Spatio-Temporal Anomaly Detection for the DQM of the CMS Experiment via Graph Networks

Talk for 5th IML Workshop

Speakers:
Mulugeta W. Asres (UiA), Long Wang (CERN), and David Yu (CERN)

HCAL DPG and HCAL Ops Team

Mulugeta W. Asres
University of Agder, Norway
May 11, 2022

Topics

Introduction

Motivation: ML4DQM

ML4DQM for the HCAL: Research Gaps

Anomaly Detection Mechanism: ML4DQM for the HE

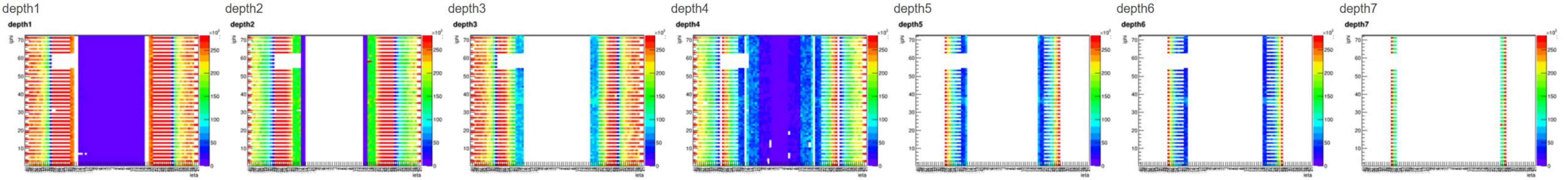
- Preprocessing: Digioccupancy Map Normalization
- Model Evaluation on Synthetic Anomalies
- Comparison with Benchmarks Models
- Detection of Real HE Channel Anomalies
- Computational Complexity

Summary

Introduction

- The HCAL Data is utilized for various physics analyses--from **low luminosity** to **high pile-up** environment.
- Our on-going effort **DEtector System Monitoring Diagnostics and Prognostics (DESMOD)** for the HCAL via ML models.
  - **DESMOD-HEngCCM**: Anomaly Detection (AD) and Prediction with output explanation from multivariate diagnostics sensors [1]
  - **DESMOD-DQMAD:** AD for the HCAL Endcap (HE) channels monitoring from DQM occupancy maps (this talk)

- The DQM of the HCAL of the CMS aims to guarantee high-quality physics data through
  - **Online monitoring** generates set of histograms following data acquisition.
  - **Offline monitoring** is used to certify data quality.

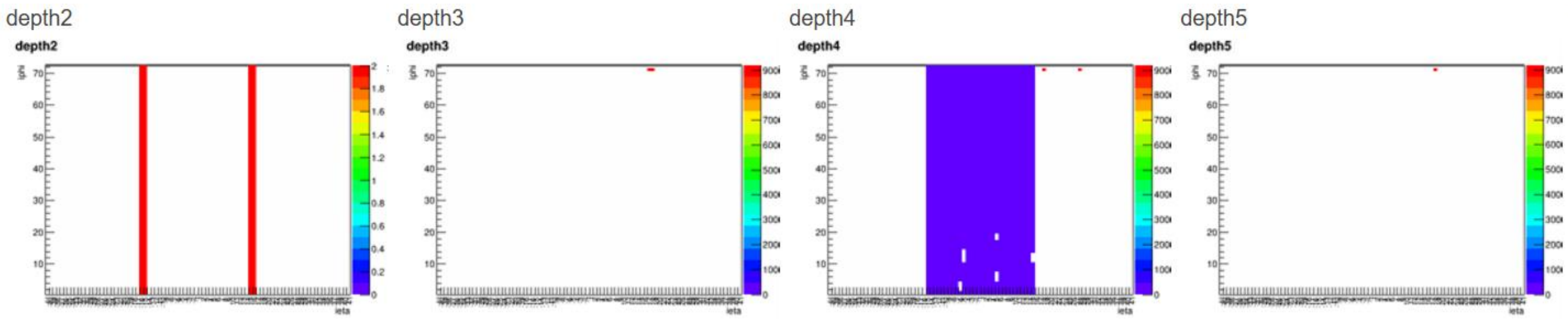


The HCAL has 3D spatial Digioccupancy maps, $[i\eta \times i\phi \times depth]$.

[1] https://ieeexplore.ieee.org/abstract/document/9687034

Motivation: ML4DQM

- **The Digioccupancy Maps of the HCAL:**
  - Contains a **digi** or **particle hit** record of a data-taking sensor (aka **channel**) of the detector.
  - Has **3D spatial map data,** $[i\eta \times i\phi \times depth]$.
  - **Each pixel** in the occupancy map belongs to **a HCAL channel**.
  - Potential **abnormal channels** can be **spotted** from **the occupancy map**.



RED: Bad Quality Channels

- **Challenge**: lack annotated anomalies covering all possible anomalies **shapes** and **sizes**—challenging to anticipate all possible failure modes.
- **Semi-/Un-supervised ML as potential solution:** robust **anomaly detection (AD)** and **localization**

Motivation: ML4DQM - HCAL

DQM-HCAL Data
- 3D histogram maps
- High dimensional spatial data
- Detector channels share common RBX

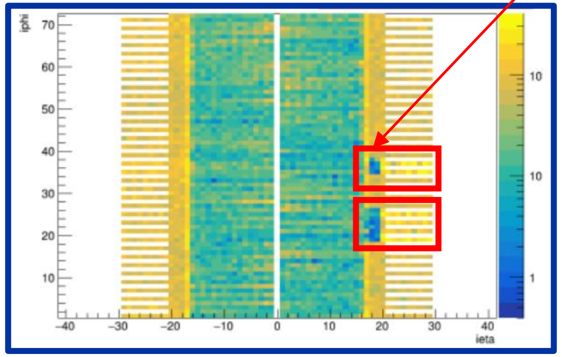
DQM Histogram Map Normalization
- Dependency on experiment settings (e.g. Luminosity, Event number, etc.)

Temporal AD
- Faulty channels persists over time
- AD within temporal context

DQM-HCAL Challenges

Degrading Channels
- May impact physics data quality
- Relevant for Predictive Maintenance (PdM)

HBHE DIGI occupancy Map [2]



In Oct 2021, Run346247
- **Issue: Non-uniformity in the HE Digi-occupancy distributions.**
- **Cause: improperly tuned SiPMs Bias Voltage for HEP06, 07, and 10 sectors**

[2]https://indico.cern.ch/event/1141023/contributions/4791854/attachments/2421439/4144738/hcal_pfg_cmsweek_apr2022.pdf

Automated **AD with ML model** has the potential to detect such faults instantly.

Anomaly Detection Mechanism: ML4DQM for the HE

- **Autoencoder ($\mathcal{F}$):** spatio-temporal γ data (X) reconstruction.
- **Intuition:** $\mathcal{F}$ trained on healthy maps would struggle to reconstruct anomalies and thus, will higher reconstruction error.

