AI4EIC Hackathon: PID with the ePIC dRICH

Cristiano Fanelli[7,5,*], *James* Giroux[7], *Diana* McSpadden[5], *Kishansingh* Rajput[5], *Karthik* Suresh[7], *Evaristo* Cisbani[4], *Wouter* Deconinck[6], *Eric* Walter[7], *Andrea* Bressan[3], *Markus* Diefenthaler[5], *Tanja* Horn[2], and *Torre* Wenaus[1]

[1]Brookhaven National Lab, Upton, NY, 33973, USA
[2]Catholic University of America, Washington, DC, 20064, USA
[3]INFN Trieste and University of Trieste, Department of Physics, Trieste, 34127, Italy
[4]Istituto Superiore di Sanitá, INFN, viale Regina Elena 299, Rome, 00161, Italy
[5]Thomas Jefferson National Accelerator Center, Newport News, VA, 23606, USA
[6]University of Manitoba, Physics, Winnipeg, MB, R3T 2Ns, Canada
[7]William and Mary, Williamsburg, VA, 23185, USA

Abstract.
The inaugural AI4EIC Hackathon unfolded as a high-point satellite event during the second AI4EIC Workshop at William & Mary. The workshop itself boasted over two hundred participants in a hybrid format and delved into the myriad applications of Artificial Intelligence and Machine Learning (AI/ML) for the Electron-Ion Collider (EIC). This workshop aimed to catalyze advancements in AI/ML with applications ranging from advancements in accelerator and detector technologies—highlighted by the ongoing work on the ePIC detector and potential development of a second detector for the EIC—to data analytics, reconstruction, and particle identification, as well as the synergies between theoretical and experimental research. Complementing the technical agenda was an enriched educational outreach program that featured tutorials from leading AI/ML experts representing academia, national laboratories, and industry. The hackathon, held on the final day, showcased international participation with ten teams from around the globe. Each team, comprising up to four members, focused on the dual-radiator Ring Imaging Cherenkov (dRICH) detector, an integral part of the particle identification (PID) system in ePIC. The data for the hackathon were generated using the ePIC software suite. While the hackathon presented questions of increasing complexity, its challenges were designed with deliberate simplifications to serve as a preliminary step toward the integration of machine learning and deep learning techniques in PID with the dRICH detector. This article encapsulates the key findings and insights gained from this unique experience.

1 Introduction

The AI4EIC Hackathon was a significant part of the second AI4EIC Workshop, which took place at William & Mary from October 10 to 14, 2022. This important gathering explored a wide variety of active and prospective applications in Artificial Intelligence (AI), Machine Learning (ML), and Deep Learning (DL) that are particularly relevant to the Electron-Ion Collider (EIC). Furthermore, the workshop acted as a showcase for ongoing research within

[*]e-mail: {cfanelli@wm.edu}

the newly-formed ePIC Collaboration, emphasizing the EIC program. The event was designed to encourage interdisciplinary collaboration by bridging theoretical concepts and experimental applications.

The workshop featured five interconnected sessions. The first session, AI/ML for Design, focused on the use of AI and ML in the design of detectors and accelerators. This was followed by the second session, Experiment/Theory Connections, which examined the symbiosis between theoretical frameworks and experimental results facilitated by AI and ML technologies. The third session, Reconstruction and Particle Identification (PID), discussed innovative techniques in reconstructing event scenarios and identifying particles. The fourth session, AI/ML Infrastructure and Frontiers, delved into the infrastructural requirements for transitioning from prototype to production, as well as emerging AI and ML technologies. Finally, the fifth session, AI/ML in Streaming Readout (SRO), looked into how AI and ML can be seamlessly integrated into real-time data processing to align offline and online analyses. Complementing the workshop's core themes was a series of in-depth tutorials. These tutorials not only served as a robust educational cornerstone for the workshop, but they were also helmed by a distinguished panel of experts drawn from academia, industry, and national research laboratories. Topics covered included Multi-Objective Optimization, offering nuanced insights into optimizing multiple objectives simultaneously; Graph Neural Networks, discussing the advantages and applications of this advanced neural network architecture; MLFlow, exploring the tools available for effective ML lifecycle management; and Unfolding with Deep Learning, demonstrating how to transform observed data back to its original distribution. For a more comprehensive overview of the AI4EIC Workshop, readers are encouraged to refer to the cited work [1].

