GSoC 2023 Project Proposal

Organisation : ML4SCI

CMS : Graph Neural Networks for Particle Momentum Estimation in the CMS Trigger System

MENTORS

Suzanne Rozenzweig, Efe Yigibasi, Darin Acosta, Sergei Gleyzer, Ali Hariri

Personal Information:

- Name: Ishaan Watts
- University: IIT Delhi, Engineering Physics
- Email: wattsishaan18@gmail.com
- GitHub: WattsIshaan
- LinkedIn: Ishaan-Watts
- Mobile: +91-8527946043
- Website: wattsishaan.github.io

1 Abstract

This project aims to improve the performance of the CMS experiment by implementing and evaluating the use of deep learning algorithms, such as graph networks (GNNs), for the Level-1 trigger system. The CMS experiment uses machine learning algorithms to estimate the momentum of particles like Muons that pass through it. At present, the trigger system uses a discretized boosted decision tree algorithm. However, CMS is exploring the use of GNNs for the same purpose. The challenge is that the deep learning algorithms need to provide inference results in microseconds to be useful for the trigger system. The project will focus on implementing and testing the performance of GNNs for the trigger inference task. The goal is to optimize the inference process to achieve high accuracy while meeting the stringent time requirements of the Level-1 trigger system.

2 Introduction

The Compact Muon Solenoid (CMS) is a particle detector located at the Large Hadron Collider (LHC) at CERN, which is used to investigate the fundamental nature of matter and the universe by colliding protons or heavy ions at high energies and observing the resulting particles. One of the key particles that CMS is interested in detecting is the muon, which is an elementary particle similar to an electron but with a much greater mass.

Muon detection is crucial in studying high-energy particle physics as muons are produced in high-energy collisions and can travel through matter more easily than other charged particles. Detecting and measuring the momentum of muons is important for understanding the behavior of high-energy particles and studying various physical phenomena, such as the Higgs boson.

The CMS experiment uses a Level-1 trigger system to quickly filter out interesting events from the vast amount of data produced by the detector. The Level-1 trigger system uses a machine learning algorithm to estimate the momentum of particles such as muons that pass through the detector. Currently, a discretized boosted decision tree algorithm [1] is used for this task, but CMS is exploring the use of deep learning algorithms to potentially improve accuracy.

Deep learning algorithms, such as graph networks (GNNs), have shown promising results in tasks involving structured data, such as graphs or networks. In this project, we aim to improve the performance of the Level-1 trigger system by exploring the use of GNNs for the momentum estimation task.

The project will involve several steps, starting with the selection and implementation of a suitable GNN architecture. The detector data can be represented as a graph, where each node represents a particle and the edges represent the relationships between them. The GNNs will be trained on a large dataset of simulated detector data and evaluated for accuracy and speed. Benchmarking will involve comparing the performance of the GNN-based approach to the current algorithm used in the Level-1 trigger and other potential approaches such as other types of deep learning architectures or traditional machine learning algorithms.

The goal of this project is to improve the accuracy and speed of the momentum estimation task in the Level-1 trigger system, potentially enabling new discoveries and insights in the study of high-energy particle physics. By exploring the use of GNNs for this task, we aim to push the boundaries of what is possible with machine learning and contribute to ongoing efforts to understand the fundamental nature of matter and the universe.

3 Evaluation Tasks

3.1 Task 1 : Electron/photon classification

Use a deep learning method of your choice to achieve the highest possible classification on this dataset (we ask that you do it both in Keras/Tensorflow and in PyTorch). Please provide a Jupyter notebook that shows your solution. The model you submit should have a ROC AUC score of at least 0.80.

Dataset Description

32x32 matrices (two channels - hit energy and time) for two classes of particles electrons and photons impinging on a calorimeter. photons and electrons

Model Architecture

Used a modified ResNet-15 model and hyper-parameters as explained in the paper End-to-End Physics Event Classification with CMS Open Data [2].

Results

Implementation	Test Accuracy	F1 score	ROC-AUC
Keras	0.737	0.737	0.8078
PyTorch	0.736	0.738	0.8058

Table 1: Results of Task 1

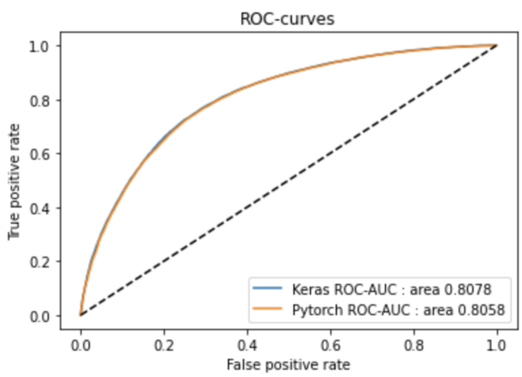

Figure 1: ROC Curve for Task 1

3.2 Task 2 : Graph Neural Networks

Choose 2 Graph-based architectures of your choice to classify jets as being quarks or gluons. Provide a description on what considerations you have taken to project this point-cloud dataset to a set of interconnected nodes and edges. Discuss the resulting performance of the 2 chosen architectures. Dataset link.