Input: $\boldsymbol{X} \in \mathbb{R}^{T\times N_{i\eta}\times N_{i\phi}\times N_d\times N_f}$

Output: $\overline{\boldsymbol{X}} \in \mathbb{R}^{T\times N_{i\eta}\times N_{i\phi}\times N_d\times N_f}$

$$\overline{\boldsymbol{X}} = \mathcal{F}_d\,(\mathcal{F}_e(\boldsymbol{X}))$$

Training loss function (MSE):

$$\mathcal{L}(\boldsymbol{x}_i, \overline{\boldsymbol{x}}_i) = \|\boldsymbol{x}_i - \overline{\boldsymbol{x}}_i\|_2^{\,2}$$

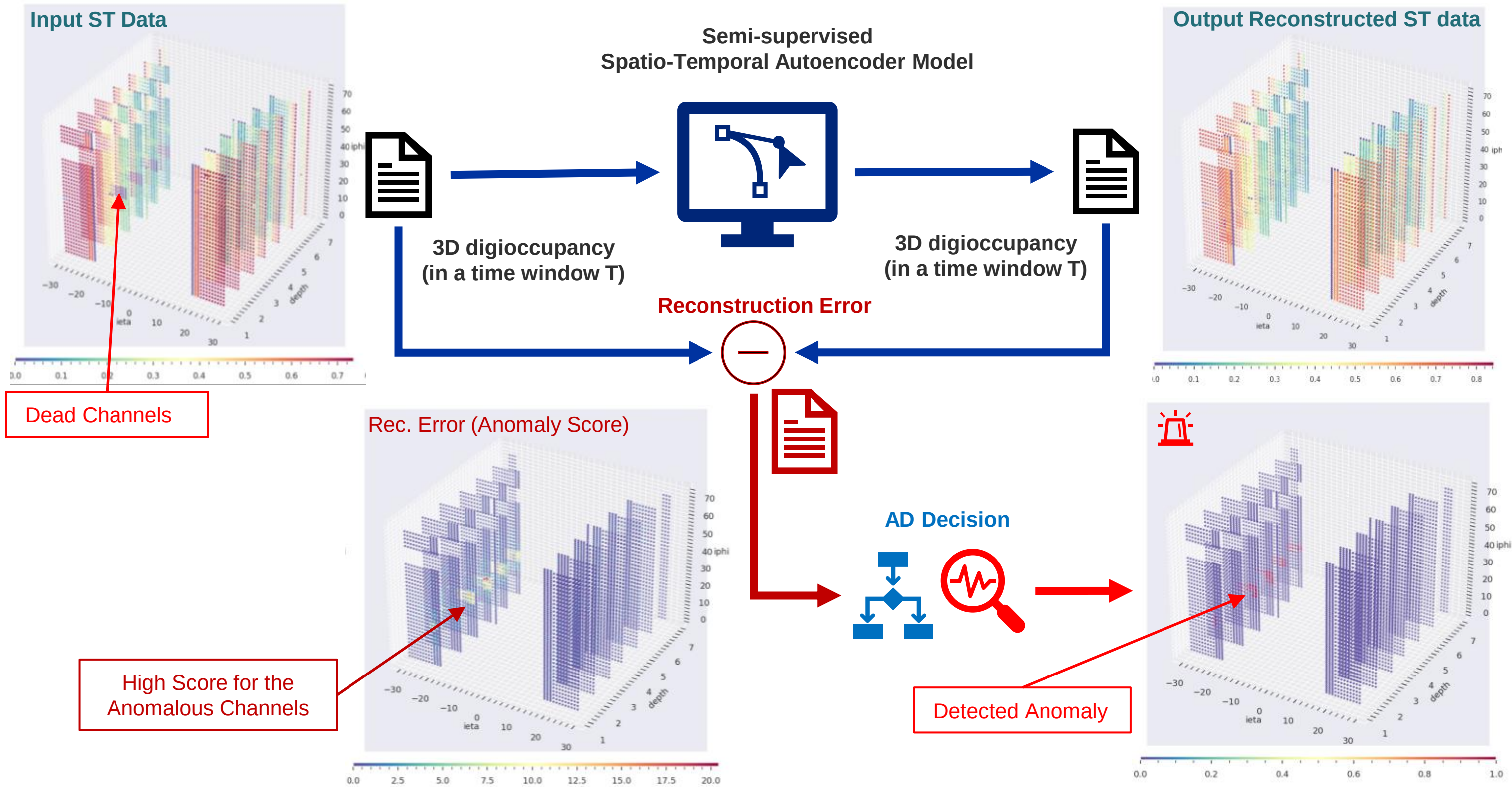
Anomaly Score with Rec. Correction: $\boldsymbol{a}_i = \frac{\boldsymbol{e}_i}{\boldsymbol{\sigma}_i}$

Where $\boldsymbol{a}_i$ is a standardize reconstruction error of $\boldsymbol{e}_i$

$$\boldsymbol{e}_i = |\boldsymbol{x}_i - \overline{\boldsymbol{x}}_i|$$

$\boldsymbol{\sigma}_i$ is std the reconstruction error for the $\boldsymbol{i}^{th}$ channel estimated from the training set.

Anomaly Decision: $AD(\boldsymbol{a}_i) = \boldsymbol{a}_i > \boldsymbol{k}$, single tunable threshold $\boldsymbol{k}$ for all channels.



DESMOD-DQMAD Model

Preprocessing: Digioccupancy Map Normalization

- **The digioccupancy value (γ):**
  - is **number of digi per channel in a given LS**.
  - is determined by the experiment luminosity $\boldsymbol{\xi}$ and number of events $\boldsymbol{N_e}$ settings.

$$\gamma\,(c) \in [0, N_e]$$

- **γ Normalization Regression Model (DNN):**
  - **Deep regression model $\mathcal{R}$** to harmonize the variation in the $\boldsymbol{\xi}$ and $\boldsymbol{N_e}$ at each HCAL depth.

$$\bar{\gamma}_l = \underset{\mathbb{E}[(\gamma_l - \mathcal{R})^2]}{\arg\min} \{\mathcal{R}(N_e, \xi)\}$$

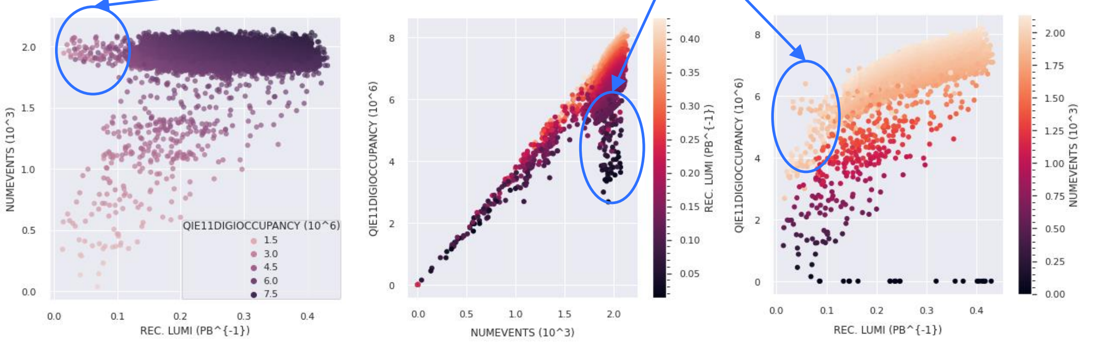
$$\gamma_l = \sum_{\forall c} \gamma(l, c)$$

$$\hat{\gamma}(l, c) = \frac{K * \gamma(l, c)}{\bar{\gamma}_l}$$

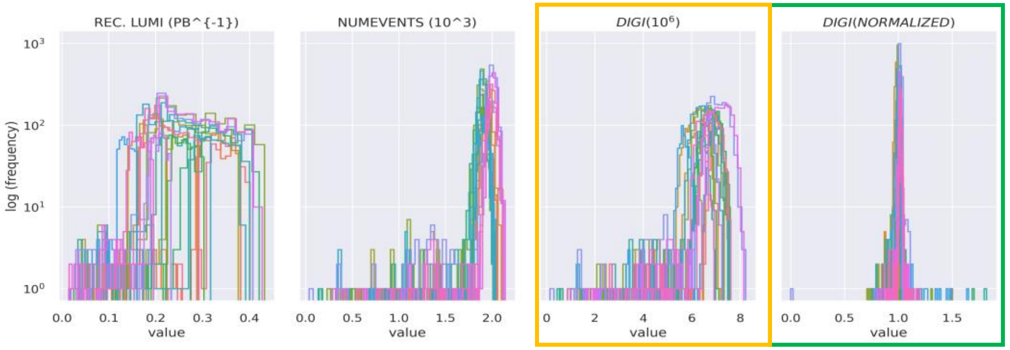
Where:
- γ_l is total γ per depth at l^{th} LS.
- $\hat{\gamma}\,(l, c)$ is the normalized γ value of the channel c at l^{th} LS.
- K is a scaling factor to compensate the difference in the number of channels per depth.

