GSoC 2023 Project Proposal

Organisation : ML4SCI

E2E : Graph Neural Networks for End-to-End Particle Identification with the CMS Experiment

MENTORS

Ruchi Chudasama, Emanuele Usai, Shravan Chaudhari, Sergei Gleyzer, Michael Andrews, Eric Reinhardt, Samuel Campbell

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

The End-to-End Deep Learning (E2E) project aims to develop deep learning approaches to reconstruct and identify particles and events of interest in collision events at the Large Hadron Collider (LHC). This project will specifically focus on the development of end-to-end graph neural networks for particle identification, specifically for low-momentum tau identification. Additionally, the project will involve testing and benchmarking the performance of the developed inference engine on GPUs. The end goal is to integrate the developed approaches into reconstruction algorithms for use in offline and high-level trigger systems of the CMS experiment.

2 Introduction

Particle identification is a crucial task in the field of high-energy physics as it helps in the identification of new particles and phenomena that might reveal new physics beyond the current understanding of the Standard Model of particle physics. In this project, the focus is on the identification of particles called taus.

Taus are short-lived particles that are produced in high-energy collisions and quickly decay into other particles. The identification of taus is challenging because their decay products can be difficult to distinguish from the products of other particles. Low-momentum tau identification specifically refers to the identification of taus that have a low momentum, which makes the identification even more challenging.

The reconstruction of events of interest involves analyzing the data collected from particle collisions and identifying the specific events that contain particles or phenomena that are of particular interest. These events may include the production of rare particles, the discovery of new phenomena, or the verification of theoretical predictions. Reconstruction involves reconstructing the trajectory and properties of the particles produced in the collision event based on the measurements made by the detector. This is a complex task that requires sophisticated algorithms and approaches.

The End-to-End Deep Learning (E2E) project aims to develop new deep learning approaches to improve the identification and reconstruction of particles and events of interest. The project will focus on the development of end-to-end graph neural networks, which are a type of neural network that can handle non-linear relationships between particles and identify patterns in the data that may not be visible through traditional methods. The goal is to improve the accuracy and efficiency of the identification and reconstruction tasks, and ultimately to enable the discovery of new physics at the LHC.

3 Related Work

The two papers "End-to-End Physics Event Classification with CMS Open Data" [1] and "End-to-End Jet Classification of Quarks and Gluons with the CMS Open Data" [2] demonstrate the use of end-to-end image-based classifiers for event classification in particle physics experiments using simulated data from the CMS detector at CERN. The classifiers directly use low-level detector data as input, allowing them to learn from the energy deposits and spatial distributions of particles in the detector, rather than relying on reconstructed particle-level information.

In the first paper [1], the authors applied end-to-end classifiers to discriminate between the decay of the Higgs boson into two photons and its leading background sources. The classifiers were able to learn about the angular distribution of photon showers and the energy of their constituent

hits, even when the underlying particles were not fully resolved. The authors demonstrated that these classifiers can be used for complex event topologies, particularly for highly boosted and merged topologies that arise in many beyond-standard-model (BSM) models. They also showed the scalability and flexibility of the end-to-end classifiers when dealing with multiple detector images and networks, where they exhibited robustness against the presence of underlying events and pile-up.

In the second paper [2], the authors constructed end-to-end jet image classifiers to discriminate between quark- and gluon-initiated jets. They used high-fidelity detector images that gave the classifiers direct access to the maximum recorded event information about the jet, differing fundamentally from conventional jet images constructed from reconstructed particle-level information. They achieved effective feature extraction and obtained performance competitive with current state-of-the-art quark versus gluon taggers based on traditional particle-level inputs. The authors noted that precise spatial resolution was of paramount importance, highlighting the critical role played by the track information. They also explored classifying di-quark versus di-gluon QCD events to illustrate ways in which end-to-end jet classifiers can be used to build event classifiers. Finally, they showed that full detector-view event classifiers were robust and versatile against underlying event and pileup outside the jet region-of-interest, making them a compelling tool for complex, multi-body event topologies.

These papers demonstrate the potential of end-to-end image-based classifiers for event classification in particle physics experiments, particularly in cases where traditional reconstruction approaches are difficult or where precise spatial information is crucial. The techniques described in these papers can be applied to a wide range of event topologies and have the potential to improve the sensitivity and accuracy of searches for new physics at the LHC.

4 Evaluation Tasks

4.1 Task 1 : Electron/photon classification

Use a deep learning method of your choice to achieve highest possible classification on this dataset (we ask that you do it both in Keras and in PyTorch). Please provide a Jupyter notebook that shows your solution. The model you submit should have a ROC AUC score of atleast 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 hyperparameters as explained in the paper End-to-End Physics Event Classification with CMS Open Data.

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

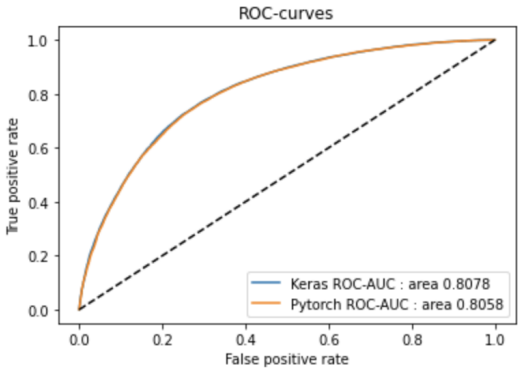

Figure 1: ROC Curve for Task 1

4.2 Task 2 : Deep Learning based Quark-Gluon Classification

Use a Convolutional Neural Network (CNN) architecture of your choice to achieve the highest possible classification on this dataset (in your preferred choice of framework for example: Tensorflow/Keras or Pytorch). Please provide a Jupyter notebook that shows your solution.

