RICH AI - MDS Team Proposal (DRAFT)

Nico Van den Hooff, Mukund Iyer, Rakesh Pandey, Shiva Jena

May 13, 2022

Contents

1 Executive summary

The capstone project titled "*RICH AI*" floated by TRIUMF (Canada's national particle accelerator center) is aimed at improving Particle Identification (PID) performance of the Ring Imaging Cherenkov (RICH) by developing a classification model based on advanced Machine Learning (ML).

The background of the project lies in the NA62 experiment at CERN (European organization for nuclear research), which aims to measure an ultra-rare meson decay reaction to identify a pion, which is produced along with a muon and a positron. The main challenge is to increase pion efficiency, that is, the fraction of the matched detected pions out of total number of pions or the true positive rate.

The goal of our Capstone project is to build a deep learning model that can accurately classify a particle produced in a decay event. The formal deliverable for our project is a Python package that encapsulates this model. The data for our project was generated as part of the 2018 NA62 experiments performed at CERN. There are a total of 11 million labelled decay events, that contain features to be used in building our model.

2 Introduction

2.1 Particle physics

Particle physics involves the study of subatomic particles that are the most elementary constituents of matter. However these entities are extremely small and exist at very high energies, and therefore complex instruments are needed to study and characterize them. The NA62 experiment at CERN is one such experiment which is focused on the study of the decay a Kaon particle into a pion, muon, or positron. The objective of this experiment is to verify the theoretical Standard Model which explains this rate of decay by precisely comparing the relationships between the quarks (Gil et al. 2017).

The current challenges of this study arise due to the rarity of a pion decay event, and the stochastic and uncontrolled nature of each particle decay. Further, the probability of each decay is non-uniform. Hence, the precision in identifying the pion after the decay process is of key interest.

2.2 RICH detector

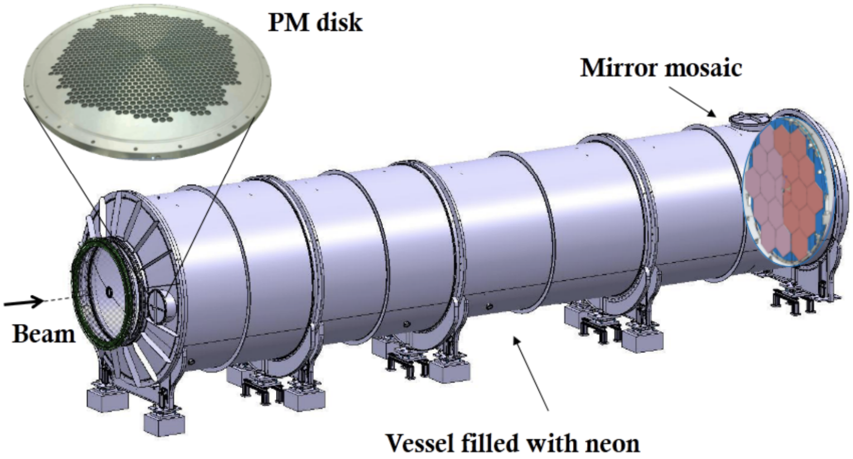

Figure 1: RICH AI beam and detector set up.

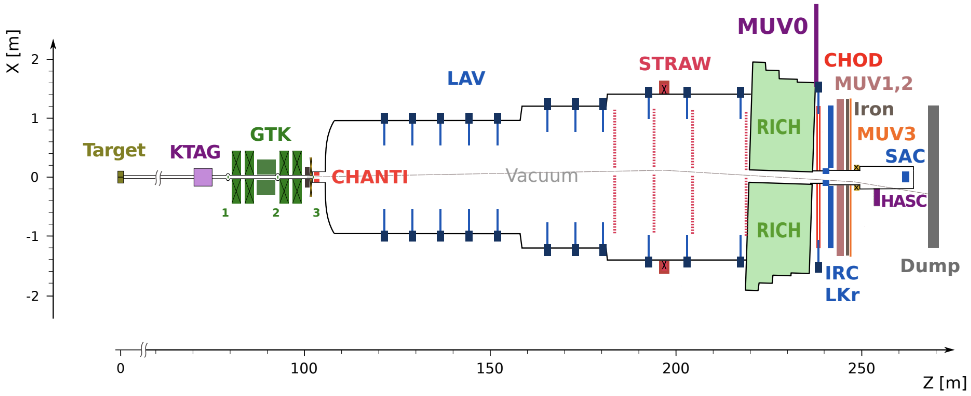

Figure 2: NA62 experiement beam and detector set up.