- Normalization enables training ML models with **smaller datasets** with effective **generalization** for Runs with previously **unseen experiment settings**.

Non-linearity in the LHC:
Unusual high Number of Events for low Luminosity



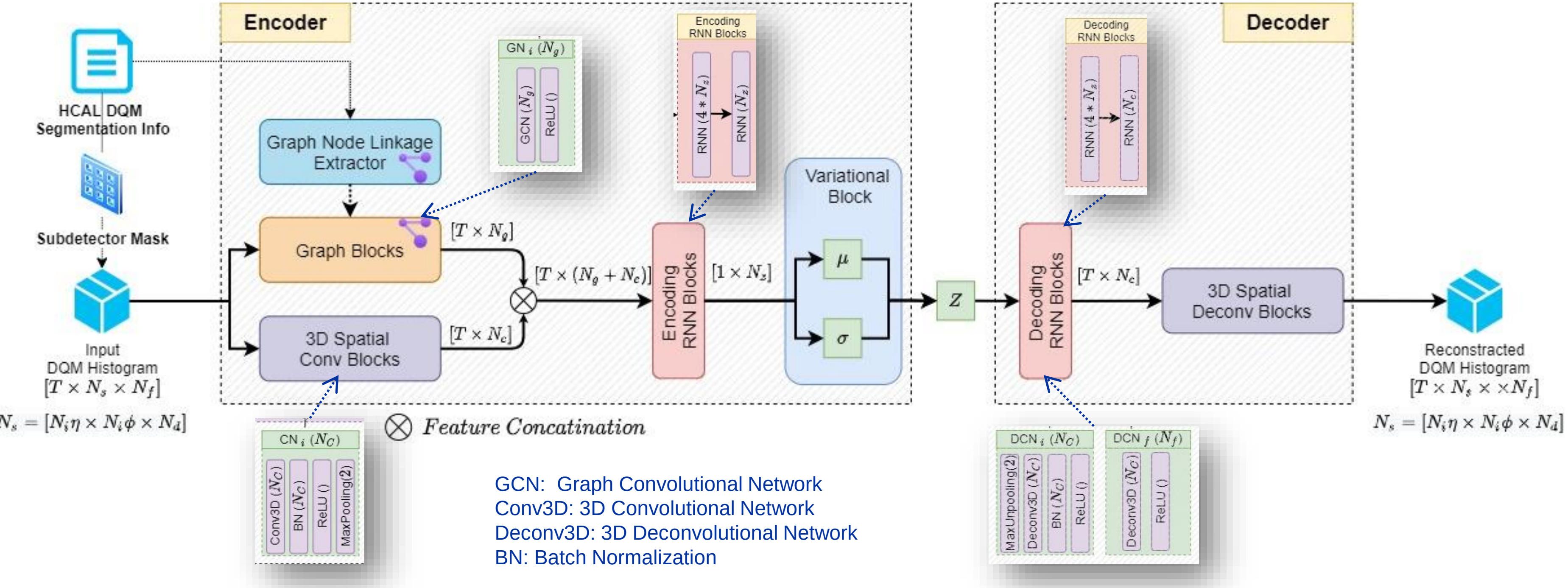
γ is depends on ξ and N_e



Total digioccupancy per LS
of several runs

AD Model Design with Graph Autoencoder

- **Model(DESMOD-DQMAD):**Multilayered networks semi-supervised autoencoder for **ML4DQM AD of the HE**.
- **Convolutional**, **graph**, and **recurrent neural networks** are integrated to capture spatial and temporal characteristics.
  - **Euclidean** and **non-Euclidean spatial characteristics** of HE digioccupancy map:
    - **Conv3D**: Proximally arranged channels (Euclidean distance) are exposed to **particle hits around their region**.
    - **GCN**: Channels share a **common backend RBX** that results in a non-Euclidean spatial distance.



Graph Network $G(v, \varepsilon)$ to learn shared local variations due **interconnected backend circuit** and **environmental impact** in a common RBX.

$$A(u, v) = \begin{cases} 1, & \text{if } RBX(u) = RBX(v) \\ 0, & \text{otherwise} \end{cases}$$

The v denotes the HE channel nodes and ε is the edges in adjacency matrix A, respectively. An edge $(u, v) \in \varepsilon$ connects a pair of nodes u *and* v in the same RBX.

Channels in a given RBX share backbone :
- **Environmental factors:** **temperature and humidity inside RBX.**
- **Local variations per RBX:** **intrinsic variations of the custom-built electronic components.**

Model Training and Validation

- **Dataset: /ZeroBias/Run2018*-v1/RAW**
  - Golden JSON lumis: “GOOD” data , around **20K** **3D digioccupancy histogram map** (γ) of **HCAL Endcap (HE).**
  - Around **1750-2250 events** per LS and luminosity ranging up to **0.4 $\boldsymbol{PD^{-1}}$.**

- **Training set** ($LS \in [\mathbf{1}, \mathbf{500}]$)**:** ~10K GOOD histograms
- **Validation set** ($LS \in [\mathbf{500}, \mathbf{1500}]$)**:** ~10K histograms
  - **10K** histograms with synthetic anomalies (**dead ($\boldsymbol{\gamma = 0}$)** and **hot ($\boldsymbol{\gamma = 2 * \gamma_{expected}}$)** anomalies, each **5K**)
  - **5K** histograms with synthetic **degrading** anomaly ($\boldsymbol{\gamma = D * \gamma_{expected}}$).
    - Decaying factor $\boldsymbol{D = [0.8, 0.6, 0.4, 0.2, 0.0]}$**.**

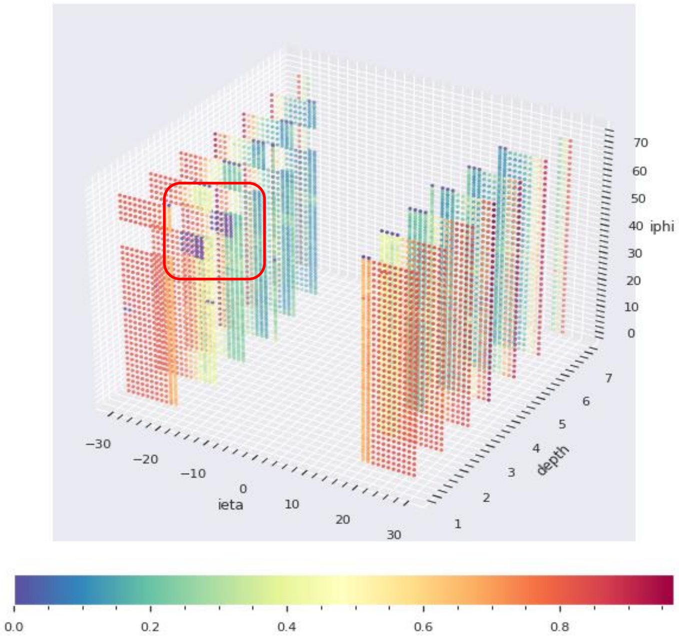
- **Model setting:**
  - Sliding time-window size: **5 LSs**

- **Temporal Anomaly Evaluation:**
  - Anomalies affecting only an **isolated LS.**
  - Anomalies affecting **consecutive LSs in a time window:**
    - Anomaly score is estimated from mean absolute error (MAE) in time window.