The workshop culminated on the final day with the inaugural AI4EIC hackathon, which is described thoroughly in this document. Conducted in a hybrid and international format, the hackathon welcomed approximately 40 participants from diverse geographic regions such as America, Asia, and Europe. Participants were organized into 10 distinct teams, each vying to solve the challenges presented to them. For this hackathon, the focus was on particle identification (PID) using the dual-RICH detector, a critical instrument for charged particle detection in the hadronic endcap of the ePIC detector at the EIC. The event provided an opportunity to implement deep learning approaches, complete with a template to facilitate cloud-based distributed training of models. Deep learning offers the ability to accurately reconstruct complex topologies, such as events with multiple tracks, and can be seamlessly integrated into a streaming environment for near real-time applications. The problem chosen for the workshop was designed with an initial level of complexity that was manageable for participants, serving as an entry point for more advanced deep learning-based solutions. As the event progressed, the questions and challenges escalated in complexity, thereby paving the way for more sophisticated deep learning implementations.

The structure of this document is organized for clarity and depth. Section 2 delves into the intricacies of the hackathon problem, offering a detailed account of the problem statement, underscoring the significance of PID with dual-RICH in the ePIC experiment, and outlining the criteria and benchmarks for solution evaluation. Section 3 furnishes an overview of the available datasets, recommended programming languages, and libraries, while also providing insights into the computational resources, platforms, and tools used, along with a discussion of the methodologies and approaches adopted by participants. Finally, Section 4 encapsulates our learnings and insights, and offers a perspective on avenues for future work.

2 The hackathon problem: PID with dual-RICH

Background/Motivation

Cherenkov detectors constitute the backbone of PID at EIC [2]. When charged particles traverse the radiator medium—with appropriate refractive index and optical transparency—of a Cherenkov detector at speeds exceeding the phase velocity of light in that medium, they give rise to Cherenkov radiation, which emanates in a distinctive conical pattern.

The dRICH (dual-radiator Ring Imaging Cherenkov detector) [3] serves as a PID detector within the ePIC experimental setup. Designed for discerning hadronic particles ($\pi/K/p$) with a statistical separation better than 3σ, the dRICH operates over a momentum range from a few GeV/c up to approximately $50 GeV/c$ in the direction of the hadron trajectory. Architecturally, the dRICH comprises six uniform, laterally open sectors. The dRICH is equipped with a pair of radiators: aerogel and C_2F_6 gas, which share a common outward-focusing mirror and readout planes. These planes are outfitted with highly-segmented photosensors, featuring 3 mm × 3 mm pixels and situated beyond the acceptance zone for charged particles. To correct for optical aberrations, the photosensor tiles are configured on a curved surface. The open geometry of the dRICH sectors allows for the dispersion of photons from a single Cherenkov cone across multiple sectors. A schematic of the dRICH configuration is illustrated in Fig. 1.

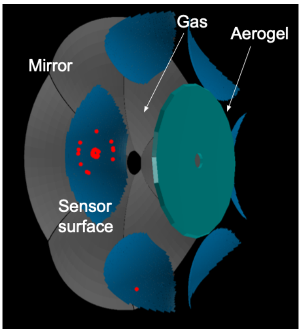


Figure 1: **Simulated Events in the dRICH**: The red points depict Cherenkov photons striking the sensor surface, generated by charged particles like π or K. In the displayed event, these photons are reflected off a mirror before arriving at one sector of the dRICH. The dRICH system utilizes two distinct radiators, aerogel and gas. Given the specific momentum of the charged track in this scenario, two concentric rings are produced: the smaller ring is attributable to gas radiation, while the larger ring is a consequence of aerogel radiation.

When working with imaging Cherenkov detectors such as the dRICH, researchers often face a series of complex challenges. For example, simulations frequently demand extensive computational resources, and the process of reconstruction necessitates sophisticated pattern recognition to decipher noisy and sparse photon rings, especially when dealing with complicated topologies such as those found in DIRC (Detection of Internally Reflected Cherenkov Light) configurations [4]. Additionally, events featuring closely-spaced tracks can easily lead to misidentification of particles.