The main detector in the NA62 experiment is called a RICH detector as shown in Figure 1 (Anzivino et al. 2020). The RICH detector collects data on light that is emitted during each particle decay. Other detectors as shown in Figure 2 (Anzivino et al. 2020) measure additional data such as particle momentum, velocity, and time of passage.

The data collected by the RICH detector forms an image of a ring, as it is the 2D projection of the conical light emitted due to the Cherenkov radiation (Gil et al. 2017). The light is emitted in packets of energy called photons, and the image produced can be represented as a scatter plot of x and y coordinates.

2.3 How data science can help

The goal of the NA62 experiment at CERN can be expressed as a classification problem that can be solved using data science techniques. Specifically, a classification machine learning or deep learning model can be built using the data generated in the experiment. The goal in building the model would be to accurately classify each particle decay, with an emphasis on pion decays.

2.4 Project deliverable

The project deliverable is a Python package that allows for training of the model that is developed, as well as evaluation on new data. A knowledge transfer session will also be performed, so that the package can be utilized and further developed by the NA62 project team at TRIUMF.

We plan to utilize `poetry` in developing our package so that the respective dependencies are effectively managed. The product will be designed so that it can be applied to the NA62 data from any year with the same input structure. The package scripts will be modular so that the package can easily be incorporated into the existing architecture at TRIUMF.

3 Data science techniques

3.1 Data

3.1.1 Data generation process

In order to generate the NA62 data, several experiment "runs" are performed. For each run, the experiment configuration is fixed and the following steps are performed:

1. A beam rich in kaon particles is delivered in "bursts" every four or five seconds into the beam and detector set up as shown in Figure 2.
2. During a burst, several particle decays occur. Each particle decay has an individual "event" ID associated with it.
3. The product of the decay is accelerated through a chamber of neon gas in the RICH detector and produces/emits a cone of light. The RICH detector is shown in Figure 1.
4. The cone of light is reflected by a mosaic of mirrors onto an array of photomultiplier tubes ("PMT"). In an ideal situation, the cone of light forms a "ring" on the PMT array.
5. Each individual PMT in the array records whether or not it was hit by light, as well as the time of arrival for each hit of light.
6. The hodoscope counters (CHOD) detector shown in Figure 2 records the time that the particle decay occurs.

3.1.2 RICH AI dataset

Our project will be utilizing data as part of the 2018 NA62 experiment. The dataset contains information on approximately 11 million particle decays and is stored in HDF5 format.

- Decay label (pion, muon, positron)
- PMT data
  - Hit locations (x, y)
  - Hit times
- Particle momentum
- CHOD time
- Features from TRIUMF's maximum likelihood fit:
  - Ring radius
  - Ring centre (x, y)

    - Ring likelihood

- Metadata (Run ID, Burst ID, Event ID etc.)

The total number of decays for each class is as follows:

- 10,281,301 muon decays
- 1,169,556 pion decays
- 113,112 positron decays

3.2 Machine learning

3.2.1 Task

The task of our project is to build a machine learning model that can accurately classify an individual particle decay event as a pion, muon, or positron. The rarest and most interesting decay is a pion, whereas muon and positron decays are more common and less interesting. We note that the rarity of pion decays is demonstrated by the imbalanced dataset.

3.2.2 TRIUMF's current approach

TRIUMF currently employs an analytical approach without the use of machine learning to classify particle decays. Specifically, maximum likelihood estimation is used to fit a ring to the PMT hit data, and the ring radius is used to classify the particle decay. With this methodology, TRIUMF achieves the following results:

1. pion efficiency of 95% (true positive rate for pion classification)
2. Misclassification of a pion as a muon 1% of the time.

3.2.3 Goals

In building our machine learning model we have the following goals:

1. Achieve a pion efficiency of at least 95% or greater.
2. Reduce pion/muon misclassification by 10x to 0.01% of the time.