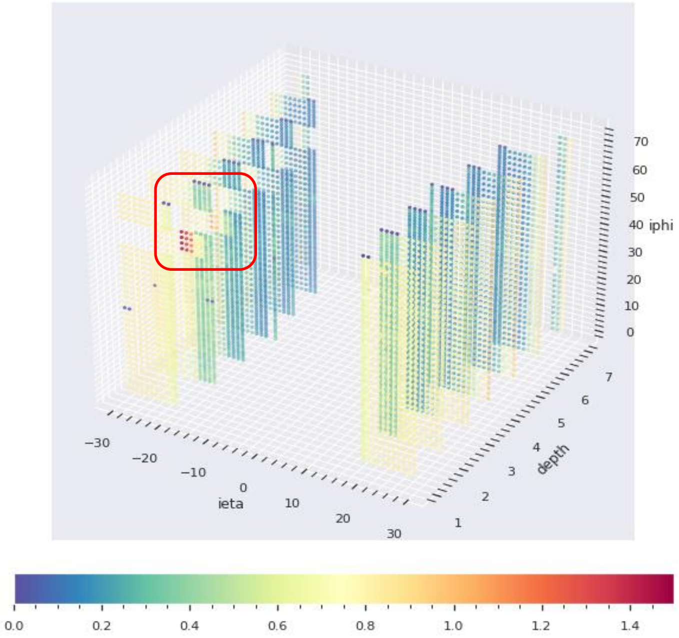
Model Evaluation: Anomaly on Isolated LS

- **10K** histograms with **dead** and **hot channels**, monitored ~**32M** channels (**335K (1.05%)** anomalous) for each anomaly type.

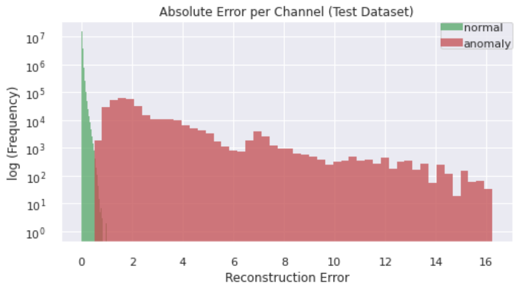
Sample γ histogram map with DEAD channel anomaly



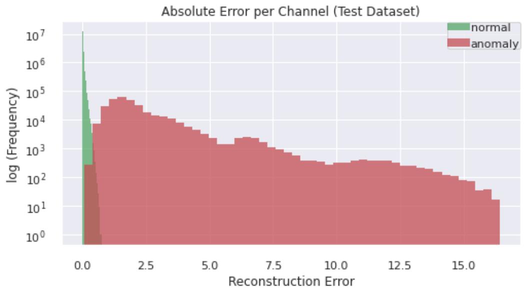
Sample γ histogram map with HOT channel anomaly



Reconstruction error distribution DEAD channel anomaly



Reconstruction error distribution HOT channel anomaly



Anomaly Type	Captured Anomalies	P	R	F1	FPR
Dead Channel	99%	0.999	0.99	0.995	6.722×10^{-6}
	95%	1.000	0.95	0.974	3.102×10^{-6}
	90%	1.000	0.90	0.947	2.068×10^{-6}
Hot Channel	99%	0.999	0.99	0.994	9.113×10^{-6}
	95%	1.000	0.95	0.974	1.939×10^{-6}
	90%	1.000	0.90	0.947	1.196×10^{-6}

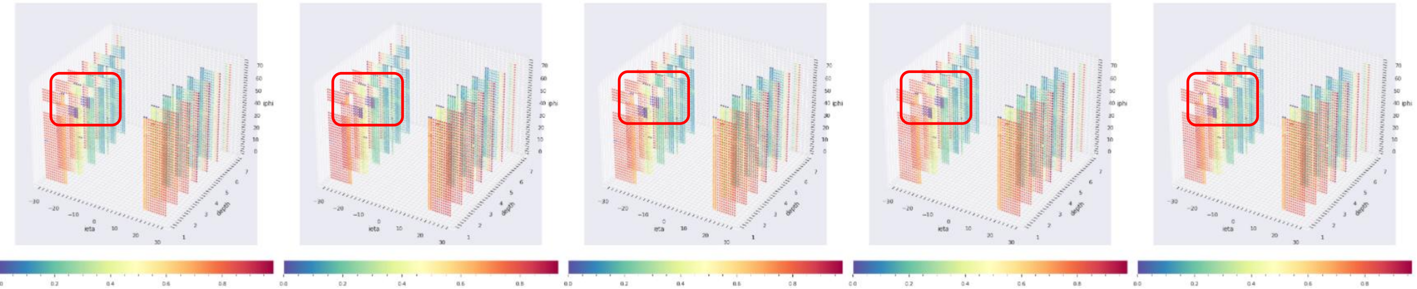
The proposed AD system has achieved a promising high performance with precise localized detection of the faulty channels, i.e., **0.99 precision** while detecting **99%** of the **335K faulty channels**.

P- Precision, R- Recall, F1 – F1 score, FPR– False Positive Rate

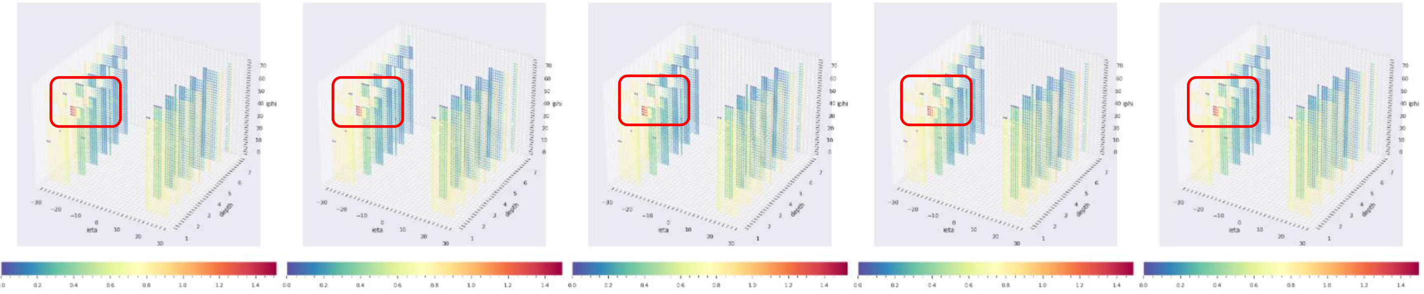
Model Evaluation: Time Persistent Anomaly

- **10K** histograms with **dead and hot** channels in a **time window of 5 LSs (50K)**, monitored ~**156M** channels (**1.68M (1.05%)** anomalous) for each anomaly type.
- Anomaly scores are estimated using **mean absolute error (MAE)** across the LSs in a time window.

