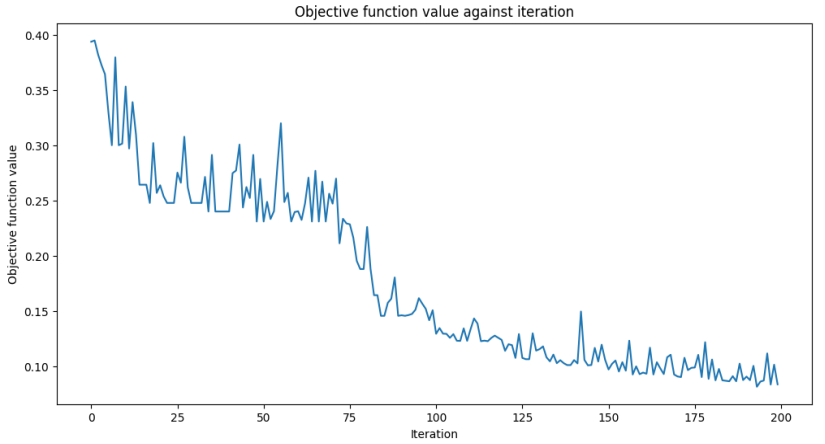


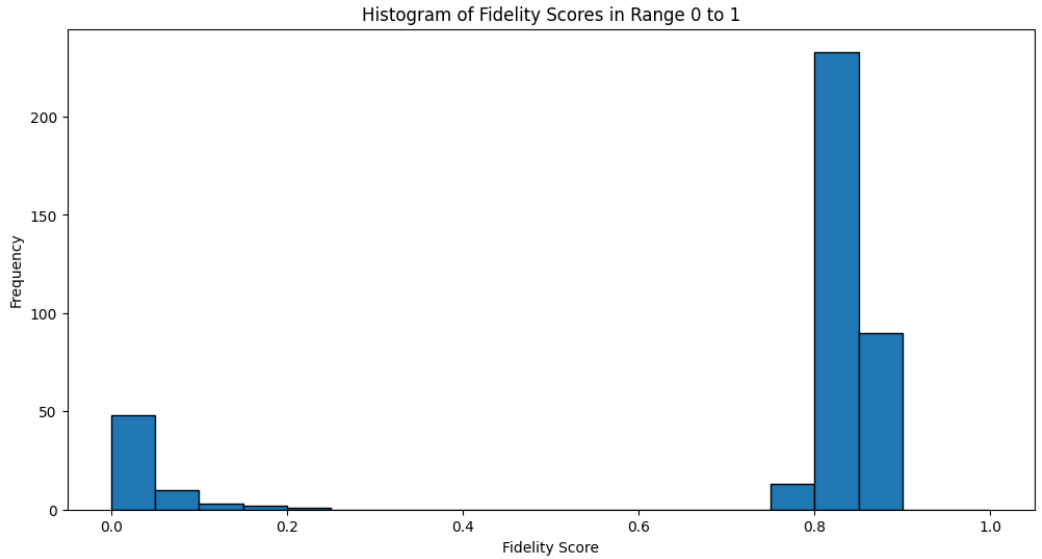


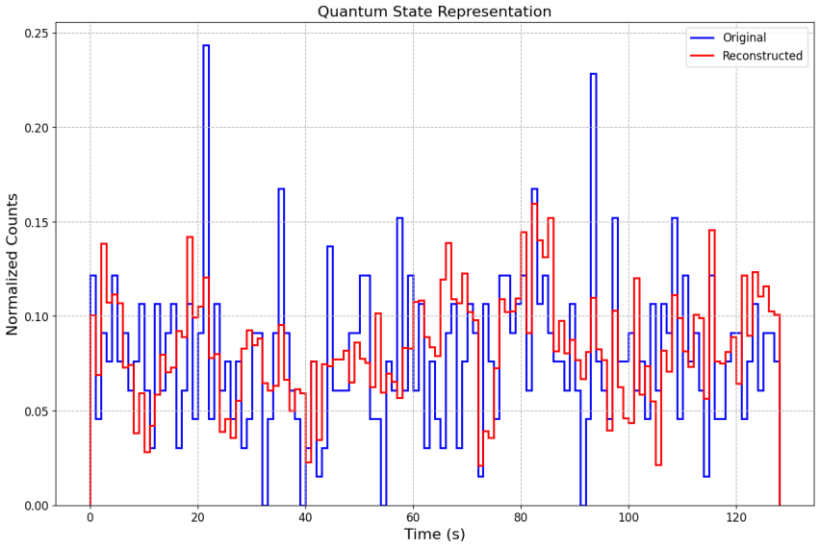
Quantum Approach (3) - Results

- The loss function used to train the autoencoder is the fidelity score between the trash state and the reference state.

- While we achieved a good fidelity score for some states in the test set, the output quantum state showed significant deviations from the input quantum state for several other cases.