Within this challenging landscape, ML/DL technologies offer a promising avenue for more efficient event-level reconstruction, as well as computational speed-up. The dRICH detector in the ePIC experiment is designed to facilitate efficient hadron identification across a broad momentum range, making it the focal point for our hackathon challenge. The central question we aimed to address through the hackathon is *whether ML/DL techniques can be effectively employed for PID based on low-level features derived from imaging Cherenkov de-*

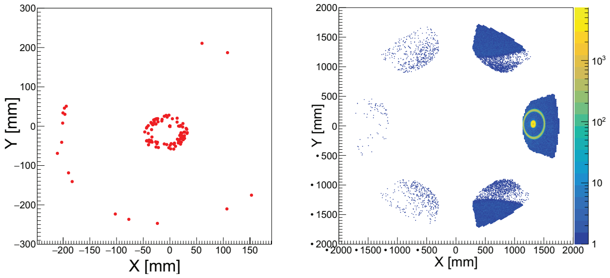


Figure 2: **Reconstructed rings in X-Y projections:** The hit pattern for a single π^+ corresponds to the kinematics presented in Question 1, as detailed in Table 1 (left). The smaller ring is produced by the gaseous radiator due to its lower refractive index. In contrast, the larger ring originates from the aerogel radiator, which possesses a higher refractive index. On the right, the hit pattern relates to Question 2, as shown in Table 1 (middle), and is based on data from 100k π^+ collected.

tectors. This initiative serves as an introductory step towards broader applications of ML/DL for PID within the realm of Cherenkov detectors.

Hackathon Challenges

Three questions of increased complexity have been proposed for the hackathon event.

For the first question (Question #1), the dataset comprises 1.5 million training events with distinct kinematic settings, as summarized in Table 1 (top). Each event unfolds in a magnetic field environment of approximately 1.5 T. All kinematic parameters—momentum, θ, and ϕ—are measured at the interaction point, denoted as the origin. The primary challenge here lies in distinguishing between pions (π^+ with PID=211) and kaons (K^+ with PID=321). Participants are encouraged to analyze the positions of optical photon hits in the dRICH detector using a test dataset of 100,000 events. Predictions can be submitted through the hackathon leaderboard.

Similarly, the second question (Question #2) provides participants with a more complex scenario, involving 3 million training events and continuous kinematic ranges as outlined in Table 1 (middle). The challenge remains to accurately identify pions and kaons based on the dRICH data.

The third question (Question #3) introduces an additional layer of complexity by incorporating random noise hits into the dataset, illustrated in Table 1 (right). Even in this noisy environment, the task remains to effectively categorize π^+ and K^+ particles. For each of these challenges, a test dataset of 100,000 events is available, and participants are invited to submit their solutions to the online leaderboard. Fig. 2 displays examples of events in an XY plane for charged π^+ corresponding to the increased complexity scenarios described in the text.

Question 1	
Training Events	1.5M
p	15 GeV/c
θ	20 °
ϕ	0 °

Question 2	
Training Events	3M
p	15-20 GeV/c
θ	15-16 °
ϕ	0-5 °

Question 3	
Training Events	3M
p	15-20 GeV/c
θ	15-16 °
ϕ	0-5 °

Table 1: **Kinematics for Questions 1, 2, and 3:** Events in Question 1 are generated at a fixed Kinematics, in contrast to Question 2 which are generated over a continuous space. Events in Question 3 also includes random noise hits, which average to about 1/10th of the expected number of optical photons.

eventID	PID	p	θ	ϕ	X_0	$\cdots$	X_{59}	Y_0	$\cdots$	Y_{59}	Z_0	$\cdots$	Z_{59}
0	211 (π^+) or 321 (K^+)	[GeV/c]	[deg]	[deg]	[mm]	[mm]	[mm]	[mm]	[mm]	[mm]	[mm]	[mm]	[mm]
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
N	211 or 321	[GeV/c]	[deg]	[deg]	[mm]	[mm]	[mm]	[mm]	[mm]	[mm]	[mm]	[mm]	[mm]

Table 2: **Summary of Data Format for Hackathon Problems**: This table provides a comprehensive overview of the data formats employed for the hackathon challenges, serving as a quick reference for participants to understand the structure and attributes of the datasets.