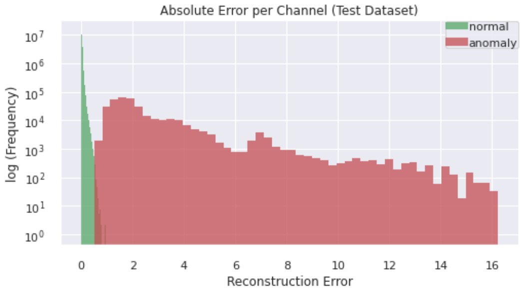
Sample γ histogram maps with DEAD channel anomaly persisted across LSs



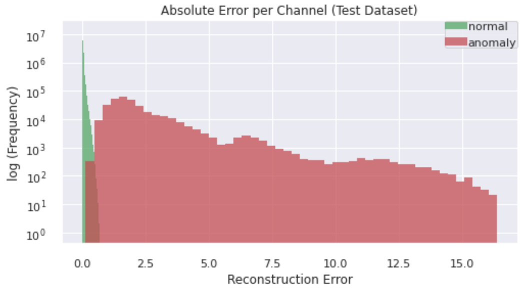
Sample γ histogram maps with HOT channel anomaly persisted across LSs



Reconstruction error distribution
DEAD channel anomaly



Reconstruction error distribution
HOT channel anomaly



Anomaly Type	Captured Anomalies	P	R	F1	FPR
Dead Channel	99%	0.999	0.99	0.995	7.691×10^{-6}
	95%	1.000	0.95	0.974	2.715×10^{-6}
	90%	1.000	0.90	0.947	1.616×10^{-6}
Hot Channel	99%	0.999	0.99	0.995	5.461×10^{-6}
	95%	1.000	0.95	0.974	1.357×10^{-6}
	90%	1.000	0.90	0.947	7.756×10^{-7}

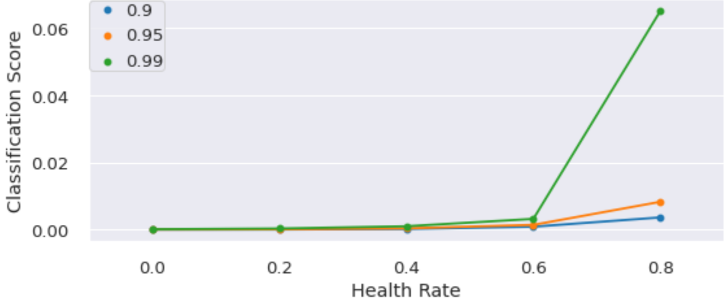
Time-persistent anomalies are easier to detect as the FPR improves by 13%-23% and 28%-40% for the dead and hot anomalies, respectively, as compared to detecting anomalies affecting isolated LSs.
Generally, capturing **99% of the anomaly** is relatively more challenging as **certain channels may have a very small expected γ.**

Model Evaluation: Degrading Channels, Time Persistent

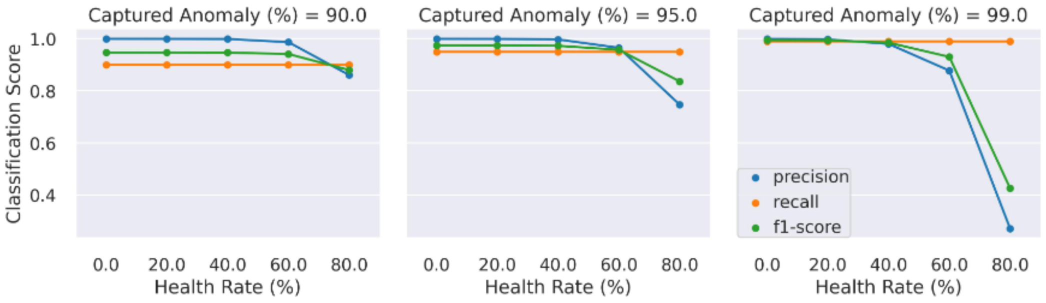
- **5K** histograms with **degrading to death channel** anomalies in a **time window of 5 LSs (25K)**, monitored ~**156M** channels (**1.74K (1.11%)** anomalous)
- **Degrading** anomaly is when $\gamma = D * \gamma_{expected}$), with decaying factor $D = [0.8, 0.6, 0.4, 0.2, 0.0]$, where $D = 0$ denotes **a dead channel.**
- **1K** histograms for each **5K** histograms.

Health Rate = Decay Factor

Anomaly Type	Health Rate	FPR (90%)	FPR (95%)	FPR (99%)
Decaying Channel	80%	1.636×10^{-3}	3.614×10^{-3}	2.988×10^{-2}
	60%	1.329×10^{-4}	3.834×10^{-4}	1.550×10^{-3}
	40%	8.405×10^{-6}	2.764×10^{-5}	2.242×10^{-4}
	20%	2.263×10^{-6}	5.173×10^{-6}	2.505×10^{-5}
	0%	9.699×10^{-7}	1.778×10^{-6}	6.142×10^{-6}



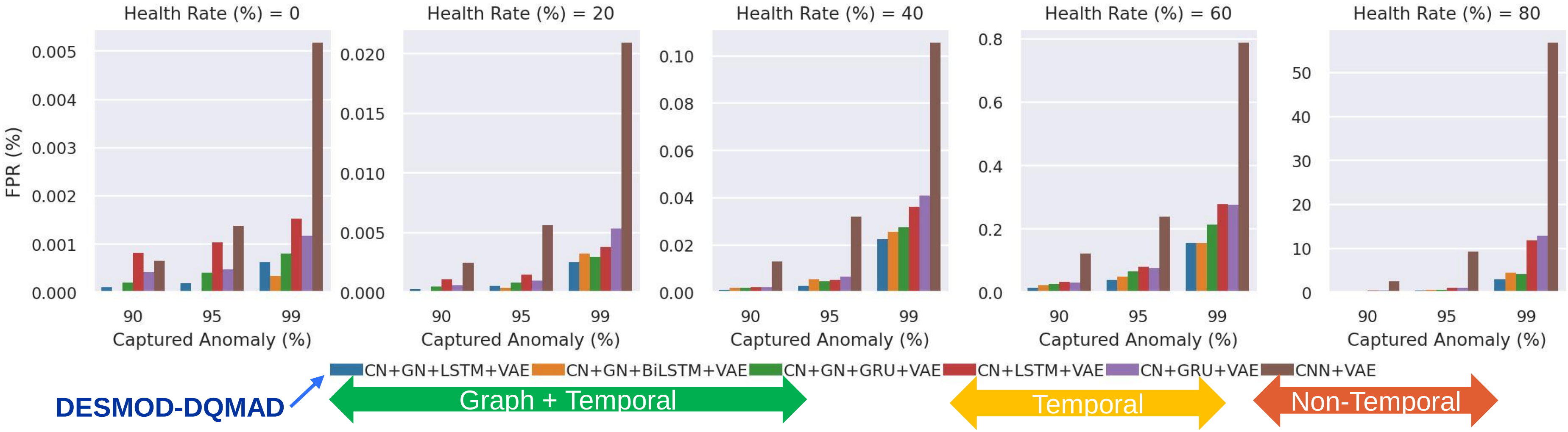
- Promising high performance in detecting the degrading faulty channels.
- The FPR to capture **99%** of the anomalies is **2.988%, 0.155%, 0.022%, 0.002%,** and **0.001%** when channels operate at **80%, 60%, 40%, 20%,** and **0%** of its expected capacity, respectively.



- The **relatively lower precision** for the health rate of **80%** signifies there are still a very few anomalous channels that are challenging to capture.
  - This is because a channel operating at **80%** is an **inlier** to the normal operating ranges, and it becomes even **more challenging** if the expected γ **is very low**.
  - However, the performance **significantly improves** by **88%** and **95%** when the percentage of the target **anomaly to captured** is reduced to **95%** and **90%**, respectively.