Model Architecture

Trained two GNN architectures for quark-gluon classification. Input feature are a matrix of size (N, M, 4) where,

- N is the number of jets.
- M is the maximum particles in a jet.
- 4 are the features of each particles denoting (pt, eta, phi, pid) values.

The graph for each jet is constructed in the following way:

- First for each jet we remove all particle having all feature values zero i.e., remove padding.
- Next we calculate the pair-wise euclidean distance between the nodes using this metric,

$$R = \sqrt{\Delta\eta^2 + \Delta\phi^2}$$

I have set a threshold for R above which we do not consider an edge between the nodes. The chosen value after experimenting is 0.05.

- Next, edge-index are formed and the reverse edges are also concatenated.
- R-values are set as the edge-weights and all the 4 feature are set as the node-features.

- **Architecture-1**
  - 3-layer GCN network with ReLu for aggregation of node-level features.
  - Readout layer as global mean pooling for graph-level embedding.
  - A dropout layer followed by linear layer.

- **Architecture-2**
  - Use of GraphConv layer in-place of GCN layer. It adds skip connections in the network to preserve central node information and omits neighborhood normalization completely.
  - The same readout layer with global mean pooling is used. I also tried using a combination of global mean and global max pool but it lead to decrease in performance.
  - This is followed by an additional linear layer with ReLu. Then a dropout and final linear layer.

Results

Architecture	Test Accuracy	F1 score	ROC-AUC
1 - GCN	0.768	0.762	0.841
2 - GraphConv	0.792	0.788	0.872

Table 2: Results of Task 2

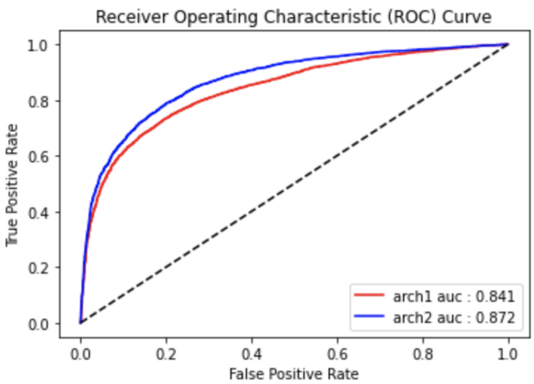

Figure 2: ROC Curve for Task 2

4 Proposed Deliverables

1. A detailed report outlining the GNN architecture selection process, training procedures, and performance evaluation metrics. The report should also include a comparison of the GNN-based approach with the current algorithm used in the Level-1 trigger system and other potential approaches such as other types of deep learning architectures or traditional machine learning algorithms.
2. A fully functional GNN-based momentum regression model that can be integrated into the Level-1 trigger system for real-time inference. The model should be trained on a large dataset of simulated detector data and should achieve high accuracy and speed requirements specified by CMS.
3. Benchmarking results comparing the performance of the GNN-based approach with the current algorithm used in the Level-1 trigger system and other potential approaches such as other types of deep learning architectures or traditional machine learning algorithms. The benchmarking results should include accuracy, inference speed, and memory requirements.
4. Code documentation and codebase to ensure easy integration into the existing CMS software infrastructure.
5. A presentation summarizing the project goals, methodology, and results for the CMS team and other stakeholders.

Overall, the deliverables for this project aim to improve the accuracy and speed of the momentum estimation task in the Level-1 trigger system, potentially enabling new discoveries and insights in the study of high-energy particle physics.

5 Project Timeline

May 4 - 28: Community Bonding Period

- Read relevant literature and documentation on CMS experiment and Level-1 trigger system.
- Familiarize with the ROOT framework and datasets used in the CMS experiment.
- Discuss project goals and expectations with mentor and develop a detailed plan for the project, including selecting and implementing a suitable GNN architecture for the momentum regression task.

May 29 - June 11: Week 1-2

- Implement the selected GNN architecture in Python using pytorch-geometric. Train the GNN on a small dataset and verify its correctness.
- Explore various hyper-parameters and optimize the GNN's performance on the small dataset.

June 12 - June 25: Week 3-4

- Train the GNN on a larger dataset of simulated detector data. Evaluate the GNN's performance on the training data, including accuracy and inference time.
- Explore methods for further optimizing the GNN's performance, such as reducing the number of layers or nodes in the network.

June 26 - July 9: Week 5-6

- Begin testing the GNN on real detector data from the CMS experiment.
- Evaluate the GNN's performance on the real data, including accuracy and inference time.