3 Infrastructure, Resources and Methodology

The AI4EIC hackathon, being geographically distributed, required a robust infrastructure. This encompassed both a leaderboard and submission website as well as the AWS cloud computing infrastructure. The associated dataset can be found in [5]. Teams joined the hackathon both in-person at William & Mary and remotely from various global locations. This setup mandated video conferencing capabilities at the hackathon's inception and conclusion. It also facilitated communication between the hackathon organizers and the remote teams. Further details about the hackathon, including information about datasets, training infrastructure, leaderboard and submission procedures, scoring mechanisms, and computational resources, are elaborated upon in the subsequent sections.

Datasets

The hackathon's training and test data are provided in `.csv` format, maintaining a consistent data structure across all questions. Specifically, each dataset comprises 185 columns. A single row within these datasets represents an event featuring a particular type of particle, either π^+ or K^+. The initial five columns contain essential event information: `eventID` for identifying the event, `PID` for the type of particle involved (note that this column is absent in test data), and kinematic variables—`momentum` in GeV, `theta` and `phi` in degrees. These kinematic variables correspond to values at the interaction point where beam collisions occur. Subsequent columns, comprising a total of 180, capture the positions of detected optical photons on the photo-sensor due to the particle tracks. These positions are labeled as `X0, X1, ..., X59`, `Z0, Z1, ..., Z59`, and `Y0, Y1, ..., Y59`, and represent the coordinates in millimeters. Each coordinate triplet (X_N, Y_N, Z_N) indicates the location of the N^{th} detected optical photon. To ensure uniform data formatting, we restrict the maximum number of detected optical photons to 60. In instances where fewer than 60 photons are detected, the remaining coordinates are padded with zeroes.

In addition to the initial `.csv` data format, a more compact and efficient `.h5` data format has been made available post-hackathon. Like the `.csv` data, each row in the `.h5` dataset also represents an event where a single charged particle is introduced into the dRICH detector. The dataset structure in the `.h5` format comprises fields such as `eventID` (an integer indicating the event ID), `PID` (an integer signifying the type of particle, either 211 or 321), and kinematic variables like `momentum`, `theta`, and `phi`—all represented as `float32`. These kinematic variables correspond to values at the interaction point where beam collisions occur. Moreover, arrays of `float32` are used for capturing the positions of detected optical photons due to particle tracks in the dRICH detector. Specifically, `positionX`, `positionY`, and `positionZ` store the X, Y, and Z coordinates, respectively, of photon hits resulting from the particle's interaction with the radiators. For a more detailed overview of both the `.csv` and `.h5` data formats, please refer to Table 2.

Training infrastructure

A specialized training infrastructure was constructed for both PyTorch and Tensorflow users. Both environments provided detailed training templates, example inference scripts, and optimized data loading pipelines to limit resource contention on an individual AWS instance. While resource usage was ultimately left for groups to manage, our pipelines minimized memory consumption during training such that extensive data exploration could be conducted in parallel if desired. Users were provided with instructions on how to allocate multiple GPUs in order to facilitate distributed training. The adaptation of code for such use cases required minimal additions and was left to the participant. Fig. 3 shows the workflow that has been implemented. For an in-depth explanation of the scripts, including their architecture and functionalities, we refer the reader to our previous work [5].

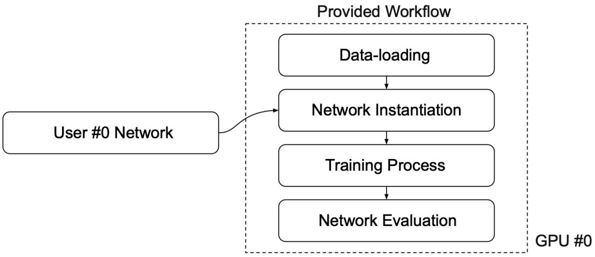


Figure 3: **Simplified workflow for ML/DL training during the hackathon**: This figure illustrates the user-provided workflow integrating seamlessly into our specialized training infrastructure. The process encompasses Data-loading, Network Instantiation, Training, and Network Evaluation. Our setup accommodates both PyTorch and TensorFlow, offering optimized data pipelines and the flexibility for distributed training across multiple GPUs with minimal code adaptation.