Comparison with Benchmark Models on Degrading Channels

- The benchmark models follow overall similar **autoencoder architecture** as the proposed **DESMOD-DQMAD**, but have **different modeling layers**.
- **Integration of the graph network** has a **significant performance gain** from **1.6** to **3.9X**.
- **Temporal models** have achieved a **3** to **5-fold** **boost over** the **non-temporal spatial AD** model, **CNN+VAE**.
  - For channel **degraded** by **only 20%,** the **DESMOD-DQMAD** improvement is by **25X**.
- **Spatio-temporal learning mechanism** enhances the **context to capture degrading channels.**



CN: convolutional network,
GN: graph network,
BiLSTM: bidirectional LSTM,
GRU: gated recurrent unit.

CNN+VAE: the RNN blocks are replaced with FC layers.

Real Channel Anomalies Detected with DESMOD-DQMAD Model

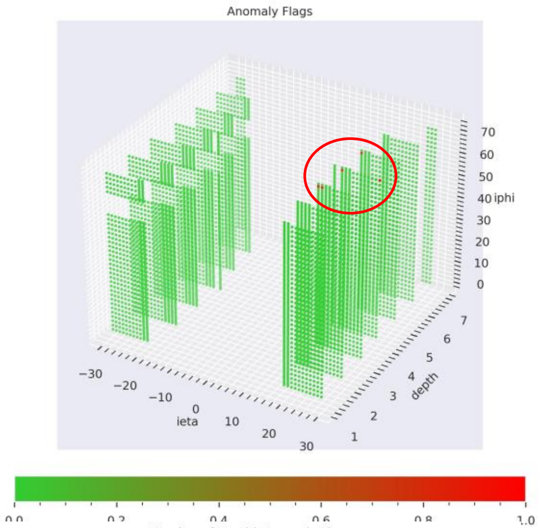
- The proposed **DESMOD-DQMAD model** detected **REAL bad channels** in the HE.

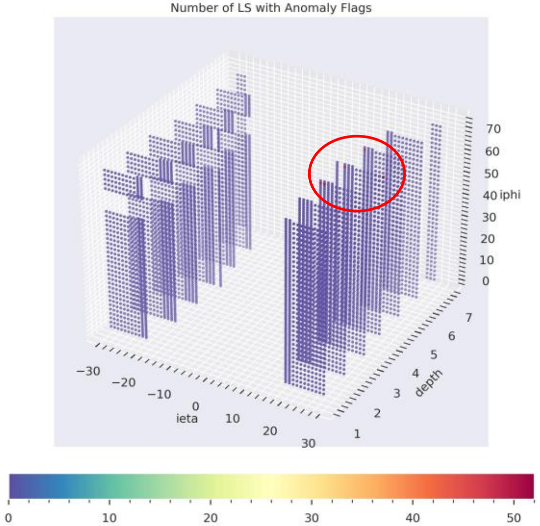
Run: 324841

Anomaly Types: Dead + Degraded Channels

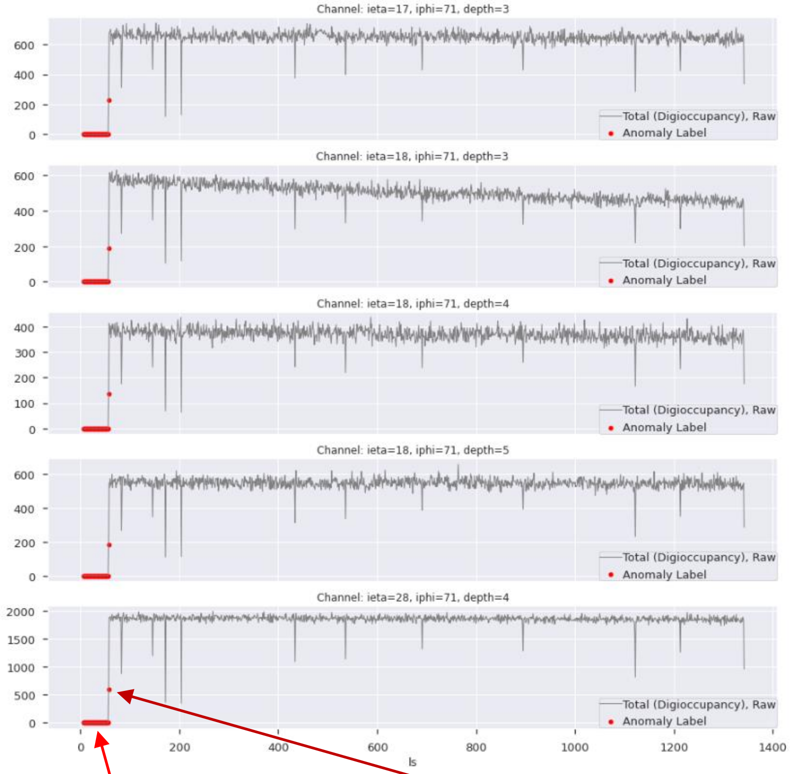
Bad channels:

'ieta=17, iphi=71, depth=3',
'ieta=18, iphi=71, depth=3',
'ieta=18, iphi=71, depth=4',
'ieta=18, iphi=71, depth=5',
'ieta=28, iphi=71, depth=4'