Dataset Description

125x125 matrices (three channel images) for two classes of particles quarks and gluons impinging on a calorimeter. For description of 1st dataset please refer to the link provided for the dataset.

Model Architecture

Simple vanilla CNN architecture.

- 2x Conv → Relu → MaxPool
- Linear → Relu → Linear
- Adam optimizer with learning rate 5e-4 which halves every 10 epochs.
- Defining CrossEntropyLoss function

Results

- Testing Accuracy 0.935
- F1 score: 0.935
- ROC-AUC: 0.9558
- ROC Curve

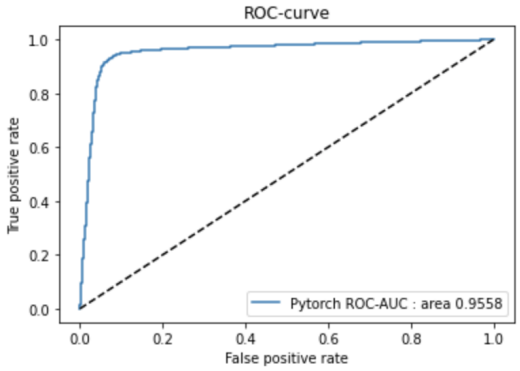

Figure 2: ROC Curve for Task 2

4.3 Specific Task : 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 (Same as Task2).

Model Architecture

Graph Construction -

- Treat all 125x125 pixels as nodes of the graph.
- Keep only the nodes having non-zero absolute sum of channel values. This helps convert the image to a point-cloud representation.
- Now, for each node take the nearest k(=8) neigbours as the edge indices.
- The node features are set as the channel values hence we get 3 features per node.

Architecture-1

- 3-layer GCN network with relu for aggregation of node-level features.
- Readput 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 tp preserve central node information and omits neighborhood normalization completely.
- The same readout layer with gloab 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	0.693	0.654	0.765
2 - GraphConv	0.696	0.730	0.771

Table 2: Results of Task 3

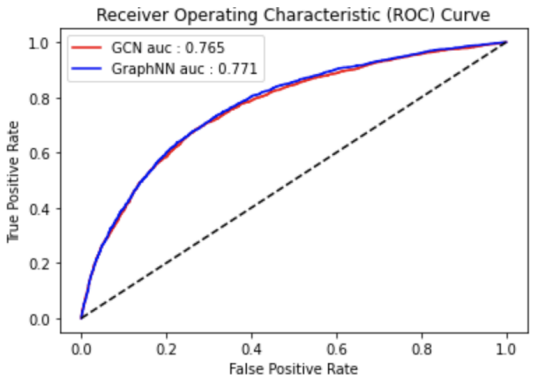

Figure 3: ROC Curve for Task 3

5 Proposed Deliverables

1. A working end-to-end graph neural network model for low-momentum tau identification.
2. A CMSSW inference engine that integrates the developed neural network for use in offline and high-level trigger systems of the CMS experiment.
3. A benchmark report that evaluates the performance of the developed model and inference engine, including testing on GPUs.
4. Documentation that describes the developed model and inference engine, along with instructions for how to integrate and use them in the CMS experiment.
5. A final report that summarizes the project's accomplishments, challenges faced, and lessons learned. This report could also include potential future directions for the project.

6 Project Timeline

May 4 - 28: Community Bonding Period

- Get to know my mentor and the CMS experiment community.
- Read up on documentation related to the project and familiarize myself with the tools and software being used.
- Set up my development environment and make sure I have access to necessary data and resources.

May 29 - June 11: Week 1-2

- Begin development of end-to-end graph neural networks for low-momentum tau identification.
- Explore different architectures and models for graph neural networks.
- Begin implementing code for training and testing the models.

June 12 - June 25: Week 3-4

- Continue development of the graph neural networks.
- Fine-tune the model and experiment with different hyperparameters.
- Start working on code for CMSSW inference engine for use in reconstruction algorithms.

June 26 - July 9: Week 5-6

- Complete development of the graph neural networks.
- Test and benchmark the inference engine on GPUs.
- Begin writing documentation and creating demos and tutorials for users.

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 - July 28: Week 7-8

- Implement any necessary changes to the GNN architecture or training process based on the evaluation results and mentor feedback.
- Continue development and testing of the inference engine and graph neural networks.
- Integrate the inference engine with existing CMS software.
- Work on improving performance and scalability of the models.

July 29 - August 11: Week 9-10

- Refine and finalize the implementation of the inference engine.
- Work on optimizing code and improving efficiency.
- Begin testing and validation of the complete system.

August 12 - August 21: Week 11-12

- Finalize testing and validation of the system.
- Address any issues and bugs that arise during testing.
- Finalize documentation, demos and tutorials for users.

August 21 - August 28: Final week

- Submit final work product and final mentor evaluation.
- Participate in any final presentations or events related.

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.

7 Other Information

7.1 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.

7.2 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.

7.3 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 embeddings. To achieve this, I designed and implemented a Multi-Relational Heterogeneous Graph Autoencoder 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 codebases 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 End-to-End Particle Identification with the CMS Experiment project.

7.4 Technical Knowledge

Programming languages:

- Experienced in Python and Java
- 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 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.

7.5 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.

I am grateful for the mentorship provided by Ruchi Chudasama, Emanuele Usai, Shravan Chaudhari, Sergei Gleyzer, and Eric Reinhardt. Their expertise and feedback have been invaluable in guiding me through the proposal writing process.

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.

8 References

1. End-to-End Physics Event Classification with CMS Open Data. link

2. End-to-End Jet Classification of Quarks and Gluons with the CMS Open Data link
3. Link to GitHub Repository. link