Leaderbord and submission infrastructure

To enhance the hackathon experience, an interactive leaderboard and a user-friendly dashboard that allowed teams to seamlessly access and submit their solutions have been provided. This framework was developed utilizing Flask in conjunction with SQLite databases, ensuring secure storage of team credentials and comprehensive submission histories. Every participant gained exclusive access to their personalized dashboard, which empowered them to monitor their solution submissions and seamlessly upload their entries to tackle the hackathon's challenging problems. Behind the scenes, the system evaluated and graded these submissions, subsequently updating the dashboard in real-time. This dynamic and transparent process heightened the level of enthusiasm among participants, driving them to engage in a more competitive and spirited manner throughout the hackathon.

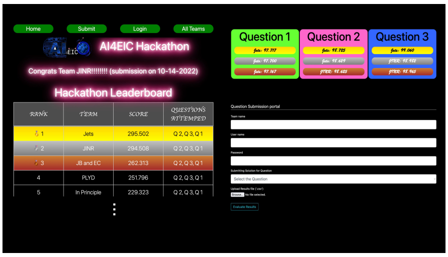
Fig. 4 shows a snapshot of the hackathon's submission portal, highlighting its interactive leaderboard, team member profiles, dedicated questions section, and an integrated submission area with automated solution validation, all designed to enhance team collaboration, real-time ranking, and immediate feedback on submissions.

Computational resources

Access to cloud computing resources has been provided during the event, and each team was endowed with an Amazon Web Services (AWS) g5.12xlarge instance, 4 Nvidia a10g GPUs, 48 vCPUs, 192GB Ram, 3.9 TB of disk.

Scoring mechanism

Participants will find an `eventID` column in the `test` dataset provided to them. Their task is to make predictions exclusively using the Particle Data Group (PDG) codes for π^+ (211)



AI4EIC Hackathon

Congrats Team JINR!!!!!!!! (submission on 10-14-2022)

Hackathon Leaderboard

RANK	TEAM	SCORE	QUESTIONS ATTEMPED
1	Jets	295.502	Q 2, Q 3, Q 1
2	JINR	294.508	Q 2, Q 3, Q 1
3	JB and EC	262.313	Q 2, Q 3, Q 1
4	PLYD	251.796	Q 2, Q 3, Q 1
5	In Principle	229.323	Q 2, Q 3, Q 1

⋮

Question 1	Question 2	Question 3
Jets: 97.717	Jets: 98.725	Jets: 99.060
Jets: 97.700	Jets: 98.629	JINR: 98.982
Jets: 97.167	JINR: 98.625	JINR: 98.965

Question Submission portal

Team name

User name

Password

Submitting Solution for Question

Select the Question

Upload Results file ('.csv')

Browse... No file selected.

Evaluate Results

Figure 4: **AI4EIC hackathon submission portal**: A snapshot of the hackathon's submission portal (available at ai4eichackathon.pythonanywhere.com). The portal features an interactive leaderboard, team member profiles, a questions section, and a submission area with automated solution validation. This infrastructure was developed to facilitate seamless team collaboration, real-time ranking, and immediate feedback on submissions.

and K^+ (321). These predictions should be submitted in `.csv` format, the only format supported by our submission portal. Importantly, the `.csv` files should not contain any headers. Typically, the prediction files have a manageable size, under 1MB, comfortably within the portal's file size limit of 3MB.

Submissions are welcome throughout the duration of the hackathon, without restrictions on the number of attempts. Evaluation criteria are primarily based on prediction accuracy, as described by the formula:

$$\text{ACCURACY} = \frac{\Sigma_{N_{test}} |y_{\text{pred}} == y_{\text{true}}|}{N_{\text{test}}}$$

The problems are organized by escalating complexity, and scores are adjusted accordingly. Each problem has an accuracy threshold, predetermined via a custom benchmark study. If a submission's accuracy falls below this threshold, it scores a zero but should be seen as a stepping stone, not a setback. On the other hand, for accuracies above the threshold, scores are calculated between 50 and 100 based on the formula:

$$\text{SCORE} = 50.0 + 50.0 \times \frac{(\text{ACCURACY} - \text{THRESHOLD})}{100.0 - \text{THRESHOLD}}$$

The thresholds for individual problems are provided in a dedicated table.