Number of affected LS:
52



Dead channel anomaly:
LS: 6-56

Degraded channel anomaly:
LS: 57

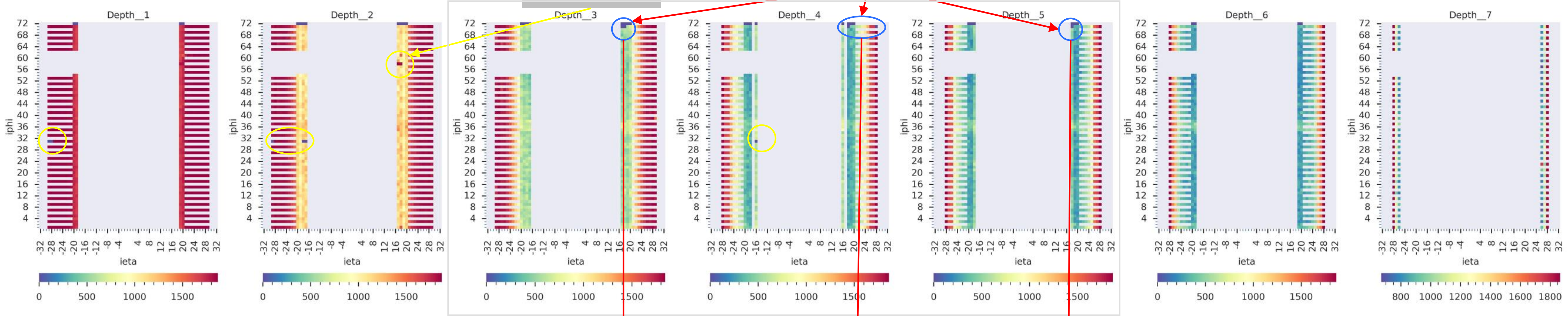
Detected Channel Anomalies: Detection

Run: 324841, LS: 10, Dead Channel

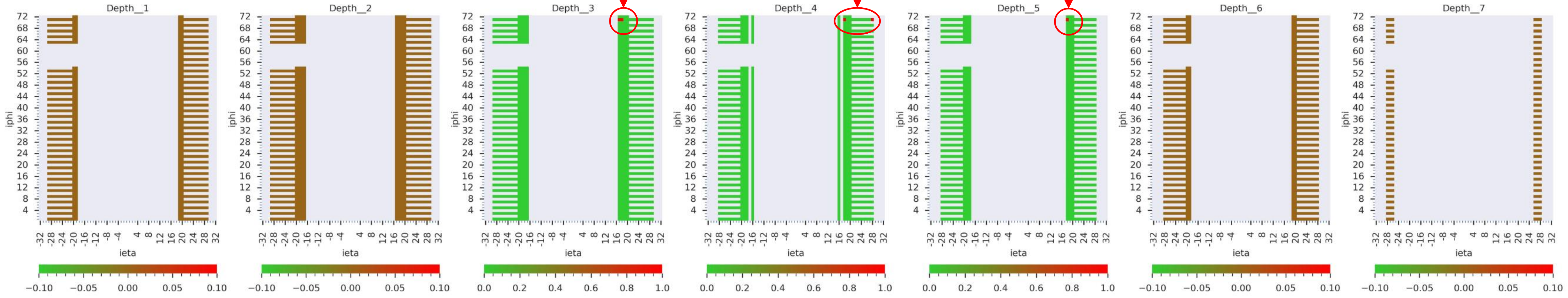
Previously Known Bad Channels

New Dead Channels

LS-10, digioccupancy map

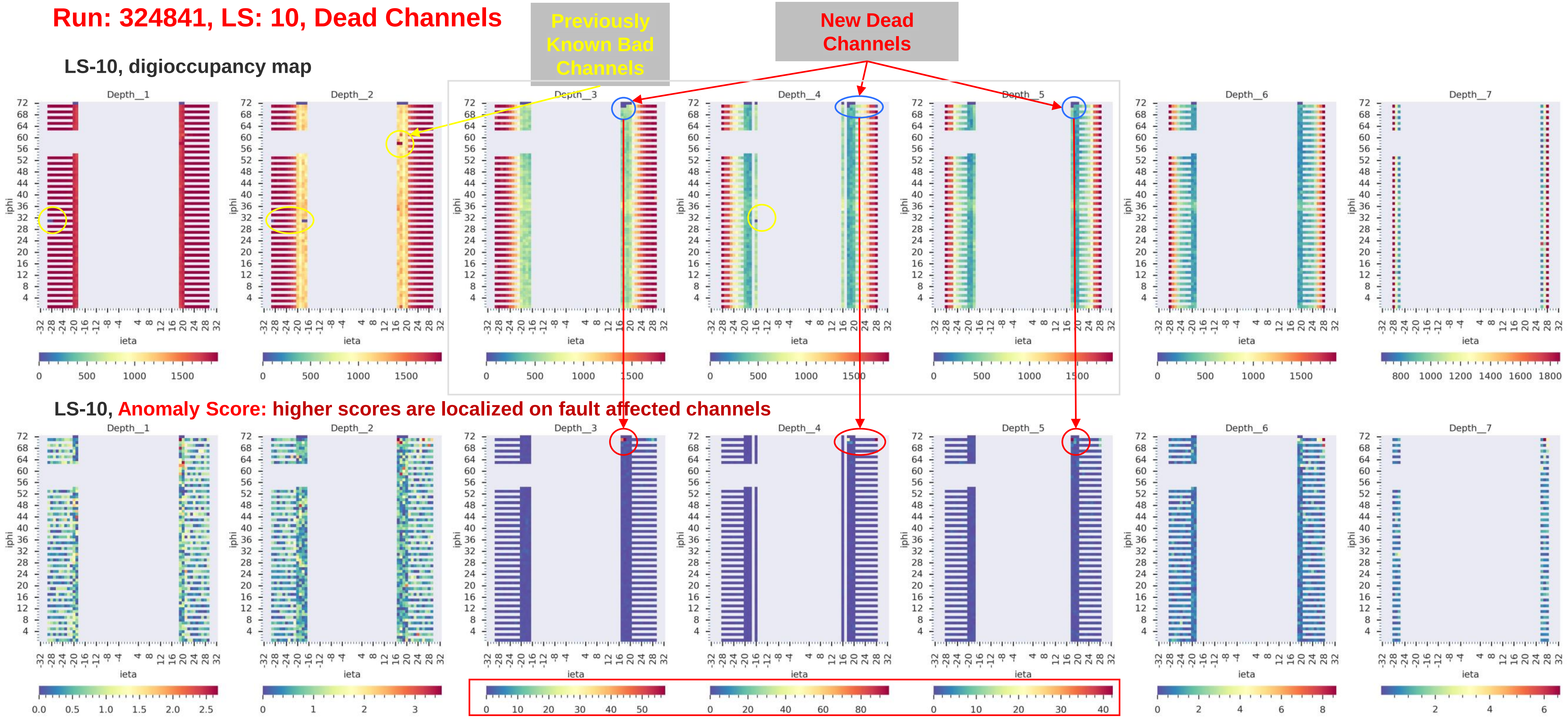


LS-10, Anomaly Flags (RED)