3.2.4 Metrics

In building our machine learning model, we will attempt to maximize pion efficiency, which is defined as:

$$\text{pion efficiency} = \frac{\text{total number of pions detected}}{\text{total number of pions in dataset}}$$

In comparing multiple machine learning models, we will utilize receiver operating characteristic ("ROC") curves.

3.2.5 Baseline model

A gradient boosted tree will serve as our baseline model. We will try both XGBoost and LightGBM. These models will be trained on raw feature data, and the one that performs better will be selected as the baseline.

3.2.6 Deep learning models

In building a more complicated machine learning model we will employ deep learning. We note that the image data produced by the RICH detector is sparse. Hence a traditional CNN would be disadvantageous as a significant portion of the image contains no information.We have identified two potential model architectures that may work well in achieving our task and goals:

1. PointNet (Qi et al. 2017)
2. Graph Convolutional Neural Networks ("Graph CNN") (Zhou et al. 2018)

The first model architecture is called PointNet, which was originally designed to work with point cloud coordinate data. We will first try out the "vanilla" version of PointNet, and if needed we will also implement the more complex updated architecture called PointNet++. We will also try out Graph CNNs on our data.

3.2.7 Implementation

The baseline model will be implemented with the python libraries for XGBoost (Chen and Guestrin 2016) or LightGBM (Ke et al. 2017), respectively. The deep learning models will be built with PyTorch (Paszke et al. 2019) and PyTorch Geometric (Fey and Lenssen 2019).

4 Communication and timeline

4.1 Overall project timeline

The total time allocated for our capstone project is eight weeks. We propose the following high level project timeline:

Week 1 (May 2 to May 6)

- MDS hackathon
- Initial meetings with TRIUMF to learn about data generation process, machine learning options and helper scripts provided
- Brainstorming problem solutions
- Prepare and complete proposal presentation to MDS faculty

Week 2 (May 9 to May 13)

- Perform exploratory data analysis
- Create input data pipeline for machine learning models
- Begin implementation of machine learning models
- Prepare and submit formal written proposal to TRIUMF

Week 3, 4, 5 (May 16 to June 3)

- Build machine learning models

Week 6 (June 13 to June 17)

- Evaluate models on test set against target metrics
- If needed, further iterate and improve models
- Modularize code
- Create final deliverable python package

Week 7 (June 20 to 24)

- Document code base and steps required to run in production
- Write final project report

Week 8 (June 27 to June 30)

- Submit final product to TRIUMF and perform handover session

References

Anzivino, G, M Barbanera, A Bizzeti, F Brizioli, F Bucci, A Cassese, P Cenci, et al. 2020. "Light Detection System and Time Resolution of the Na62 RICH." *Journal of Instrumentation* 15 (10): P10025.

Chen, Tianqi, and Carlos Guestrin. 2016. "XGBoost." In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM. https://doi.org/10.1145/2939672.2939785.

Fey, Matthias, and Jan Eric Lenssen. 2019. "Fast Graph Representation Learning with PyTorch Geometric." arXiv. https://doi.org/10.48550/ARXIV.1903.02428.

Gil, E. Cortina, E. Martin Albarran, E. Minucci, G. Nüssle, S. Padolski, P. Petrov, N. Szilasi, et al. 2017. "The Beam and Detector of the NA62 Experiment at CERN." *Journal of Instrumentation* 12 (05): P05025–25. https://doi.org/10.1088/1748-0221/12/05/p05025.

Ke, Guolin, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. "Lightgbm: A Highly Efficient Gradient Boosting Decision Tree." *Advances in Neural Information Processing Systems* 30: 3146–54.

Paszke, Adam, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, et al. 2019. "PyTorch: An Imperative Style, High-Performance Deep Learning Library." In *Advances in Neural Information Processing Systems 32*, edited by H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, and R. Garnett, 8024–35. Curran Associates, Inc. http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

Qi, Charles R, Hao Su, Kaichun Mo, and Leonidas J Guibas. 2017. "Pointnet: Deep Learning on Point Sets for 3d Classification and Segmentation." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 652–60.

Zhou, Jie, Ganqu Cui, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, and Maosong Sun. 2018. "Graph Neural Networks: A Review of Methods and Applications." *CoRR* abs/1812.08434. http://arxiv.org/abs/1812.08434.