- Compare the performance of the GNN to the current algorithm used in the Level-1 trigger and other potential approaches.

July 10 - July 14: Midterm Evaluation Period

- Submit midterm evaluation to mentor.
- Refine project plan based on feedback from mentor.
- Discuss progress with mentor and provide a midterm evaluation of my work.

July 15 - August 7: Week 7-10

- Implement any necessary changes to the GNN architecture or training process based on the evaluation results and mentor feedback.
- Fine-tune the GNN's hyper-parameters to improve performance on the real detector data.
- Evaluate the GNN's performance on a larger dataset of real detector data.

August 8 - August 21: Week 11-12

- Finalize the GNN implementation and prepare the final report.
- Write detailed documentation and instructions for using the GNN in Level-1 trigger system.
- Work with mentor to prepare the final presentation and demo for submission.

August 21 - August 28: Final week

- Submit the final report, code, and documentation to the mentor.
- Complete the final mentor evaluation.
- Participate in the final presentation and demo.

This timeline is subject to change based on mentor feedback and project progress, but it provides a general outline of the tasks to be completed each week. Documentation and weekly-logs of the work are also updated regularly.

6 Other Information

6.1 Why ML4SCI?

As a final year physics major with a minor in computer science and extensive machine learning experience, I am thrilled to join ML4SCI. The opportunity to work on existing scientific collaborations and contribute to solving important scientific challenges using machine learning is truly inspiring. Additionally, collaborating with researchers and motivated students who share my passion for this field is a great opportunity for me to learn and grow. Joining ML4SCI would not only allow me to contribute to important scientific challenges and collaborate with like-minded individuals, but it would also be a valuable asset to my graduate school application. Being a part of such an open-source community would showcase my passion for machine learning and science, as well as my ability to work in a team and apply my skills to real-world problems.

6.2 Education

I am a final year student majoring in Engineering Physics and minoring in Computer Science at the prestigious Indian Institute of Technology, Delhi. With a strong academic background and a CGPA of 9.11, I have always been interested in Machine Learning and Deep Learning. I have a solid understanding of the fundamental principles of physics, which I believe is a valuable asset in understanding complex datasets in ML. My education has provided me with a strong foundation in mathematics, statistics, and data analysis, which are essential skills in the field of Machine Learning. I am confident that my educational background and hands-on experience will enable me to contribute significantly to the project's success.

6.3 Past Experience

During my previous work experiences, I have gained a deep understanding of a variety of machine learning techniques and frameworks. As a Data Scientist intern at Udaan, I developed a Graph Neural Network (GNN) framework to generate holistic embedding. To achieve this, I designed and implemented a Multi-Relational Heterogeneous Graph Auto-encoder Network, where I also customized the loss function based on relevant research papers. I also received an LoR from the Data Science Lead for my contribution.

In my role as an ML Engineer at Torch Investment, I was responsible for performing sentiment analysis on Twitter data and optimizing existing codebase for regression models. Additionally, I have worked as a Research Intern at Griffith University, where I leveraged Graph Convolutional Networks (GCN) for malware detection.

During my B.Tech project, I had the opportunity to work under Professor Abhishek Iyer, where I utilized LHC datasets generated using the ROOT framework. In this project, I performed tasks such as anomaly detection and electron-photon classification.

My past experiences have given me the skills and knowledge required to tackle complex problems in machine learning and data science. I believe that this experience will allow me to make meaningful contributions to the Graph Neural Networks for Particle Momentum Estimation in the CMS Trigger System project.

6.4 Technical Knowledge

Programming languages:

- Experienced in Python
- Familiar with C++

Machine learning:
Completed evaluation tasks demonstrating knowledge of CNN and GNN techniques.

Data analysis:

- Experience working with the ROOT framework for data generation.
- Domain knowledge in particle physics, including the behavior of high-energy particles and the role of muons in the CMS experiment.

Overall, I believe that my skills in programming languages, machine learning, and data analysis, combined with my domain knowledge in particle physics, make me well-suited for this project.

6.5 Interaction with Mentors

Throughout the process of writing the proposal I have had the privilege of interacting with several mentors. I have frequently contacted them on LinkedIn and the mailing list to seek their guidance on the project evaluation tasks, and to clarify any doubts I had regarding the proposal.

Moreover, I have also had the opportunity to connect and interact with other students who have previously contributed to GSoC. These interactions have helped me gain a deeper understanding of the project requirements and have allowed me to learn from their experiences. Overall, these interactions have provided me with a broader perspective on the project and have enabled me to improve my proposal.

7 References

1. Boosted Decision Trees in the Level-1 Muon Endcap Trigger at CMS. link
2. End-to-End Physics Event Classification with CMS Open Data. link
3. Link to GitHub Repository. link