Detected Channel Anomalies: Fault Localization

Run: 324841, LS: 10, Dead Channels



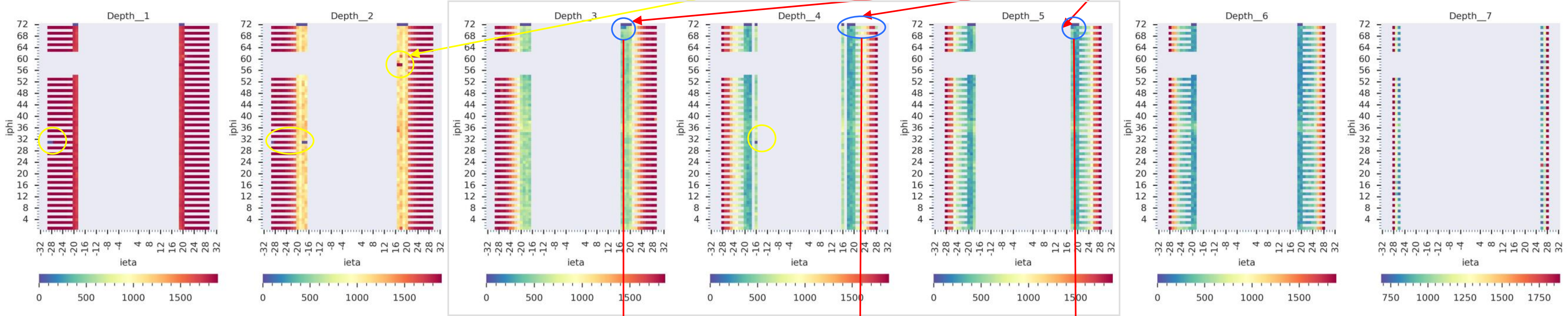
Detected Real Anomalies: Degrading

Run: 324841, LS: 57, Degraded Channels

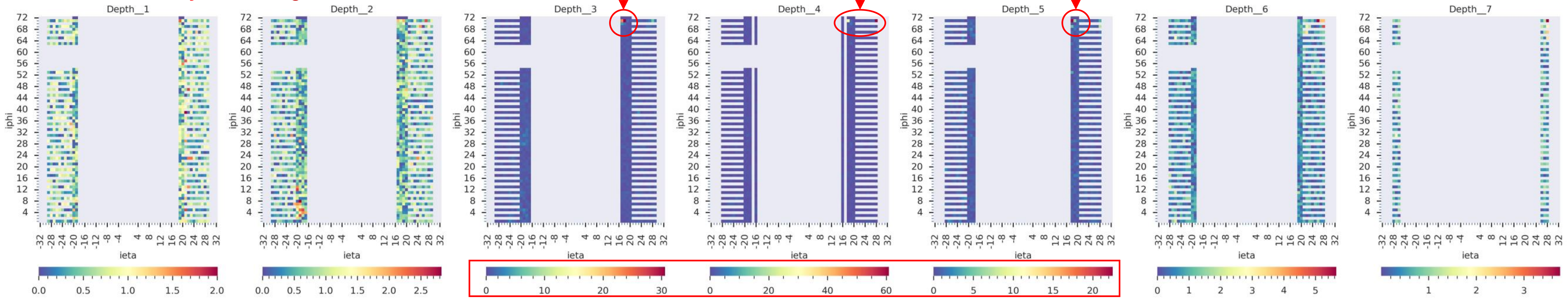
Previously Known Bad Channels

Degrading Bad Channels:
The digioccupancy is much lower than expected

LS-10, digioccupancy



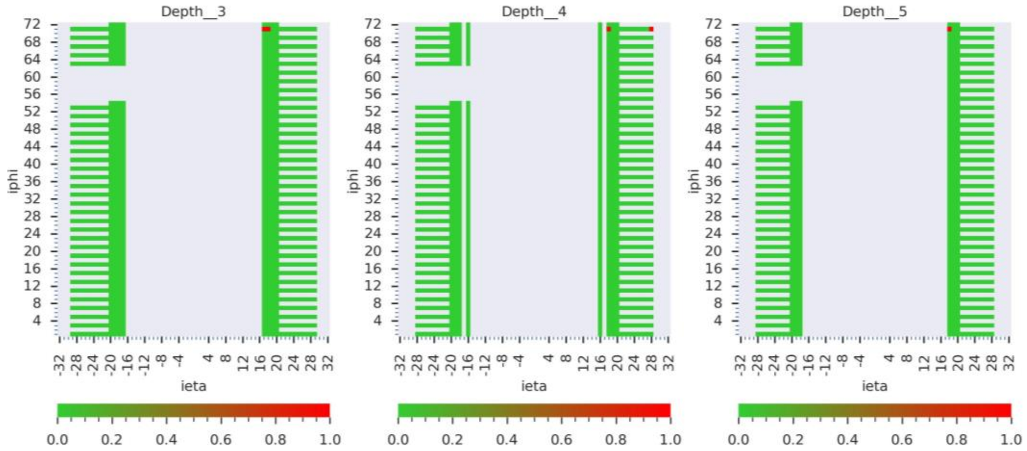
LS-10, Anomaly Score: higher scores are localized on fault affected channels



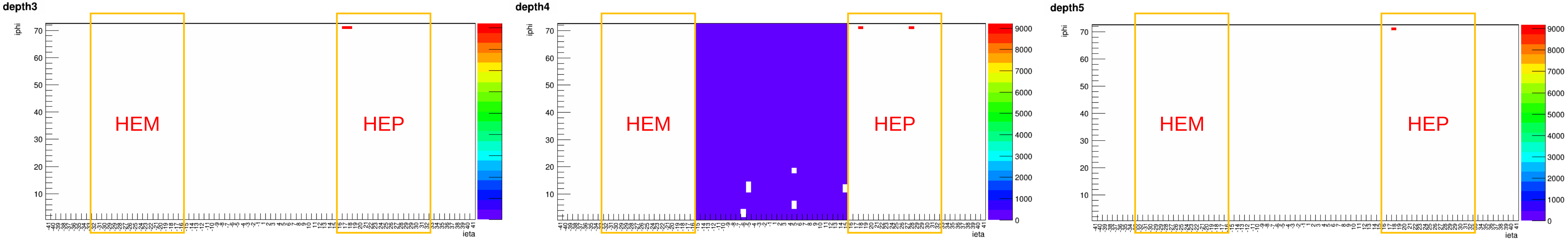
Detected Channel Anomalies: Central DQM vs DESMOD-DQMAD

- The Central DQM has also spotted the bad channels through **analysis at the end of the run**.
- Our approach detects the bad channels on streaming **instantly in a LS granularity**.

DESMOD-DQMAD based on LS level detection
Run: 324841



The Central DQM: uses run granularity detection
Run: 324841



Computational Complexity

- **Model Training:**
  - The models was developed with **PyTorch** and trained on **four GPUs of NVIDIA Tesla V100 SXM3 32 GB and Intel(R) Xeon(R) Platinum 8168 CPU @2.70GHz**.
  - Training time: around **45 sec/epoch** with **batch size B=8.**

- **Model Inference:**
  - **Inference time** on **a single GPU** is around $\boldsymbol{\mu = 0.05 \pm \sigma = 0.006}$ **seconds** where $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ is the median and a standard deviation of the inference time.

Summary

- The proposed model has **achieved promising performance** in capturing several types of anomalies from the **digioccupancy histogram map**.

- Our study expands the effort on **ML4DQM** on **temporal**, **digioccupnacy map normalization**, **learning non-Euclidean spatial behavior**, and **degrading channel detection**.

- The model's capability in detecting degrading channels will aid in **prognostics and predictive intervention.**

- The progress is currently on **model integration** in to the DQM production after fine tuning the model with occupancy maps of RUN-III.

- Previous DESMOD models are hosted in http://www.demond.cern.ch for pre-testing.

BACK-UP

Graph Convolutional Network (GCN)

- **GCN** is performs **convolution-like operations** directly on **graphs**.
- Given a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ data structure consisting of nodes $\mathcal{V}$ and edges $\mathcal{E}$ components, the graph convolution operation produces a **normalized aggregation of the node feature of the neighbors**:
- The layer-wise propagation mechanism:

$$h_i^{(l+1)} = \sigma\left(W^{(l)} \frac{1}{d_i} \sum_{j \in \mathcal{N}(i)} h_j^{(l)}\right)$$

$$\underbrace{H^{(l+1)} = \sigma\left(D^{-\frac{1}{2}} A D^{-\frac{1}{2}} H^{(l)} W^{(l)}\right) \quad D = diag\left(\sum_j A_{ij}\right)}_{\text{With symmetric normalized Laplacian matrics}}$$

$\boldsymbol{\mathcal{N}(i)}$ is the set of one-hop neighbors of the i^{th} node v_i with self-looping to include v_i in the set.
$\boldsymbol{d_i}$ is the number of neighboring nodes and used as a normalization constant
$\boldsymbol{\sigma}(\cdot)$ denotes an activation function such as $\boldsymbol{ReLU}(\cdot) = \boldsymbol{max}\ (\boldsymbol{0},\ \cdot)$
$\boldsymbol{h_i^l} \in \mathbb{R}^{\boldsymbol{1 \times M}}$ is the node feature vector of $\boldsymbol{i^{th}}$ node at the $\boldsymbol{l^{th}}$ layer, where $\boldsymbol{M}$ is the feature dimension. H is matrix equivalent.
$\boldsymbol{W^{(l)}}$ is a shared weight matrix for node-wise feature transformation

Previous Efforts on ML4DQM

- **Previous efforts on ML4DQM:**
  - Adrian et al. (2018) Detector monitoring with ANN at the CMS experiment (for the ECAL) [1]
  - Azzolin et al. (2019) Improving DQM via a partnership between the CMS and industry (for the ECAL) [2]
  - Abhirami Harilal (2021) ML based AD for the ECAL online DQM [3]

[1]https://arxiv.org/abs/1808.00911
[2]https://doi.org/10.1051/epjconf/201921401007
[3]https://indico.cern.ch/event/1045606/contributions/4458491/attachments/2288758/3890699/ML4DQM_MLTownhall_28July21AbhiramiH.pdf