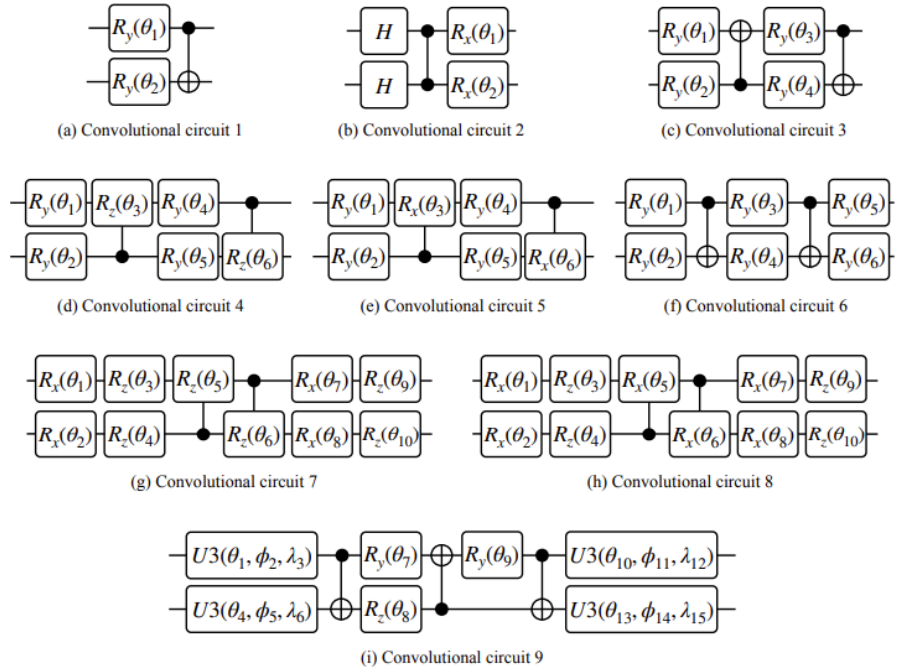


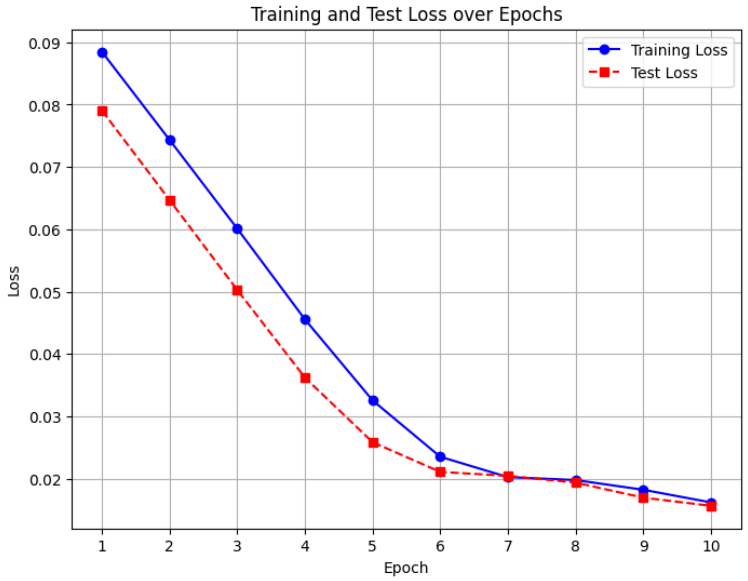


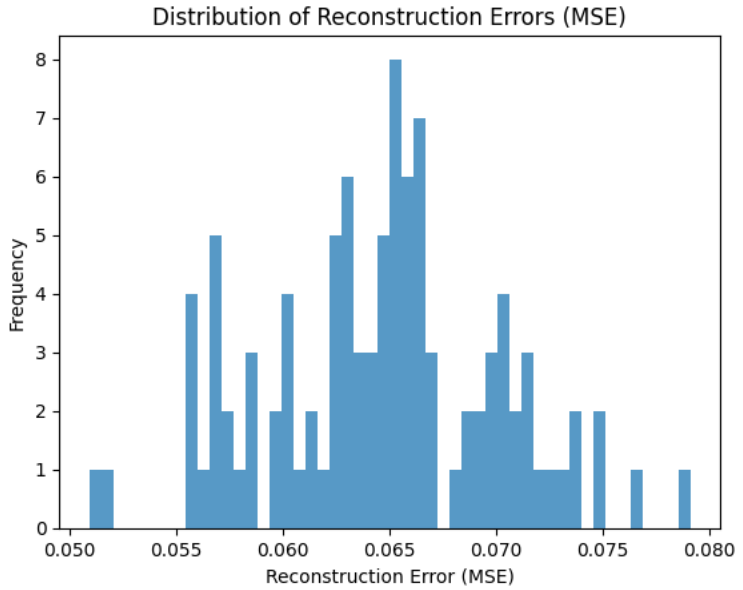
Quantum Approach (4) - Results

- Using PennyLane we tested a Quantum Convolutional Autoencoder followed by three layers of 1D Transposed Convolutions, therefore leading to a **Hybrid Quantum-Classical Convolutional Neural Network**.

- The obtained results, as seen in the distribution of reconstruction errors on the test set, were not very satisfactory.







Quantum Approach (5) – Results & Comparisons

- This table shows the results obtained from the different approaches presented, comparing those achieved by the quantum models with those of the classical and hybrid models.
- **The reconstruction error** is measured on the test set using the **Mean Squared Error** between the original and the reconstructed lightcurves.

Approach	Framework	Model	Parameters	Reconstruction Error (Test Set)	Qubits
Classical	Keras & Tensorflow	Vanilla Autoencoder	> 2M	1.5e-4	//
		Variational Autoencoder	> 2M	1.4e-3	
		Smaller Autoencoder	705	1.1e-2	
Hybrid	PennyLane & Keras	Quantum Convolutional Autoencoder + 1D Transposed Convolutions	51 (quantum) + 140K (classical)	6.47e-2	8
Fully Quantum	Qiskit	Quantum Autoencoder	42	6.53e-3	7
		Smaller Quantum Autoencoder	18	8.82e-3	3

Conclusion & Future Work

- The quantum autoencoder implementation fails to reconstruct the light curves properly and, therefore, does not perform well in this anomaly detection task.
  Possible reasons for this include:
  - **Encoding error**: Embedding large, high-dimensional classical data into a quantum state can lead to loss of information or inaccuracies.
  - **Noisy Encoding**: Quantum data encoding is sensitive to noise, which is problematic in current **Noisy Intermediate-Scale Quantum (NISQ)** devices → **Fault Tolerance**, errors caused by factors such as decoherence, gate errors and measurement errors can significantly affect quantum autoencoders → This is crucial because even minor errors during encoding, processing or decoding can lead to significant deviations in the reconstructed output.

- However, it is worth noting that as the number of parameters in the classical model is reduced, the quantum model tends to achieve better reconstruction errors on the test set, effectively leveraging quantum mechanical properties such as entanglement and superposition.

- Try implementing more sophisticated variational quantum circuits by experimenting with different circuit designs and using more qubits.

Thanks for your attention