Time constraints

The hackathon commences at 10:00 AM E.T on October 14, 2022, with the submission window closing promptly at 5:00 PM E.T the same day. Although late submissions won't be eligible for prizes, the portal remains open until 5:00 PM E.T on October 17, 2022, for those wishing to further refine their solutions for personal enrichment.

Methodology

An interactive leaderboard was seamlessly integrated into the submission portal, fostering both competition and real-time performance feedback. Participants actively engaged with this feature, submitting their solutions for automatic, real-time grading. This instant evaluation mechanism not only intensified competition but also encouraged an atmosphere of continuous improvement and learning. The portal's functionality and real-time feedback mechanisms were so well-received that participants continued to submit and refine their solutions even after the official conclusion of the hackathon event.

Emerging victorious in this competitive landscape was the "JINR" team. Their strategy leveraged the CatBoostClassifier [6], an advanced machine learning algorithm known for its robustness against overfitting and its high accuracy in complex classification tasks.

In a close second place was the "Jets" team, who opted for a Convolutional Neural Networks (CNN) approach [7]. Given that CNNs excel in image and spatial data tasks, they are a natural choice for challenges that can be interpreted as image classification problems. Their strategy demonstrated both innovation and effectiveness, earning them their high placement. In an intriguing post-hackathon development, submissions remained open for an additional day. Remarkably, the "Jets" team managed to refine their model further, ultimately achieving the highest score in an absolute sense during this extended period.

In recognition of their achievements, both teams have been invited to share their methods, insights, and lessons learned at the forthcoming AI4EIC workshop.

4 Conclusions

The AI4EIC hackathon served as an exemplary platform for fostering interdisciplinary collaboration, attracting participants from diverse fields such as computer science, data science, and physics. This symbiosis led to innovative projects that are aligned with the overarching goals of the EIC. With a participation of 10 teams around the world, the event highlighted the power of collective problem-solving. Importantly, the winning solutions showcased the immense utility of Machine Learning and Deep Learning techniques in addressing complex issues, particularly in the realm of particle identification using imaging Cherenkov detectors. These successes substantiate the role of advanced algorithms as not just ancillary tools but potentially central elements in the next wave of R&D within the PID landscape. In light of the hackathon's achievements and the evident potential for Machine Learning and Deep Learning in this field, AI4EIC hackathons will now be an annual event. The next event is already scheduled for December 2023 at The Catholic University of America (CUA) in Washington D.C. The series aims to continue acting as an accelerator for targeted and effective research, propelling advancements in PID technologies through machine learning.

Acknowledgments

The authors express their gratitude to AWS for sponsoring the cloud computing infrastructure essential for the hackathon's success. Our appreciation also extends to William and Mary, whose generous sponsorship not only supported the hackathon organization but also funded the prizes awarded.

References

[1] C. Allaire, R. Ammendola, E.C. Aschenauer, M. Balandat, M. Battaglieri, J. Bernauer, M. Bondì, N. Branson, T. Britton, A. Butter et al., arXiv preprint arXiv:2307.08593 (2023)

[2] C. Fanelli, A. Mahmood, Journal of Instrumentation **17**, C07011 (2022)

[3] E. Cisbani, A. Del Dotto, C. Fanelli, M. Williams et al., Journal of Instrumentation **15**, P05009 (2020)

[4] C. Fanelli, J. Pomponi, Machine Learning: Science and Technology **1**, 015010 (2020)

[5] C. Fanelli, J. Giroux, D. McSpadden, K. Rajput, K. Suresh, *AI4EIC Hackathon: PID with the dRICH and possibilities for Machine Learning and Deep Learning approaches* (2022)

[6] L. Prokhorenkova, G. Gusev, A. Vorobev, A.V. Dorogush, A. Gulin, Advances in neural information processing systems **31** (2018)

[7] W. Rawat, Z. Wang, Neural computation **29**, 2352 (2017)